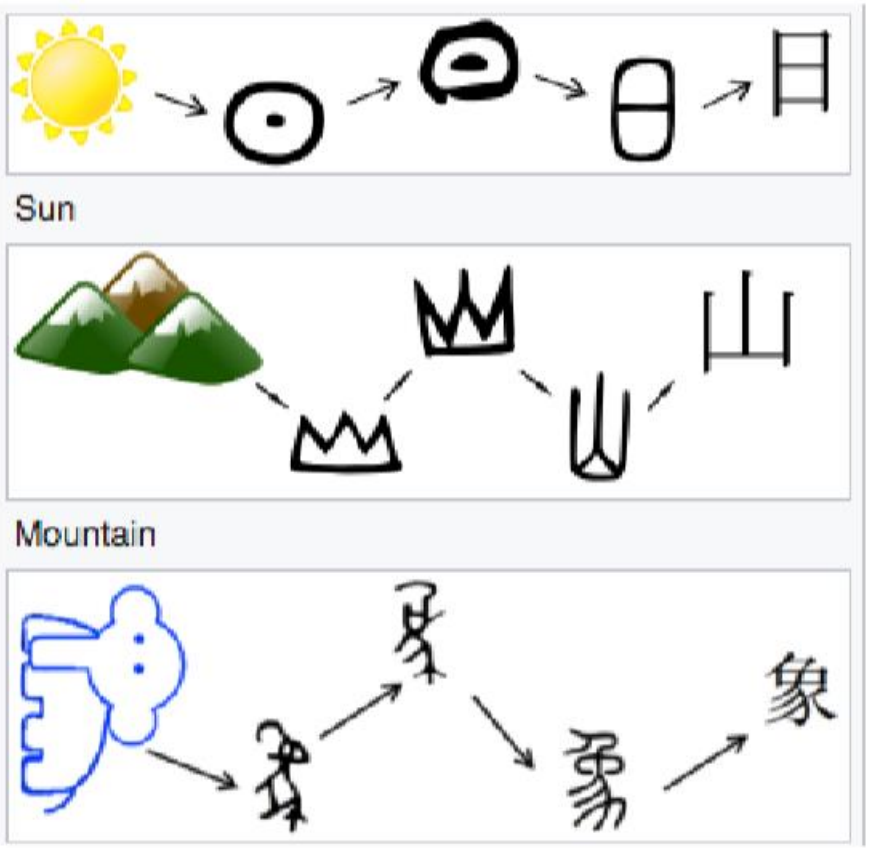
Glyce: Glyph-vectors for Chinese Character Representations

沈彧 1500012713

2018.2.28

Glyph



	oracle bone *jiaguwen*	greater seal *dazhuan*	lesser seal *xiaozhuan*	clerkly script *lishu*	standard script *kaishu*
rén (*nin) human					
nǚ (*nraʔ) woman					
ěr (*nəʔ) ear					
mǎ (*mrāʔ) horse					
yú (*ŋa) fish					
shān (*srān) mountain					

