CS 7650: Natural Language Processing

Wei Xu

Administrivia

- Course website: https://cocoxu.github.io/CS7650_fall2025/
  - homework release, slides, readings
  - course policies

- Piazza:
  - for all class announcements, homework discussion, and contacting teaching staff
  - TA will start a mega-thread when release each assignment, and post a sign-up list for OH, etc

- Gradescope:
  - for homework submission and grading

Instructor

Wei Xu

Office Hours: Monday after class

Teaching Assistants

Duong Minh Le

Jerry Zheng

Joseph Thomas

Rohan Phadnis

Course Goals

- Cover fundamental machine learning techniques used in NLP

- Understand how to look at language data and approach linguistic phenomena

- Cover modern NLP problems encountered in the literature:

- Make you a “producer” rather than a “consumer” of NLP tools

  - The four programming assignments should teach you what you need to know to understand nearly any system in the literature

Course Requirements

- **Probability** (e.g. conditional probabilities, conditional independence, Bayes Rule)
- **Linear Algebra** (e.g., multiplying vectors and matrices, matrix inversion)
- **Multivariable Calculus** (e.g., calculating gradients of functions with several variables)
- **Programming / Python experience** (medium-to-large scale project, **debug** PyTorch codes when there are no error messages)
- Prior exposure to machine learning

There will be a lot of math and programming!

Some Example Slides

Sequential Models - e.g., Conditional Random Fields

- Model: $$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z}\prod_{i=2}^{n}\exp(\phi_t(y_{i-1}, y_i))\prod_{i=1}^{n}\exp(\phi_e(y_i, i, \mathbf{x}))$$
$$P(\mathbf{y}|\mathbf{x}) \propto \exp w^\top \left[\sum_{i=2}^{n} f_t(y_{i-1}, y_i) + \sum_{i=1}^{n} f_e(y_i, i, \mathbf{x})\right]$$
- Inference: argmax P(**y**|**x**) from Viterbi
- Learning: run forward-backward to compute posterior probabilities; then

$$\frac{\partial}{\partial w}\mathcal{L}(\mathbf{y}^*, \mathbf{x}) = \sum_{i=1}^{n} f_e(y_i^*, i, \mathbf{x}) - \sum_{i=1}^{n}\sum_{s} P(y_i = s|\mathbf{x}) f_e(s, i, \mathbf{x})$$

Some Example Slides

Training CRFs

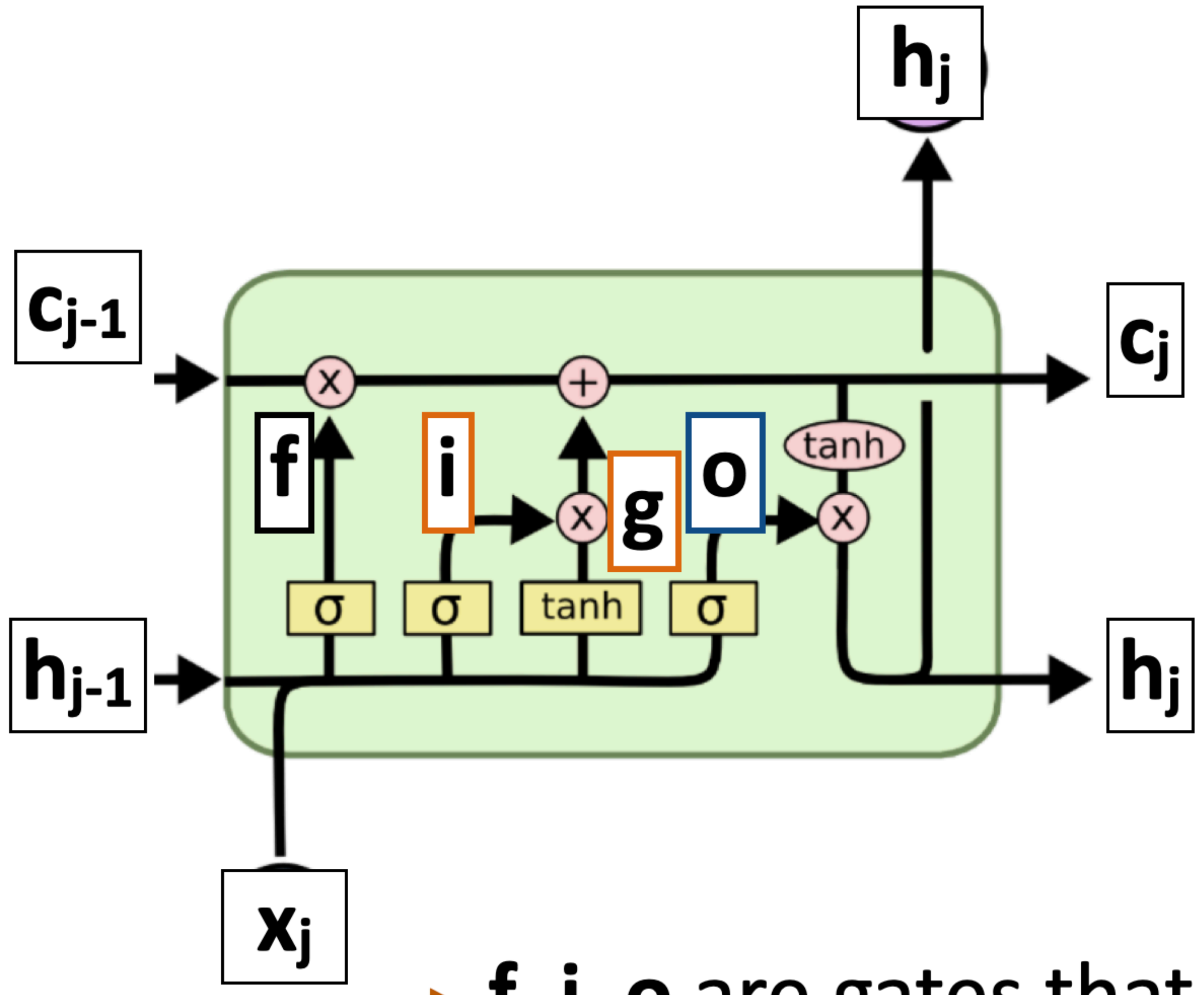
$$\frac{\partial}{\partial w}\mathcal{L}(\mathbf{y}^*, \mathbf{x}) = \sum_{i=2}^{n} f_t(y_{i-1}^*, y_i^*) + \sum_{i=1}^{n} f_e(y_i^*, i, \mathbf{x})$$
$$-\mathbb{E}_{\mathbf{y}}\left[\sum_{i=2}^{n} f_t(y_{i-1}, y_i) + \sum_{i=1}^{n} f_e(y_i, i, \mathbf{x})\right]$$

