

Word Normalization and Stemming

/ Нормализация, лемманизация и СТЕММИНГ

**Speech and Language Processing (3rd ed. raft),
Dan Jurafsky and James H. Martin. Глава 2.3, стр.
11.**

Ерофеев Илья
24.03.2017

Normalization

- Need to “normalize” terms
 - Information Retrieval: indexed text & query terms must have same form.
 - We want to match **U.S.A.** and **USA**
- We implicitly define equivalence classes of terms
 - e.g., deleting periods in a term
- Alternative: asymmetric expansion:
 - Enter: **window** Search: **window, windows**
 - Enter: **windows** Search: **Windows, windows, window**
 - Enter: **Windows** Search: **Windows**
 - Enter: **Снеговик** Search: **Снеговик, снеговики**
- Potentially more powerful, but less efficient

Case folding

- Applications like IR: reduce all letters to lower case
 - Since users tend to use lower case
 - Possible exception: upper case in mid-sentence?
 - e.g., *General Motors*
 - *Fed* vs. *fed*
 - *SAIL* vs. *sail*
 - *МегаФон* vs. мегафон
- For sentiment analysis, MT, Information extraction
 - Case is helpful (*US* versus *us* is important)

Lemmatization

- Reduce inflections or variant forms to base form
 - *am, are, is* → *be*
 - *car, cars, car's, cars'* → *car*
- Lemmatization: have to find correct dictionary headword form
- Machine translation
 - Spanish **quiero** ('I want'), **quieres** ('you want') same lemma as **querer** 'want'
- *the boy's cars are different colors* → *the boy car be different color*
- Мы ели суп, а вдоль аллеи стояли раскидистые ели -> я есть суп, а вдоль аллея стоять раскидистый ель

Morphology

- **Morphemes:**
 - The small meaningful units that make up words
 - **Stems:** The core meaning-bearing units
 - **Affixes:** Bits and pieces that adhere to stems
 - Often with grammatical functions

Приведите примеры аффиксов

Stemming

- Reduce terms to their stems in information retrieval
- *Stemming* is crude chopping of affixes
 - language dependent
 - e.g., ***automate(s), automatic, automation*** all reduced to ***automat.***
 - Например, чистый, чистка сведутся к «ЧИСТ».

*for example compressed
and compression are both
accepted as equivalent to
compress.*



for exampl compress and
compress ar both accept
as equal to compress

Porter's algorithm

The most common English stemmer

Step 1a

sses	→	ss	caresses	→	caress
ies	→	i	ponies	→	poni
ss	→	ss	caress	→	caress
s	→	∅	cats	→	cat

Step 2 (for long stems)

ational	→	ate	relational	→	relate
izer	→	ize	digitizer	→	digitize
ator	→	ate	operator	→	operate
...					

Step 1b

(*v*)ing	→	∅	walking	→	walk
			sing	→	sing
(*v*)ed	→	∅	plastered	→	plaster
...					

Step 3 (for longer stems)

al	→	∅	revival	→	reviv
able	→	∅	adjustable	→	adjust
ate	→	∅	activate	→	activ
...					

Viewing morphology in a corpus

Why only strip `-ing` if there is a vowel?

```
(*v*)ing → ∅ walking → walk  
sing → sing
```

Как в большинстве случаев узнать, надо ли отбрасывать `ing`?

Viewing morphology in a corpus

Why only strip `-ing` if there is a vowel?

```
(*v*)ing → ∅ walking → walk  
sing → sing
```

```
tr -sc 'A-Za-z' '\n' < shakes.txt | grep 'ing$' | sort | uniq  
-c | sort -nr
```

1312 King	548 being
548 being	541 nothing
541 nothing	152 something
388 king	145 coming
375 bring	130 morning
358 thing	122 having
307 ring	120 living
152 something	117 loving
145 coming	116 Being
130 morning	102 going

```
tr -sc 'A-Za-z' '\n' < shakes.txt | grep '[aeiou].*ing$' | sort  
| uniq -c | sort -nr
```

Объясните работу данных команд?

Dealing with complex morphology is sometimes necessary

- Some languages requires complex morpheme segmentation
 - Turkish
 - **Uygarlastiramadiklarimizdanmissinizcasina**
 - `(behaving) as if you are among those whom we could not civilize`
 - **Uygar** `civilized` + **las** `become`
 - + **tir** `cause` + **ama** `not able`
 - + **dik** `past` + **lar** `plural`
 - + **imiz** `p1pl` + **dan** `abl`
 - + **mis** `past` + **siniz** `2pl` + **casina** `as if`

Basic Text Processing

Word Normalization and
Stemming

Литература, статьи:

- Диалог. Лемматизация слов русского языка в применении к распознаванию слитной речи. Саввина Г.В., Саввин И.В.
<http://www.dialog-21.ru/digest/2001/articles/savvina/>
- Stanford NLP Group. Stemming and lemmatization.
<https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html>
- Alexander Gelbukh. Computational Linguistics and intelligent Text Processing. 2006
- Саввина Г.В. Распознавание ключевых слов в потоке слитной речи. Искусственный интеллект , №3 2000 г., с.543-551.