Tokenization

NLP: Fall 2024

Anoop Sarkar

Word structure and subword models

- NLP used to model the vocabulary in simplistic ways based on English
- Tokenize based on spaces into a sequence of "words"
- All novel words at test time were mapped to [UNK] (unknown token)

	word	index	embedding
	hat	hat	🟦
	learn	learn	🟦
spell errors	laern	[UNK]	🟥
variations	taaasty	[UNK]	🟥
neologisms	Transformerify	[UNK]	🟥

cs224n-2023-lecture9-pretraining.pdf

cs224n-2023-lecture9-pretraining.pdf

Byte Pair Encoding algorithm

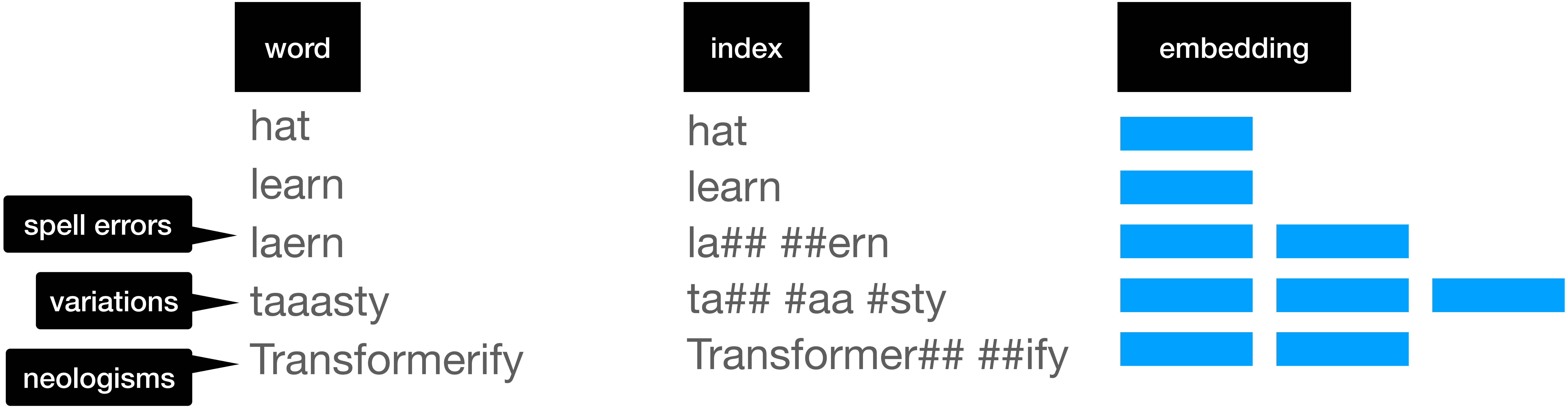
- Learn a vocabulary of parts of words (subwords)
- Vocabulary of subwords is produced before training a model on the training dataset (larger the better)
- At training and test time the vocabulary is split up into a sequence of known subwords
- Byte Pair Encoding (BPE) algorithm (takes max merges as input)
    - Init subwords with individual characters/bytes and "end of word" token.
    - Using the training data find most common adjacent subwords, merge and add to list of subwords
    - Replace all pairs of characters with new subword token; iterate until max merges

See bpe.ipynb

https://arxiv.org/abs/1508.07909

Word structure and subword models

- Common words are kept as part of the vocabulary (ignore morphology)
- Rarer words are split up into subword tokens
- In the worst case, words are split up into characters (or bytes)



word	index	embedding
hat	hat	▬
learn	learn	▬
laern	la## ##ern	▬ ▬
taaasty	ta## #aa #sty	▬ ▬ ▬
Transformerify	Transformer## ##ify	▬ ▬

spell errors

variations

neologisms

cs224n-2023-lecture9-pretraining.pdf