- Let's focus on emission feature expectation

$$\mathbb{E}_{\mathbf{y}}\left[\sum_{i=1}^{n} f_e(y_i, i, \mathbf{x})\right] = \sum_{\mathbf{y}\in\mathcal{Y}} P(\mathbf{y}|\mathbf{x})\left[\sum_{i=1}^{n} f_e(y_i, i, \mathbf{x})\right] = \sum_{i=1}^{n}\sum_{\mathbf{y}\in\mathcal{Y}} P(\mathbf{y}|\mathbf{x}) f_e(y_i, i, \mathbf{x})$$
$$= \sum_{i=1}^{n}\sum_{s} P(y_i = s|\mathbf{x}) f_e(s, i, \mathbf{x})$$

Some Example Slides

Neural Network Models — e.g., LSTMs



$$c_j = c_{j-1} \odot f + \boxed{g \odot i}$$

$$f = \sigma(x_j W^{xf} + h_{j-1} W^{hf})$$

$$g = \tanh(x_j W^{xg} + h_{j-1} W^{hg})$$

$$i = \sigma(x_j W^{xi} + h_{j-1} W^{hi})$$

$$h_j = \tanh(c_j) \odot o$$

$$o = \sigma(x_j W^{xo} + h_{j-1} W^{ho})$$

- **f**, **i**, **o** are gates that control information flow
- **g** reflects the main computation of the cell

Hochreiter & Schmidhuber (1997)

http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Some Example Slides

Computing Gradients: Backpropagation

$$\mathcal{L}(\mathbf{x}, i^*) = W\mathbf{z} \cdot e_{i^*} - \log \sum_{j=1}^{m} \exp(W\mathbf{z} \cdot e_j)$$

$\mathbf{z} = g(Vf(\mathbf{x}))$

Activations at hidden layer

- Gradient with respect to *V*: apply the chain rule

$$\frac{\partial \mathcal{L}(\mathbf{x}, i^*)}{\partial V_{ij}} = \frac{\partial \mathcal{L}(\mathbf{x}, i^*)}{\partial \mathbf{z}} \boxed{\frac{\partial \mathbf{z}}{V_{ij}}} \qquad \frac{\partial \mathbf{z}}{V_{ij}} = \boxed{\frac{\partial g(\mathbf{a})}{\partial \mathbf{a}}} \boxed{\frac{\partial \mathbf{a}}{\partial V_{ij}}}$$

$\mathbf{a} = Vf(\mathbf{x})$

- First term: gradient of nonlinear activation function at ***a*** (depends on current value)
- Second term: gradient of linear function
- Straightforward computation once we have *err*(**z**)

Background Test

- Problem Set 0 (math background) is released, **due Thursday Aug 21**.
- Project 0 (programming - logistic regression) is also released, **due Aug 29**.
- Take **CS 4641/7641 Machine Learning** and (Math 2550 or Math 2551 or Math 2561 or Math 2401 or Math 24X1 or 2X51 - equivalent) before this class.
- If you want to understand the lectures better and complete homework with more ease, taking also CS 4644/7643 Deep Learning before this class.

Wait List

- If you plan to take the class, please complete and submit Problem Set 0 by Thursday Aug 21.
- If you get off the wait list, you will be automatically added to Gradescope after about a day. If not, post a message on Piazza to get the access to Gradescope.
- If you cannot access Gradescope by the due date, please email your submission to the instructor.

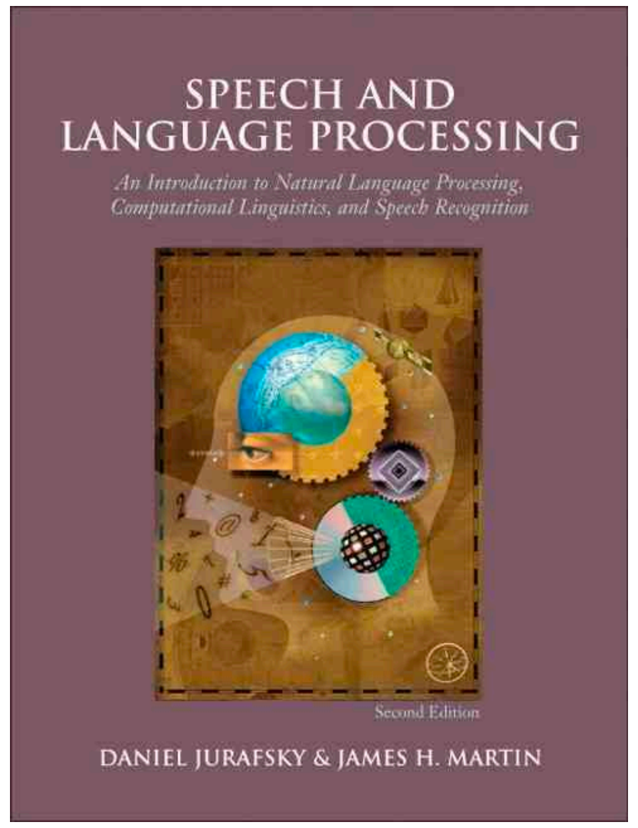
Free Textbooks!

