Natural Language Understanding, Generation, and Machine Translation

Lecture 1: Introduction

Alexandra Birch, Frank Keller, and Laura Perez

Week of 17 January 2022 (week 1)

School of Informatics
University of Edinburgh
a.birch@ed.ac.uk,{keller,lperez}@inf.ed.ac.uk

Overview

Reading: Manning (2015).

Introduction

What is Natural Language Understanding?

Application: Digital assistants

Input: A command in natural language

Output: Code to carry out that command

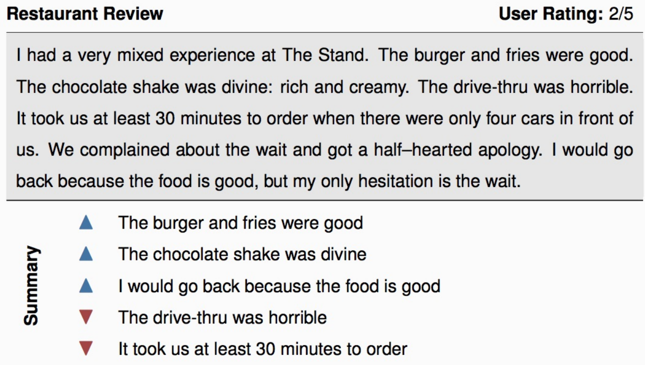
What is Natural Language Understanding?

Application: Question answering

Input: A question in natural language

Output: A natural language answer to that question

What is Natural Language Understanding?



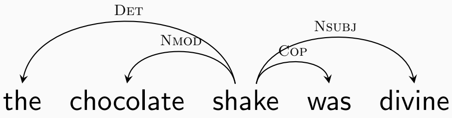
Source: Stefanos Angelidis, *Weakly Supervised Sentiment Analysis and Opinion Extraction*, 2019 PhD thesis

Application: Sentiment analysis

Input: Natural language sentence

Output: Classification of sentence as positive, negative, or neutral towards its subject

What is Natural Language Understanding?



Core NLP problem: syntactic parsing

Input: A natural language sentence

Output: A dependency analysis of the sentence

What is Natural Language Understanding?

What countries border France?

$$\lambda x.\mathrm{country}(x) \cap \mathrm{borders}(x, \mathrm{France})$$

Core NLP problem: semantic parsing

Input: A natural language sentence

Output: A logical form expressing the meaning of that sentence

Summary of Natural Language Understanding

Broadly: any computational problem where the *input* is natural language, and the *output* is structured information that a computer can store (e.g. in a database) or execute (e.g. a command to a digital assistant).

NLU often requires a system to resolve (either implicitly or explicitly) many of the same question that human readers efficiently resolve when they read a text. *Who is doing what to whom, and when?* The answers depend on a variety of cues from morphology, syntax, semantics, discourse, and inferences based on common sense.

Text $\Longrightarrow$ Analyses (syntactic relationships, logical forms, named entities, coreference, user intents, sentiment, etc.)

What is Natural Language Generation?

Pollen Concentration Data for Scotland

AreaID	Value
1 (North)	6
2 (North West)	5
3 (Central)	5
4 (North East)	6
5 (South West)	8
6 (South East)	8

Monday looks set to bring another day of relatively high pollen counts, with values up to a very high eight in the Central Belt. Further North, levels will be a little better at a moderate to high five to six. However, even at these lower levels it will probably be uncomfortable for Hay fever sufferers.

Application: Data-to-text generation

Input: Structured data (e.g. database tables)

Output: A natural language description of that data

What is Natural Language Generation?

Two small dogs run through the grass.

Application: Image captioning

Input: Image

Output: A natural language description of that image

Summary of Natural Language Generation

Broadly: any computational problem where the *input* is non-linguistic data (e.g. data, images, sound) and the *output* is a natural language description of the input.

Non-linguistic input (logical forms, database entries, images, etc.) $\Longrightarrow$ Text

What about tasks where *both the input and output are text?*

What is Machine Translation?

Så varför minskar inte vi våra utsläpp?

So, why are we not reducing our emissions?

Example: Swedish-English machine translation

Input: A sentence in Swedish

Output: A sentence in English expressing the same meaning

What is Machine Translation?

Owls are the order Strigiformes, comprising 200 bird of prey species.

An owl is a bird. There are about 200 kinds of owls.

Application: English text simplification

Input: A sentence in English

Output: Sentences in basic English expressing the same meaning

What is Machine Translation?

Doing some traveling this year and I am looking to build the ultimate travel kit . . . So far I have a Bonavita 0.5L travel kettle and AeroPress. Looking for a grinder that would maybe fit into the AeroPress. This way I can stack them in each other and have a compact travel kit.

TL;DR: What grinder would you recommend that fits in AeroPress?

Application: summarization

Input: A paragraph or document

Output: A sentence that summarizes the key content of the input

Summary of Machine Translation

Both *input* and *output* are text that convey the same meaning, but written in a different language or style.