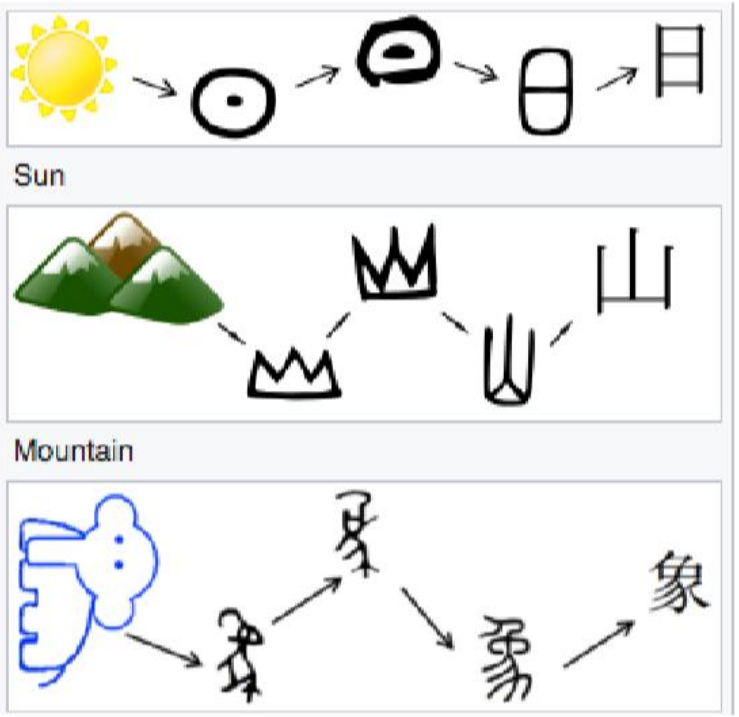
Related work

- Mainstream algorithms learn embedding representations on word/character level (word2vec…)

- Recent work on character logos:
  - Dai and Cai, 2017 —— Negative results
  - Tan et al 2018 —— Significant improvements only on word semantic evaluation
  - Zhang and LeCun 2017——Little improvements on text classification

Why fail?

- Not using the right versions of script
  - Simplified Chinese has lost pictographic Information
- Not using the proper CNN architecture
  - ImageNet 800*600, Chinese words 12*12
- No auxiliary function



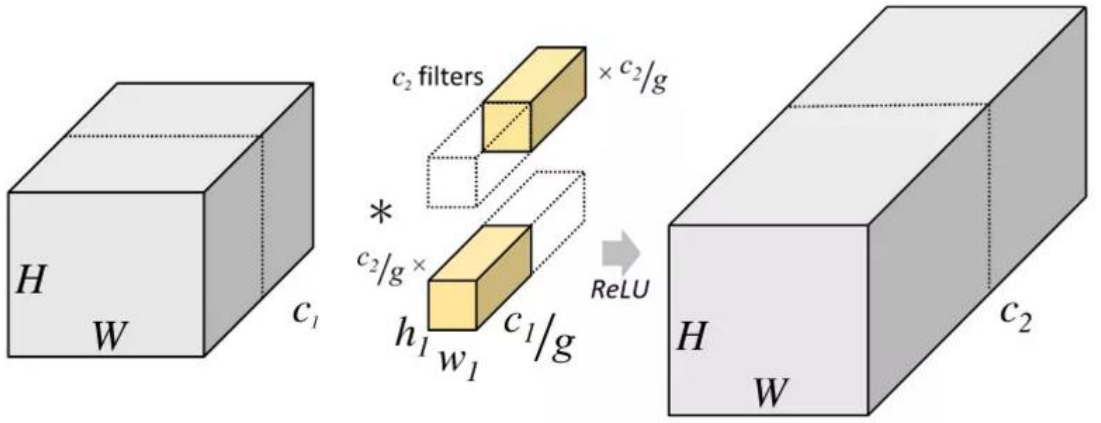
Dataset

Chinese	English	Time Period
金文	Bronzeware script	Shang dynasty and Zhou dynasty (2000 BC – 300 BC)
隶书	Clerical script	Han dynasty (200BC-200AD)
篆书	Seal script	Han dynasty and Wei-Jin period (100BC - 420 AD)
魏碑	Tablet script	Northern and Southern dynasties 420AD - 588AD
繁体中文	Traditional Chinese	600AD - 1950AD (mainland China). still currently used in HongKong and Taiwan
简体中文(宋体)	Simplified Chinese - Song	1950-now
简体中文(仿宋体)	Simplified Chinese - FangSong	1950-now
草书	Cursive script	Jin Dynasty to now

Table 1: Scripts and writing styles used in Glyce.

CNN Structure

- Input shape in the last layer = [2, 2, channel] (inspired by Tianzige)
- Using group convolution to prevent overfitting



CNN Structure

layer	kernel size	feature size
input		n× 12× 12
conv	5	1024×8×8
relu		1024×8×8
maxpool	4	1024× 2×2
8-group conv	1	256×2× 2
16-group conv	2	1024×1× 1

Table 2: The tianzige-CNN structure in Glyce.

