Intro Lecture: Accompanying Notes

Noah A. Smith
CSE 447 and CSE 517: Natural Language Processing – University of Washington

Winter 2022

What do we mean by "natural language"? The word *language* has a formal definition in CS (a set of strings over some finite alphabet Σ), and an everyday meaning. When we talk about natural languages, we tend to refer to abstractions like *English* and *Korean*. This terminology is not perfect; there is considerable variation among speakers of any one of these "languages" and each speaker uses their language differently in different social contexts. Languages also change—sometimes drastically—over time. And what about communities of speakers who mix different languages together when they communicate? Complicating things further are the *modalities* in which language is used: some languages are spoken (and some of those can also be written), some are signed. This class focuses just on language in digitized, textual form (e.g., sequences of Unicode characters).

There is no "spec" for natural languages. Language has emerged among our species. Whenever you hear about "rules" of a language, you should remember that these are post hoc. In some cases the rules are *prescriptive*, telling people how they *should* use language. In other cases, they are *descriptive*, trying to explain how people *do* use language. Linguists tend to be most interested in eliciting the latter; what are the rules that explain what language users in a community will say and mean, to what extent are these rules universal (or specific to particular communities), and to what extent are they innate (or learned)? This is in stark contrast to programming languages, whose syntax and semantics are strictly and deliberately defined. Whatever rules we ascribe to a natural language, they are always conjectures, subject to argument and revision as new evidence emerges.

Natural language *processing* is about automating analysis and generation of text. Sometimes people use the word *understanding* to refer to what I call *analysis*: mapping a textual input into a data type that captures some aspects of its meaning (referred to in the lecture as "$\mathcal{R}$"). NLP systems are derived jointly from (1) text data (which comes from human language users) and (2) their human designers, who know about language, its use, and the desired system behavior. A major ongoing debates in the field of NLP is:

- How much should NLP system development rely on language data vs. human knowledge about language? Specifically, is $\mathcal{R}$ something to be designed by us, or will it emerge from the data alone, through sophisticated automation?

This debate is closely related to a larger philosophical debate between *rationalism* and *empiricism*. The most successful NLP researchers and practitioners understand and draw from both perspectives.

A pragmatic view: start from the application. Once we know what we want the final system to do, we can focus on getting *enough* information about language into the system. Note that "knowing" a language is not a binary property: you can know "a little bit" of German, and small children are in the process of aquiring their first language. You probably don't even know "all" of your first language (e.g., words you

haven't encountered yet because they're too new or too old). The ways in which your computer program can partially "know" a language are probably different from the ways in which humans partially know languages.

We teach a whole class on NLP because it's freaking hard. Even simple ideas like "words" are non-trivial when you start looking at real language. Examples:

- Segmenting text into words (e.g., Thai example in lecture)
- Some languages have intricate systems of word formation, leading to huge numbers of different surface forms, too many to list (e.g., Turkish and Hebrew examples in lecture)
- Words with multiple meanings: *bank*, *mean*
- Context-specific meanings: *latex*
- Multiword expressions where multiple words combine to mean something that's not merely the "sum of the parts": *make a decision*, *take out*, *make up*, *bad hombres*
- New words (e.g., *covid-19*) and changing meanings (e.g., *bagel* is now a verb)

Even if we can recognize all the words and "know" all of their meanings, more ambiguity emerges when we put them together into sentences and longer texts. (*Ambiguity* refers to the situation where a piece of text has more than one possible meaning.) For many kinds of ambiguity, resolution requires humans to use context, common sense, or world knowledge. A few examples were given in the lecture.

It's also difficult to know what we need NLP to do, exactly. There is no single definition of "meaning" that will satisfy every application builder (let alone every philosopher or linguist!). In some cases, we might be able to formalize meaning using tools from math and CS: giving commands to a robot with a relatively narrow set of capabilities, or querying an already-defined database (cases where the system only needs to reason about a closed, grounded world). But language also gets used to express opinions, propose policies, explore scientific hypotheses, talk about hypothetical events, and much more. Formalizations start to break down when we get into these more "human" aspects of language, and when we start considering non-literal meaning.