- Two really awesome textbooks available
  - There will be assigned readings from both
  - Both freely available online

Speech and Language Processing **(3rd ed. draft)**

Dan Jurafsky **and** **James H. Martin**

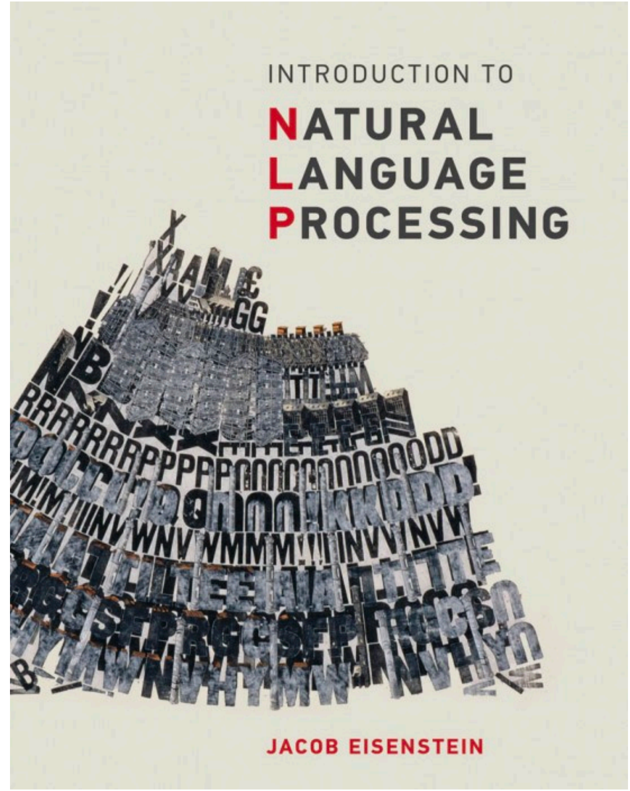
Here's our December 30, 2020 draft! Includes:



Introduction to Natural Language Processing

By Jacob Eisenstein

Published by The MIT Press
Oct 01, 2019 | 536 Pages | 7 x 9
| ISBN 9780262042840



Coursework Plan

- Four programming projects (25%)
  - Implementation-oriented
  - 1.5~2 weeks per assignment
  - fairly substantial implementation effort except P0
- Three written assignments (20%) + midterm exam (20%)
  - Mostly math and theoretical problems related to ML / NLP
- Final project (28%) + in-class presentation of a recent research paper (2%)
- Participation/Attendance (5%)

Programming Projects

- Modern NLP methods require non-trivial computation
  - Training/debugging neural networks can take a long time (**start early!**)
  - Most programming will be done with **PyTorch** library (can be tricky to debug)
  - You will want to use a GPU (Google Colab; pro account for $10/month)
  - The programming projects are designed with Google Colab in mind

Final Project

- In-class presentation of a recent research paper (2%)

- Final project (30%)
  - Groups of 2-4 student preferred, 1 student is also possible with permission.
    - 4-5 page project report (similar to ACL/NAACL/EMNLP short papers: https://arxiv.org/search/?query=EMNLP+short+paper&searchtype=comments&source=header)
    - Final project presentation
    - Good idea to run your project idea with me during office hour.

Final Project

- Grading rubrics
  - Clarity (1-5): For the reasonably well-prepared reader, is it clear what was done and why? Is the report well-written and well structured?
  - Originality / Innovativeness (1-5): How original is the approach? Does this project break new ground in topic, methodology, or content? How exciting and innovative is the work that it describes?
  - Soundness / Correctness (1-5): First, is the technical approach sound and well-chosen? Second, can one trust the claims of the report – are they supported by proper experiments, proofs, or other argumentation?
  - Meaningful Comparison (1-5): Does the author make clear where the problems and methods sit with respect to existing literature? Are any experimental results meaningfully compared with the best prior approaches?
  - Substance (1-5): Does this project have enough substance, or would it benefit from more ideas or results? Note that this question mainly concerns the amount of work; its quality is evaluated in other categories.
  - **Overall (1-5) - Overall quality/novelty/significance of the work. <u>Not</u> a sum of aspect-based scores.**

Late Policy

- Late Policy
  - 6 flexible days to use over the duration of the semester for homework assignment only.
  - These flexible days should be reserved for emergency situation only.
  - Homework submitted late after all flexible days used up will receive penalty (5% deduction per day).
- No make-up exam for midterm. No late submission for final project report.
  - Unless under emergency situation verified by the Office of the Dean of Students

Outline of the Course

ML and structured prediction for NLP

Deep Learning (Neural Networks)

