Applied Deep Learning



More on Embeddings

April 7th, 2020 http://adl.miulab.tw





2 Handling Out-of-Vocabulary

- One of the main problems of using pre-trained word embeddings is that they are unable to deal with out-ofvocabulary (OOV) words, i.e. words that have not been seen during training.
- Typically, such words are set to the **UNK** token and are assigned the same vector, which is an ineffective choice if the number of OOV words is large.



Subwords and characters

4 Subword Embeddings

- separating unseen or rare words into common subwords, potentially address OOV issue
- "AppleCare" = "Apple" + "Care", "iPhone11" = "iPhone" + "11"

5 Why Subwords?

- "台灣大學生喜歡深度學習"
- suboptimal word segmentation system
- ambiguity in word segmentation: "深度學習" or "深度" "學習"
- informal spelling: "So gooooooood.", "lolllllllll"

Subword Embeddings

- Possibility of leveraging morphological information
- In speech, we have phonemes; in language, we have morphemes.
- Morphemes (語素): smallest semantic units
- -s: noun plural, -ed: verb simple past tense, pre-, un-...



Morphological Recursive Neural Network



8 How to Decide Subwords?

- by simple n-gram: Apple = [App, ppl, ple]
- Byte Pair Encoding: an algorithm to build the vocabulary

Byte Pair Encoding

- Originally a compression algorithm: most frequent byte pair

 → a new byte.
- Used as a word segmentation algorithm
- Start with a unigram vocabulary of all (Unicode) characters in data
- Most frequent ngram pairs \mapsto a new ngram



- Start with a unigram vocabulary of all (Unicode) characters in data
- Ost frequent ngram pairs → a new ngram

Dictionary



- Start with a unigram vocabulary of all (Unicode) characters in data
- Ost frequent ngram pairs → a new ngram



Dictionary

Vocabulary

1, 0, w, e, r, n, w, s, t, i, d.

Add "es" with frequency (6+3)

tes



- Start with a unigram vocabulary of all (Unicode) characters in data
- Ost frequent ngram pairs → a new ngram

5 low 2 lower 6 newest 3 widest

Dictionary

Vocabulary

+ est Add "est" with frequency (6+3)

Byte Pair Encoding

- Have a target vocabulary size and stop when you reach it
- Automatically decides vocab for system

Oharacter-Level Embeddings

- Interval information word-level representation by character-level information
- completely solve OOV problem
- Output dynamically infer representation

15— Character-Level Embeddings

compositional character to word (C2W) model







- Optimizing towards pretrained embeddings
- no need to access the originating corpus



Figure 1: MIMICK model architecture.



- An extension of the word2vec skip-gram model with character n-grams
- Represent word as char n-grams augmented with boundary symbols and as whole word: Apple = [<Ap, App, ppl, ple, le>, Apple]
- Prefix, suffixes and whole words are special
- Supervised objective: text classification



Sentences and documents

19— Sentence/Document Embedding

- It is to extend to sentence/document-level?
- simply averaging word embeddings, inferring by trained models, ... etc.
- Itraining objective?



- extend skip-gram concept to sentence-level
- inspired by the distributional hypothesis: sentences that have similar surrounding context are likely to be both semantically and syntactically similar





- Change the objective to classification problem
- Ithe model can choose to ignore aspects of the sentence that are irrelevant in constructing a semantic embedding space

Spring had come.
$$\rightarrow Enc(f)$$
 $\rightarrow \bullet \bullet \bullet \bullet$
They were so black. $\rightarrow Enc(g)$ $\rightarrow \bullet \bullet \bullet \bullet - 1$ $\rightarrow \bullet \bullet \bullet \bullet - 1$
And yet his crops didn't grow. $\rightarrow Enc(g)$ $\rightarrow \bullet \bullet \bullet \bullet - 2$
He had blue eyes. $\rightarrow Enc(g)$ $\rightarrow \bullet \bullet \bullet \bullet - 3$



- Itrained on natural language inference (NLI) task
- NLI is the task of determining whether a "hypothesis" is true (entailment), false (contradiction), or undetermined (neutral) given a "premise".









- https://www.aclweb.org/anthology/W13-3512.pdf
- http://web.stanford.edu/class/cs224n/slides/cs224n-2019lecture12-subwords.pdf
- http://www.aclweb.org/anthology/D15-1176
- https://arxiv.org/pdf/1508.07909.pdf
- https://arxiv.org/pdf/1707.06961.pdf
- https://github.com/Separius/awesome-sentence-embedding
- https://openreview.net/pdf?id=rJvJXZb0W
- https://arxiv.org/pdf/1607.01759.pdf



https://arxiv.org/pdf/1705.02364.pdf