Automatic Construction of Sememe Knowledge Bases via Dictionaries

Fanchao Qi[1,2], **Yangyi Chen**[2,4*], **Fengyu Wang**[1,2], **Zhiyuan Liu**[1,2,3], **Xiao Chen**[5], **Maosong Sun**[1,2,3†]
[1]Department of Computer Science and Technology, Tsinghua University, Beijing China
[2]Beijing National Research Center for Information Science and Technology
[3]Institute for Artificial Intelligence, Tsinghua University, Beijing, China
[4]Huazhong University of Science and Technology [5]Huawei Noah's Ark Lab
qfc17@mails.tsinghua.edu.cn

Abstract

A sememe is defined as the minimum semantic unit in linguistics. Sememe knowledge bases (SKBs), which comprise words annotated with sememes, enable sememes to be applied to natural language processing. So far a large body of research has showcased the unique advantages and effectiveness of SKBs in various tasks. However, most languages have no SKBs, and manual construction of SKBs is time-consuming and labor-intensive. To tackle this challenge, we propose a simple and fully automatic method of building an SKB via an existing dictionary. We use this method to build an English SKB and a French SKB, and conduct comprehensive evaluations from both intrinsic and extrinsic perspectives. Experimental results demonstrate that the automatically built English SKB is even superior to HowNet, the most widely used SKB that takes decades to build manually. And both the English and French SKBs can bring obvious performance enhancement in multiple downstream tasks. All the code and data of this paper (except the copyrighted dictionaries) can be obtained at https://github.com/thunlp/DictSKB.

1 Introduction

A word is the smallest linguistic element that can be used on its own with a particular meaning, but not the smallest semantic unit (O'Grady et al., 1997). The meaning of a word can be divided into smaller components. In linguistics, a *sememe* is defined as the minimum semantic unit of human languages (Bloomfield, 1926). Some linguists believe that meanings of all words can be expressed by a limited set of predefined sememes (Goddard and Wierzbicka, 1994). For example, the basic meaning of "boy" can be expressed by the compositions of human, male and child, while the meaning of "girl" can be expressed by human, female and child, where human, male, female and child are predefined sememes. It is even deemed that this sememe-based semantic system as well as the sememe set is universal among different languages, in which case sememes are also named *universal semantic primitives* (Wierzbicka, 1996).

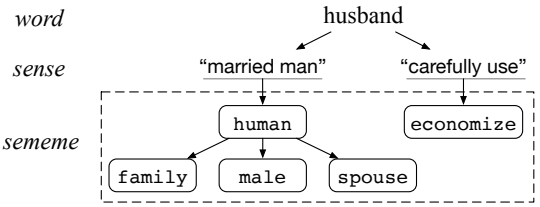

Figure 1: Sememe annotations of the word "husband" in HowNet.

Sememes are implicit in words and cannot be directly used in natural language processing (NLP). Dong and Dong (2006) make a seminal contribution and put the sememe-based semantic system into practice. They define a set of about 2,000 sememes and use them to annotate senses of over 100,000 Chinese and English words, whereupon a sememe knowledge base (SKB) named HowNet is built up. Figure 1 illustrates an example of how words are annotated with sememes in HowNet.

As a sememe-based lexical knowledge base, HowNet is very different from most other lexical knowledge bases like WordNet (Miller, 1998), which extensionally explain meanings of words by word-level relations, e.g., hyponym and meronym. In contrast, HowNet provides intensional definitions using infra-word sememes. This distinctness brings special advantages to HowNet. First, the sememe-to-word semantic compositionality endows HowNet with particular suitability for integration into neural networks (Qi et al., 2019a; Li et al., 2019). The sememes of a word can be re-

*Work done during internship at Tsinghua University
†Corresponding author. Email: sms@tsinghua.edu.cn

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 4673–4686
August 1–6, 2021.

garded as semantic labels and easily incorporated into the neural processing unit of the word, e.g., a cell of RNN (Qin et al., 2020). Second, the nature that a limited set of sememes are used to express meanings of unlimited words makes HowNet very useful in low-data regimes, e.g., improving embeddings of rare words (Sun and Chen, 2016; Niu et al., 2017), where sememes serve as a bridge between high-frequency and rare words. Thus far a large body of research has demonstrated the usefulness of HowNet in various NLP tasks (Qi et al., 2020b).

HowNet is distinctive and valuable, but it covers only two languages. Most languages have no SKBs like HowNet, which deprives NLP in those languages of benefit from sememes. An obvious solution to this problem is to build an SKB for each language manually, but it is not realistic because it would be unimaginably time-consuming and labor-intensive.[1] To address the challenge, previous studies try to extend HowNet to other languages by automatically predicting sememes for words in those languages (Qi et al., 2018, 2020a). However, existing methods are not effective enough, and manual effort is necessary to ensure the correctness of their sememe prediction results.

In this paper, we explore a fully automatic way to build an SKB for a language via dictionaries with a *controlled defining vocabulary*. A dictionary, especially a learner's dictionary, usually uses a well-chosen list of words to construct all its definitions, and the word list is named controlled defining vocabulary (CDV) (Atkins and Rundell, 2008). A CDV is composed of high-frequency words that not only cover the vast majority of texts but also form a semantic basis so as to express meanings of all other words (Nation and Waring, 2004). To some extent, words in a CDV can fit the definition of sememes (Wierzbicka, 1996). This discovery inspires us to utilize a dictionary to build an SKB by regarding the words in its CDV as sememes.

We design a quite simple and quick process for automatically building SKBs based on dictionaries. First, a sememe set is constructed based on the CDV of a dictionary by removing words that are not suitable as sememes (e.g., stop words), then sememes of words are extracted from corresponding definitions, and finally an SKB composed of words annotated with sememes is established. We adopt the process to build an English SKB and a French SKB and conduct both intrinsic and extrinsic evaluations. In intrinsic evaluation, we find that both the SKBs possess high sememe annotation consistency, and the English SKB performs even better than the English part of HowNet. In extrinsic evaluation, we apply the dictionary-based SKBs to several sememe-incorporated models originally designed for HowNet and carry out experiments on different downstream tasks. Experimental results show that incorporating the SKBs can bring consistent performance enhancement, and the English SKB-incorporated models even outperform HowNet-incorporated models. These results demonstrate the usefulness and effectiveness of the dictionary-based SKBs as well as the feasibility of building SKBs via dictionaries.

To conclude, our contributions are threefold: (1) discovering the similarity between sememes and words in the controlled defining vocabulary, which is the first time as far as we know; (2) proposing to automatically build an SKB via a dictionary, which can be achieved by a simple and quick process; and (3) building an English SKB and a French SKB based on dictionaries and demonstrating their effectiveness in multiple downstream tasks.