Language Models

	Topic	Projects	Problem Sets
8/18/2025	Course Overview	Proj. 0 Out	PS0 Out
8/20/2025	Machine Learning Recap - Naive Bayes, MLE		PS0 Due (8/21), PS1 O
8/25/2025	Machine Learning Recap - logistic regression, perceptron, SVM		
8/27/2025	Machine Learning Recap - multi-class classification	Proj. 0 Due (8/29)	
9/1/2025	**No class - Labor Day holiday**	PyTorch Tutorial	
9/3/2025	Neural Networks - feedforward network, training, optimization	Proj. 1 Out	PS1 Due (9/5)
9/8/2025	Word Embeddings		
9/10/2025	Sequence Labeling		
9/15/2025	Conditional Random Fields	Proj. 1 Due (9/19)	
9/17/2025	Recurrent Neural Networks	Proj. 2 Out	
9/22/2025	Convolutional Neural Networks, Neural CRF		
9/24/2025	Encoder-Decoder	Instructions Out for 2-min presentation per indi	
9/29/2025	Attention	Instructions Out for a (lightweight) project prop	
10/1/2025	Transformer	Proj. 2 Due (10/3)	PS2 Out
10/6/2025	**No class - Fall Break**		
10/8/2025	Pretrained Language Models (part 1 - BERT), midterm review		
10/13/2025	Pretrained Language Models (part 2 - BART/T5, GPT2/3, instruction tuning), Ethics		PS2 Due (10/14)
10/15/2025	Pretrained Language Models (part 3 - Post-training of Language Models)	Proj. 3 Out	
10/20/2025	Pretrained Language Models (part 4 - Open-source Language Models)	2-min presentation Due (10/21)	
10/22/2025	**student in-class presentation**	Course Project Proposal Due (10/24)	
10/27/2025	**No class - reading day** 📖		10/25 withdraw deadlir
11/3/2025	**Midterm** 📝		
11/5/2025	**student in-class presentation, Midterm** 📝		
11/10/2025	**student in-class presentation, Midterm** 📝	Proj. 3 Due (11/11) - last homework to use flex	
11/12/2025	**student in-class presentation, Midterm** 📝		
11/17/2025	**student in-class presentation**		
11/19/2025	Guest Lecture		
11/24/2025	Guest Lecture		
11/26/2025	**No class - school recess**		
12/1/2025	**No class - reading day** 📖		
12/5/2025	**2:40pm Final Project (written report + oral presentations)**	Final Project Due - no late submission allowed	

tentative plan

(subject to change)

* Link to this Google spreadsheet on course website:
https://docs.google.com/spreadsheets/d/1CIc2FTHgTR_lL71W4ON3J01WVYgC51fxgsfYmn2pwtQ/edit?usp=sharing

FAQ

- Q: The class is full, can I still get in?

  Depending on how many students will drop the class. The course registration system and office controls the process and priority order.

- Q: I am taking CS 4641/7641 ML class this same semester, would that be sufficient?

  A: No. You need to take 4641/7641 (or equivalent) **before** taking this class. NLP is at the very front of technology development. This is one of the most advanced classes. This course will be more work-intensive than most graduate or undergraduate courses at Georiga Tech, but will be comparable to NLP classes offered at other top universities.

- Q: How much grades I need to pass the class?

  A: Students need to receive 50% grade to pass the class.

FAQ

- Q: I want to understand the lectures better, what can I do?

  A: Read the required reading before the class. Taking deep learning class first will greatly help too. The lectures are designed to cover state-of-the-art material in class, while lower-level details will be "taught" through written and programming homework assignments. (similar design to NLP classes at other top universities, e.g., Stanford/ Berkeley/Princeton)

- Q: I want to learn more about LLMs, what can I do?

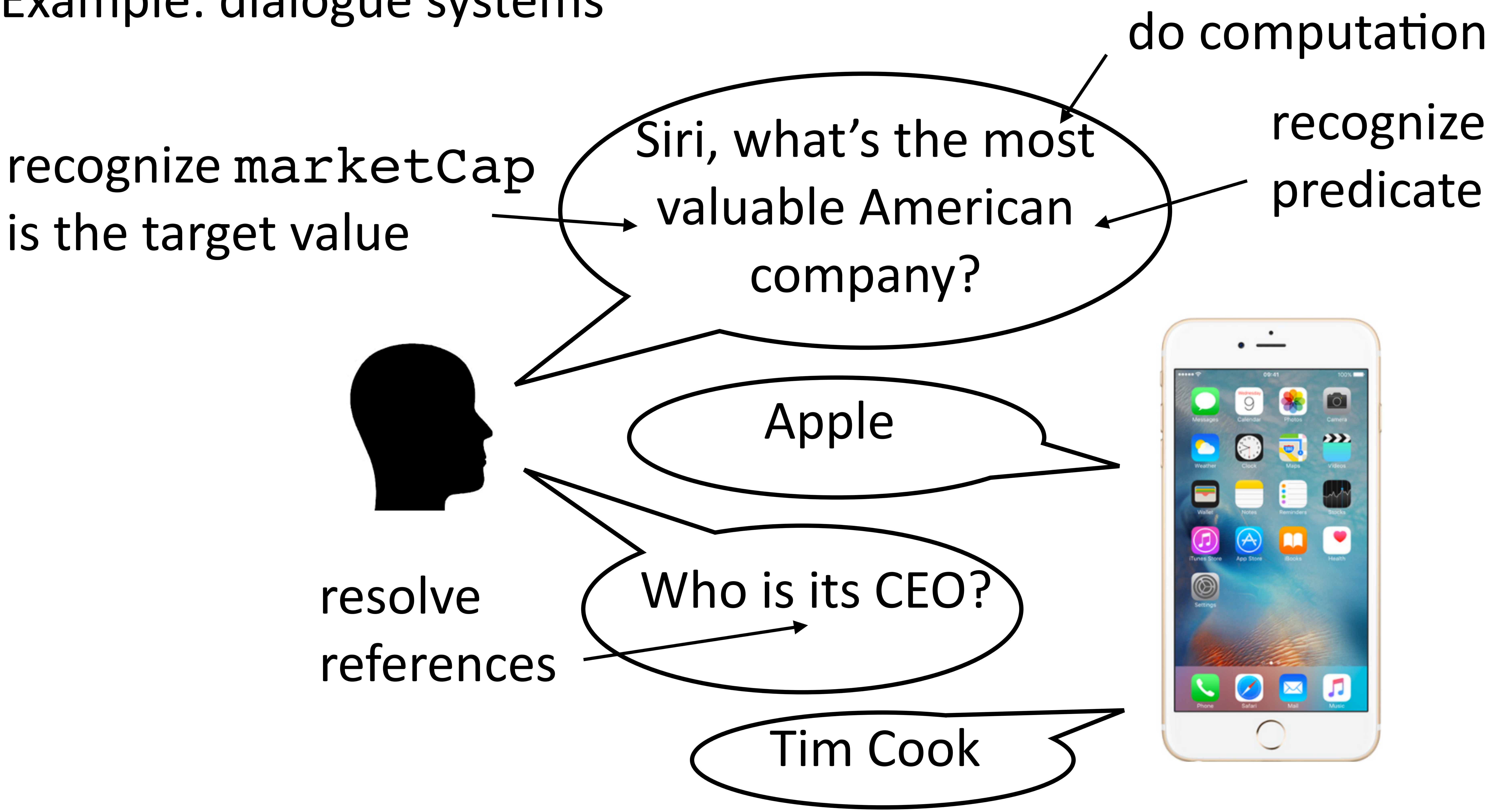
  A: CS 8803-LLM "Large Language Model" (Fall 2024)

