Department of Computing

Imperial College of Science, Technology and Medicine

Introduction Lecture for NLP and some ML baselines

THe introductory lecture for Natural Language Processing and some core concepts for Natural Language Processing with supplementary material regarding basic ML concepts required for this course

Author: Anton Zhitomirsky

Contents

1 Dealing with natural language

1.1 Complexities

- Ambiguity at word level

  > "Can you bring me the file"

- Syntactic ambiguity (prepositional phrase attachment ambiguity)

  > "I saw the boy with a telescope"

- Semantic ambiguity

  > "I haven't slept for 10 days" "The rabbit is ready for lunch"

- Referential ambiguity

  > "We gave the monkeys bananas because they were more than ready to eat."

- Non-literal meaning

  > "Call me a cab, it's raining cats and dogs"

1.2 The role of Deep Learning

- The creation and maintenance of linguistic rules often is infeasible or impractical.
- We'd much rather learn functions from data instead of creating rules based on intuition.
- deep learning is very flexible, learnable framework fro representing information from many different modalities.

1.3 Composites of Language

1.3.1 Lexicon - morphological analysis

Definition: Words: segmentation, normalisation, morphology

1. Word Segmentation (tokenization, decompounding):

   > "For example, most of what we are going to do with language relies on first separating out or tokenizing words from running text, the task of tokenization. English words are often separated from each other tokenization by whitespace, but whitespace is not always sufficient. "New York" and "rock 'n' roll" are sometimes treated as large words despite the fact that they contain spaces, while sometimes we'll need to separate "I'm" into the two words "I and am"" [2]

2. Word normalization (capitalization, acronyms, spelling variants)
3. Lemmatisation (reduce to base form = valid word)

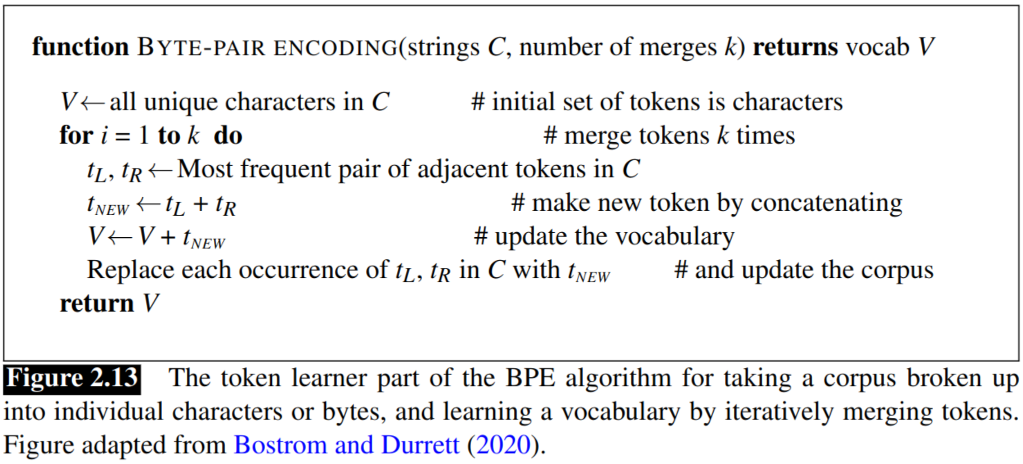
   > "Another part of text normalization is lemmatization, the task of determining lemmatization that two words have the same root, despite their surface differences. For example, the words sang, sung, and sings are forms of the verb sing. The word sing is the common lemma of these words, and a lemmatizer maps from all of these to sing" [2]

4. Stemming (reduce to root = not always valid word)

   > “Stemming refers to a simpler version of lemmatization in which we mainly stemming just strip suffixes from the end of the word” [2].

5. Byte-pair encoding (BPE) and wordpieces

   > “Instead of defining tokens as words (whether delimited by spaces or more complex algorithms), or as characters (as in Chinese), we can use our data to automatically tell us what the tokens should be” [2].



```
function BYTE-PAIR ENCODING(strings $C$, number of merges $k$) returns vocab $V$

  $V \leftarrow$ all unique characters in $C$          # initial set of tokens is characters
  for $i = 1$ to $k$ do                                # merge tokens $k$ times
    $t_L, t_R \leftarrow$ Most frequent pair of adjacent tokens in $C$
    $t_{NEW} \leftarrow t_L + t_R$                     # make new token by concatenating
    $V \leftarrow V + t_{NEW}$                         # update the vocabulary
    Replace each occurrence of $t_L, t_R$ in $C$ with $t_{NEW}$     # and update the corpus
  return $V$
```

Figure 2.13 The token learner part of the BPE algorithm for taking a corpus broken up into individual characters or bytes, and learning a vocabulary by iteratively merging tokens. Figure adapted from Bostrom and Durrett (2020).

> “The BPE token learner begins BPE with a vocabulary that is just the set of all individual characters. It then examines the training corpus, chooses the two symbols that are most frequently adjacent (say ’A’, ’B’), adds a new merged symbol ’AB’ to the vocabulary, and replaces every adjacent ’A’ ’B’ in the corpus with the new ’AB’. It continues to count and merge, creating new longer and longer character strings, until k merges have been done creating k novel tokens; k is thus a parameter of the algorithm. The resulting vocabulary consists of the original set of characters plus k new symbols” [2].

6. Part-of-speech tagging (Recognize category of word): verb, noun, adverb, adjective, determiner, preposition...

   It is the process of assigning a part-of-speech to each word in a text. This is a disambiguation task; words are ambiguous - have more tha one possible part-of-speech - and the goal is to find the correct tag for the sitauation. E.g. ‘book’ can be a verb (‘book that flight’) or a noun (‘hand me that book’). The POS-tagging resolves these ambiguities by choosing the proper tag for the context [2].