2 Related Work

2.1 HowNet and Its Applications

Since HowNet was published (Dong and Dong, 2003), it has attracted considerable attention of NLP researchers. In the era of statistical NLP, it plays a very important role in various NLP tasks including word similarity computation (Liu and Li, 2002), word sense disambiguation (Zhang et al., 2005; Duan et al., 2007), text classification (Sun et al., 2007), sentiment analysis (Zhu et al., 2006; Fu et al., 2013), etc.

When deep learning becomes the mainstream approach of NLP, the usefulness of HowNet is also proved in diverse tasks including word representation learning (Sun and Chen, 2016; Niu et al., 2017), language modeling (Gu et al., 2018), semantic composition (Qi et al., 2019a), sequence modeling (Qin et al., 2020), reverse dictionary (Zhang et al., 2020), word sense disambiguation (Hou et al., 2020), textual adversarial attacking (Zang et al., 2020) and backdoor attacking (Qi et al., 2021).

2.2 Expansion of HowNet

To tackle the challenge that many new words are not contained in HowNet, Xie et al. (2017) present

[1] The construction of HowNet takes several linguistic experts more than two decades.

the task of lexical sememe prediction, aiming to expand HowNet by automatically predicting sememes for new words. They propose two simple and effective sememe prediction methods inspired by recommendation system. Jin et al. (2018) further incorporate Chinese characters into sememe prediction and achieve higher performance when predicting sememes for Chinese words.

Another research line focuses on extending HowNet to other languages. Qi et al. (2018) propose the task of cross-lingual lexical sememe prediction, aiming to extend HowNet to a new language by predicting sememes for words in that language. Qi et al. (2020a) present a more efficient way to extend HowNet to other languages, i.e., building a multilingual SKB based on BabelNet (Navigli and Ponzetto, 2012). BabelNet is composed of multilingual synsets that contain synonyms in many languages. Words (synonyms) in a synset have the same meaning and hence the same sememes. Therefore, they propose to predict sememes for the multilingual synsets, by which all the words in synsets will obtain predicted sememes at the same time.

Limited by the accuracy of sememe prediction, manual examination is necessary if we want to put the above HowNet expansion methods into service. In contrast, our proposed dictionary-based SKB construction method is completely automatic and can build a usable SKB very quickly.

2.3 Applications of Dictionaries

Dictionaries are handy and high-quality resources for NLP research. A main application of dictionaries is word sense disambiguation, where dictionaries play the role of sense inventory, and their definitions provide abundant semantic information for each sense (Lesk, 1986; Luo et al., 2018a,b; Kumar et al., 2019; Huang et al., 2019; Du et al., 2019; Blevins and Zettlemoyer, 2020). The semantic information in dictionary definitions is also used to improve word representation learning (Tissier et al., 2017; Bahdanau et al., 2017; Bosc and Vincent, 2018; Scheepers et al., 2018). In addition, dictionary definitions are also utilized in reverse dictionary (Hill et al., 2016; Pilehvar, 2019; Zhang et al., 2020), knowledge graph embedding (Zhong et al., 2015; Xie et al., 2016), reading comprehension (Long et al., 2017), etc. As far as we know, this paper is the first work to utilize dictionaries to build SKBs.

3 Building an SKB via a Dictionary

In this section, we detail the process of building an SKB via a dictionary. We take the building process of an English SKB based on *Longman Dictionary of Contemporary English* (LDOCE) (Bullon, 2006), a highly influential English learner's dictionary, as an example, and the building method can be readily generalized to other languages or dictionaries.[2]

3.1 Constructing the Sememe Set

We first construct the sememe set from the CDV of LDOCE by removing some words. LDOCE uses an approximately $2,000$-word CDV named Longman Defining Vocabulary (Bullock, 2011), which is developed from General Service List (West, 1953), a famous high-frequency word list for English learners. The CDV includes some stop words such as "that" and "to", which bear insignificant meanings and are not suitable as sememes. Thus, we filter them out according to the stop word list of NLTK (Loper and Bird, 2002). But negators like "not" are retained because they are critical to the meanings of words. In addition, according to previous work (Xie et al., 2017; Qin et al., 2020), sememes that are annotated to too many or too few words are usually uninformative and ineffective to downstream applications. Therefore, we count the frequencies of words in the CDV occurring in all definitions and empirically remove the most frequent 1% and the infrequent 10%. So far we have obtained the sememe set that is composed of $2,046$ sememes.

3.2 Extracting Sememes from Definitions

Next, we extract sememes for each sense of each word from its definition. We take the word "beautiful" as a running example to illustrate the process of sememe extraction, as shown in Figure 2.

"beautiful" has two senses in LDOCE, and both of them are adjective. For each sense, we first use NLTK to normalize its definition including tokenization and lemmatization. For example, the definition of its first sense is normalized into a sequence of tokens: {"someone", "or", "something", "that", "be", "beautiful", "be", "extremely", "attractive", "to", "look", "at"}. Then we remove the tokens that are not in the sememe set. In the above example, "someone", "something", "or", "that", "be", "to" and "at" are removed. So far we obtain the sememes

[2]The building process and evaluation results of the French SKB are given in Appendix A and B.

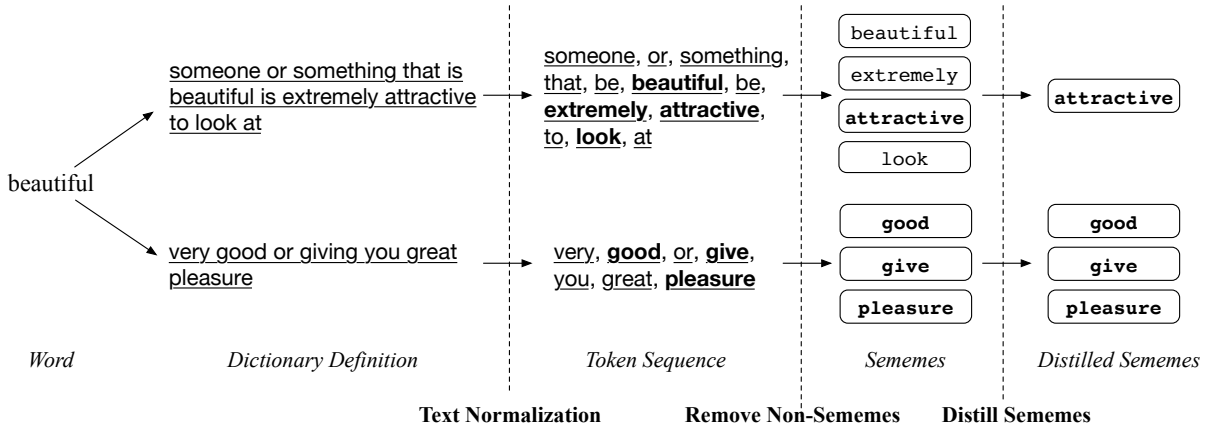

Figure 2: The process of extracting sememes from dictionary definitions for the word "beautiful".