  https://cocoxu.github.io/CS8803-LLM-fall2024/

QA Time

What's the goal of NLP?

- Be able to solve problems that require deep understanding of text
- Example: dialogue systems



Automatic Summarization

POLITICS

Google Critic Ousted From Think Tank Funded by the Tech Giant

WASHINGTON — In the hours after European antitrust regulators levied a record $2.7 billion fine against Google in late June, an influential Washington think tank learned what can happen when a tech giant that shapes public policy debates with its enormous wealth is criticized.

...

But not long after one of New America's scholars posted a statement on the think tank's website praising the European Union's penalty against Google, Mr. Schmidt, who had been chairman of New America until 2016, communicated his displeasure with the statement to the group's president, Anne-Marie Slaughter, according to the scholar.

...

Ms. Slaughter told Mr. Lynn that "the time has come for Open Markets and New America to part ways," according to an email from Ms. Slaughter to Mr. Lynn. The email suggested that the entire Open Markets team — nearly 10 full-time employees and unpaid fellows — would be exiled from New America.

compress text

provide missing context

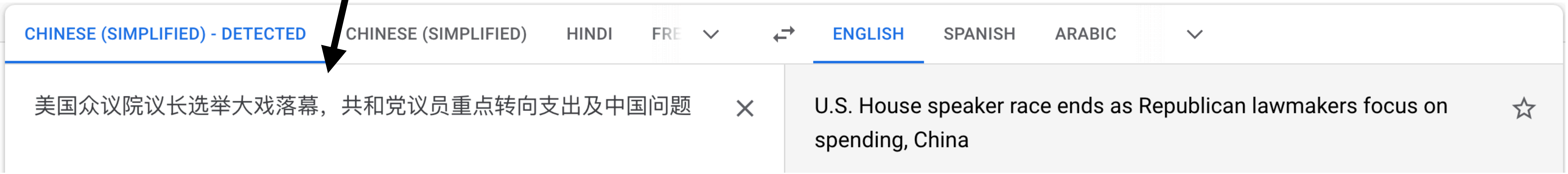
One of New America's writers posted a statement critical of Google. Eric Schmidt, Google's CEO, was displeased.

The writer and his team were dismissed.

paraphrase to provide clarity

Machine Translation

- Working very well now for high-resource languages.
- Some language pairs more difficult (e.g. English-Japanese)
- Still a number of challenges (scaling up to thousands of languages, etc.)



African Languages!

- AfroLID, a neural LID toolkit for 517 African languages and varieties.

Figure 1: All 50 African countries in our data, with our 517 languages/language varieties in colored circles overlayed within respective countries. More details are in Appendix E.

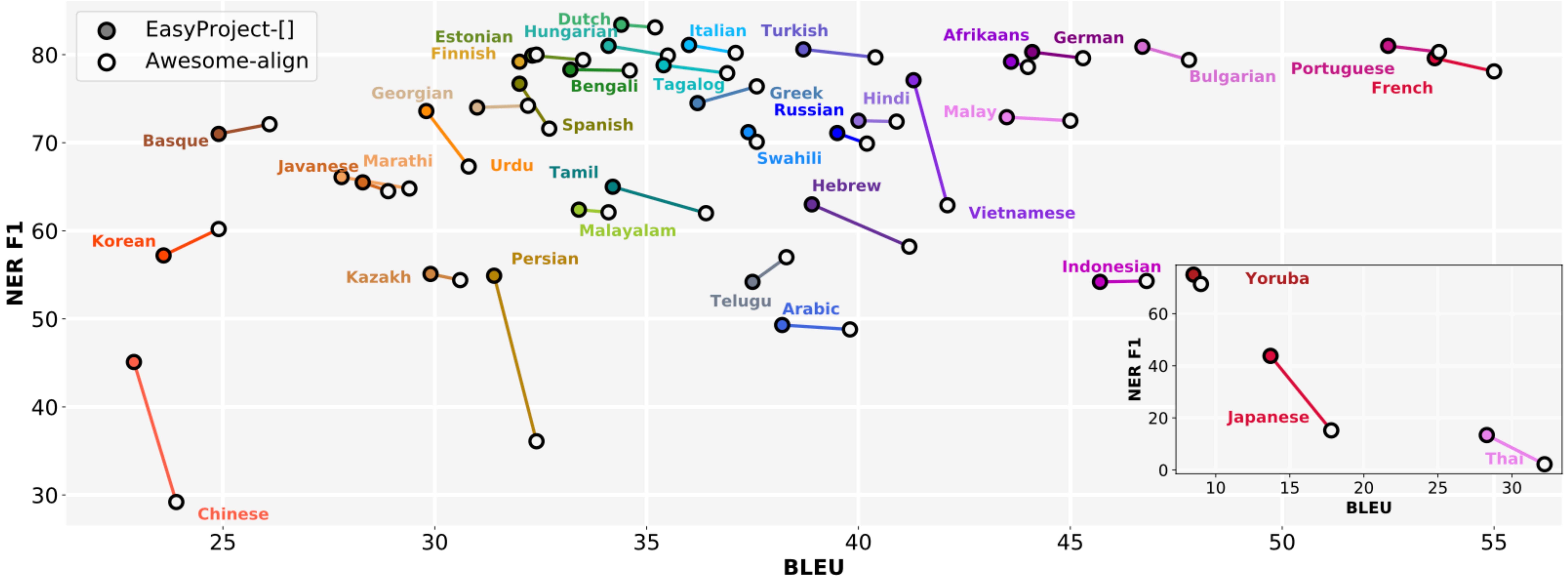
Word Order	Example Languages
SVO	Xhosa, Zulu, Yorùbá
SOV	Khoekhoe, Somali, Amharic
VSO	Murle, Kalenjin
VOS	Malagasy
No-dominant-order	Siswati, Nyamwezi, Bassa

