

CS 533: Natural Language Processing

General Introduction

Karl Stratos



Rutgers University

Modern Natural Language Processing (NLP)

NLP is everywhere



Other examples?

Short-Term Goals

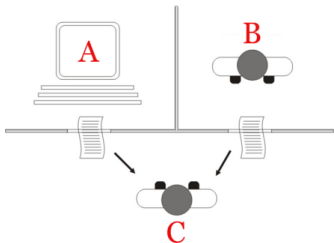
Make machines understand human language



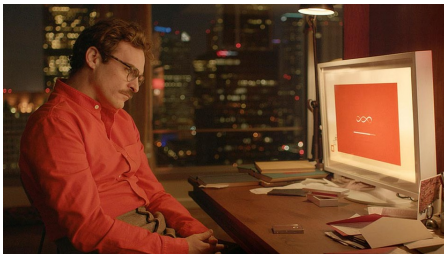
Countless applications: machine translation (MT), personal assistant, crucial component in any AI system (e.g., autonomous driving)

Long-Term Goals

Make machines as intelligent and conscious as humans (or more)



The Turing test



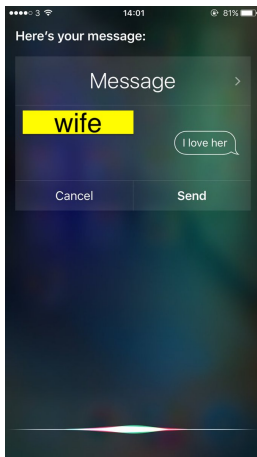
Her (2013)

Some History

- ▶ 1950: Alan Turing proposes the Turing test
- ▶ 1954: Georgetown–IBM experiment (rule-based MT)
 - “Within three or five years, machine translation will be a solved problem”
- ▶ 50-90s: focus on rule-based AI systems (e.g., SHRDLU)
- ▶ **From early 90s: Rise of statistical/data-driven NLP**
 - ▶ IBM: statistical MT and speech recognition
 - “Every time I fire a linguist, the performance of the speech recognizer goes up” -Fred Jelinek
 - ▶ UPenn/AT&T: statistical techniques for tagging and parsing
- ▶ 2011: IBM Watson wins *Jeopardy!* against human champions
- ▶ **From early 2010s: Rise of deep learning for NLP**
 - ▶ “Human-level” MT: The Great A.I. Awakening (*NYT*, 2016)
 - ▶ “Human-level” conversation: Google Duplex (2018)

Reality

“Hey Siri, tell my wife I love her”

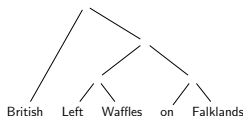
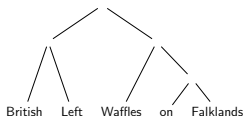


Why NLP is Hard: Ambiguity

Actual headline in *Guardian* (1982)

“British Left Waffles on Falklands”

- **Syntactic ambiguity**



- **Lexical ambiguity:** Every single word
- **Semantic ambiguity**



Why NLP is Hard: Nonsmoothness

A single word can completely change the meaning



Jack Black **Black Jack Black**



Black Jack Black playing BlackJack



**Jack Black Playing BlackJack
with BlackJack Black**

Why NLP is Hard: World Knowledge

Winograd (1972)

- ▶ The city councilmen refused the demonstrators a permit because **they** feared violence.
- ▶ The city councilmen refused the demonstrators a permit because **they** advocated violence.

NLP and Machine Learning (ML)

- ▶ **(Supervised) ML:** Extract patterns from past observations that can generalize to future observations
 - ▶ Contrasts with rule-based approaches (manually specify patterns based on human expertise)
- ▶ **Training data:** Bunch of examples of (**input**, **output**)
 - ▶ Classification: Output is discrete (one of k objects).
 - ▶ Regression: Output is continuous (e.g., 3.141592).
- ▶ **Model:** Mapping **input** \mapsto **output** that can be applied to any input (e.g., outside the training data)
- ▶ **Generalization:** The model must make correct predictions to all future unseen inputs!
- ▶ Can NLP be framed as an application of ML?