SKB	#Word/Phrase	#Sense	#Sememe	#AvgSem
HowNet	50,879	111,519	2,187	2.26
EDSKB	70,218	105,160	2,046	6.03
EDSKB*	70,218	105,160	1,682[3]	2.04

Table 1: Statistics of EDSKB, its distilled version EDSKB* and the English part of HowNet.[4] #AvgSem denotes the average sememe number per sense.

of the first sense of "beautiful": {beautiful, extremely, attractive, look}. In a similar way, we can obtain the sememes of its second sense: {good, give, pleasure}.

By repeating this process on all the words of LDOCE, we obtain an English SKB that we name EDSKB. Its statistics are shown in Table 1.

3.3 Distilling Sememes of Senses

By comparison with HowNet, we find that the sememe set of EDSKB is smaller (EDSKB $2,046$ vs. HowNet $2,187$), but its average sememe number per sense is much larger (EDSKB 6.03 vs. HowNet 2.26), which means the sememes of EDSKB are utilized more fully and effectively. Moreover, annotating a sense with more sememes can explain the sense more accurately and finely. Nevertheless, it would also increase the distinguishability between different senses/words, which has an adverse effect on some downstream tasks. For example, word-level textual adversarial attacking conducts word substitution to generate adversarial examples, in which fewer substitute words usually lead to lower attack success rates (Wang et al., 2019; Zang et al., 2020). In a sememe-based word substitution strategy (Zang et al., 2020), more sememes per sense mean fewer substitute words that share the same sememes can be found, which will decrease the final adversarial attack success rate. To address this problem, we intend to craft an extra distilled version of EDSKB by distilling its sememes of senses.

To this end, we need to determine the importance of each sememe of a sense, and remove the relatively unimportant sememes. Here we resort to dependency parsing (Kubler et al., 2009). Dependency parsing is used to analyze syntactic structures of a sentence by identifying the word that another word is "dependent" on, e.g., the adjective is dependent on the noun in an adjective-noun phrase. We believe that the words with more dependents are more important in a definition. Hence, we define the importance score of a sememe for a sense as the number of the dependents of its original word in the definition.

Then, we empirically remove the sememes whose importance scores are below the highest importance score minus t for the senses having m or more sememes. Here t and m are two hyper-parameters and are tuned to 1 and 4 respectively, based on the performance on the validation sets of downstream tasks, especially adversarial attacking. For example, for the first sense of "beautiful", by using AllenNLP (Gardner et al., 2018) to conduct dependency parsing on its definition, we obtain the numbers of dependents of all the words in the definition. Correspondingly, we get the importance scores of the four sememes {beautiful, extremely, attractive, look}, which are 2, 0, 6 and 0 respectively. The highest impor-

[3]Notice that its sememe set shrinks because some sememes are not annotated to any senses anymore.

[4]The data of HowNet are obtained from OpenHowNet (Qi et al., 2019b).

tance score is 6 and thus the sememes whose importance scores are less than 5 are removed, i.e., `beautiful`, `extremely` and `look`. Finally, the remaining sememe of the first sense of "beautiful" is {`attractive`}. As for the second sense, it has only 3 sememes and all of them are retained (the sememe number threshold for sememe reduction is $m = 4$). Therefore, its final sememes after reduction are still {`good`, `give`, `pleasure`}.

By repeating the above process on all the words, we obtain a distilled version of EDSKB (signified by EDSKB*), whose average sememe number per sense is comparable with HowNet (2.04 vs. 2.26). Its detailed statistics are also shown in Table 1.

Later experiments (on both English and French) show that the full version outperforms the distilled version in some downstream tasks while not in others. In practice, we can build both full and distilled versions and conduct experiments to see which one is better in a specific task. It is affordable to build and evaluate two versions.

4 Intrinsic Evaluation

In this section, we conduct an intrinsic evaluation to assess the sememe annotation consistency of EDSKB. Sememe annotation consistency measures how compatible the sememe annotations for different words/senses are, e.g., whether two synonyms are annotated with exactly the same sememes. The sememe annotation consistency of an SKB not only reflects its intrinsic quality but also has impact on its effectiveness in downstream tasks.

We evaluate both full and distilled versions of EDSKB, and the English part of HowNet for comparison. We adopt a sememe consistency assessment method named CCSA (Liu et al., 2020), which is designed for HowNet originally but can be used for any SKB. This method is motivated by the idea that semantically close senses should have similar sememes, which conforms to the linguistic definition of sememes. It actually implements a sememe prediction process that predicts sememes for a small proportion of senses according to the sememe annotations of the other senses. The sememe prediction method it adopts is based on collaborative filtering (Xie et al., 2017), and tends to predict the sememes that are annotated to semantically close senses to the target sense. Therefore, higher sememe prediction performance means the semantically close senses are annotated with more similar sememes, and the sememe annotations are more consistent. Correspondingly, the sememe annotation consistency of an SKB is measured by two sememe prediction evaluation metrics, namely mean average precision (MAP) and F1 score.

SKB	MAP	F1
HowNet	0.93	<u>0.91</u>
EDSKB	0.88	0.86
EDSKB*	**0.95**	<u>0.91</u>

Table 2: Sememe annotation consistency results. The **boldfaced** results show statistically significant improvement over the best results from baselines with $p<0.1$ given by t-test, and the <u>underlined</u> results represent having no significant difference.[5]

Table 2 lists the evaluation results of sememe annotation consistency. We can see that the distilled version of EDSKB has overall higher consistency than HowNet, and the full version of EDSKB yields lower consistency results. It is not strange because CCSA is based on sememe prediction and according to previous work (Qi et al., 2020a), senses with more sememes usually have lower prediction performance. Since the full version of EDSKB has much more sememes per sense than HowNet, it is actually not fair to compare their consistency using CCSA. The distilled version of EDSKB has a similar average sememe number as HowNet, and its superior results can demonstrate the great consistency of the dictionary-based SKB.

5 Extrinsic Evaluation

In this section, we conduct extrinsic evaluations to assess the effectiveness of EDSKB in downstream tasks. We pick three representative sememe-incorporated neural network models that are used for language modeling, sequence modeling and textual adversarial attacking tasks, respectively. All of them are originally designed for HowNet and have demonstrated efficacy on their respective tasks.