Table 1: Sentential word order in our data.

Adebara et al. (2022)

Cross-Lingual Transfer Learning

- Marker-based label projection is especially promising for low-resource languages & languages that are written in non-Latin scripts.



Yang Chen, Chao Jiang, Alan Ritter, Wei Xu. "Frustratingly Easy Label Projection for Cross-lingual Transfer" (ACL 2023 Findings)

Why is language hard?

(and how can we handle that?)

Language is Ambiguous!

- Hector Levesque (2011): "Winograd schema challenge" (named after Terry Winograd, the creator of SHRDLU)

they advocated

The city council refused the demonstrators a permit because they _______ violence

they feared

- This is so complicated that it's an AI challenge problem! (AI-complete)
- Referential/semantic ambiguity

Language is Ambiguous!

- Ambiguous News Headlines:
  - Teacher Strikes Idle Kids
  - Hospitals Sued by 7 Foot Doctors
  - Ban on Nude Dancing on Governor’s Desk
  - Iraqi Head Seeks Arms
  - Stolen Painting Found by Tree
  - Kids Make Nutritious Snacks
  - Local HS Dropouts Cut in Half
- Syntactic/semantic ambiguity: parsing needed to resolve these, but need context to figure out which parse is correct

slide credit: Dan Klein

Language is **Really** Ambiguous!

- There aren't just one or two possibilities which are resolved pragmatically

il fait vraiment beau →

It is really nice out
It's really nice
The weather is beautiful
It is really beautiful outside
He makes truly beautiful
He makes truly boyfriend
It fact actually handsome

- Combinatorially many possibilities, many you won't even register as ambiguities, but systems still have to resolve them

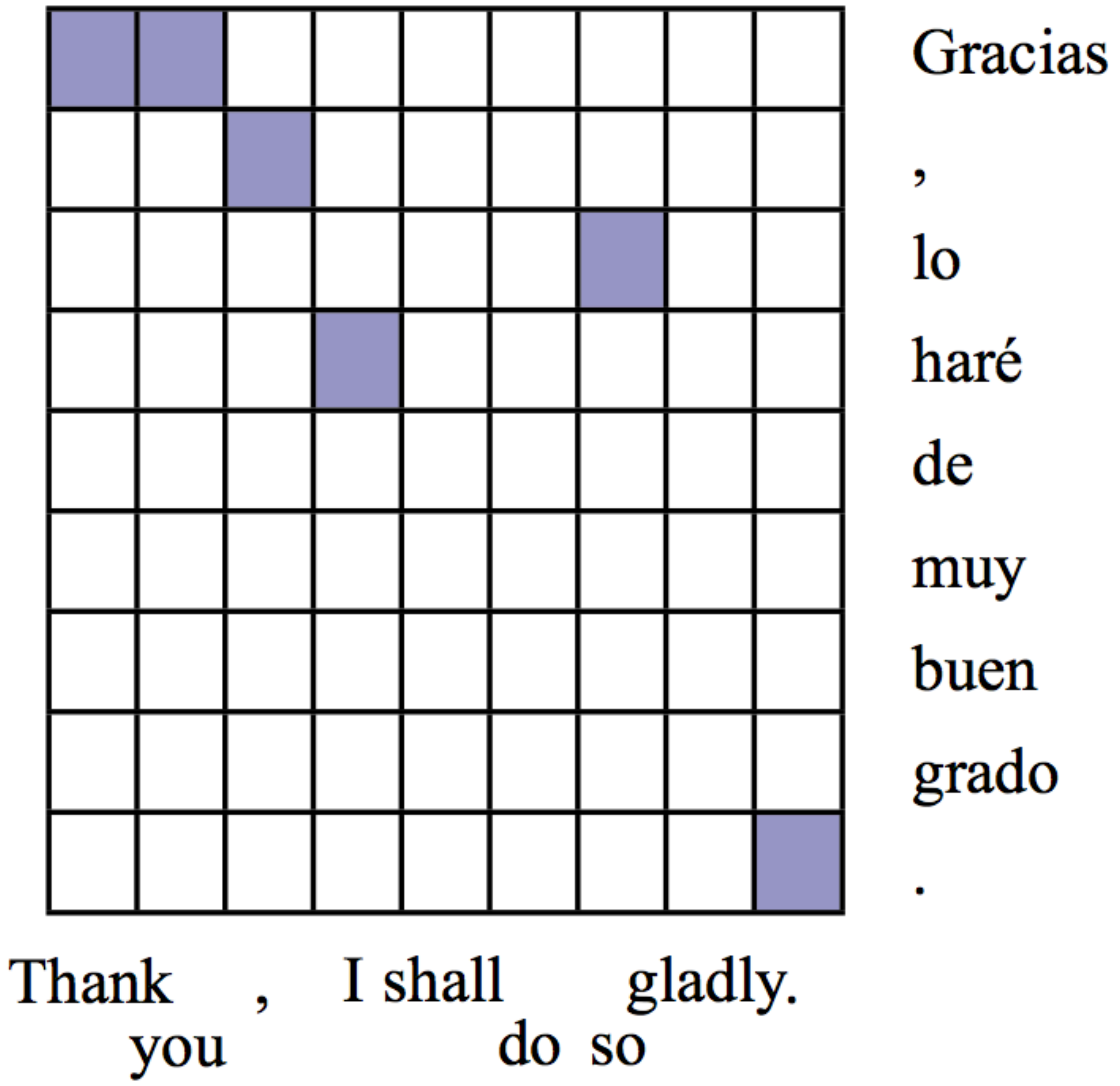
What do we need to understand language?