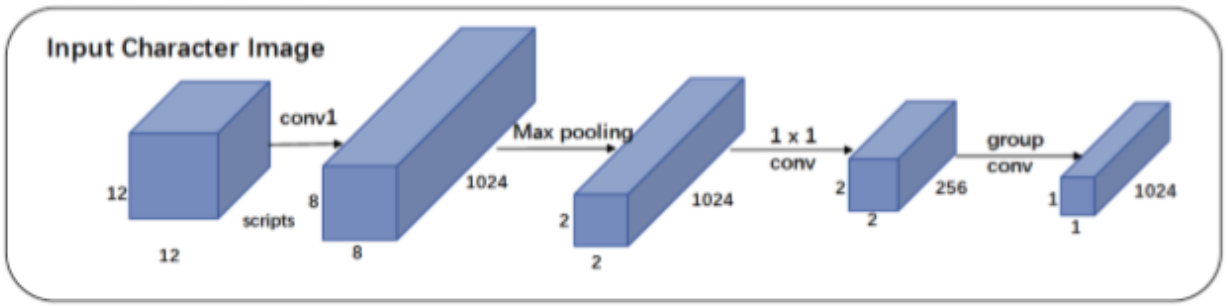


Figure 3: The CNN structure for Glyce.

Loss function

- Task loss+ Image classification loss

$$\begin{aligned}\mathcal{L}(\text{cls}) &= -\log p(z|x) \\ &= -\log \text{softmax}(W \times h_{\text{image}})\end{aligned} \quad (1)$$

$$\mathcal{L} = (1-\lambda(t))\,\mathcal{L}(\text{task}) + \lambda(t)\mathcal{L}(\text{cls}) \quad (2)$$

