Glyph Enhanced Chinese Character Pre-Training for Lexical Sememe Prediction

Boer Lyu, Lu Chen; Kai Yu∗
X-LANCE Lab, Department of Computer Science and Engineering
MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
Shanghai Jiao Tong University, Shanghai, China
State Key Lab of Media Convergence Production Technology and Systems, Beijing, China
{boerlv, chenlusz, kai.yu}@sjtu.edu.cn

Abstract

Sememes are defined as the atomic units to describe the semantic meaning of concepts. Due to the difficulty of manually annotating sememes and the inconsistency of annotations between experts, the lexical sememe prediction task has been proposed. However, previous methods heavily rely on word or character embeddings, and ignore the fine-grained information. In this paper, we propose a novel pre-training method which is designed to better incorporate the internal information of Chinese character. The Glyph enhanced Chinese Character representation (GCC) is used to assist sememe prediction. We experiment and evaluate our model on HowNet, which is a famous sememe knowledge base. The experimental results show that our method outperforms existing non-external information models.

1 Introduction

In linguistics, sememes are defined as the minimum semantic units for human language (Bloomfield, 1926), which describe the semantic meaning of concepts. HowNet (Dong and Dong, 2003) is one of the most well-known sememe knowledge bases (KB), which has been widely used in many NLP tasks (Qi et al., 2021), such as semantic similarity computation (Liu, 2002), sentiment analysis (Fu et al., 2013; Huang et al., 2014), language modeling (Gu et al., 2018), word representation learning (Niu et al., 2017) and short text matching (Lyu et al., 2021).

In order to free human experts from the laborious sememe annotating job, Xie et al. (2017) propose the task of sememe prediction, which intends to automatically select related sememes from a closed sememe set for each word. They propose two frameworks based on word embedding and matrix factorization. But these methods usually fail to deal with the prediction problem of low-frequency words.

Motivated by this, Jin et al. (2018) present character-enhanced sememe prediction (CSP), taking advantage of both internal character information and external context information of words. However, CSP is an ensemble model which still relies on word and character representation, and ignores the fine-grained information.

For internal structural information of words, many researchers believe that only using characters is not sufficient for capturing the semantic information (Yu et al., 2017; Cao et al., 2018; Sun et al., 2019; Meng et al., 2019). For instance, the words “森林(forest)” and “木头(wood)” are semantically related. But these two words share no information since they consist of different characters. To address this problem, we split each Chinese character into several components, and regard component as the minimum unit to express the meaning of the character. We believe that fine-grained units can share more information between semantically related words, which helps model prediction. Take Figure 1 for example, the characters of word “濒海(near the sea)” have components “步(step)” and “氵(water)”, which are related to its sememes, namely “靠近(BeNear)” and “水域(waters)”, respectively.

In order to better incorporate the internal information of Chinese character, we pre-train a Glyph
enhanced Chinese Character embedding (GCC) for sememe prediction task. More specifically, we use the same model structure as BERT (Kenton and Toutanova, 2019), but change the input unit and the masking scheme. First, we regard Chinese words as our training samples and take components of each character in the word to form the input sequence. Second, we mask random tokens and predict the modified tokens as well as all characters in the sample.

We evaluate our model on HowNet sememe KB. Experimental results demonstrate that our model outperforms the baseline model. In summary, our contributions include:

• To the best of our knowledge, we are the first to use masked language model (MLM) objective to force the model to learn the internal information of characters.

• We propose a novel sememe prediction framework considering both internal and contextual character information.

• Our method is particularly useful for low-frequency words and shows the effectiveness and robustness on the dataset.

2 Methodology

In this section, we first introduce the architecture of pre-training model. Then, we describe how to incorporate pre-trained representation into sememe prediction task.

2.1 Pre-Training Model Architecture

As shown in Figure 2, the framework of our pre-training model includes an embedding layer and a masked transformer encoder layer.