Philosophically and technically, machine translation requires both NLU and NLG.

Introduction to the NLU+ Course

Central Question of the Course

Suppose your goal is to implement an NLG, NLU, or MT system.

For concreteness, suppose it is Swedish-English translation.

Q1: How would you write a function to translate Swedish to English?
Q1: How should we deal with ambiguity?
Q2: How should we deal with morphosyntactic differences?
Q3: How should we deal with . . .

Suppose I give you many examples of Swedish-English translation:

Q: How can we *learn* a function to translate Swedish to English?

Overall Objective of the Course

NLU+ covers advanced machine learning methods for functions whose input and/or output is natural language.

In slightly more formal terms:

Task	Input type	Output type
Question answering	string	string
Sentiment analysis	string	label
Syntactic parsing	string	tree
Semantic parsing	string	graph (logical form)
Generation	table	string
Image captioning	image	string
Machine translation	string	string

Fundamental Methods of the Course

Our primary tool will be *probabilistic models* parameterized by *deep learning architectures* such as:

- feed-forward neural networks
- recurrent neural networks
- transformers
- convolutional networks

... applied primarily to *structured prediction* problems in NLP.

The first few weeks will focus on the mathematical foundations of these models, motivated by the problem of machine translation, and setting the stage for other applications.

Courseworks will focus on the fundamentals of these models.

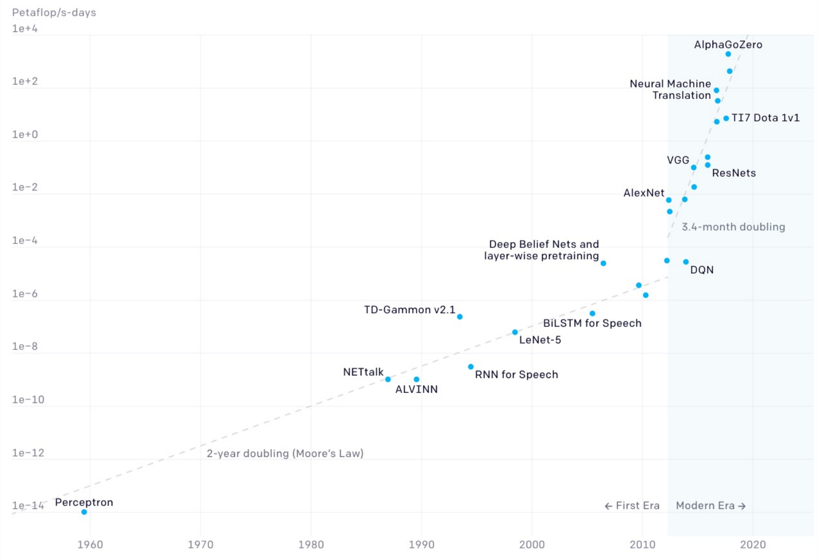
Idea: Deep learning simplifies machine learning

Why has deep learning taken over NLP?

- Deep learning simplifies the design of probabilistic models, by replacing complex dependencies and independence assumptions with *universal function approximators.*
- Deep learning gives us *representation learning:* data representations are learned rather than engineered.
- Learned representations are easy to obtain and reusable, enabling *multi-task learning.*
- Deep learning provides a uniform, flexible, trainable framework that can easily mix and match different data types: strings, labels, trees, graphs, data, and images.

In short: deep learning solves the difficulties of *applying* machine learning to NLP... *But it does not solve NLP!*

Problem: Deep learning technology is energy intensive



What is the carbon cost of a state-of-the-art ML model?

Problem: Ethical practice lags technical practice

Modern NLP originated in laboratory experiments with machine learning methods on linguistically annotated text. But NLP has escaped the lab, and can have a direct effect on people's lives:

- An Alexa chatbot responded to "Should I sell my house?" with "Sell sell sell!"
- It also responded to "Should I kill myself?" with "Yes."
- Facebook's emotional contagion experiment manipulated people on a large scale, without their consent.
- NLP is now used to recommend products, services, jobs, loans, medical treatments, prison sentences . . .

There are many wider *ethical concerns* about ML/data science, e.g., privacy. We'll focus on NLP in the course, but specific problems in NLP often reflect more general problems.

Applications of Fundamental Ideas

The second half of the course will focus on the application of deep models to a variety of core NLP problems:

- machine translation
- word embeddings
- pretrained language models
- syntactic parsing
- semantic parsing

And applications:

- paraphrasing
- question answering
- summarization
- data-to-text generation
- sentiment analysis

How the Course will be Run

Relationship to other Courses

This is not an introductory course. You must have previous experience with NLP, ML, and programming.

You must have background in natural language processing:

- *Requires* Accelerated Natural Language Processing OR Foundations of Natural Language Processing (pre-req, *not* a co-req!);

Machine learning and programming:

- IAML, MLPR, or MLP (can be taken concurrently);
- CPSLP or equivalent programming experience.

