Natural Language Processing

CSC 495-012 and CSC 791-012

Munindar P. Singh, Professor
singh@ncsu.edu

Department of Computer Science
North Carolina State University

Fall 2022

Bio Highlights and Humble Bragging

- Students
  - Graduated PhD: 29; MS: 39
  - Inaugural Alumni Hall of Fame: Nirmit Desai, Pınar Yolum
  - Inaugural Faces of Computer Science (EB2 hall): Chris Hazard
  - Rising Star Alumnus: Chris Hazard, Anup Kalia
  - Associate Editors: Amit Chopra, Michael Maximilien, Pınar Yolum
  - CGS MS Thesis Award: Payal Chakravarty; nominee: Anup Kalia
  - Dept awards. 2021: Amanul Haque, Parth Diwanji; 2020: Hui Guo; 2019: Nirav Ajmeri; 2017: Nirav Ajmeri, Hui Guo, Pradeep Murukannaiah; 2016: Pradeep Murukannaiah
- NCSU Internal
  - Outstanding Graduate Faculty Mentor Award
  - Research Leadership Academy
  - Alumni Distinguished Graduate Professor
  - Outstanding Research Achievement Award
- External
  - Member (honoris causa), Academia Europaea
  - Fellow, American Association for the Advancement of Science
  - Fellow, Association for the Advancement of Artificial Intelligence
  - Fellow, Association for Computing Machinery
  - Fellow, Institute of Electrical and Electronics Engineers
  - ACM/SIGAI Autonomous Agents Research Award
  - IEEE TCSVC Research Innovation Award
  - IFAAMAS Influential Paper Award
  - Editor in Chief
    - ACM Transactions on Internet Technology, 2012–2018
    - IEEE Internet Computing, 1999–2002

My Goal and Request for Your Help

- Introduce you to deep concepts, some years in the making in the research and advanced development community
- Introduce you to critical thinking
- Boost your confidence in taking on technical challenges
  - You might hesitate to take on otherwise
  - Your peer group might find overwhelming
- Offer free advice (worth every penny$^{\text{SM}}$) about your
  - Education
  - Career
- How you can help
  - Don't take ethically dubious actions
  - Stay engaged
  - Communicate with me personally, especially about
    - Explanations and motivations
    - Improvements to the course, in general

Mechanics

- Scope
- Grading
- Policies
  - Especially, academic integrity
  - Don’t help; don’t take help; don’t collude

Bloom's Taxonomy of Learning Domains (Cognitive)

I emphasize the upper categories

Creating	Build new structures
Evaluating	Make judgments
Analyzing	Identify elements
Applying	Use on a problem
Understanding	State in own words
Remembering	Recall

- `http://www.nwlink.com/~donclark/hrd/bloom.html`

Scope of this Course

- Directed at computer science students
  - Non-CSC students with a strong humanities and social science background can do well—ask me
- Addresses foundational ideas of language and how to compute with them
  - Emphasizes concepts and theory
  - Involves tools in assignments
  - Involves discussions of challenges
- Requires a moderate amount of work
  - Fairly easy if you don't let your tasks slip

What Makes Human Languages Interesting?

- Connecting minds: how one person's thoughts reach into another's mind

- Gender assignment to words, explicit in some languages
- Even in English, think of pronouns and given names
  - Cat
  - Book
  - Faith
  - Hope

What Makes Human Languages Challenging?

- Sarcasm
- Versus logic
  - No no
  - Yeah yeah (Sidney Morgenbesser's famous retort to John L. Austin)
- Accommodation
- Interpretations shift to make sense
  - Beer is a mass noun (liquid), so we can't count it, but this works:
    Give me three beers
- Winograd schema (use of world knowledge)
  The trophy didn't fit in the suitcase because it was too big [small]

Applications of NLP

What makes NLP so valuable?

Brief Historical Look

- Ad hoc
- Inspired by cognitive science
- Knowledge-based
- Statistical
- Speech

Hierarchy of Language Concepts

Not to be taken too seriously

Discourse	Passage
Sentence	Assertion
Word	Unit of meaning
Morpheme	Meaning component
Phoneme	Language sound
Audio	Signal

- How would you pronounce *project*?
- Verb vs. noun

Language as a Symbolic System

Also called semiotics

Pragmatics	Meaning based on words and context
Semantics	Meaning based on syntax
Syntax	Structure of symbols
Symbol	Token (morpheme, phoneme, lexeme)

- Holy grail: to express meaning compositionally
  - Meaning of whole = combination of meanings of parts

Text Normalization

- Tokenization
  - Punctuation
  - Abbreviations
  - Number, date, email address, . . .
  - Clitics: not standalone, e.g., n’t
  - Case to mark names, e.g., mark vs. Mark
  - Hyphenated words
- Normalization
  - Case folding
  - Stemming: remove affixes
  - Porter stemming: popular but heavy-handed application of rules
  - Lemmatization: standard root, even if superficially different, e.g., $\{\text{am, is}\} \Rightarrow$ *be*
- Challenges
  - Scripts such as Chinese

Minimum Edit Distance

Illustration of dynamic programming

- Source string $X[n]$, prefixes $X[1..i]$, $i \in [1..n]$
- Target string $Y[m]$, prefixes $Y[1..j]$, $j \in [1..m]$
- Edit distance $D(i,j)$ between $X[1..i]$ and $Y[1..j]$
- $D(0,0) = 0$; for $i \in [1..n]$ and $j \in [1..m]$:

$$D(i,j) = \min \begin{cases} D(i-1,j) + \text{del-cost}(X[i]) \\ D(i,j-1) + \text{ins-cost}(Y[j]) \\ D(i-1,j-1) + \text{sub-cost}(X[i],Y[j]) \end{cases}$$

- Levenshtein values

$$D(i,j) = \min \begin{cases} D(i-1,j) + 1 \\ D(i,j-1) + 1 \\ D(i-1,j-1) + \begin{cases} 2 & X[i] \neq Y[j] \\ 0 & X[i] = Y[j] \end{cases} \end{cases}$$

- $D(n,m)$ is the answer; compute path from (n,m) back to $(0,0)$

Levenshtein Example

There (Source) ⇒ Their (Target)

		Target					
		0	1	2	3	4	5
Source		#	T	H	E	I	R
0	#						
1	T						
2	H						
3	E						
4	R						
5	E						