5.1 Language Modeling

In this subsection, we try to apply EDSKB to the task of language modeling. We use SDLM (Gu et al., 2018), a sememe incorporation method for language models, to incorporate EDSKB into two representative language models based on recurrent neural networks (RNNs).

Language modeling is aimed at predicting the next word given previous context (Bengio et al.,

[5]The same is true for the following tables.

2003). Language models based on RNNs, especially LSTMs (Hochreiter and Schmidhuber, 1997), are very popular, which use RNNs to encode the previous text into a vector and then feed the vector to a classifier to predict the next word. SDLM reforms the prediction process. Instead of directly predicting the next word, SDLM predicts sememes first, then senses and finally the next word.

Base Models We use two representative LSTM-based language models as the base models into which EDSKB is incorporated by SDLM.

- **Tied LSTM** (Zaremba et al., 2014), which enhances a vanilla two-layer LSTM language model by introducing dropout and weight tying. We use its large version whose word embedding and hidden vector sizes are $1,500$.
- **AWD-LSTM** (Merity et al., 2018), which adopts several regularization and optimization strategies including DropConnect (Wan et al., 2013) and non-monotonically triggered average stochastic gradient descent, and is a very strong baseline language model. Its hidden vector size is $1,150$ and word embedding size is 400.

Baseline Methods In addition to the two original base models, we additionally use SDLM to incorporate HowNet into the base models as baseline methods.

Datasets We choose two benchmark language modeling datasets for evaluation, namely Penn Treebank (PTB) (Marcus et al., 1993) and WikiText-2 (Merity et al., 2017). PTB consists of news stories from the Wall Street Journal. The numbers of tokens in its training, validation and test sets are $887,521$, $70,390$ and $78,669$, respectively. WikiText-2 is made up of Wikipedia articles, and it has $2,088,628$, $217,646$ and $245,569$ tokens in its training, validation and test sets.

Experimental Settings In our experiments, we use the official implementation of SDLM and its default hyper-parameters as well as training methods. The evaluation metric is perplexity. The lower perplexity a language model computes, the better the language model is.

Experimental Results Table 3 lists the perplexity results on the two datasets. We observe that the models incorporated with EDSKB, especially the full version, consistently outperform the two base models without sememe incorporation and even the HowNet-incorporated models. These results demonstrate the effectiveness of EDSKB in language modeling.

Dataset	PTB		WikiText-2	
Model	Valid	Test	Valid	Test
Tied LSTM	63.92	63.98	53.10	51.41
+HowNet	58.93	58.95	48.83	47.28
+EDSKB	**58.81**	**58.82**	**43.38**	**42.15**
+EDSKB*	60.17	60.15	45.18	42.59
AWD-LSTM	58.89	59.24	45.29	44.13
+HowNet	58.95	58.92	46.84	45.29
+EDSKB	**56.94**	**57.13**	**42.44**	**41.25**
+EDSKB*	58.63	58.59	43.85	43.95

Table 3: Perplexity results of different language models on the validation and test sets of PTB and WikiTex-2.

5.2 Sequence Modeling

In this subsection, we incorporate EDSKB into RNNs to improve their sequence modeling ability by SememeCell (Qin et al., 2020), a sememe incorporation method for enhancing RNNs.

SememeCell uses a special RNN cell to encode sememes of a word into a latent vector and transmits it to the corresponding RNN cell of the word, aiming to inject the semantic information of sememes into RNNs. It has demonstrated its effectiveness in improving the sequence modeling ability of RNNs in multiple downstream tasks, including natural language inference, sentiment analysis and paraphrase detection (Qin et al., 2020).

Base Models Following Qin et al. (2020), we choose two most representative RNNs, namely LSTM, GRU (Cho et al., 2014), and their bidirectional versions (BiLSTM and BiGRU) as the base models, into which sememes are incorporated by SememeCell.

Baseline Methods In addition to the vanilla and HowNet-incorporated RNNs, we also design another two baseline methods.

- **+Pseudo**. RNNs incorporated with either EDSKB or HowNet have a little more parameters than vanilla RNNs. To eliminate the possible effect brought by more parameters, we build a pseudo-SKB named Pseudo. Specifically, for each sense in EDSKB, we substitute its sememes with the same number of meaningless labels. The labels are randomly sampled from a label set with the same size as the sememe set of ED-

SKB. We use SememeCell to incorporate this pseudo-SKB into the two base models as baselines, which have exactly the same numbers of parameters as EDSKB-incorporated models.

- **+Definition**. EDSKB is obtained from dictionary definitions by the transformation from a sequence of words (definition) into several discrete semantic labels (sememes). We intend to compare the EDSKB-incorporated models and models incorporated with the complete dictionary definitions. Since SememeCell only takes a vector (i.e., the sum of sememe embeddings) as input, we can leverage it to incorporate definitions into RNNs by encoding definitions into vectors with a sentence encoder. Specifically, we choose the powerful pre-trained language model BERT (Devlin et al., 2019) as the sentence encoder and use the hidden vector of the `[CLS]` token as the definition embedding. The definition-incorporated RNN models are also baselines.

Downstream Tasks and Datasets RNNs are basic sequence encoders and can be used in many downstream NLP tasks. Following Qin et al. (2020), we choose two representative tasks to evaluate the sentence modeling ability of EDSKB-incorporated RNNs.

- Natural language inference (NLI), which is aimed at determining whether a natural language hypothesis can be inferred from a premise. It is a typical sentence pair classification task. We use the SNLI dataset (Bowman et al., 2015) for evaluation. SNLI contains about $570,000$ English premise-hypothesis pairs, and each pair is manually labeled one of three relation labels, namely "entailment", "contradiction" and "neutral".
- Sentiment analysis, which aims to recognize the sentiment orientation of a sentence and is a typical single sentence classification task. Following Qin et al. (2020), we use the CR dataset (Hu and Liu, 2004) for evaluation. It contains about $8,000$ product reviews and each review is labeled with "positive" or "negative".

Experimental Settings We use the official implementation of SememeCell (Qin et al., 2020) and the default hyper-parameter settings and training methods, where the embedding size (for both word and sememe embeddings) is 300 and hidden size is $2,048$. In the baseline method +Definition, to keep the definition vector size comparable with sememe embedding size, we choose the medium version of BERT, which has 512-dimensional hidden vectors and 8 layers.[6] As for evaluation metrics, we use accuracy for both NLI and sentiment analysis.

