

# TDCEN IZAEON

Tokenization  $\rightarrow$  part of text normalization

1. Segmenting/Tokenizing words
2. Normalizing words format
3. Segmenting sentences

Words

- \* Type : an element of the vocabulary
- \* Token : an instance of that type

! do wh main - mainly  
business level processing } how many  
fragments } words?

Titch's cast in the hat  
is different from other casts

- \* lemma: same stem, part of speech, rough word sense
  - ↳ cat & cats is same lemma
- \* word form: full inflected surface form
  - ↳ cat & cats is different word forms



## How many words?

they lay back on the San Francisco grass and looked at the stars and their

- **Type:** an element of the vocabulary.
- **Token:** an instance of that type in running text.

dozen  $\rightarrow$  12 dozens Cor (4) "San", "Francisco"

Type  $\rightarrow$  13 types (or 12) "The" "They" "Heir"  
Cor 11 "San Francisco" "San Francisco"

In general,

$N$  = number of tokens

V = Vocabulary

= set of types,

with  $|V|$  is the size of

There are some issues in tokenization:

- use of "what're", "finland's", "isn't"
- use of - "Hawlett-Packard"

the vocabulary

"state-of-the-art"

- "San Francisco", 1 or 2 tokens?
- m.p.h., Ph.D.
- language issues  
(German, French, Japanese, Chinese, etc.)

## NORMALIZATION

"Normalize"

- \* IR → indexed text & query terms must have same form
- \* asymmetric expansion
- \* reduce all letters to lowercase
  - ↳ sometimes "Case" is helpful: sentiment analysis, MT, IE
  - (US and us is important)

Lemmatize → reduce inflection/  
variant forms  
to base form

- \* am, are, is ⇒ be
  - \* car, cars, car's, cars' ⇒ car
- ↳ finding correct dictionary headword form

Looking at parts of words → morphology

↓  
morphemes

↳ small meaningful units that  
make up words

↳ 2 kinds of morphemes:

- stems: core meaning-bearing units
- affixes: pieces that adhere to stems  
↳ often with grammatical function