Encoding Chinese, Japanese and Korean for Text Classification using Convolutional Networks

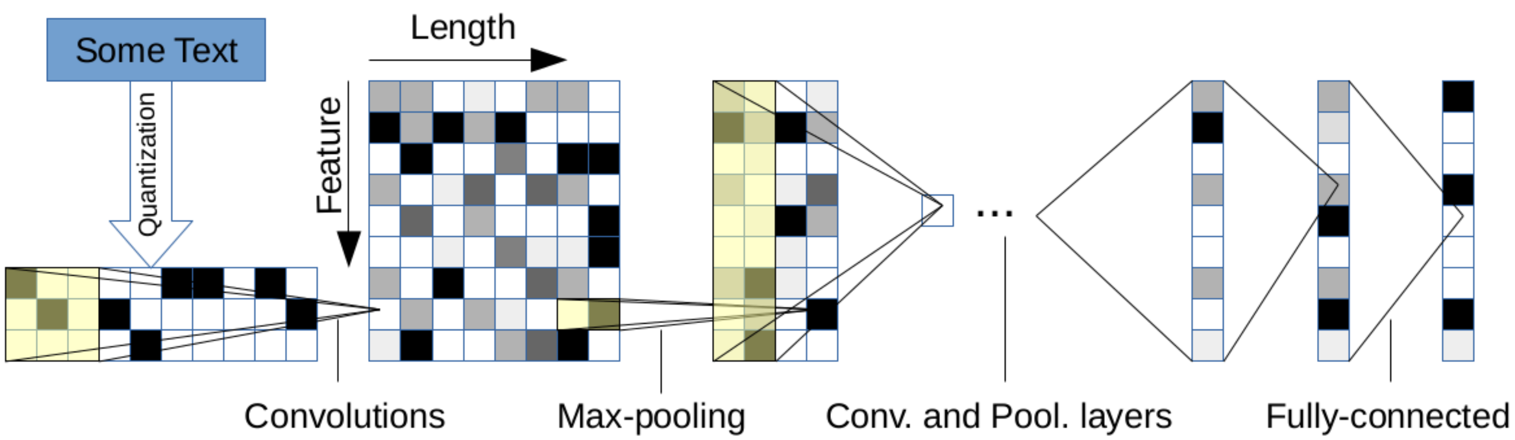
(Work in Progress)

Xiang Zhang, Yann LeCun
Courant Institute of Mathematical Sciences, New York University

Recap of Char-level ConvNets

NYU

- **Char-level ConvNets (Zhang et al 2015)**



- **Problem When Applied to Chinese, Japanese and Korean (CJK)**
  - Too many characters, explicit one-hot encoding not possible.

Solutions for Encoding CJK

- **Character Glpyhs (Choose a Font!)**
- **Byte-level Onehot Encoding**
  - Treat any text as a sequence of UTF-8 encoded bytes
- **Embedding**
  - Byte-level
  - Char-level
  - Word-level
  - Romanization Word-level
- **Also comparisons with linear models and fastText (Bojanowski et al. 2016)**
- **An alternative not yet considered: a convolutional layer that takes character indices as the input.**

We Need Data

NYU

- **I Crawled 14 Large-scale Text Classification Datasets (Seriously!)**

Dataset	Language	Type	# Classes	Training	Testing
Dianping	Chinese	Sentiment	2	2000000	500000
JD Full	Chinese	Sentiment	5	3000000	250000
JD Binary	Chinese	Sentiment	2	4000000	360000
Rakuten Full	Japanese	Sentiment	5	4000000	500000
Rakuten Binary	Japanese	Sentiment	2	3400000	400000
11st Full	Korean	Sentiment	5	750000	100000
11st Binary	Korean	Sentiment	2	4000000	400000
Amazon Full	English	Sentiment	5	3000000	650000
Amazon Binary	English	Sentiment	2	3600000	400000
Joint Full	Multilingual	Sentiment	5	10750000	1500000
Joint Binary	Multilingual	Sentiment	2	15000000	1560000
ifeng	Chinese	Topic	5	800000	50000
Chinanews	Chinese	Topic	7	1400000	112000
NYT	English	Topic	7	1400000	105000

First Results: Dianping (Chinese)

