Natural Language Processing

Adam Faulkner

Aug 26, 2025

1. Introduction
2. History of NLP
3. Course structure and content

1. Introduction
2. History of NLP
3. Course structure and content

About Me

- Currently Senior Manager, Data Science at Capital One
- Began teaching a course on Generative AI at Katz in Fall 2024
- Previously at IBM Research, Grammarly, and ETS
- Ph.D. CUNY Graduate Center, 2014

Core NLP Tasks (not exhaustive)

- The "Natural" in NLP:
  - Natural languages, such as English and Mandarin, are used in everyday discourse and have evolved organically. They have fuzzily defined rules and are ambiguity-laden.
  - Formal languages such as first-order-logic and Python have well-defined rules and are unambiguous
- NLP is an engineering discipline: The goal is to get computers to perform useful tasks involving human language—enabling human-machine communication, improving human-human communication, or simply doing useful processing of text.

Why NLP is hard

- To see why this is challenging, let's consider the following user request and the system response from a functioning conversational agent (from the film 2001) *HAL*:

```
User:Open the pod bay doors please HAL.
HAL: I'm sorry Dave, I'm afraid I can't do that
```

- What did HAL need to understand about natural language in order to properly respond?

Natural Language Understanding: Syntax

- HAL must use structural knowledge to properly string together the words that constitute its response. For example, HAL must know that the following sequence of words will not make sense to Dave, despite the fact that it contains precisely the same set of words as the original.

  `I’m I do,sorry that afraid Dave I’m can’t.`

- The knowledge needed to order and group words together comes under the heading of **syntax.**

Natural Language Understanding: Lexical Semantics

- HAL also needs to understand the meaning of words such as *doors*. Specifically, he must understand that *doors* are things that open and close – this is the word's *lexical frame*.
- Knowledge of word meaning involves knowledge of **lexical semantics**

Natural Language Understanding: Commonsense Reasoning

- HAL also has to somehow understand that a request to open a set of doors implies that the doors are currently closed – nothing in the request itself indicated this.
- World-knowledge of this sort falls under the heading of **commonsense reasoning**.

Natural Language Understanding: Discourse

- The user used “please” when making their request. HAL should reciprocate by also using a polite *register*: "I’m sorry Dave," etc."
- These are the implicit rules of **discourse** governing the interaction

Natural Language Understanding: Pragmatics

- HAL needs to understand that the imperative *Open the pod bay doors please HAL* is a request for action – this knowledge falls under the heading of **pragmatics**
- Additionally, as we learn later in the film, HAL is capable of opening the door but *won't* and could have simply replied *No* or *No, I won't open the door.* Instead, it first embellishes its response with the phrases *I'm sorry* and *I'm afraid*, and then only indirectly signals its refusal by saying *I can't*, rather than the more direct (and truthful) *I won't.*
- Again, this knowledge of the kinds of actions speakers intend by phrasing their sentences one way rather than another is an example of **pragmatic** knowledge.

Ambiguity: A core problem in NLP

- A surprising fact about these categories of linguistic knowledge is that most tasks in NLP can be viewed as resolving ambiguity at one of these levels.
- Consider the sentence *I made her duck*. Here are a few possible interpretatons:
  1. I cooked waterfowl for her. (*d*
  2. I cooked waterfowl belonging to her. (*her* is ambiguous)
  3. I created the (plaster?) duck she owns. (*make* is ambiguous)
  4. I caused her to quickly lower her head or body. (*make* is ambiguous)

Core NLP Tasks (not exhaustive)