First, we use the file¹ about structures of Han Ideographs and refer to Ke and Hagiwara² to get all the Chinese character trees. Then, we use the depth-first algorithm to convert each character tree into the format of a sequence (Nguyen et al., 2019). Note that, there are two types of tokens in the input sequence. As shown in the left block in Figure 2, the leaf nodes (position 2, 5, 6, 7) are components of Chinese character, and the inner nodes (position 1, 3, 4) are structural composition operators (such as vertical stacking) applied to children nodes. The character "near" can be serialized as \{char, x_1, x_2, x_3, x_4, x_5, x_6, x_7\} \}, where C is the set of components, T is the structural composition operator set.

2.1.1 Embedding Layer

The input embedding of the model is the sum of token embedding, type embedding, position embedding and character segmentation embedding.

For token embedding, we maintain two lookup tables (Sun et al., 2020) and use [CHAR] as the character tag which represents the entire character information, [S_M] to mask the structure type token and [C_M] to mask the component token. To distinguish them, we simply use type embedding to indicate the token types, i.e. CHAR for character tag, STC for structure type token and CPN for component token. As for position embedding, we assign a number starting from 0 to each token belonging to the same character. Finally, our model use segmentation embedding to identify different characters. For instance, the input sequence in Fig-

¹https://github.com/tomcumming/chinese-ids
²https://github.com/yuanzhiKe/Radical_CR_Encoder
ure 2 is marked with a sequence of segment tags, i.e. \{A, ..., A, B, ..., B\}. All the embeddings have the same dimension \(d\).

### 2.1.2 Masked Transformer Encoder

We use the multi-head self-attention network as the basic structure. Given the representation of sequence tokens \(X \in \mathbb{R}^{n \times d}\), where \(n\) is the number of tokens in the sequence and \(d\) is the dimension of each token. The process of masked self-attention can be formulated by

\[
A = \frac{(XW^Q)(XW^K)^T}{ \sqrt{d_k}},
\]

\[
\tilde{X} = \max (A + M)(XW^V),
\]

where \(W^Q, W^K, W^V \in \mathbb{R}^{d \times d_k}\) are learnable parameters, and \(M \in \mathbb{R}^{n \times n}\) is the attention mask matrix (Liu et al., 2020). We obtain \(M\) by setting \(M_{ij}\) to 0 when \(x_j\) is visible to \(x_i\), while setting \(M_{ij}\) to \(-\infty\) when \(x_j\) is invisible to \(x_i\). More specifically, all tokens belonging to the same character are visible to each other; and the special tags [CHAR] are also visible to each other. Thus, the output representation of [CHAR] not only contains internal component information of the character itself, but also other character information in the word.

### 2.1.3 Pre-Training Objective

MLM objective is used in our model. Generally, we mask 15% of the input sequence at random; of those, 80% are replaced by their mask token ([C_M] for component tokens, [S_M] for structure type tokens), 10% are replaced by a random token which belongs to the same token type, and 10% are kept unchanged. We train a model to predict the original tokens from the modified input. Masking component tokens helps model to learn the fine-grained information from the contextual component sequence. Masking structure type tokens helps model to learn the structural information of components.

We also predict the character of tag [CHAR]. This objective forces model to gather all useful multi-granularity information to the token [CHAR]. The advantage is that we can easily use the hidden output of [CHAR] as the character representation \(u\) for downstream tasks, such as sememe prediction task.

### 2.2 Sememe Prediction Model

Given a word \(w \in \mathcal{W}\), the goal of our model is to predict the corresponding \(P(s|w)\) for each sememe \(s \in \mathcal{S}\), where \(\mathcal{W}\) is the word set and \(\mathcal{S}\) is the set of sememes existing in HowNet. Then, we recommend sememes with high scores to \(w\).

Our sememe prediction model GCC (Figure 3) has two parts, one is an encoder which encodes the word-related information into a vector and the other is a multi-label classifier, which uses the vector to compute scores for each sememe. We use Bidirectional LSTM (Bi-LSTM) (Schuster and Paliwal, 1997) as the encoder. For each word \(w\), we concatenate the word and the characters \(c_i\) in the word as \(\{w, c_1, ..., c_n\}\), and then convert it to \(\{w, c_1, ..., c_n\}\) with the embedding trained on SogouT corpus\(^3\) using Skipgram (Mikolov et al., 2013). We incorporate our pre-trained character embedding by addition operation:

\[
\hat{c}_i = c_i + W^U u_i,
\]

where \(W^U\) is a projection matrix and \(u_i\) is the character representation mentioned in Section 2.1.3.

