AI INSTITUTE #AIISC
UNIVERSITY OF SOUTH CAROLINA

UNIVERSITY OF SOUTH CAROLINA

CSCE 771: Computer Processing of Natural Language
Lecture 3: Words, Morphology, Lexicons

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

25TH AUG 2020

Carolinian Creed: "I will practice personal and academic integrity."

Acknowledgement: Using materials by Prof. Mausam, Jurafsky & Martin, Julia Hirschberg, …

Organization of Lecture 3

- Opening Segment
  - Announcements
  - Review of previous class
  - Discussion on course projects
- Main Segment

- Concluding Segment
  - Course project
  - Reading material
  - About Next Lecture – Lecture 4

Main Section
- Understanding concepts
  - Words
  - Morphology – structure of words
  - Lexicons – collection of words
- Using them for content processing
- Dealing with multiple languages
- Concluding comments

Opening Segment

- Announcements
- Review of Lecture 2
  - We surveyed a wide variety of issues around communication and languages
  - Looked at processing of audio and visual media
  - Discussed project ideas and course logistics

Understanding Concepts - Words

What is a Word ?

- Unix command - man wc

"**A word is defined as a string of characters delimited by white space characters**."

- Example
  - Content = "CSCE 771: Computer Processing of Natural Language
    Lecture 3: Words, Morphology, Lexicons
    Prof. Biplav Srivastava, AI Institute
    31st Aug 2020 "
  - Command = "wc -w content.txt"
    Result = "20 content.txt" (stored in file - result.txt)
  - "CSCE 771: Computer Processing of Natural Language (7)
    Lecture 3: Words, Morphology, Lexicons (12)
    Prof. Biplav Srivastava, AI Institute (17)
    31st Aug 2020 " (20)

Types of Words in English

Content words (open-class – i.e., continuously changing):

– **Nouns**: student, university, knowledge,...

– **Verbs**: write, learn, teach,...

– **Adjectives**: difficult, boring, hard, ...

– **Adverbs**: easily, repeatedly,...

Function words (closed-class – fixed):

– **Articles**: a, an, the

– **Prepositions**: in, with, under,...

– **Conjunctions**: and, or,...

– **Determiners**: a, the, every,...

Another Language - Turkish

A Turkish word

uygarlaştıramadıklarımızdanmışsınızcasına

uygar_laş_tır_ama_dık_lar_ımız_dan_mış_sınız_casına

"as if you are among those whom we were not able to civilize (=cause to become civilized)"

uygar: *civilized*
_laş: *become*
_tır: *cause somebody to do something*
_ama: *not able*
_dık: past participle
_lar: plural
_ımız: 1st person plural possessive (our)
_dan: *among* (ablative case)
_mış: past
_sınız: 2nd person plural (you)
_casına: *as if* (forms an adverb from a verb)

K. Oflazer pc to J&M

Chinese: 我开始写小说 = 我 开始 写 小说
I start(ed) writing novel(s)

A strict reliance on spaces will make us miss useful parts of text

Common Definitions

- **Corpus** (plural corpora): a computer-readable corpora collection of text or speech.
- **Lemma**: A lemma is a set of lexical forms having the same stem, the same major part-of-speech, and the same word sense. Example: Cat and cats have same lemma.
- **Word form**: The word form is the full inflected or derived form of the word. Example: Cat and cats have different word forms.
- **Word type**: Types are the number of distinct words in a corpus. If the set of words is V, the number of types is the word token vocabulary size |V|.
- **Word tokens**: The total number N of running words in the sentence / document of interest.
- **Code switching**: use multiple languages in a code switching single communicative act – Example: Hindlish (Hindi English), Spanish (Spanish English)

"They picnicked by the pool, then lay back on the grass and looked at the stars."

- 16 tokens, 14 word types

Source: Jurafsky & Martin

Lexical Meaning – Common Terms

- **Synonym**: same/ similar meaning
  - start-begin, finish-end, far-distant
- **Antonym**: opposite meaning
  - Far – near, clever - stupid, high - low, big – small
- **Homonym**: identical in spelling and pronunciation
  - bear, bank, ...
- **Homophones**: sounds identical but are written differently
  - site-sight, piece-peace.
- **Homograph**: written identically but sound differently
  - Potato, tomato, lead, wind, minute
- **Polysemy**: a word or phrase which has two (or more) rated meanings
  - Duck, sharp

Source: Mausam

Knowing About Words

> Of course he wants to take the advanced course too.
> He already took two beginners' courses.