- Lots of data!

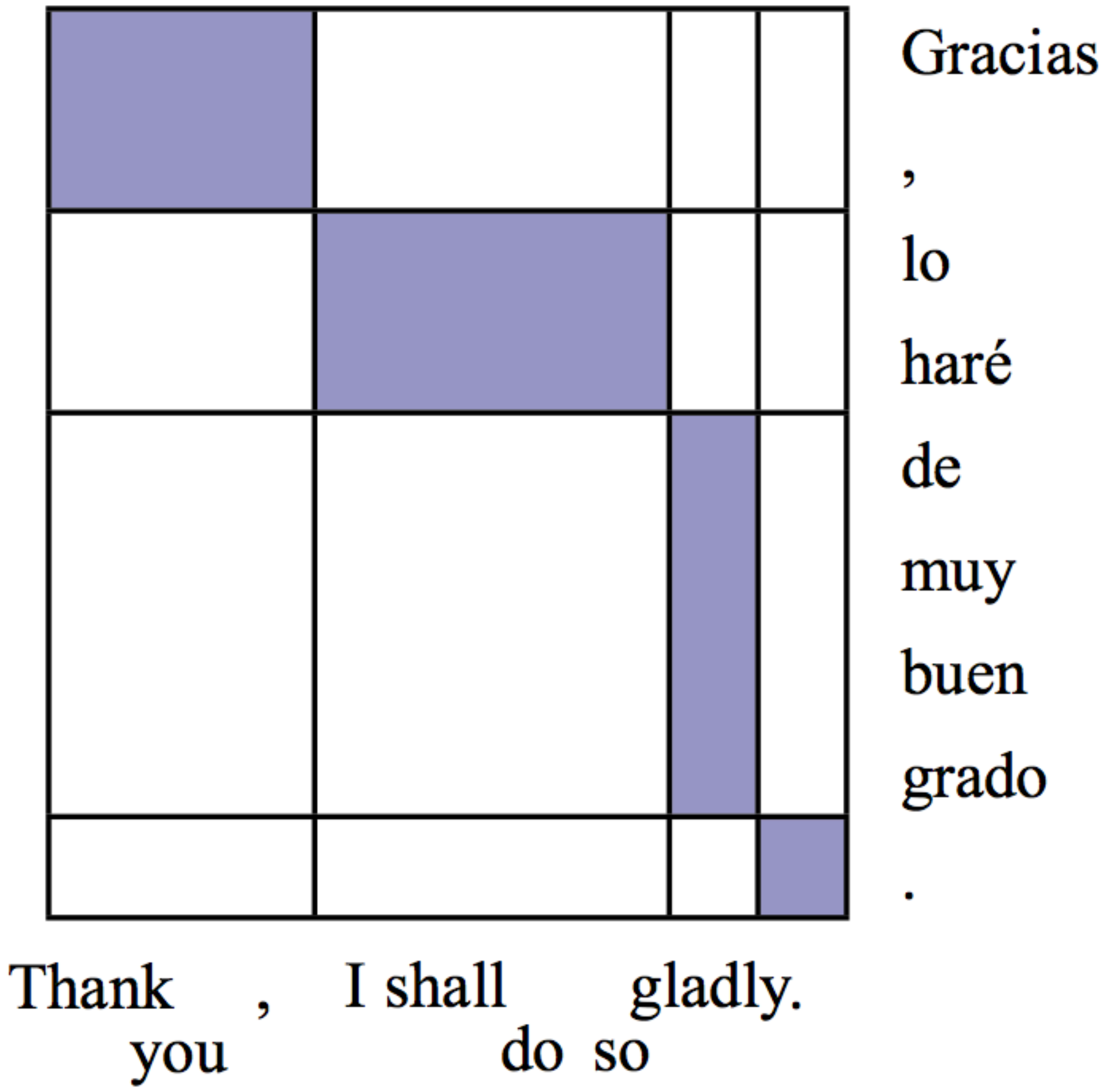
SOURCE	Cela constituerait une solution transitoire qui permettrait de conduire à terme à une charte à valeur contraignante.
HUMAN	That would be an interim solution which would make it possible to work towards a binding charter in the long term .
1x DATA	[this] [constituerait] [assistance] [transitoire] [who] [permettrait] [licences] [to] [terme] [to] [a] [charter] [to] [value] [contraignante] [.]
10x DATA	[it] [would] [a solution] [transitional] [which] [would] [of] [lead] [to] [term] [to a] [charter] [to] [value] [binding] [.]
100x DATA	[this] [would be] [a transitional solution] [which would] [lead to] [a charter] [legally binding] [.]
1000x DATA	[that would be] [a transitional solution] [which would] [eventually lead to] [a binding charter] [.]