Then, we pass it to Bi-LSTM. The concatenation of the last hidden states in both directions, denoted as \(h\), is fed to the multi-label classifier:

\[
h = \text{Bi-LSTM}(w, c_1, ..., c_n),
\]

\[
x = Wh + b,
\]

where \(W \in \mathbb{R}^{|\mathcal{S}| \times 2l}, \ x, b \in \mathbb{R}^{|\mathcal{S}|}, l\) represents the dimension of hidden states in a single direction. Each element of \(x\) is a score related to the sememe in \(\mathcal{S}\). For training, we use the multi-label one-versus-all cross-entropy loss, where \(\sigma\) is a sigmoid function and \(y_j \in \{0, 1\}\) represents whether the \(j\)-th sememe is in the sememe set of word \(w\):

\[
L = -\frac{1}{|\mathcal{S}|} \sum_{j=1}^{|\mathcal{S}|} y_j \sigma (x_j) + (1 - y_j) \sigma (-x_j).
\]

\(^3\)https://www.sogou.com/labs/resource/t.php
3 Experiments

3.1 Experimental Setup

Pre-Training Data We adopt Tencent embedding corpus (Song et al., 2018) which covers over 8 million Chinese words and phrases. We remove non-Chinese characters such as punctuation and pure digits, and finally get 7,291,828 words as our pre-training samples.

Sememe Prediction Dataset To make the results comparable, we follow Du et al. who proposed the previous state-of-the-art model. This dataset is constructed from HowNet sememe KB, where they disregard the hierarchical structures of sememes and filter out the low-frequency sememes which appear less than 5 times in HowNet. The final number of sememes we use is 1,400. The total number of words in the dataset is 48,383, which are divided into non-overlapping training, validation, and test sets in the ratio of 8:1:1.

Hyper-parameters Both pre-training and the sememe prediction models are trained by Adam with a learning rate of 0.0001 (Kingma and Ba, 2014). For pre-training, we use the structure of BERT-base and the batch size is 1024. For sememe prediction, the dimension of word embedding is 200, the dimension of Bi-LSTM hidden states is $512 \times 2$, and the batch size is 128. Our code is available at https://github.com/lbe0613/GCC.

3.2 Evaluation Metrics

Following Xie et al., we use mean average precision (MAP) as evaluation metrics. We rank all sememes according to the model output. For a word with $K$ sememes, we get MAP by

$$\text{MAP} = \sum_{k=1}^{K} \frac{k}{r_k},$$

where the rankings of the $K$ sememes are $r_1 \leq r_2 \leq \ldots \leq r_K$.

3.3 Results

In Table 1, we report average results of 5 runs to ensure the reliability of results.

<table>
<thead>
<tr>
<th>Models</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPWE (Xie et al., 2017)</td>
<td>55.04</td>
</tr>
<tr>
<td>CSP (Jin et al., 2018)</td>
<td>58.93</td>
</tr>
<tr>
<td>LD+Seq2Seq$^\dagger$ (Li et al., 2018)</td>
<td>30.49</td>
</tr>
<tr>
<td>MC$^\dagger$ (Du et al., 2020)</td>
<td>60.55</td>
</tr>
<tr>
<td>SCorP$^\dagger$ (Du et al., 2020)</td>
<td>64.65</td>
</tr>
<tr>
<td>GCC w/o pre-train (Ours)</td>
<td>58.18</td>
</tr>
<tr>
<td>GCC$\bullet$ (Ours)</td>
<td>60.23</td>
</tr>
<tr>
<td>JWE$\bullet$ (Yu et al., 2017)</td>
<td>59.03</td>
</tr>
<tr>
<td>Glyce$\bullet$ (Meng et al., 2019)</td>
<td>59.10</td>
</tr>
</tbody>
</table>

Table 1: Sememe prediction results of all models. The second part models with $^\dagger$ utilize external dictionary definition information, and the third part models with $\bullet$ consider glyph information.

models include LD+Seq2Seq, MC and SCorP. Our GCC models belong to representation-based models. We also compare GCC with other models utilizing glyph information. Here, we simply replace our GCC embedding in Figure 3 with character embedding in JWE and Glyce.