- **Words:** set of characters separated by spaces
- **Word tokens**: The total number N of running words
- **Word forms**:
  - Spelling differences - specialize v/ specialise
  - Meaning similarity/differences - Take/ took, course/ courses, two/ too
- **Word types:** distinct words

Pop Quiz: Are word tokens and word types same in the example above?

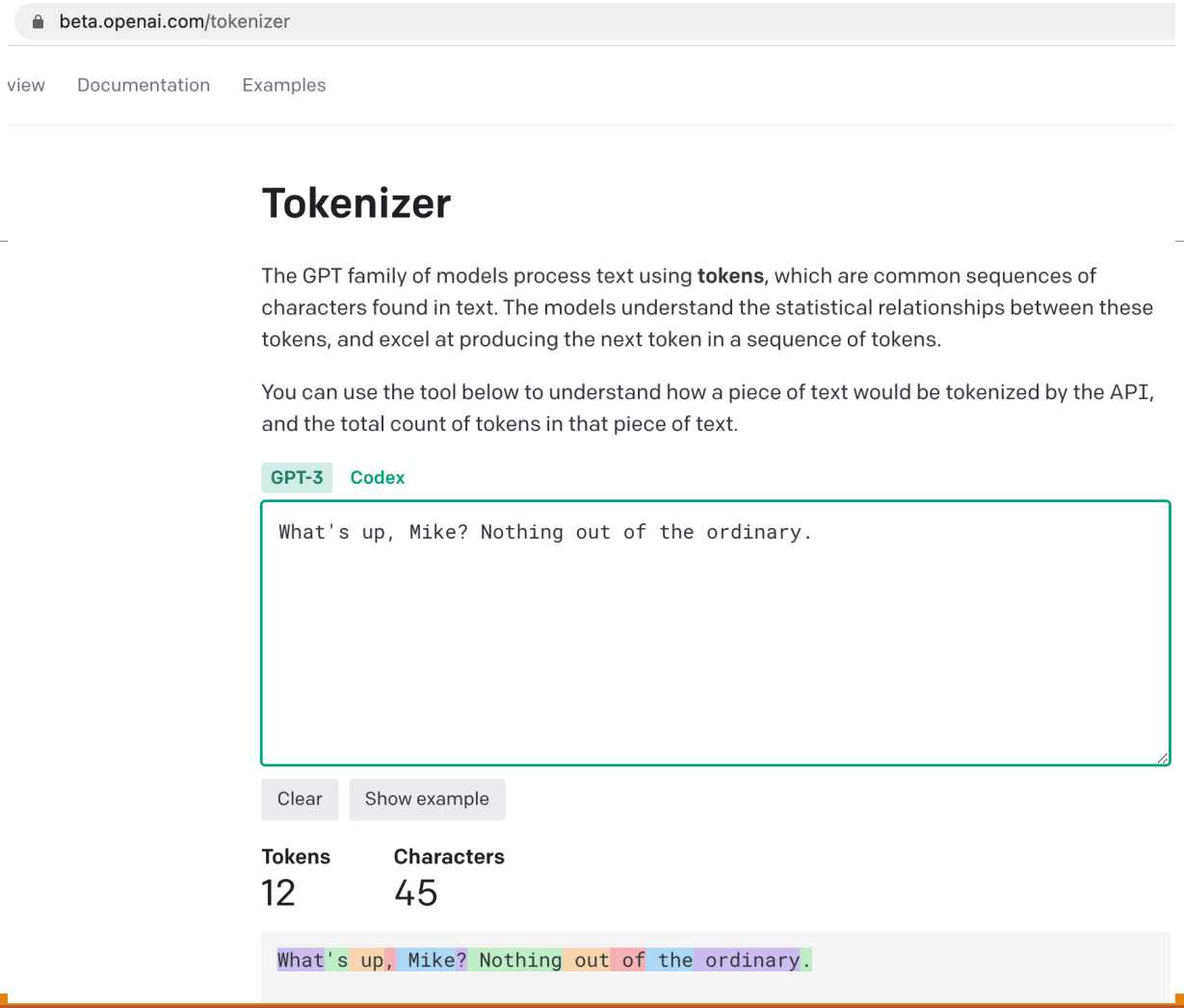
Knowing About Words

Of course he wants to take the advanced course too.
He already took two beginners' courses.