Dataset	Method	LSTM	GRU	BiLSTM	BiGRU
SNLI	vanilla	80.66	82.00	81.30	81.61
	+Pseudo	81.28	80.90	81.91	82.07
	+HowNet	81.87	82.90	82.55	83.15
	+Definition	81.62	82.80	81.10	83.22
	+EDSKB	**82.32**	**83.18**	82.54	**83.55**
	+EDSKB*	81.78	82.10	82.11	82.35
CR	vanilla	74.17	76.37	77.62	78.76
	+Pseudo	73.96	75.44	76.16	78.20
	+HowNet	76.47	78.57	77.66	76.25
	+Definition	76.29	78.20	77.19	77.77
	+EDSKB	**77.51**	**79.68**	**78.95**	**78.88**
	+EDSKB*	75.09	77.54	76.90	78.18

Table 4: Accuracy results of different models on the test sets of SNLI and CR.

Experimental Results Table 4 shows the evaluation results on the test sets of SNLI and CR. We can see that RNNs incorporated with dictionary-based SKB, especially the full version (+EDSKB), yield overall better results than vanilla RNNs, which proves that the dictionary-based SKB can improve the sequence modeling ability of RNNs. Furthermore, the +EDSKB models outperform +Pseudo models that have the same number of parameters, +Definition models that have the same semantic information source, and +HowNet models that incorporate another SKB. These results demonstrate the superiority of discrete sememes over definitions, and the advantage of dictionary-based SKB over HowNet in enhancing RNNs. +Pseudo performs slightly better than vanilla in some cases, which is probably because +Pseudo utilizes the random meaningless labels as noises. The addition of noise has been proven a regularization method for mitigating overfitting and improving performance in neural networks (Bishop, 1995).

5.3 Textual Adversarial Attacking

In this subsection, we investigate the effectiveness of EDSKB in textual adversarial attacking.

Adversarial attacking has attracted considerable research attention recently, mainly because it can reveal the vulnerability of neural network models and help improve their robustness and interpretability (Xu et al., 2020). Adversarial attacks use *ad-*

[6] https://github.com/google-research/bert

versarial examples (Szegedy et al., 2014), which are maliciously crafted by perturbing the original model input, to fool the victim model. In textual adversarial attacking, word-level attack methods, mainly based on word substitution, are a kind of popular attack method and have demonstrated overall better attack performance (Wang et al., 2019).

Zang et al. (2020) decompose the process of word-level attacks into two steps: (1) determining the substitute set for each word in the original input via a word substitution strategy, e.g., synonym-based and word embedding-based substitution strategies; and (2) searching the combinations of each original word's substitutes for adversarial examples that can successfully fool the victim model.

They also propose an adversarial attack approach that employs a sememe-based word substitution strategy and achieves strong attack performance. The sememe-based word substitution strategy essentially regards a word w_1 as the substitute of another word w_2, if one sense of w_1 has the same sememes as one sense of w_2, according to an SKB. We use this approach to conduct textual adversarial attacks and measure the attack performance.

Baseline Methods In addition to the original sememe-based attack approach that uses HowNet as the SKB, we choose some other baseline methods for comparison. Notice that all these baseline methods use the same approach to search for adversarial examples (the aforementioned step 2) and differ in word substitution strategies (step 1) only.

- **+Synonym**, the attack method that uses synonym-based word substitution strategy. Following previous work (Ren et al., 2019), we use WordNet as the thesaurus and the words in a synset can be regarded as substitutes of each other.
- **+Definition**, the attack method that uses a definition-based word substitution strategy. Inspired by word embedding-based word substitution, we encode the definition of each sense of words into a vector and define the similarity between two words as the cosine similarity between their closest definition vectors. Then, a certain number of words that are most similar to the target word are regarded as its substitutes. Specifically, we still use the medium-size BERT to encode definitions into 512-dimensional vectors. And the number of substitutes of each word is the same as that in the sememe-based substitution strategy.

Victim	Attack Method	ASR	%M	%IGE	PPL
BiLSTM	+Synonym	79.0	10.45	7.59	593.09
	+Definition	90.0	8.76	7.56	518.71
	+HowNet	93.6	9.02	2.57	**468.92**
	+EDSKB	26.5	8.27	3.77	538.46
	+EDSKB*	**94.0**	8.29	**1.27**	507.34
BERT	+Synonym	81.3	9.22	8.00	576.82
	+Definition	86.3	8.03	7.18	538.00
	+HowNet	91.2	8.25	2.08	503.06
	+EDSKB	29.7	8.10	3.36	**485.00**
	+EDSKB*	**93.3**	**7.66**	**1.07**	544.51

Table 5: Adversarial attack results of different word substitution strategies. ASR is short for attack success rate. %M, %IGE and PPL denote word modification rate, increase rate of grammatical errors and perplexity, respectively.

In this task, the +Pseudo baseline in the previous section cannot work because it would regard random words as substitutes of the target word.

Victim Models and Datasets Following Zang et al. (2020), we choose BiLSTM and BERT, specifically $\text{BERT}_{\text{BASE}}$ as the victim models we intend to attack. The evaluation task is sentiment analysis and the evaluation dataset is SST-2 (Socher et al., 2013). SST-2 comprises about $10,000$ sentences in movie reviews and each sentence is labeled with "positive" or "negative". The accuracy results of BiLSTM and BERT on the test set of SST-2 are 83.75 and 90.28.

Experimental Settings We use the official implementation of the sememe-based attack approach (Zang et al., 2020) and the default hyper-parameter settings.

Evaluation Metrics Following Zang et al. (2020), we use attack success rate to measure the effectiveness of an attack method and three metrics to assess the quality of its adversarial examples. The three metrics are (1) word modification rate, the percentage of words in the original input that are perturbed; (2) increase rate of grammatical errors in adversarial examples compared with original input, where LanguageTool grammar checker is used; and (3) perplexity given by GPT-2 (Radford et al., 2019) that is used to measure the fluency of adversarial examples. The lower the three metrics are, the better the quality of adversarial examples is.

Experimental Results According to Table 5, we find that the attack method based on EDSKB*

Word	SKB	Sememes
screenwriter	HowNet	human, occupation, entertainment, compile, shows
	EDSKB	someone, write, play, film, television
	EDSKB*	write, play, film, television
tweet	HowNet	InstitutePlace, ProperName, produce, software, LookFor, document, information, internet
	EDSKB	Sense 1: bird, make, high, small, short, sound Sense 2: service, message, network, short, send, use, social
	EDSKB*	Sense 1: bird, sound Sense 2: message, send, use, network

Table 6: Two cases of sememe annotations in HowNet, EDSKB and EDSKB*.

not only achieves the highest attack success rates but also generates adversarial examples with overall higher quality. These results show that the dictionary-based SKB EDSKB* can better capture the semantic relations between words and find appropriate substitutes for adversarial attacks. Attack success rates of the EDSKB-based method are extremely low. It is because EDSKB has too many sememes per sense, which causes the found substitutes to be very few (EDSKB 1.6, EDSKB* 12.6 and HowNet 15.3 on average), according to the sememe-based word substitution strategy that requires substitutes to have the same sememes.