NLP as a Classification Problem

text \mapsto **expected human response**

NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output
failed to not disappoint	-1 (sentiment)


NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output	
failed to not disappoint	-1	(sentiment)
the dog saw the cat	D N V D N	(POS tagging)
	<pre>graph TD PRED --> DET1[DET] PRED --> SBJ[SBJ] PRED --> OBJ[OBJ] DET1 --> the1[the] SBJ --> dog[dog] OBJ --> DET2[DET] OBJ --> cat[cat] DET2 --> the2[the]</pre>	(parsing)

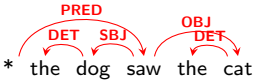
NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output	
failed to not disappoint	-1	(sentiment)
the dog saw the cat	D N V D N	(POS tagging)
		(parsing)
산을 갔다	I went to the mountain	(translation)

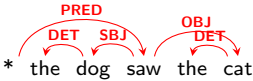
NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output	
failed to not disappoint	-1	(sentiment)
the dog saw the cat	D N V D N	(POS tagging)
		(parsing)
산을 갔다	I went to the mountain	(translation)
SONOMA, Calif. — Wine country was shrouded in a thick layer of smoky haze here on Tuesday as firefighters continued to battle wildfires that have left at least 13 people dead and have damaged or destroyed more than 1,500 structures.	Wildfires sweep across northern California; 13 dead	(summarization)

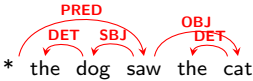

NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output	
failed to not disappoint	-1	(sentiment)
the dog saw the cat	D N V D N	(POS tagging)
		(parsing)
산을 갔다	I went to the mountain	(translation)
SONOMA, Calif. — Wine country was shrouded in a thick layer of smoky haze here on Tuesday as firefighters continued to battle wildfires that have left at least 13 people dead and have damaged or destroyed more than 1,500 structures.	Wildfires sweep across northern California; 13 dead	(summarization)
Who was the richest man in 2020?	Jeff Bezos	(QA)


NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output	
failed to not disappoint the dog saw the cat	-1 D N V D N 	(sentiment) (POS tagging) (parsing)
산을 갔다	I went to the mountain	(translation)
SONOMA, Calif. — Wine country was shrouded in a thick layer of smoky haze here on Tuesday as firefighters continued to battle wildfires that have left at least 13 people dead and have damaged or destroyed more than 1,500 structures.	Wildfires sweep across northern California; 13 dead	(summarization)
Who was the richest man in 2020?	Jeff Bezos	(QA)
Trump spent years pushing the untrue “birther” claim that the nation’s first black president was not born in the U.S.	 <p>Trump spent years pushing the untrue “birther” claim that the nation’s first black president was not born in the U.S.</p>	(linking)

NLP as a Classification Problem

text \mapsto **expected human response**

Input	Output	
failed to not disappoint the dog saw the cat	-1 D N V D N * the dog saw the cat	(sentiment) (POS tagging) (parsing)
산을 갔다	I went to the mountain	(translation)
SONOMA, Calif. — Wine country was shrouded in a thick layer of smoky haze here on Tuesday as firefighters continued to battle wildfires that have left at least 13 people dead and have damaged or destroyed more than 1,500 structures.	Wildfires sweep across northern California; 13 dead	(summarization)
Who was the richest man in 2020?	Jeff Bezos	(QA)
Trump spent years pushing the untrue “birther” claim that the nation’s first black president was not born in the U.S.	Trump spent years pushing the untrue “birther” claim that the nation’s first black president was not born in the U.S. 	(linking)
Open the pod bay doors, HAL.	I'm sorry, Dave. I'm afraid I can't do that.	(dialogue)

Classification as an Optimization Problem

- **Optimization:** Given possible answers \mathcal{X} , find one x^* that minimizes some “loss” function $f : \mathcal{X} \rightarrow \mathbb{R}$

$$x^* = \underset{\substack{x \in \mathcal{X}: \\ \text{constraint}(x) = \text{True}}}{\text{arg min}} \quad f(x)$$

(Equivalently maximization of “reward” $-f(x)$)

Classification as an Optimization Problem

- **Optimization:** Given possible answers \mathcal{X} , find one x^* that minimizes some “loss” function $f : \mathcal{X} \rightarrow \mathbb{R}$

$$x^* = \underset{\substack{x \in \mathcal{X}: \\ \text{constraint}(x) = \text{True}}}{\text{arg min}} \quad f(x)$$

(Equivalently maximization of “reward” $-f(x)$)

- **Training a classifier:** Find **model parameters** that minimize a **loss function on training data**

Classification as an Optimization Problem

- **Optimization:** Given possible answers \mathcal{X} , find one x^* that minimizes some “loss” function $f : \mathcal{X} \rightarrow \mathbb{R}$

$$x^* = \underset{\substack{x \in \mathcal{X}: \\ \text{constraint}(x) = \text{True}}}{\text{arg min}} \quad f(x)$$

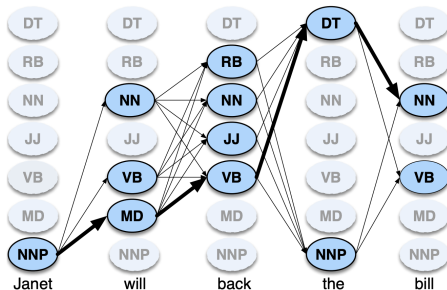
(Equivalently maximization of “reward” $-f(x)$)

- **Training a classifier:** Find **model parameters** that minimize a **loss function on training data**
- **Making prediction:** Find a valid **label** that maximizes the **chance of being correct**
 - **Unstructured label:** One of k choices (e.g., what is the topic of this document?)
 - **Structured label:** One of $O(2^k)$ choices (e.g., what is a Hindi translation of this sentence?)