The course we are taking is about preparing specialized three course meals.

- **Synonym**: same/ similar meaning [take -> participate ?]
- **Antonym**: opposite meaning [take -> give?]
- **Homonym**: identical in spelling and pronunciation [course?]
- **Homophones**: sounds identical but are written differently [two/ too]
- **Homograph**: written identically but sound differently
- **Polysemy**: a word or phrase which has two (or more) rated meanings [took]

Tokens



Credit: OpenAI

Word Variety

- **Inflection:** creates different forms of the same word
  - Verbs: to be, being, I am, you are, he is, I was,
  - Nouns: one book, two books
- **Derivation:** creates different words from the same lemma
  - grace ⇒ disgrace ⇒ disgraceful ⇒ disgracefully
- **Compounding:** new words from combinations

"ice cream", "website", "web site", "New York-based"

New words over time:
Google ⇒ Googler, to google, to ungoogle,
to misgoogle, googlification, ungooglification,
googlified, Google Maps,
Google Maps service, ...

Review: Regular Expression

Metacharacter	Explanation
^	Matches the starting position within the string
.	Matches any single character
[ ]	Matches a single character that is contained within the brackets
[^]	Matches a single character that is not contained within the brackets.
$	Matches the ending position of the string
*	Matches the preceding element zero or more times
+	Matches the preceding element one or more times
|	Separates choices

Regex	Matches any string that
hello	contains {hello}
gray|grey	contains {gray, grey}
gr(a|e)y	contains {gray, grey}
gr[ae]y	contains {gray, grey}
b[aeiou]bble	contains {babble, bebble, bibble, bobble, bubble}
[b-chm-pP]at|ot	contains {bat, cat, hat, mat, nat, oat, pat, Pat, ot}
colou?r	contains {color, colour}
rege(x(es)?|xps?)	contains {regex, regexes, regexp, regexps}
go*gle	contains {ggle, gogle, google, gooogle, goooogle, ...}
go+gle	contains {gogle, google, gooogle, goooogle, ...}
g(oog)+le	contains {google, googoogle, googoogoogle, googoogoogoogle, ...}
z{3}	contains {zzz}
z{3,6}	contains {zzz, zzzz, zzzzz, zzzzzz}
z{3,}	contains {zzz, zzzz, zzzzz, ...}

Example Source: https://cs.lmu.edu/~ray/notes/regex/

Implementation: Finding Words in Python

- Python has extended Regex specifications for convenience
- Useful for
  - Matching patterns
  - Information extraction
  - Content manipulation (e.g., substitution)
  - Error (e.g., spelling) correction

```
data = "The CSCE 771 course is taught at
        University this Fall!"
pattern = "[tT]+\w"
m = re.findall(pattern, data)
print(m)
```

['Th', 'ta', 'ty', 'th']

See more code examples on class Github

Details: https://docs.python.org/3/library/re.html

Programming Exercise

- Using regex
- [Later] Using Wordnet to find information about words

Code: https://github.com/biplav-s/course-nl-f22/blob/main/sample-code/l3-word/WordLesson-Examples.ipynb

Morphology

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

Morphemes: stems, affixes

dis-grace-ful-ly
prefix-stem-suffix-suffix

Many word forms consist of a stem plus a number of affixes (*prefixes* or *suffixes*)

Infixes are inserted inside the stem.

Circumfixes (German gesehen) surround the stem

Morphemes: the smallest (meaningful/grammatical) parts of words.

Stems (grace) are often free morphemes.

Free morphemes can occur by themselves as words.

Affixes (dis-, -ful, -ly) are usually bound morphemes.