Check above syllabi if you have taken NLP elsewhere. We *cannot* advise you on whether your outside syllabus suffices. Compare it to ANLP, and realistically assess what you know.

Required Preparation for Students on this Course

Background required for the course:

- You should be familiar with Jurafsky and Martin (2008).
- But this textbook serves as background only. Each lecture will rely on one or two book chapters or papers as the main reading. Read them and discuss or ask questions.
- You will need solid maths: probability theory, linear algebra, some calculus. But this is not a maths course: we're not going to examine you on your ability to take derivatives.

Course Team: Lecturers

Lexi Birch

Frank Keller

Laura Perez

Course Team: Teaching Assistants

Christos Baziotis

Yao Fu

Course Infrastructure

- *Blackboard Learn:* Hosts the course site with all materials, timetable, courseworks, contact information, etc.
- *Piazza:* Forum for posting questions about the course material; monitored by TAs and lecturers.
- *Mailing List:* `nluplus-students@inf.ed.ac.uk`. Used by lecturers for important announcements. You were subscribed when you enrolled for the course.
- *TurnItIn:* Coursework submission system, linked with Learn. Used for courseworks, includes plagiarism checking.
- *Gradescope:* Used for the class test (coursework 3) and for the exam.
- *Dice:* Informatics computing environment, used for courseworks. If you don't have a Dice account yet, apply for one through the ITO.

Course Mechanics

The course will have a weekly rhythm:

- *Monday* of every week: *lecture slides* and *readings* for the week released; expectations for this week clarified.
- *Three in-person lectures each week;* these are also live streamed and recorded for later viewing
- You can ask questions in person during the lectures or on the Piazza Live Q&A that runs at the same time
- In each week, there is either a *tutorial* or a *lab session.* More on the next slide.

Exception: Lectures 2 and 3 will be video-only (no in-person or live stream), due to illness of the lecturer.

Tutorials and Lab Sessions

- These run weeks 2 to 10 and *alternate weekly.* So each week has *either* a tutorial *or* a lab session.
- *Tutorials* are run in small groups led by a tutor. You should try to solve the tutorial exercises ahead of time, and then discuss your solutions in the tutorial session.
- *Lab sessions* are practical sessions in which you solve programming exercises (often in preparation for an coursework).
- Labs are run in larger groups. Two lab demonstrators are on hand to help if you get stuck.
- Both tutorial and lab exercises are issued a week before.

Tutorials will or labs will run *in person,* with a few exceptions that will use *Teams* (see course timetable).

Tutorials and Lab Sessions

- If you're enrolled for this course, you will be automatically assigned a tutorial group and a lab group.
- If you are unhappy with your assigned groups, please use the group change request form to request a change.
- If you have not yet been assigned a tutorial and a lab group, please contact the ITO.
- Tutorials start in week 2, labs in week 3.

Assessment

Assessment will consist of:

- Two courseworks, each worth 15%.
- A class test (coursework 3), worth 20%.
- A final exam, worth 50%.

When you will be assessed:

- Coursework 1 issued 24 January, due 18 February.
- Class test 2 March.
- Coursework 2 issued 4 March, due 25 March.
- Final exam in the April/May exam period (date tba).

Assessment

- *Courseworks* require you to implement and run code; the experiments can be time consuming, so start early!
- Courseworks will include intermediate milestones and recommended timelines.
- Courseworks are accompanied by *lab sessions* in which you can ask questions about the coursework.
- The *class test* is timed and will consist of multiple choice questions and short-answer questions.
- The *final exam* is also timed and will consist of longer, problem-solving questions.
- Both will emphasize *understanding* and *synthesis* of ideas, rather than rote memorization of technical details.

More on Courseworks

- Courseworks can be done in pairs.
- This means you will work together with a classmate and submit a single solution.
- Both members of the pair will receive the same mark.
- You don't *have* to work in pairs, but it's strongly encouraged.
- You can work with the *same* partner for both courseworks.
- Details on how pairs are formed will be released ahead of Coursework 1.

How to get help

Ask questions. Asking questions is how you learn.

Answer questions. Answering questions is how you demonstrate that you've learned it!

- TA drop-in office hours, starting week 2.
- Piazza forum: course staff will answer questions *once a day, Monday through Friday*. *You* can answer questions any time! Your questions can be anonymous to classmates. They can be private (use this if the question pertains *only to you*).
- *Don't ask us questions over email.* We might not see your question for days. And when we do, we will just repost it to Piazza, so that everyone can see the answer.

Preview

The rest of this week:

- Lecture 2: Introduction to Machine Translation **[video only]**
- Lecture 3: Machine Translation with n-grams **[video only]**
- Tutorial 1 issued (for next week): Probabilistic Models and Neural Nets

References

Manning, Christopher D. 2015. Computational linguistics and deep learning. *Computational Linguistics* 41(4): 701–707.