7. Morphological analysis (recognize/generate word variants):

   > “Morphology is the study of the way words are built up from smaller meaning-bearing units called morphemes. Two broad classes of morphemes can be distinguished: **stems** — the central morpheme of the word, supplying the main meaning — and **affixes** — adding “additional” meanings of various kinds. So, for example, the word fox consists of one morpheme (the morpheme fox) and the word cats consists of two: the morpheme cat and the morpheme ‘-s’” [2]

1.3.2 Syntax

Syntax comes from the Greek s’yntaxis, meaning “setting out together or arrangement”, and refers to the way words are arranged together [2].

We can form a parse tree based on some syntax rules which makes up grammar. This was typically used in applications before deep learning; there are sentences that make sense but don't follow grammar.

	Phrase Structure Rule	Example
$S \rightarrow NP \quad VP$	Sentence → Noun-phrase Verb-phrase	I prefer a morning flight
$NP \rightarrow Det\ N$	Noun-phrase → Determiner Noun	prefer a morning flight
$VP \rightarrow VNP$	Verb-phrase → Verb Noun-phrase	leave Boston in the morning
$VP \rightarrow V$	Verb-phrase → Verb	
$VP \rightarrow VPP$	Verb-phrase → Verb Propositional-phrase	leaving on Thursday
$PP \rightarrow PNP$	Preposition-phrase → Preposition Noun-phrase	from Los Angeles

1.3.3 Semantics

Definition: Meaning of words and sentences

"We also introduce word sense disambiguation (WSD), the task of determining which sense of a word is being used in a particular context [...] A sense (or word sense) is a discrete representation of one aspect of the meaning of a word." [2].

Compositional meaning understands who did what to whom, when, where, how and why. It composes the meaning of the setnece, based on the meaning of the words and the structure of the sentence. Here, the dog chased the man = The man was chased by the dog, but The dog bit the man ≠ The man bit the dog.

"Semantic role labeling (sometimes shortened as SRL) is the task of automatically finding the semantic roles of each argument of each predicate in a sentence" [2].

1.3.4 Discourse

Definition: Meaning of a text (relationship between sentences)

"language does not normally consist of isolated, unrelated sentences, but instead of collocated, structured, coherent groups of sentences. We refer to such a coherent structured group of sentences as a discourse, and we use the word coherence to refer to the relationship between sentences that makes real discourses different than just random assemblages of sentences" [2].

"Coreference resolution is the task of determining whether two mentions corefer, by which we mean they refer to the same entity in the discourse model (the same discourse entity)" [2].

1.3.5 Pragmatics

Definition: Intentions, commands; what is the intent of the text, how to react to it?

2 How to represent language to an algorithm

2.1 One-hot-encoding

- Corpus: A collection of documents, i.e. our entire dataset
- Document: one item of our corpus (e.g. a sequence)
- token: the atomic unit of a sequence (e.g. a word)
- vocabulary: the unique tokens across our entire corpus

3 ML Refresher

3.1 Linear activation function

useful for regression

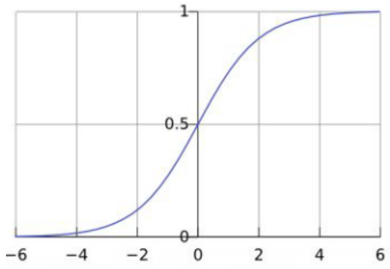
$$f(x) = x \tag{1}$$

3.2 Non-linear activation functions

3.2.1 Sigmoid

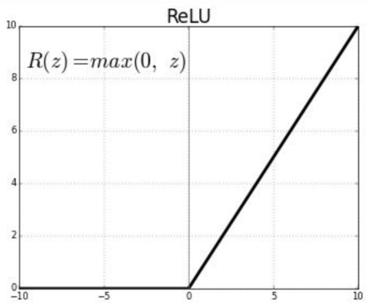
useful for binary classification
useful for multi-label classification (predicting many classes)

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}} \tag{2}$$



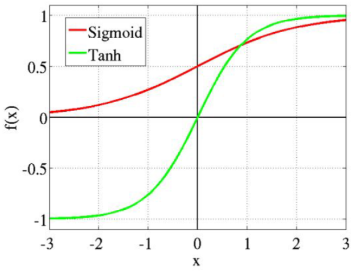
3.2.2 ReLU

$$f(x) = ReLU(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ 1 & \text{for } x > 0 \end{cases} \tag{3}$$



3.2.3 Tanh

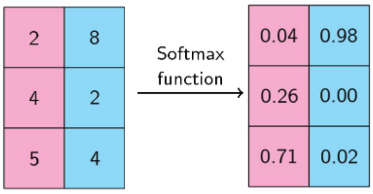
$$f(x) = \tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{4}$$



3.2.4 Softmax

useful for multi-label classification (predicting one class out of many)

$$softmax(z_i) = \frac{e^{z_i}}{\sum_k e^{z_k}} \tag{5}$$



2	8
4	2
5	4

0.04	0.98
0.26	0.00
0.71	0.02

3.3 Loss Functions

3.3.1 Mean suqared error

useful for regression

$$L = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{6}$$

3.3.2 Binary cross-entropy

useful for binary classification
useful for multi-label classification (predicting many classes)

$$L = -\frac{1}{N} \sum_{i=1}^{N} (y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \tag{7}$$

3.3.3 Categorical cross-entropy

useful for multi-label classification (predicting one class out of many)

$$L = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} y_c^{(i)} \log(\hat{y}_c^{(i)}) \tag{8}$$

3.4 Regularization

any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error. (See Chapter 7 of [1])