6 Case Study on Sememe Annotations

In this section, we give two cases of sememe annotations in EDSKB and EDSKB* as well as HowNet in Table 6.

The first case is the word "screenwriter". In HowNet, this word has only one sense that is annotated by five sememes, as listed in the second row of Table 6. As for EDSKB and EDSKB*, according to *Longman Dictionary of Contemporary English* (LDOCE), this word also has only one sense whose definition is "someone who writes plays for film or television". EDSKB provides five sememes and one (someone) is filtered out in EDSKB*. By comparison, we can find that sememes in EDSKB and EDSKB* can represent the meaning of the word more specifically, e.g, write and play, while sememes in HowNet seem to express a more general meaning.

The second case is about the word "tweet". HowNet only annotates one sense for this word, i.e., "to send a message on Twitter". As for EDSKB and EDSKB*, since LDOCE contains the basic meaning of this word, i.e., "to make the short high sound of a small bird", the sememes including bird and sound are extracted to express this meaning. In addition, for the shared sense, sememes in EDSKB and EDSKB* are more succinct than those in HowNet, e.g., message in EDSKB/EDSKB* can better describe the core meaning of "tweet" than document and information in HowNet.

From the two cases, we can see the advantage of the dictionary-based SKBs over HowNet in terms of sememe annotations. We hope that the dictionary-based SKBs can be used to perfect HowNet by supplying more senses and annotating more suitable sememes.

7 Conclusion and Future Work

In this paper, we propose to utilize a dictionary to build an SKB for the first time, which can be implemented by a simple, quick and fully automatic process. We try utilizing existing dictionaries to build an English SKB and a French SKB, and demonstrate their effectiveness on multiple NLP tasks. Extensive experimental results prove the reliability and practicality of our idea about dictionary-based SKB construction.

It is worth mentioning that although EDSKB delivers better empirical results than HowNet, HowNet has its unique advantages including better interpretability and multilinguality. In the future, therefore, we will systematically compare the sememe annotations in EDSKB and HowNet and try to use EDSKB to improve and expand HowNet. Besides, the hierarchical structures of sememes in HowNet are neglected in this paper. We will also explore to extract sememes with hierarchy from dictionary definitions.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (Grant No. 2020AAA0106502 and No. 2020AAA0106501) and Beijing Academy of Artificial Intelligence (BAAI). We also thank all the anonymous reviewers for their valuable comments and suggestions.

Ethical Considerations

In this paper, we use two copyrighted dictionaries, namely Longman Dictionary of Contemporary English and Le Petit Robert French Dictionary. We extract data from the electronic versions of the two dictionaries we bought for the research purpose only. We will not release the dictionary data. In addition, the datasets we use in downstream tasks are all open and free (actually also widely used).

The task we tackle is sememe knowledge base construction, which is not a practical application and is only related to NLP research. Therefore, the datasets we build and the models we use would not be misused by common people.

In addition, since we do not use very large models, the required energy in this work is very limited. Finally, we use no demographic or identity characteristics.

References

BT Sue Atkins and Michael Rundell. 2008. *The Oxford guide to practical lexicography*. Oxford University Press.

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

Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3(Feb):1137–1155.

Chris M Bishop. 1995. Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1):108–116.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of word sense disambiguation with gloss-informed biencoders. In *Proceedings of ACL*.

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

Tom Bosc and Pascal Vincent. 2018. Auto-encoding dictionary definitions into consistent word embeddings. In *Proceedings of EMNLP*.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of EMNLP*.

David Bullock. 2011. Nsm+ ldoce: A non-circular dictionary of english. *International Journal of Lexicography*, 24(2):226–240.

Stephen Bullon. 2006. *Longman Dictionary of Contemporary English*. Pearson Education Limited.

Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of EMNLP*.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*.

Zhendong Dong and Qiang Dong. 2003. Hownet-a hybrid language and knowledge resource. In *Proceedings of International Conference on Natural Language Processing and Knowledge Engineering*. IEEE.

Zhendong Dong and Qiang Dong. 2006. *HowNet and the Computation of Meaning (With CD-Rom)*. World Scientific.

Jiaju Du, Fanchao Qi, and Maosong Sun. 2019. Using bert for word sense disambiguation. *arXiv preprint arXiv:1909.08358*.

Xiangyu Duan, Jun Zhao, and Bo Xu. 2007. Word sense disambiguation through sememe labeling. In *Proceedings of IJCAI*.

Xianghua Fu, Guo Liu, Yanyan Guo, and Zhiqiang Wang. 2013. Multi-aspect sentiment analysis for Chinese online social reviews based on topic modeling and HowNet lexicon. *Knowledge-Based Systems*, 37:186–195.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. Allennlp: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6.

Cliff Goddard and Anna Wierzbicka. 1994. *Semantic and lexical universals: Theory and empirical findings*, volume 25. John Benjamins Publishing.

Yihong Gu, Jun Yan, Hao Zhu, Zhiyuan Liu, Ruobing Xie, Maosong Sun, Fen Lin, and Leyu Lin. 2018. Language modeling with sparse product of sememe experts. In *Proceedings of EMNLP*.

Felix Hill, KyungHyun Cho, Anna Korhonen, and Yoshua Bengio. 2016. Learning to understand phrases by embedding the dictionary. *Transactions of the Association for Computational Linguistics*, 4:17–30.

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

Bairu Hou, Fanchao Qi, Yuan Zang, Xurui Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Try to substitute: An unsupervised chinese word sense disambiguation method based on hownet. In *Proceedings of COLING*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of KDD*.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuan-Jing Huang. 2019. Glossbert: Bert for word sense disambiguation with gloss knowledge. In *Proceedings of EMNLP-IJCNLP*.

Huiming Jin, Hao Zhu, Zhiyuan Liu, Ruobing Xie, Maosong Sun, Fen Lin, and Leyu Lin. 2018. Incorporating chinese characters of words for lexical sememe prediction. In *Proceedings of ACL*.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A Smith. 2020. The multilingual amazon reviews corpus. In *Proceedings EMNLP*.

Sandra Kubler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis lectures on human language technologies*, 1(1):1–127.

Sawan Kumar, Sharmistha Jat, Karan Saxena, and Partha Talukdar. 2019. Zero-shot word sense disambiguation using sense definition embeddings. In *Proceedings of ACL*.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbe, Laurent Besacier, and Didier Schwab. 2020. Flaubert: Unsupervised language model pre-training for french. In *Proceedings of LREC*.

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

Ziran Li, Ning Ding, Zhiyuan Liu, Haitao Zheng, and Ying Shen. 2019. Chinese relation extraction with multi-grained information and external linguistic knowledge. In *Proceedings of ACL*.