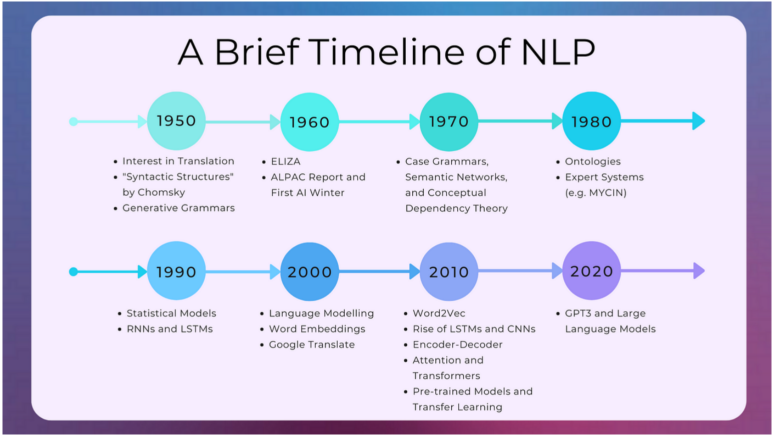
- Commonsense Reasoning
- Constituency parsing
- Coreference resolution
- Dialogue
- Grammatical error correction
- Information extraction
- Intent Detection and Slot Filling
- Language modeling
- Machine translation
- Named entity recognition
- Part-of-speech tagging
- Paraphrase Generation
- Question answering
- Relationship extraction
- Semantic textual similarity
- Semantic parsing
- Semantic role labeling
- Sentiment analysis
- Text Simplification
- Summarization
- Taxonomy learning
- Text classification

1 Introduction

2 History of NLP

3 Course structure and content

History of NLP



A Brief Timeline of NLP

1950
- Interest in Translation
- "Syntactic Structures" by Chomsky
- Generative Grammars

1960
- ELIZA
- ALPAC Report and First AI Winter

1970
- Case Grammars, Semantic Networks, and Conceptual Dependency Theory

1980
- Ontologies
- Expert Systems (e.g. MYCIN)

1990
- Statistical Models
- RNNs and LSTMs

2000
- Language Modelling
- Word Embeddings
- Google Translate

2010
- Word2Vec
- Rise of LSTMs and CNNs
- Encoder-Decoder
- Attention and Transformers
- Pre-trained Models and Transfer Learning

2020
- GPT3 and Large Language Models

History of NLP: 1940’s - 1950’s

- Post-WWII, funded by the US military, focused on developing Machine Translation systems for the automated translation of Soviet scientific articles
- **McCulloch-Pitts neuron** : A simplified model of the neuron as a kind of computing element that could be described in terms of propositional logic
- Shannon applied probabilistic models of discrete Markov processes to **automata for language**.
- Drawing on the idea of a finite-state Markov process from Shannons work, Chomsky first considered finite-state machines as a way to characterize a grammar, and defined a finite-state language as a language generated by a finite-state grammar. These early models led to the field of **formal language theory**

History of NLP: 1957 - 1970

- Symbolic, rules-based AI
- **Chomsky's generative syntax** and other parsing algorithms – top-down and bottom-up and then parsing via dynamic programming
- First use of the stochastic paradigm: Bledsoe and Browning (1959) built a **Bayesian system for text-recognition** that used a large dictionary and computed the likelihood of each observed letter sequence given each word in the dictionary by multiplying the likelihoods for each letter.
- Mosteller and Wallace (1964) applied Bayesian methods to the problem of **authorship attribution** of The Federalist papers.
- 1963-64: **Brown corpus of American English,** a 1 million word collection of samples from 500 written texts from different genres (newspaper, novels, non-fiction, academic, etc.)

History of NLP: 1970 - 1983

- Three paradigms: *stochastic, logic-based, natural language understanding-based*
- **Stochastic**: Hidden Markov Model and the metaphors of the noisy channel and decoding, developed by Jelinek, Bahl, Mercer, and colleagues at IBM
- **Logic-based**: Logical programming languages such as Prolog
- **Natural Language Understanding**: Terry Winograds SHRDLU system, which simulated a robot embedded in a world of toy blocks. The program was able to accept natural language text commands ("Move the red block on top of the smaller green one") of a hitherto unseen complexity and sophistication.

History of NLP: 1983 - 1999

- Rise of probabilistic models throughout speech and language processing, influenced strongly by the **work at the IBM on probabilistic models of speech recognition and machine translation**
- Algorithms for parsing, part-of-speech tagging, reference resolution, and discourse processing all began to incorporate probabilities, and employ evaluation methodologies borrowed from speech recognition and information retrieval
- **Increases in the speed and memory of computers** had allowed commercial exploitation of a number of subareas of speech and language processing, in particular speech recognition and spelling and grammar checking.