NYU

Model	Level of Encoding	Variant	Training Error	Testing Error
GlyphNet	Character	20 layers	24.03%	24.35%
OnehotNet	Byte	16 layers	22.42%	**23.17%**
	Romanization	16 layers	22.78%	23.53%
EmbedNet	Character	13 layers	22.97%	23.60%
	Byte	13 layers	23.33%	24.09%
	Romanization Byte	13 layers	24.66%	25.42%
	Word	13 layers	24.03%	24.55%
	Romanization Word	13 layers	23.07%	23.70%
LinearNet	Character	500K features, TF-IDF	26.72%	26.82%
	Character n-gram	1M features, TF-IDF	23.30%	23.59%
	Word	500K features, TF-IDF	23.39%	24.26%
	Word n-gram	1M features, TF-IDF	22.59%	**23.03%**
	Romanization Word	500K features, TF-IDF	28.03%	**28.02%**
	Romanization Word n-gram	1M features, TF-IDF	23.14%	23.35%

First Results: JD (Chinese)

Model	Level of Encoding	Variant	Training Error	Testing Error
GlyphNet	Character	20 layers	48.63%	48.97%
OnehotNet	Byte	16 layers	47.58%	**48.10%**
	Romanization	16 layers	47.79%	48.42%
LinearNet	Character	500K features, TF-IDF	51.30%	51.63%
	Character n-gram	1M features, TF-IDF	46.08%	48.18%
	Word	500K features, TF-IDF	46.55%	50.06%
	Word n-gram	1M features, TF-IDF	43.62%	48.30%
	Romanization Word	500K features, TF-IDF	52.61%	**52.77%**
	Romanization Word n-gram	1M features, TF-IDF	46.15%	48.47%
fastText	Character	1-gram	50.89%	51.25%
		2-gram	45.02%	48.62%
		5-gram	9.35%	52.32%
	Word	1-gram	47.50%	49.33%
		2-gram	34.62%	50.39%
		5-gram	1.60%	52.69%
	Romanization Word	1-gram	51.97%	52.21%
		2-gram	45.50%	48.66%
		5-gram	9.31%	52.42%

First Results: Rakuten (Japanese)

NYU

Model	Level of Encoding	Variant	Training Error	Testing Error
GlyphNet	Character	20 layers	46.65%	46.86%
OnehotNet	Byte	16 layers	44.63%	**45.12%**
	Romanization	16 layers	44.87%	45.21%
LinearNet	Character	500K features, TF-IDF	52.82%	**52.82%**
	Character n-gram	1M features, TF-IDF	45.57%	46.47%
	Word	500K features, TF-IDF	45.90%	47.73%
	Word n-gram	1M features, TF-IDF	43.61%	45.26%
	Romanization Word	500K features, TF-IDF	46.71%	48.31%
	Romanization Word n-gram	1M features, TF-IDF	44.17%	45.66%
fastText	Character	1-gram	51.76%	51.83%
		2-gram	43.39%	44.92%
		5-gram	16.41%	46.30%
	Word	1-gram	45.82%	46.56%
		2-gram	36.67%	44.66%
		5-gram	0.28%	47.51%
	Romanization Word	1-gram	46.58%	47.07%
		2-gram	38.97%	**44.25%**
		5-gram	1.28%	47.62%

First Results: 11st (Korean)

NYU

Model	Level of Encoding	Variant	Training Error	Testing Error
GlyphNet	Character	20 layers	31.72%	32.78%
OnehotNet	Byte	16 layers	28.84%	**32.56%**
	Romanization	16 layers	29.35%	32.73%
LinearNet	Character	500K features, TF-IDF	47.63%	**48.35%**
	Character n-gram	1M features, TF-IDF	43.16%	43.42%
	Word	500K features, TF-IDF	42.20%	45.05%
	Word n-gram	1M features, TF-IDF	40.62%	43.44%
	Romanization Word	500K features, TF-IDF	35.30%	44.77%
	Romanization Word n-gram	1M features, TF-IDF	35.29%	45.66%
fastText	Character	1-gram	42.83%	43.09%
		2-gram	34.37%	39.20%
		5-gram	9.70%	40.21%
	Word	1-gram	38.44%	40.84%
		2-gram	20.92%	40.81%
		5-gram	1.26%	40.34%
	Romanization Word	1-gram	17.33%	43.83%
		2-gram	1.79%	43.22%
		5-gram	0.54%	43.06%

Look to the Future

- **Complete Results in ~ 2 weeks**
  - We will have a more complete picture of the differences
- **From current results, byte-level ConvNet seems to be the best**
  - Byte sequences have complete information of the text
  - Allow the network to start associating context is beneficial
- **What is going on with Korean?**
  - Does long-term dependency matter more in Korean than other languages?
  - Or do I have a bug?