Qun Liu and Sujian Li. 2002. Word similarity computing based on HowNet. *International Journal of Computational Linguistics & Chinese Language Processing*, 7(2):59–76.

Yangguang Liu, Fanchao Qi, Zhiyuan Liu, and Maosong Sun. 2020. Research on consistency check of sememe annotations in hownet. *Journal of Chinese Information Processing*.

Teng Long, Emmanuel Bengio, Ryan Lowe, Jackie Chi Kit Cheung, and Doina Precup. 2017. World knowledge for reading comprehension: Rare entity prediction with hierarchical lstms using external descriptions. In *Proceedings of EMNLP*.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*.

Fuli Luo, Tianyu Liu, Zexue He, Qiaolin Xia, Zhifang Sui, and Baobao Chang. 2018a. Leveraging gloss knowledge in neural word sense disambiguation by hierarchical co-attention. In *Proceedings of EMNLP*.

Fuli Luo, Tianyu Liu, Qiaolin Xia, Baobao Chang, and Zhifang Sui. 2018b. Incorporating glosses into neural word sense disambiguation. In *Proceedings of ACL*.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2).

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018. Regularizing and optimizing lstm language models. In *Proceedings of ICLR*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *Proceedings of ICLR*.

George Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

Paul Nation and Robert Waring. 2004. Vocabulary size, text coverage and word lists.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Yilin Niu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2017. Improved word representation learning with sememes. In *Proceedings of ACL*.

William O'Grady, Michael Dobrovolsky, and Francis Katamba. 1997. *Contemporary linguistics*.

Mohammad Taher Pilehvar. 2019. On the importance of distinguishing word meaning representations: A case study on reverse dictionary mapping. In *Proceedings of NAACL-HLT*.

Fanchao Qi, Liang Chang, Maosong Sun, Sicong Ouyang, and Zhiyuan Liu. 2020a. Towards building a multilingual sememe knowledge base: Predicting sememes for babelnet synsets. In *Proceedings of AAAI*.

Fanchao Qi, Junjie Huang, Chenghao Yang, Zhiyuan Liu, Xiao Chen, Qun Liu, and Maosong Sun. 2019a. Modeling semantic compositionality with sememe knowledge. In *Proceedings of ACL*.

Fanchao Qi, Yankai Lin, Maosong Sun, Hao Zhu, Ruobing Xie, and Zhiyuan Liu. 2018. Cross-lingual lexical sememe prediction. In *Proceedings of EMNLP*.

Fanchao Qi, Ruobing Xie, Yuan Zang, Zhiyuan Liu, and Maosong Sun. 2020b. Sememe knowledge computation: a review of recent advances in application and expansion of sememe knowledge bases. *Frontiers of Computer Science*.

Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Qiang Dong, Maosong Sun, and Zhendong Dong. 2019b. Openhownet: An open sememe-based lexical knowledge base. *arXiv preprint arXiv:1901.09957*.

Fanchao Qi, Yuan Yao, Sophia Xu, Zhiyuan Liu, and Maosong Sun. 2021. Turn the combination lock: Learnable textual backdoor attacks via word substitution. In *Proceedings of ACL-IJCNLP*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020c. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of ACL*.

Yujia Qin, Fanchao Qi, Sicong Ouyang, Zhiyuan Liu, Cheng Yang, Yasheng Wang, Qun Liu, and Maosong Sun. 2020. Improving sequence modeling ability of recurrent neural networks via sememes. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of ACL*.

Paul Robert, Alain Rey, and Josette Rey-Debove. 2015. *Dictionnaire Le Petit Robert 2016*. Dictionnaires Le Robert.

Thijs Scheepers, Evangelos Kanoulas, and Efstratios Gavves. 2018. Improving word embedding compositionality using lexicographic definitions. In *Proceedings of WWW*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*.

Jingguang Sun, Dongfeng Cai, Dexin Lv, and Yanju Dong. 2007. HowNet based Chinese question automatic classification. *Journal of Chinese Information Processing*, 21(1):90–95.

Maosong Sun and Xinxiong Chen. 2016. Embedding for words and word senses based on human annotated knowledge base: Use HowNet as a case study. *Journal of Chinese information processing*, 30(6).

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *Proceedings of ICLR*.

Julien Tissier, Christophe Gravier, and Amaury Habrard. 2017. Dict2vec: Learning word embeddings using lexical dictionaries. In *Proceedings of EMNLP*.

Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. 2013. Regularization of neural networks using dropconnect. In *Proceedings of ICML*.

Xiaosen Wang, Hao Jin, and Kun He. 2019. Natural language adversarial attacks and defenses in word level. *arXiv preprint arXiv:1909.06723*.

Michael Philip West. 1953. *A general service list of English words: with semantic frequencies and a supplementary word-list for the writing of popular science and technology*. Longmans, Green.

Anna Wierzbicka. 1996. *Semantics: Primes and universals: Primes and universals*. Oxford University Press, UK.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of NAACL-HLT*.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions. In *Proceedings of AAAI*.

Ruobing Xie, Xingchi Yuan, Zhiyuan Liu, and Maosong Sun. 2017. Lexical sememe prediction via word embeddings and matrix factorization. In *Proceedings of IJCAI*.

Han Xu, Yao Ma, Hao-Chen Liu, Debayan Deb, Hui Liu, Ji-Liang Tang, and Anil K Jain. 2020. Adversarial attacks and defenses in images, graphs and text: A review. *International Journal of Automation and Computing*, 17(2):151–178.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of ACL*.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Lei Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2020. Multi-channel reverse dictionary model. In *Proceedings of AAAI*.

Yuntao Zhang, Ling Gong, and Yongcheng Wang. 2005. Chinese word sense disambiguation using HowNet. In *Proceedings of International Conference on Natural Computation*.

Huaping Zhong, Jianwen Zhang, Zhen Wang, Hai Wan, and Zheng Chen. 2015. Aligning knowledge and text embeddings by entity descriptions. In *Proceedings of EMNLP*.

Yan-Lan Zhu, Jin Min, Ya-qian Zhou, Xuan-jing Huang, and Li-De Wu. 2006. Semantic orientation computing based on HowNet. *Journal of Chinese information processing*, 20(1):14–20.

A Building Process of a French SKB

In this section, we describe the building process of a dictionary-based French SKB that we call FDSKB. We choose Le Petit Robert French Dictionary 2016 edition (Robert et al., 2015), a very popular French dictionary, as the base dictionary.

Constructing the Sememe Set We first construct a sememe set from the defining vocabulary of the dictionary. Similar to EDSKB, we remove the most frequent and infrequent words that appear in definitions as well as some stop words, and obtain a sememe set comprising 2,919 sememes (defining words).

