Augmentation of Chinese Character Representations with Compositional Graph Learning (Student Abstract)

Jason Wang,[1] Kaiqun Fu,[2] Zhiqian Chen,[3] Chang-Tien Lu[4]

[1] Harvard University
[2] South Dakota State University
[3] Mississippi State University
[4] Virginia Tech
jasonwang1@college.harvard.edu, kaiqun.fu@sdstate.edu, zchen@cse.msstate.edu, ctlu@vt.edu

Abstract

Chinese characters have semantic-rich compositional information in radical form. While almost all previous research has applied CNNs to extract this compositional information, our work utilizes deep graph learning on a compact, graph-based representation of Chinese characters. This allows us to exploit temporal information within the strict stroke order used in writing characters. Our results show that our stroke-based model has potential for helping large-scale language models on some Chinese natural language understanding tasks. In particular, we demonstrate that our graph model produces more interpretable embeddings shown through word subtraction analogies and character embedding visualizations.

Introduction

Chinese characters are logographic, meaning that a word is represented by a single symbol or character that has evolved over time from a pictorial representation. There are eight basic types of calligraphic strokes that compose Chinese characters. These strokes combine to form radicals—similar to how roots function in many other languages—and provide either semantic or phonological meaning to the character. Previous studies have attempted to extract information with CNNs, but have found that the introduction of such compositional information via CNNs is mostly ignored (Dai and Cai 2017) and provides minor performance boosts overall (Meng et al. 2019). We hypothesize that because Chinese characters are composed of a very limited set of geometric strokes, if a character can be represented as a graph of strokes, we can condense the data to fewer features. Furthermore, the strictly prescribed stroke order of Chinese characters provides unrealized temporal information which may aid the segmentation of semantic-rich radicals.

Our research problem is thus stated: can we use graph representation learning methods on graphs of Chinese characters to create useful character embeddings for NLP tasks? The main contributions of our paper are: 1) designing a novel graph structure that provides new insights to Chinese character composition, 2) implementing embeddings that can build on top of existing language models, and 3) demonstrating that our graph embeddings are more interpretable than prior image-based embeddings.

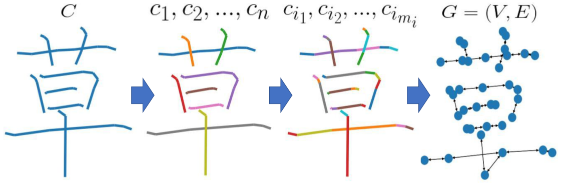

Figure 1: Character Graph Construction

Problem Statement and Methods

Character Graph Construction Let a Chinese character C be composed of ordered strokes $c_1, c_2, ..., c_n$, each of which is further composed of one or more ordered line segments $c_{i_1}, c_{i_2}, ..., c_{i_{m_i}} \forall i \in \mathbb{N}, i \leq n$, where n denotes the number of strokes in C and m_i denotes the number of line segments in c_i. We construct graph $G = (V, E)$ from C where V is a set of length $\sum_{i=1}^{n} m_i$ containing nodes v_{i_j} representing each line segment c_{i_j}, described by a vector $(x_{mid}, y_{mid}, l, \theta, i, j)$, where the first four features denote spatial information and the last two features denote temporal information. E then represents the bidirectional edges of G connecting all v_{i_j} in which the line c_{i_j} it represents intersects another line. Figure 1 illustrates the constructed character graph.

Graph Representation Learning Given the constructed character graph, the task is to create a representative embedding g for each Chinese character graph to aid the learning of the function $F(x^*, g^*) = y$ where x^* is the list of character tokens, g^* is the list of corresponding graph embeddings, and y is the task-specific output (i.e., character classification, sentence classification, or sentence pair classification).

We approach the creation of g by first pre-training node embeddings $z_v \forall v \in V$ through an unsupervised run of the GraphSAGE algorithm over all Chinese character graphs. We calculate $z_v = h_v^D$, where at a particular depth d, $h_v^d = \sigma(W^d \cdot \mathbf{C}(h_v^{d-1}, \mathbf{A}_d(h_u^{d-1} \forall u \in \mathbf{N}(v))), \forall v \in V$, in which D is the number of GraphSAGE layers, W is the weight matrix, $\mathbf{C}$ is concatenation, $\mathbf{A}$ is a single dense layer followed by a max pool, and $\mathbf{N}$ are the neighboring nodes. The unsupervised loss for any given node z_u is defined as $J_{G(z_u)} = -\log(\sigma(z_u^\top z_v)) - Q \cdot \mathbb{E}_{v_n \sim P_n(v)} \log(\sigma(-z_u^\top z_{v_n}))$, in which v is a co-occuring node with u from a random walk, Q is the number of negative samples, and P_n is a negative sampling distribution (Hamilton, Ying, and Leskovec 2017).