Language is impossibly rich. The converse problem of ambiguity is that a given meaning can be expressed through countless different choices of language. The choices we make in how we convey an idea in language sometimes add shades of meaning, sometimes don't, and often different listeners might disagree about those shades of meaning! Further, we can use language to talk about a huge range of things, from politics to science to lunch. Finally, there is always**variation** in how a language is used, which often aligns with aspects of a speaker's and listener's identities. There's been a lot of recent discussion about how choices in the design of an NLP system might affect how well it serves different users (e.g., speakers of different dialects).

Doing NLP is always a balancing act. Key desiderata for NLP systems (in no particular order):

1. Sensitivity to a wide range of the phenomena and constraints in human language
2. Generality across different languages, genres, styles, and modalities
3. Computational efficiency at construction time and runtime
4. Strong formal guarantees (e.g., convergence, statistical efficiency, consistency, etc.)

5. High accuracy when judged against expert annotations and/or task-specific performance
6. Explainable to human users

Despite what you may read in the popular press, NLP systems today do not deliver all of the above.

NLP is not (just) machine learning. ML is about building programs from examples. It is used ubiquitously in NLP and the two fields have a lot of common roots (e.g., 1940s information theory). To be successful, a machine learner needs bias/assumptions; for NLP, that might be linguistic and/or domain theory/representations. Supervised learning for NLP tends to require manual data creation. Symbolic, probabilistic, and connectionist ML have all seen NLP as a source of inspiring applications, though arguably computer vision is more widely considered the "default" application area. If you haven't taken an ML course, you'll learn some basics here. If you have, notice how NLP's demands of ML are different from other kinds of applications.

NLP is not (just) linguistics. Linguistics is about how language works, e.g.:

- How are languages related to each other? How and why do they change?
- What principles explain grammaticality of sentences and the mapping between strings and meanings; are they shared across languages? Universally?
- How do people learn their first language? Their second?

NLP and linguistics pay attention to each other but are not identical. To be useful in the real world, NLP must contend with natural language data as it is found in the world; depending on the research question, linguists might or might not need to do that. NLP is often equated to **computational linguistics**, i.e., linguistics carried out using computational techniques. NLP inventions sometimes get used by linguists in their work.

NLP is a key part of artificial intelligence. AI aims to automate human mental capacities. Language is a fundamental part of human mental function, and the only tool we have for communicating about much of what we think about. Therefore, natural language may be our best bet for solving the knowledge bottleneck. Example:

- The cat doesn't fit in the box because **it** is too big.
- The cat doesn't fit in the box because **it** is too small.

This means NLP is linked to fields that study other parts of intelligence, such as computer vision!

Doing NLP requires us to consider ethical questions. Who's using the NLP system? Whose language does it model? How do social biases expressed in language data get built into systems that use those data? What is the role of NLP technology in free expression and in surveillance?

NLP is changing fast. A few years ago, Hirschberg and Manning [1] noted some trends: Increases in computing power; the rise of the web, then the social web; advances in machine learning; advances in understanding of language in social context. Since then, the field has been heavily influenced as well by consumer and investor demand, and by emerging ethical questions around deployment.

Your instructor's approach to teaching NLP.

- Application tasks are difficult to define formally and are always evolving, so I focus on useful **abstractions** (with examples).
- Objective evaluations of performance are always up for debate, so I discuss shortcomings and emphasize **tradeoffs**.
- Different applications require different $\mathcal{R}$, so I encourage **openmindedness**.
- People who succeed at NLP for long periods of time understand many tools and how they relate to each other.
- This class doesn't teach you how to solve any one particular problem; it gives you tools that will help you understand and tackle new problems.

References

[1] Julia Hirschberg and Christopher D. Manning. Advances in natural language processing. *Science*, 349 (6245):261–266, 2015. URL https://www.sciencemag.org/content/349/6245/261.full.