Bound morphemes have to combine with others to form words.

- Plural nouns add -s to singular:
  - book-books,
- but:
  - box-boxes, fly-flies, child-children
- Past tense verbs add -ed to infinitive:
  walk-walked,
- but:
  - like-liked, leap-leapt

Source: Julia Hirschberg

Morphological Generation

- Generate legal variations.
  - For **grace** (**stem**): graceful, gracefully, disgrace, disgraceful, disgracefully, ungraceful, ungracefully, undisgraceful, undisgracefully,…
- But avoid ungrammatical variations
  - *gracelyful, *gracefuly, *disungracefully,...

Source: Julia Hirschberg

Word Variety – Creating New Words

- **Clitics** - *a clitic is a morpheme that has syntactic characteristics of a word, but depends phonologically on another word or phrase. In this sense, it is syntactically independent but phonologically dependent …*

English: “doesn’t” , “I’m” ,
Italian: “dirglielo” = dir + gli(e) + lo // tell + him + it

New words over time:
Google ⇒ Googler, to google, to ungoogle, to misgoogle, googlification, ungooglification, googlified, Google Maps,
Google Maps service, …

Advanced Topic –Language Formalism

An alphabet $\sum$ is a set of symbols:

e.g. $\sum = \{a, b, c\}$

A string ω is a sequence of symbols, e.g $\omega = abcb$.

The empty string ε consists of zero symbols.

The Kleene closure $\sum^*$ ('sigma star') is the (infinite) set of all strings that can be formed from $\sum$:

$\sum^* = \{\varepsilon, a, b, c, aa, ab, ba, aaa, \ldots\}$

A language $L \subseteq \sum^*$ over $\sum$ is also a set of strings.

Typically we only care about proper subsets of $\sum^*$ $(L \subset \Sigma)$.

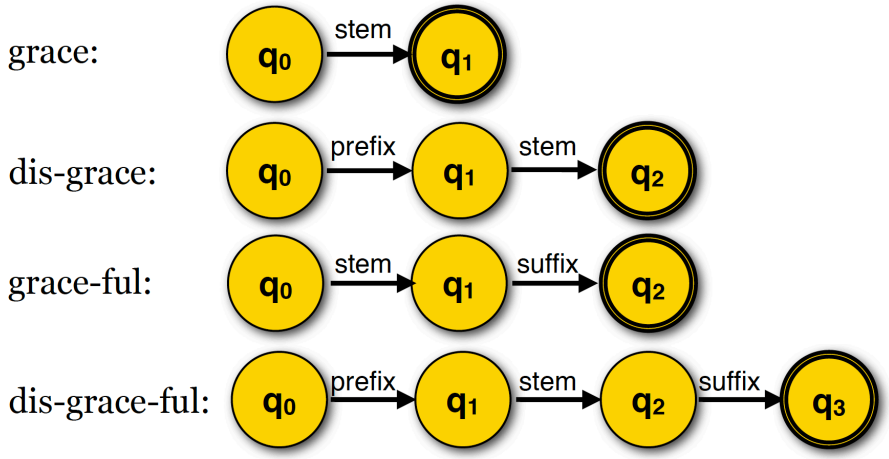
- Automata
- Finite State Automata
- Deterministic Finite State Automata(DFSA)
- Non-Deterministic Finite State Automata(NDFSA)

Source: Julia Hirschberg

Advanced Topics – Recognizing as Automata

- Automata –
  - an abstract model of a computer which reads an input string, and changes its internal state depending on the current input symbol. It can either accept or reject the input string.
  - Hence, an automata defines a language
- Finite State Automata – regular expressions

Source: Julia Hirschberg



Lexicon

George A. Miller (1995). WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41.