Extracting Sememes from Definitions We use Stanza (Qi et al., 2020c) to tokenize and lemmatize the definitions of all words in the dictionary and extract the sememes of each sense of each word according to the sememe set. So far, we have obtained the full version of FDSKB, whose statistics are shown in Table 7.

Distilling Sememes of Senses We adopt a similar way to EDSKB to distill the sememes of senses. Specifically, we use Stanza to conduct dependency parsing for every definition and obtain the importance score of each sememe. Then we empirically remove the unimportant sememes according to the experimental results of downstream tasks. In this way, we obtain the distilled version of FDSKB (FDSKB*), whose statistics are also in Table 7.

B Evaluation of FDSKB

In this section, similar to EDSKB, we conduct both intrinsic and extrinsic evaluations for FDSKB and FDSKB*. Notice that since HowNet covers only English and Chinese, there are no available HowNet-based baseline methods for French.

SKB	#Word/Phrase	#Sense	#Sememe	#AvgSem
FDSKB	55,836	113,722	2,919	4.32
FDSKB*	55,836	113,722	2,919	1.97

Table 7: Statistics of FDSKB and its distilled version FDSKB*. #AvgSem denotes the average sememe number per sense.

Dataset	French News		FR-Wikipedia	
Model	Valid	Test	Valid	Test
Tied LSTM	17.02	18.35	17.23	15.75
+FDSKB	<u>15.72</u>	<u>16.85</u>	17.14	15.60
+FDSKB*	<u>15.70</u>	<u>16.89</u>	**16.50**	**15.15**
AWD-LSTM	18.41	19.71	16.76	15.30
+FDSKB	15.41	16.47	**15.45**	15.72
+FDSKB*	**14.03**	**15.14**	15.90	**14.10**

Table 8: Perplexity results on the validation and test sets of French News and FR-Wikipedia. The **boldfaced** results show statistically significant improvement over the best results from baselines with $p<0.1$ given by t-test, and the <u>underlined</u> results represent having no significant difference.[7]

B.1 Intrinsic Evaluation

We still use CCSA (Liu et al., 2020) to measure the sememe annotation consistency. The MAP and F1 score for FDSKB are 83.47 and 80.51 respectively, and those for FDSKB* are 90.03 and 90.01 respectively. These results are comparable to those of EDSKB and can prove good sememe annotation consistency of FDSKB. Besides, similar to EDSKB, the distilled version FDSKB* delivers better sememe annotation consistency than FDSKB because it has fewer sememes per sense.

B.2 Extrinsic Evaluation

We conduct extrinsic evaluation for FDSKB and FDSKB* on three tasks including language modeling, natural language inference (NLI) and text classification.

Language Modeling

Similar to EDSKB, we use SDLM (Gu et al., 2018) to incorporate FDSKB into Tied LSTM (Zaremba et al., 2014) and AWD-LSTM (Merity et al., 2018). The experimental settings are the same as those in English experiments.

We choose two evaluation datasets: (1) **French News**[8], which comprises French news articles from

[8] https://webhose.io/free-datasets/french-news-articles/

Dataset	Method	LSTM	GRU	BiLSTM	BiGRU
XNLI	vanilla	61.14	60.88	61.56	61.36
	+Pseudo	60.96	61.46	61.10	61.12
	+Definition	61.64	61.10	61.80	61.86
	+FDSKB	62.38	61.54	**62.52**	61.44
	+FDSKB*	**62.52**	**61.74**	62.16	**62.06**
MARC	vanilla	79.98	79.16	80.35	80.64
	+Pseudo	80.35	79.38	79.16	80.53
	+Definition	81.10	79.39	80.65	80.80
	+FDSKB	81.39	80.58	80.65	**82.14**
	+FDSKB*	**81.43**	**81.10**	**80.98**	81.84

Table 9: Accuracy results of different models on the test sets of XNLI and MARC.

popular news sites. It has $2,131,774$ / $358,972$ / $370,059$ tokens in its training / validation / test sets. (2) **FR-Wikipedia**[9], which is composed of French Wikipedia articles. The token numbers in its training / validation / test sets are $3,252,094$ / $520,333$ / $517,669$.

The experimental results are given in Table 8. We observe that both FDSKB and FDSKB* bring decreases of perplexity, which demonstrates the effectiveness of the French dictionary-based SKB in language modeling and the practicality of our dictionary-based SKB building method. Notice that since the perplexity results of original Tied LSTM and AWD-LSTM on the two French datasets are quite good, the enhancement brought by FDSKB is comparatively less than that in English.

NLI and Text Classification

Similar to EDSKB, we use SememeCell (Qin et al., 2020) to incorporate FDSKB into RNNs and measure the improvement of sequence modeling ability on the tasks of NLI and text classification.

The base models are still LSTM, GRU, BiLSTM and BiGRU. And the baseline methods are also +Pseudo and +Definition. Here we use FlauBERT (Le et al., 2020), a French pre-trained language model, to encode definitions into 768-dimensional vectors. The other experimental settings are the same as English.

As for evaluation datasets, we use XNLI (Conneau et al., 2018) and MARC (Keung et al., 2020) respectively. XNLI is a cross-lingual NLI dataset in 15 languages. It is based on another NLI dataset MNLI (Williams et al., 2018) and constructs its training set by machine translation, which has $361,469$ sentence pairs. It has $2,500$ and $5,000$ sentence pairs in the validation and test sets which are manually translated from English. MARC (Multilingual Amazon Reviews Corpus) is a large corpus of Amazon reviews in 6 languages. We use its French part for product category classification. It has $40,000$ / $1,323$ / $1,345$ reviews in its training / validation / test sets.

Table 9 shows the accuracy results on the test sets of XNLI and MARC. The results are basically consistent with the experimental results in English datasets. The incorporation of the dictionary-based SKB can improve the performance of RNN models on the two different tasks, which reflects that the SKB has enhanced the sequence modeling ability of RNNs. Moreover, the results also demonstrate the usefulness and effectiveness of our dictionary-based SKB and its building method.

C Experiment Running Environment

For all the experiments, we use a server whose major configurations are as follows: (1) CPU: Intel(R) Xeon(R) E5-2680 v4 @ 2.40GHz, 56 cores; (2) RAM: 125GB; (3) GPU: 8 Nvidia RTX2080 GPUs, 12GB memory. The operation system is Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-108-generic x86_64). We use PyTorch[10] v1.5.0 and Python v3.6.9 as the programming framework for the experiments on neural network models.

[9] http://redac.univ-tlse2.fr/corpora/wikipedia_en.html

[10] https://pytorch.org/