Structured Prediction: Highlight of NLP

Labels: sequences/trees rather than one-of- k items

- ▶ Impossible to enumerate all $O(\text{length}^{\text{vocab}})$ translations!
- ▶ Solutions: approximation (beam search), exact search by dynamic programming (Viterbi, CKY)



(Example from Jurafsky and Martin)

Supervised ML Pipeline

- ▶ **Problem definition:** Define a supervised learning problem.
- ▶ **Data collection:** Start with training data for which we know the correct outcome.
- ▶ **Representation:** Choose how to represent the data.
- ▶ **Modeling:** Choose a **hypothesis class** – a set of possible explanations for the connection between input (e.g., image) and output (e.g., class label).
- ▶ **Estimation:** Find best hypothesis you can in the chosen class. This is what people usually think of as “learning”.
- ▶ **Model selection:** If we have different hypothesis classes, we can select one based on some criterion.

Example: Topic Classification

- Classify a document $x \in \mathcal{X}$ to a topic $y \in \mathcal{Y}$

$$\mathcal{Y} = \{\text{business, politics, sports, ...}\}$$

- Training data $D = \{(x_1, y_1) \dots (x_N, y_N)\}$: real-world documents labeled with correct topics
- Each document represented as a binary vector $x \in \{0, 1\}^d$ indicating which words are present (“bag-of-words”)
- Hypothesis class: linear classifier. Parameters $w_y \in \mathbb{R}^d$ for $y \in \mathcal{Y}$. Given any x , predict

$$y^*(x) = \arg \max_{y \in \mathcal{Y}} w_y^\top x$$

- **Learning**: Fit $\theta = \{\hat{w}_y\}_{y \in \mathcal{Y}}$ by minimizing some $\text{loss}_D(\theta)$
- Try other models (e.g., decision tree), deploy one that yields highest accuracy on some held-out evaluation data

Example: Machine Translation

- ▶ Translate Arabic sentence $x \in \mathcal{X}$ to English sentence $y \in \mathcal{Y}$

\mathcal{Y} = All possible English sentences

- ▶ Training data $D = \{(x_1, y_1) \dots (x_N, y_N)\}$: real-world translations (e.g., UN documents)
- ▶ Each sentence represented as a list of vectors $u_1 \dots u_n \in \mathbb{R}^d$
- ▶ Hypothesis class: transformers. Large number of parameters θ . Given any x , use beam search to predict

$$y^*(x) \approx \arg \max_{y \in \mathcal{Y}} \Pr(y|x, \theta)$$

- ▶ **Learning**: Fit θ by minimizing some $\text{loss}_D(\theta)$
- ▶ Try other models (e.g., LSTMs), deploy one that achieves highest translation score on some held-out evaluation data

Scope of the Course

Space of models in machine learning

- ▶ Decision trees
- ▶ Kernel machines (aka. support vector machines)
- ▶ Differentiable functions (aka. neural networks)
- ▶ Many others!

Space of learning methods

- ▶ Combinatorial optimization
- ▶ Method of moments
- ▶ (Stochastic) Gradient descent
- ▶ Many others!

Scope of the Course

Space of models in machine learning

- ▶ ~~Decision trees~~
- ▶ ~~Kernel machines (aka. support vector machines)~~
- ▶ Differentiable functions (aka. neural networks)
- ▶ ~~Many others!~~

Space of learning methods

- ▶ ~~Combinatorial optimization~~
- ▶ ~~Method of moments~~
- ▶ (Stochastic) Gradient descent
- ▶ ~~Many others!~~

We will primarily focus on deep learning.

Goals of the Course

1. Understanding the goals, capabilities, and principles of NLP
2. Acquiring mathematical tools to formalize NLP problems
3. Acquiring implementation skills to build practical NLP systems
4. Obtaining an ability to critically read and accurately evaluate conference papers in NLP

This is largely an engineering course, but we will need a nontrivial mathematical background to solve problems.