Glyce-char(word) Embedding

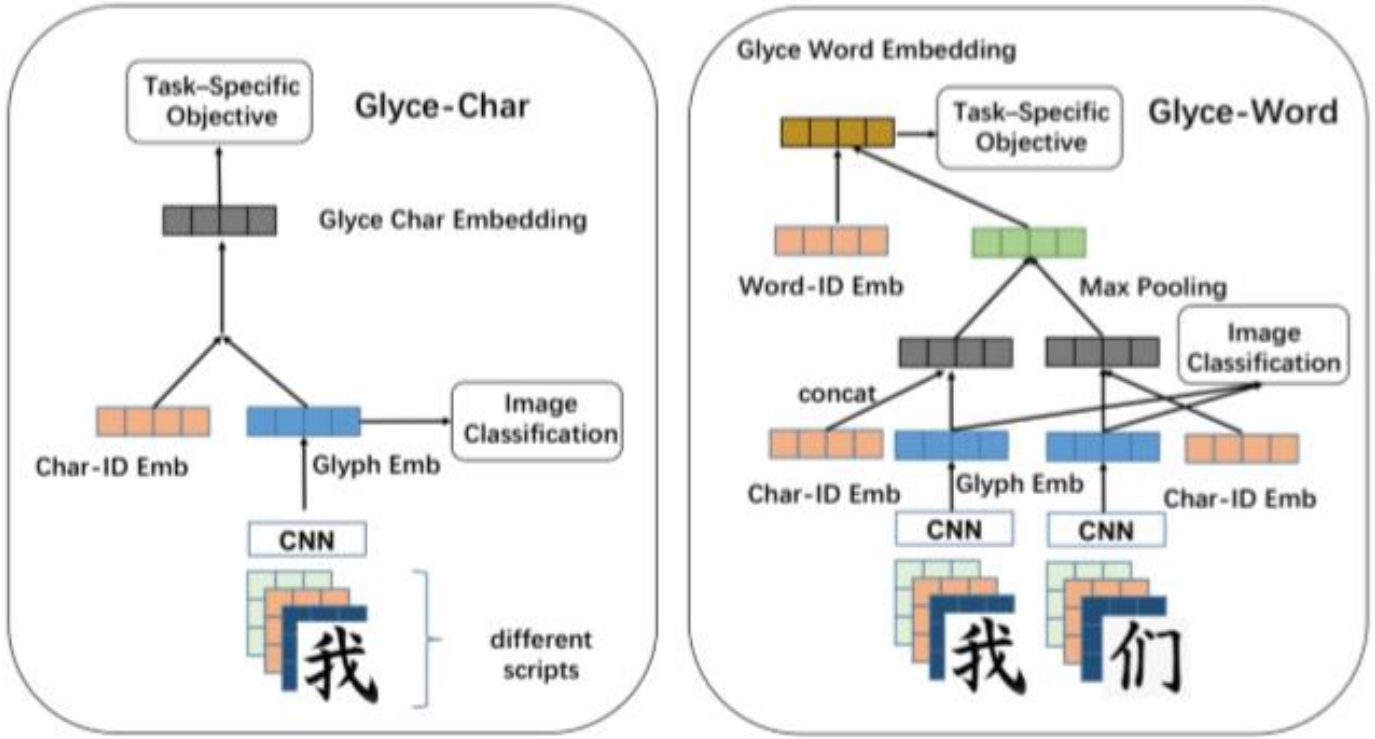


Figure 5: An overview of the Glyce character embedding and the Glyce word embedding.

Experiments

- Char(Word)-level language model with LSTMs
- Dataset: CTB6

d=2048		
model	ppl	# parameter
charID	52.86	51.6M

d=1024		
model	ppl	# para
charID	53.87	17.4M
glyph (1 script, no img cls)	53.70	13.9M
glyph (8 script, no img cls)	172.56	19.2M
glyph (1 script, with img cls)	53.40	18.3M
glyph (8 script, with img cls)	53.10	23.6M
charID+glyph (1 script, no img cls)	51.64	24.6M
charID+glyph (8 script, no img cls)	391.29	30.1M
charID+glyph (1 script, img cls)	51.38	29.0M
charID+glyph (8 script, img cls)	**50.67**	34.5M

d=512

model	ppl	# parameter
wordID	199.9	28.6M
charID	193.0	18.9M
glyph	181.0	19.2M
wordID+charID	188.4	32.4M
wordID+glyph	175.1	38.5M
wordID+charID+glyph	176.0	36.0M

Experiments

- Name Entity Recognition, Chinese Word Segmentation

OntoNotes			
Model	P	R	F
CRF-LSTM	74.36	69.43	71.81
Lattice-LSTM	76.35	71.56	**73.88**
Lattice-LSTM+Glyce	82.06	68.74	**74.81** (+0.93)
MSRA			
Dong et al. (2016)	91.28	90.62	90.95
CRF-LSTM	92.97	90.80	91.87
Lattice-LSTM	93.57	92.79	**93.18**
Lattice-LSTM+Glyce	93.86	93.92	**93.89** (+0.71)
resume			
CRF-LSTM	94.53	94.29	94.41
Lattice-LSTM	94.81	94.11	**94.46**
Lattice-LSTM+Glyce	95.72	95.63	**95.67** (+1.21)

Models	CTB6	PKU	Weibo
Zhou et al. (2017)	96.2	96.0	-
Yang et al. (2017)	96.2	**96.3**	**95.5**
Lattice+Word	**96.3**	95.9	95.1
Lattice+Subword	96.1	95.8	95.3
Lattice+Glyce-Char	**96.6** (+0.3)	**96.3** (+0.0)	**96.0** (+0.5)

Experiments

- Chinese-English Machine Translation
- Metric: BLEU

TestSet	Seq2Seq +Attn	Seq2Seq +Attn+BOW	Glyce+Seq2Seq +Attn+BOW
MT-02	34.71	39.77	**40.56** (+0.79)
MT-03	33.15	38.91	**39.93** (+1.02)
MT-04	35.26	40.02	**41.54** (+1.52)
MT-05	32.36	36.82	**38.01** (+1.19)
MT-06	32.45	35.93	**37.45** (+1.52)
MT-08	23.96	27.61	**29.07** (+1.46)
Average	31.96	36.51	**37.76** (+1.25)

Conclusion

- Contributions:
  - Historical Chinese scripts
  - Tianzige-CNN-architecture
  - Image classification as an auxiliary task objective

- It works well ☺

Q&A