As shown in Table 1, the models considering glyph information perform better than all traditional representation-based models, which demonstrates that glyph can enhance Chinese character embedding for sememe prediction task. Especially, GCC has an absolute improvement of 2.05% compared to GCC baseline without pre-training and significantly outperforms JWE and Glyce. The reason is that firstly Chinese characters are pictographic characters, and glyphs express the meaning of the word to a certain extent, which is related to the sememes of the word. Secondly, pre-training enables GCC to better integrate fine-grained information into Chinese character representation.

In addition, since experts refer to dictionary definitions when annotating sememes (Dong and Dong, 2003), it is very powerful semantic information for sememe prediction. Even though, our model is still comparable to MC and even better than LD+Seq2Seq when only using the information in words.

3.4 Influence of Word Frequency

Figure 4 shows the evaluation results of different frequencies on four strong models. We can see that GCC is superior to other models in all word frequency ranges. In addition, word frequency has great impacts on sememe prediction. Since low-frequency words are usually unrelated to each other and contain fewer and simpler sememes, the per-
formance of the model is drastically reduced when facing low-frequency words. However, our model GCC is particularly helpful in improving the performance of them. When the word frequency is less than 50, the MAP increases by 3.31\% after utilizing glyph enhanced character embedding. Compared with other models using glyph information (JWE and Glyce), it has an increase of at least 2.3\%, which is greater than that of all other word frequency ranges.

### 3.5 Case Study

![Example of 牛腩 "tenderloin"](image)

<table>
<thead>
<tr>
<th>Model</th>
<th>Sememe/rank</th>
<th>Average Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>GCC w/o pre-train</td>
<td>食品 food/0, 養殖 livestock/4, 部件 part/11, 肉 flesh/27</td>
<td>44.82</td>
</tr>
<tr>
<td>GCC</td>
<td>食品 food/0, 部件 part/2, 肉 flesh/6, 养生 livestock/7</td>
<td>64.88</td>
</tr>
</tbody>
</table>

![Example of 葬身 "be buried"](image)

<table>
<thead>
<tr>
<th>Model</th>
<th>Sememe/rank</th>
<th>Average Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>GCC w/o pre-train</td>
<td>死 die/0, 人 human/6, 埋入 bury/62</td>
<td>44.44</td>
</tr>
<tr>
<td>GCC</td>
<td>死 die/0, 人 human/6, 埋入 bury/6</td>
<td>80.95</td>
</tr>
</tbody>
</table>

The examples in Figure 5 show how glyph information assist sememe prediction. We present the sememe labels with their corresponding ranks, and average precision score of each model. Average precision refers to the accuracy of a single sample. The model recommends low-rank sememes to words. In Figure (a), the meaning of component “月(moon)” in Chinese is related to “肉(flesh)”. Thus, the rank of sememe flesh is raised from 27 to 6 when incorporating glyph information. And the average precision score increases from 44.82 to 64.88.

In Figure (b), the component “艹(grass)” is the same as grass, which is related to bury, because objects can be buried by grass. And the sememe die is also the component of the character “葬(burial)”, which demonstrates the glyphs are related to the semantics of the word. The result is also convincing. The rank of sememe bury is raised from 62 to 6 while the average precision score increases from 44.44 to 80.95.

### 4 Conclusion

In this work, we pre-train a Glyph enhanced Chinese Character embedding (GCC) for sememe prediction. The model is evaluated on HowNet sememe KB and outperforms existing non-external information models. Our experiments show that glyph information can enhance the semantic expression of words, and has a better performance on low-frequency words.

### Acknowledgments

We thank the anonymous reviewers for their thoughtful comments. This work has been supported by the China NSFC Projects (No. 62120106006 and No. 62106142), CCF-Tencent Open Fund and Startup Fund for Youngman Research at SJTU (SFYR at SJTU).
References


