is450 Cheat Sheet

by cheatingcvrlo via cheatography.com/183658/cs/38238/

preprocessing pipeline

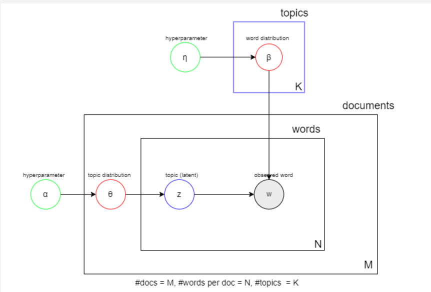
```
def corpus2docs(corpus):
       fids = corpus.fi leids()
        docs1 = []
       for fid in fids:
                doc_raw = corpus.ra w(fid)
               doc = nltk.w ord _to ken ize (do c_raw)
                doc s1.a pp end (doc)
        docs2 = [[w.lo wer() for w in doc] for doc in docs1]
        docs3 = [[w for w in doc if re.sea rch ('^ [a- z]+$', w)] for doc in docs2]
        docs4 = [[w for w in doc if w not in stop_list] for doc in docs3]
        docs5 = [[stem mer.st em(w) for w in doc] for doc in docs4]
        return docs5
def docs2v ecs (docs, dictio nary):
       # docs is a list of documents returned by corpus 2docs.
       # dictionary is a gensim.co rpo ra.D ic tionary object.
        vecs1 = [dicti ona ry.d oc 2bo w(doc) for doc in docs]
        tfidf = gensim.mo del s.T fid fMo del (vecs1)
        vecs2 = [tfidf [vec] for vec in vecs1]
        return vecs2
```

POS tags

N (noun)	dog, cat, chair
V (verb)	read, write, get
ADJ (adjective)	pretty, smart, blue
ADV (adverb)	gently, carefully, extremely
P (preposition)	in, on, by, with, about
PRO (pronoun)	I, me, mine, it, they...
CON (conju-nction)	and, or, but, while, because
INT (interjection)	ooh, wow, yeah
DET (deter-miner)	all, his, they
AUX (auxiliary verb)	**have** done, **might** do
PAR (particle)	look **up**, get **on**
NUM (numeral)	one, two, three

Context-free grammar

LDA



LDA

gibbs sampling	1. random word-to-topic assignment
	2. re-assign each word to a topic, one by one, **assuming all other assignments are correct**
hyperp-arameters	high $alpha$ --> documents feature a mixture of most topics
	high eta --> topics feature a mixture of most words
evaluation	coherence (PMI), human eval

Sentiment-Topic Model (Plate Notation)

Discourse Markers

causal	because
consequence	as a result
conditional	if
temporal	when
additive	and
elaboration	[exemplification, re-wording]
contrastive/con-cessive	but

Preparation for NLTK classifier

#doc_tuple = (doc_representation, label)

`> ({'police':1, 'lawye r':1, 'court e')`

#train_set = [doc_tuple1, doc_tuple2, ...]

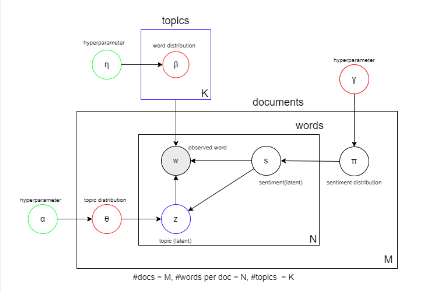
```
Grammar = {
        obj ects: [
                Wor ds/ tokens:
terminals,
                Right pos above:
tags,
                syntactic Above:
tags,
                sentence Above:
        ];
        Rules: [
               X: node name, #eg
" VP" (verb phrase)
               Y: sequence of
objects that make up X #eg
(V+NP)
        ]
}
```

Morphemes

stems, affixes (prefix/suffix). Useful for POS tagging and text normalization

Semantics

synonyms	diff words, same meaning
polyseme	same word, diff meaning
hypernym/-hyponym	category >>> specific
meronym/m-etonym	part >>> whole



Cluster Purity

$$P_i = \max_j P_{ij} \qquad P_{ij} = \frac{\#docs(class = j, cluster = i)}{\#docs(cluster = i)}$$

Overall purity

$$\sum_{i=1}^{K} \left[\frac{\#docs(cluster = i)}{\#docs} \cdot P_i \right] \qquad K = \#clusters$$

Cluster Entropy

$$e_i = -\sum_{j=1}^{L} P_{ij} \log_2 P_{ij}$$

Pointwise Mutual Information

$$PMI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)\, p(w_j)}$$

C

By **cheatingcvrlo**

cheatography.com/cheatingcvrlo/

Published 16th April, 2023.
Last updated 17th April, 2023.