Tentative Syllabus

- ▶ Topic classification
 - Linear classification, score function, softmax, loss function, stochastic gradient descent, regularization
- ▶ Deep learning background
 - Nonlinearity/universality of neural networks, backpropagation
- ▶ Sequence-to-sequence models
 - Chain rule, perplexity, training and beam search
- ▶ Structured prediction
 - Markov random fields, conditional random fields, Viterbi, CKY
- ▶ Self-supervised representation learning
 - Word embeddings, contextual word embeddings
- ▶ Unsupervised learning
 - Latent-variable models, expectation maximization, VAEs
- ▶ Special topics on various applications
 - Information extraction, QA, text generation

Course Format

Flipped classroom. Two independent tracks in parallel

1. Weekly lecture videos/slides (asynchronous)
 - ▶ Already available! But pace yourself: no need to “study ahead”. Assignments/quizzes on covered materials only
 - ▶ Slides are self-contained. There are optional reading recommendations (all publicly available).

Course Format

Flipped classroom. Two independent tracks in parallel

1. Weekly lecture videos/slides (asynchronous)
 - ▶ Already available! But pace yourself: no need to “study ahead”. Assignments/quizzes on covered materials only
 - ▶ Slides are self-contained. There are optional reading recommendations (all publicly available).
2. In-class paper discussion (1:00-2:30pm every Wednesday, tentatively)
 - ▶ Discuss a recent research paper together (chosen from a provided list)
 - ▶ **Student-led:** a group of up to 3 students (single person is fine) leads the discussion
 - ▶ Need to present and discuss *as if* we have already covered the background

Tips on Reading a Research Paper

- ▶ Answer the following 5 questions ([Widom](#))
 1. What is the problem?
 2. Why is it interesting and important?
 3. Why is it hard? (E.g., why do naive approaches fail?)
 4. Why hasn't it been solved before? (Or, what's wrong with previous proposed solutions?)
 5. What are the key components of the approach and results?
Also include any specific limitations.
- ▶ This requires understanding previous works (and *their* previous works, etc.) by literature review
 - ▶ If you lack the background, you must look up the paper(s) that it builds on and understand that first
 - ▶ Picking up the background to understand a single paper can easily take a week
- ▶ Read nonlinearly: While it's important to understand the motivation, don't get hung up on introductions. Focus on the method/data/technical parts
 - ▶ Numbers/tables/plots are far more important than words

Expectations in Paper Discussion

- ▶ The discussion leader(s) are expected to **guide** the class through the paper
 - ▶ Explain what the problem is, explain related works, explain the paper
 - ▶ Majority of time should be spent on technical elements
 - ▶ You can either make slides or just go through the paper together.
- ▶ Examples of “technical elements”
 - ▶ The paper develops a novel method B in a general framework A (e.g., latent-variable models, sequence-to-sequence learning): give backgrounds on A and derive B from A.
 - ▶ The paper defines a new problem and provides a new dataset. Qualitatively examine examples of the data and make observations. Give statistics and understand the scale of the dataset.
 - ▶ For an empirical paper, it's recommended that you download the data/model (e.g., from GitHub) and get a feel for it.
 - ▶ The best way to understand a paper, if you have time, is to replicate the results from scratch (your final project).

Prerequisites and Grading

Prereqs

- ▶ **Mathematical maturity** to use tools in linear algebra, calculus, and probability
- ▶ Entrance quiz: 1/19 (Wed) 2:30-3:10pm
- ▶ For undergrads: M250, 112, 206 (recommended: M251, 461)
- ▶ **Programming skills** (we will use Python)

Grading

- ▶ Assignments (3–4): 40%
- ▶ Project: 30%
- ▶ Quizzes (including the entrance quiz): 20%
- ▶ Paper discussion: 10%

Project

More info later in the semester, but for now you can expect to

- ▶ Submit a **proposal** (about a month before the end)
- ▶ Submit a **final report** and a **video presentation**

For the project

- ▶ Select a recent conference paper (ACL, EMNLP, NAACL): must be approved by me
- ▶ Replicate from scratch, and possibly build on its results
- ▶ Collaboration permitted up to 3 people

Assignments

- ▶ 3–4 assignments, each a mix of written problems and coding in Python
- ▶ You may collaborate as long as you (1) write your own solution entirely on your own, and (2) specify names of student(s) you collaborated with in your writeup.

All writeups (project, assignments) must be written in **LaTeX**

- ▶ We will provide a LaTeX template for you to use.
- ▶ No exception

Course Staff

- ▶ Instructor: Karl Stratos (office hours 4-5pm Wednesday)
- ▶ TA: Wenzheng “Vincent” Zhang (office hours 11-12pm Thursday)
- ▶ Grader: SohailAbbas “Sohail” Saiyed