**Regex**

Re.search() finds first match anywhere in string (match object)

Lazy = ‘The lazy dog went to sleep’

Result = re.search(‘cat’, lazy)

Re.match() finds first match at beginning of string (match object)

Lazy = ‘Dog The lazy dog went to sleep’

Result = re.search(‘dog’, lazy)

Re.findall()

Single-character regex

Any char (except newline = \n)

‘F..m:’

YES Farm

YES Foom

NO Foam!

\s Whitespace

‘Pine\sapple’

YES ‘Pine apple’

YES ‘Pine\tapple’

NO ‘Pineapple’

\S non-whitespace

‘Pine\Spple’

YES ‘Pineapple’

NO ‘Pine pple’

\d Digit, 0-9

‘Apple \d\d\d’

YES ‘Apple 123’

NO ‘Apple ABC’

c “word” character (letter, digit, underscore)

‘cat\w’

YES ‘cats’

NO ‘cat!’

\W non-word character

Escape sequences

Want to look for $ . ^ \

‘\$19\.99’ will match ‘$19.99’

‘\*’ zero or more of the previous thing

Python: a\*

Cat, caaaat

‘.\*’ one or more of anything

+ one or more of the previous thing

Ex. Two punctuations in a row

Set of characters []

[aeiou]`1 o

Re.finditer() find all matches anywhere in a string

Pipe |

**Natural Language Processing**

Bytes to a string

A string is a sequence of charactes

A character set specifies numbers should be interpreted as character symbols

**Unicode**

one universal character set

Contains ISO Latin first: first 256 characters the same

Code points: every letter in alphabet is assigned to magic number

Encoded in a file or string in different ways, depending on efficiency

Character set maps character to numbers (code points)

Encoding stores number in particular format in a sequence of bytes (file/m emory)

Font has instructions for displaying the visual form of a charcter given a code point

**Normalization**

Str.lower()

**Tokenization**

Nltk.word\_tokenize(sentence)

Outputs list of words in the sentence

**Stemming**

Fast but inaccurate