	BERT		Glyce		Graph		Combined	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc
NER	95.67	98.02	95.63	97.78	95.31	98.15	**95.81**	**98.41**
POS	**96.33**	97.04	96.19	97.03	96.12	97.05	96.31	**97.18**
CWS	96.63	96.57	**96.75**	**97.17**	96.63	96.86	96.67	96.74
Sent Pair	84.22	84.22	84.39	84.39	**84.80**	**84.80**	84.34	84.34
Sentiment	99.46	99.46	99.35	99.35	**99.51**	**99.51**	99.40	99.40

Table 1: Character Embedding Performance on 5 NLP Tasks

Root	Subtraction Pairs	Glyph	Graph
阝	阴–月, 阳–日 阴–月, 阳–房	0.06	**0.28**
艹	草–早, 芋–于 草–早, 芋–房	0.06	**0.39**
辶	送–关, 通–甬 送–关, 通–房	0.03	**0.18**

Table 2: Word Subtraction Analogy Strength[1]

Experiment

Stroke and Task Data Information. To create our stroke-order dataset, we extracted the strokes of 9,574 Chinese characters in regular script font from *hanzi-writer*[2], which we have made publicly available with our experiment code[3]. We evaluated our novel stroke order character embeddings on the **Resume** dataset (Zhang and Yang 2018) for NER, Chinese Treebank 5.0 **(CTB5)** (Palmer et al. 2005) for POS Tagging, **PKU** dataset for Chinese Word Segmentation, **BQ** corpus (Chen et al. 2018) for Sentence Pair Classification, and **Fudan** corpus (Li 2011) for Sentiment Analysis.

Comparison Method. We compared our stroke-based character embeddings with previous SOTA Glyce character embeddings (Meng et al. 2019), which boost task performance through multiple historical fonts. We tested the following four character embedding strategies: BERT, BERT+Glyce, BERT+Graph, BERT+Glyce+Graph.

Results. The graph model produces the best accuracies and the combined model produces the best F1 scores. The best F1 increase over BERT was 0.58% on BQ with our graph model. However, most other margins between the models are within a few tenths of a percent (Table 1).

Case Study. We quantified the semantic strength of a radical through word subtraction analogies, which subtract the phonological radical from a character ("radical arithmetic"). Strong compositional embeddings will preserve semantic strength such that the cosine similarity between subtracted pairs of the same radical will be higher than subtracted pairs of dissimilar radicals. We found that graph embeddings obey radical arithmetic while glyph embeddings do not (Table 2).

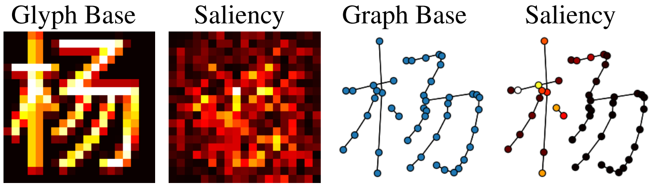

Figure 2: Saliency Map and Saliency Graph

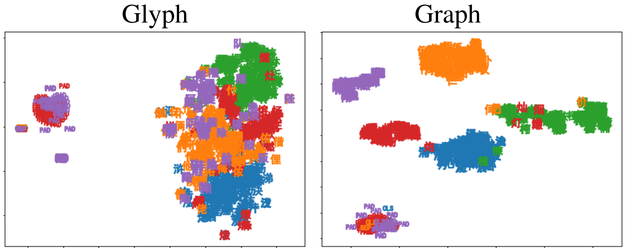

Figure 3: t-SNE Plots of 5 Radical Groups

We visualized the embeddings with dimensionality reduction using PCA and t-SNE (Figure 3), which show that clusters of radical groups in graph embeddings are farther apart (i.e., more distinct) than in glyph embeddings. We also generated saliency maps depicting the gradient (i.e., importance) of a pixel or node with respect to the downstream task (Figure 2). Glyph embeddings have noisy, uninterpretable gradients, but graph embeddings have consistently interpretable gradients corresponding to the character's semantic radical.

Conclusion

We developed a novel graph representation for Chinese characters and curated a publicly available dataset of 9,574 Chinese characters in our graph form. We then used GraphSAGE for graph representation learning, and evaluated our proposed graph model against other compositional models. Our results show that image-based and graph-based compositional models do not provide significant gains in five NLP tasks. However, we show that graph embeddings are more interpretable than image embeddings.

References

Chen, J.; Chen, Q.; Liu, X.; Yang, H.; Lu, D.; and Tang, B. 2018. The BQ Corpus: A Large-scale Domain-specific Chinese Corpus For Sentence Semantic Equivalence Identification. In *EMNLP*, 4946–4951.

Dai, F. Z.; and Cai, Z. 2017. Glyph-aware Embedding of Chinese Characters. In *SWCN@EMNLP*, 64.

Hamilton, W. L.; Ying, R.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *NIPS*, 1025–1035.

Li, R. 2011. Fudan Corpus for Text Classification. NLP Group, Fudan University.

Meng, Y.; Wu, W.; Wang, F.; Li, X.; Nie, P.; Yin, F.; Li, M.; Han, Q.; Sun, X.; and Li, J. 2019. Glyce: Glyph-vectors for Chinese Character Representations. In *NIPS*, 2746–2757.

Palmer, M.; Chiou, F.-D.; Xue, N.; and Lee, T.-K. 2005. Chinese Treebank 5.0. Philadelphia: Linguistic Data Consortium.

Zhang, Y.; and Yang, J. 2018. Chinese NER Using Lattice LSTM. In *ACL*, 1554–1564.

[1] The data in the table is the absolute difference between the cosine similarity of the sensical first subtraction pair and the cosine similarity of the nonsensical second subtraction pair.

[2] https://github.com/chanind/hanzi-writer

[3] https://github.com/jsonW0/StrokeOrderEmbeddings