History of NLP: 1999 - 2013

- Dominance of ML-based approaches: **Maximum Entropy Estimation, Naive Bayes, Logistic Regression, SVMs**
- Datasets: **Penn Treebank, PropBank, Penn Discourse Treebank.** These datasets layered standard text sources with various forms of syntactic, semantic and pragmatic annotations. The existence of these resources promoted the trend of casting more complex traditional problems, such as parsing and semantic analysis, as problems in supervised machine learning.
- Widespread availability of high-performance computing systems facilitated the training and deployment of NLP models
- **Rise of unsupervised approaches:** Statistical approaches to machine translation and topic modeling demonstrated that effective applications could be constructed from systems trained on unannotated data alone. This directly led to the shift toward language model-based NLP.

History of NLP:2013 - present

- **Deep Learning and GPUs**: RNNs and LSTMs show massive performance gains relative to traditional ML approaches leading to the Deep Learning revolution (all made possible by the commercial availability of GPU chips)
- **word2vec**, **GloVE**
- **Transformers**, small LMs, **BERT**, the finetuning paradigm
- **GPT2,** NLP tasks recast as word prediction tasks
- **GPT3**, first LLMs, post-training regimes: RLHF, DPO
- **ChatGPT, RAG, Agents**

1 Introduction

2 History of NLP

3 Course structure and content

Course structure: Part 1

- Review of classic approaches to NLP tasks using supervised ML approaches such as Naive Bayes and Logistic Regression–this roughly covers the techniques in the slide *History of NLP: 1999 - 2013.* (These techniques aren't simply historical curiosities: Logistic Regression, softmax, and a general understanding of neural networks are all relevant to an understanding of the Deep Learning architecture that currently dominates AI, the Transformer)
- N-gram language modeling, sparse vector-based representations of text such as bag-of-words
- Deep Learning-based modeling techniques and vector representations such as LSTMs and dense-vector representations.
- The Transformer and language-model-based solutions to NLP tasks via frameworks such as finetuning.

Course structure: Part 2

- Review of traditional NLP tasks such as classification (Sentiment Analysis, Intent Detection, etc.), sequence labeling (Named Entity Recognition, syntactic and semantic parsing, etc.), and text generation (Machine Translation, Question-Answering, Summarization, etc.).
- This section ends with a review of text generation tasks. This is intentional since, in contemporary NLP, all tasks, including classification and sequence labeling, are now framed as text generation tasks

Course structure: Part 3

- Large Language Models (LLMs), the dominant paradigm of contemporary NLP
- Reframing of NLP tasks described in section 2, which were traditionally solved using the ML techniques described in section 1, as word-prediction tasks.
- Autoregressive language modelling objective, post-training regimes such as Reinforcement Learning from Human Feedback
- Retrieval Augmented Generation (RAG), Agents, LLM-as-a-Judge, LLM interpretability, and hallucination detection
- Additional topics: Automated detection of LLM-generated text, What do LLMs really understand?

Course content: Github repo

- In addition to Canvas, all course content can be accessed in the course Github repo.

Course content: Slides and readings

- The bulk of the course content consists of the lecture slides and the week’s reading material (textbook chapters and representative technical papers)

Course content: Juptyer notebooks

- Each lecture has an accompanying notebook that will be run live (with instructor commentary) during the class. These notebooks can also be run before or after class to get a better understanding of the code and the lecture material

Assignments

- Five assignments that delve into the math underlying the concepts presented in the lectures. These assignments must be printed, completed by hand, and submitted online (via scan or photo) or handed in to me directly
- Note that most of these assignments contain calculations that we'll walk through together during the class – if you take good notes/screenshots during these sessions you should be able to complete these assginments quickly
- Group assignment. The final group project can be completed singly or as a group (max 5 members). The project should consist of an NLP application that implements one or more of the concepts described in class. The application will be demoed on the last day of class as a 5-10 minute presentation

Next class: Sept 2

Topics

- Naive Bayes
- Logistic Regression
- The bag-of-words vector representation

Reading

- Jurafsky & Martin Chapter 4: Naive Bayes, Text Classification, and Sentiment
- Jurafsky & Martin Chapter 5: Logistic Regression