WordNet

Home page: https://wordnet.princeton.edu/

- Find senses of a word, definition and examples
- Find synonyms and antonyms
- Convenient program-level integrated usage supported in tools like NLTK

See code examples on class Github

Programming Exercise

- Using Wordnet to find information about words

Code: https://github.com/biplav-s/course-nl-f22/blob/main/sample-code/l3-word/WordLesson-Examples.ipynb

SentiWordNet

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010 SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. LREC-2010

Home page: http://sentiwordnet.isti.cnr.it/

- All WordNet synsets automatically annotated for degrees of positivity, negativity, and neutrality/objectiveness
- [estimable(J,3)] "may be computed or estimated"

  `Pos 0 Neg 0 Obj 1`
- [estimable(J,1)] "deserving of respect or high regard"

  `Pos .75 Neg 0 Obj .25`

Source: Jurafsky & Martin

Scherer's Typology of Affective States

Emotion: relatively brief episode of synchronized response of all or most organismic subsystems in response to the evaluation of an event as being of major significance

angry, sad, joyful, fearful, ashamed, proud, desperate

Mood: diffuse affect state …change in subjective feeling, of low intensity but relatively long duration, often without apparent cause

cheerful, gloomy, irritable, listless, depressed, buoyant

Interpersonal stance: affective stance taken toward another person in a specific interaction, coloring the interpersonal exchange

distant, cold, warm, supportive, contemptuous

Attitudes: relatively enduring, affectively colored beliefs, preferences predispositions towards objects or persons

liking, loving, hating, valuing, desiring

Personality traits: emotionally laden, stable personality dispositions and behavior tendencies, typical for a person

nervous, anxious, reckless, morose, hostile, envious, jealous

Source: Jurafsky & Martin

The General Inquirer

Philip J. Stone, Dexter C Dunphy, Marshall S. Smith, Daniel M. Ogilvie. 1966. The General Inquirer: A Computer Approach to Content Analysis. MIT Press

- Home page: http://www.wjh.harvard.edu/~inquirer
- List of Categories: http://www.wjh.harvard.edu/~inquirer/homecat.htm
- Spreadsheet: http://www.wjh.harvard.edu/~inquirer/inquirerbasic.xls

Categories:

- Positiv (1915 words) and Negativ (2291 words)
- Strong vs Weak, Active vs Passive, Overstated versus Understated
- Pleasure, Pain, Virtue, Vice, Motivation, Cognitive Orientation, etc

Free for Research Use

Source: Jurafsky & Martin

Lecture 3: Concluding Segment

Course Project

- Choice 1: student proposed. Suggested themes
  - Environment: understanding regulations, impact of global warming
  - Health (e.g., COVID-19): e.g., impact of disease, prevalence of masks, availability of health services
  - Finance: economy, growth of a company
  - NLP methods: language models, explanation
- Choice 2: instructor proposed. AI for water
  - Write your name in spreadsheet: “Water - <region>”
  - Use or collect-and-add documents in regulations repository: https://drive.google.com/drive/folders/1H23Afgb3VS1yUe9uKiYH8--RoqBRZ9aV?usp=sharing

Choosing a Project – Some Considerations

- Scope: what is the problem?
- Current-state: what happens in the problem today?
- Who cares: who will benefit with the problem being solved?
- Desired-state: what will be the future situation if your project succeeds?
- Resources/ dataset: do you have reasonable data and compute resources to do the work?
- Evaluation: how will we measure goodness of the work?

Discussion: Course Project

- Expectations
  - Apply methods learned in class or of interest to a problem of interest
  - Be goal oriented: aim to finish, be proactive, be innovative
  - Do top-class work: code, writeup, presentation
- Typical pitfalls
  - Not detailing out the project, assuming data
  - Not spending enough time

Review Current List of Selections

- What will be awarded
  - Results and efforts (balance)
  - Challenge level of problem

Lecture 3: Reading Material

- Paper:
  "Contextual Word Representations: Putting Words into Computers", by Noah Smith, CACM June 2020
- Discussion next week - Thursday

Lecture 3: Concluding Comments

- We looked at structure of building block of text, i.e., words, from an English and non-English perspective
- We reviewed morphology
- We also looked at useful lexicons that can help simplify language tasks

About Next Lecture – Lecture 4

Lecture 4:

- NLP tasks - basics
  - Normalization of text
  - Lemmatization, stemming
  - Parts of speech tagging
- Evaluation methods
- Review of projects proposed by students
- Clarify doubts on reading material
  - “Contextual Word Representations: Putting Words into Computers”, by Noah Smith, CACM June 2020