slide credit: Dan Klein

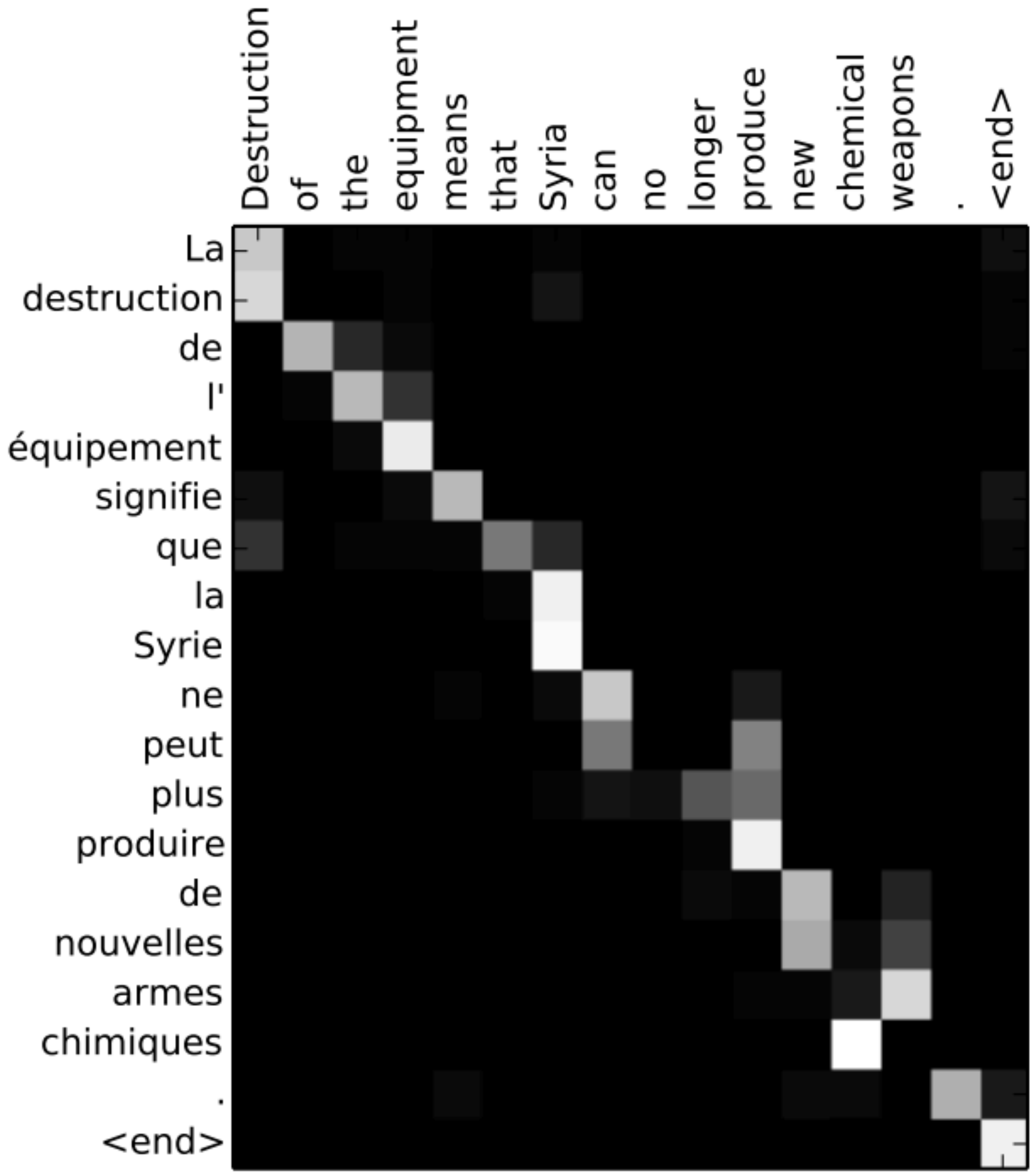
Less Manual Structure?



(a) example word alignment



(b) example phrase alignment



DeNero et al. (2008)

Bahdanau et al. (2014)

What techniques do we use?

(to combine data, knowledge, linguistics, etc.)

A brief history of (modern) NLP

"AI winter"
rule-based,
expert systems

earliest stat MT
work at IBM

Penn
treebank

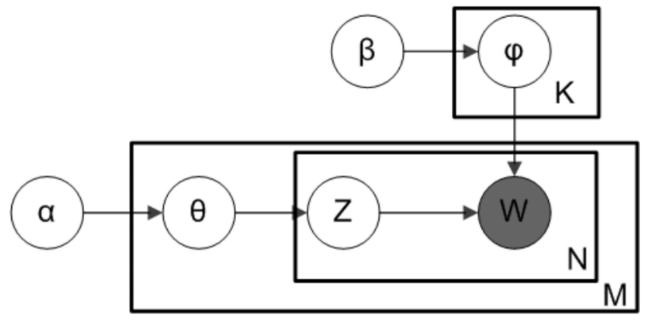
S
NP VP

Ratnaparkhi
tagger

NNP VBZ

Collins vs.
Charniak
parsers

Unsup: topic
models, etc.

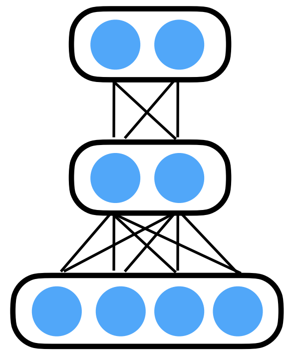


Sup: SVMs,
CRFs, NER,
Sentiment

Semi-sup,
structured
prediction

Pre-training
(ELMo, BERT, GPT)

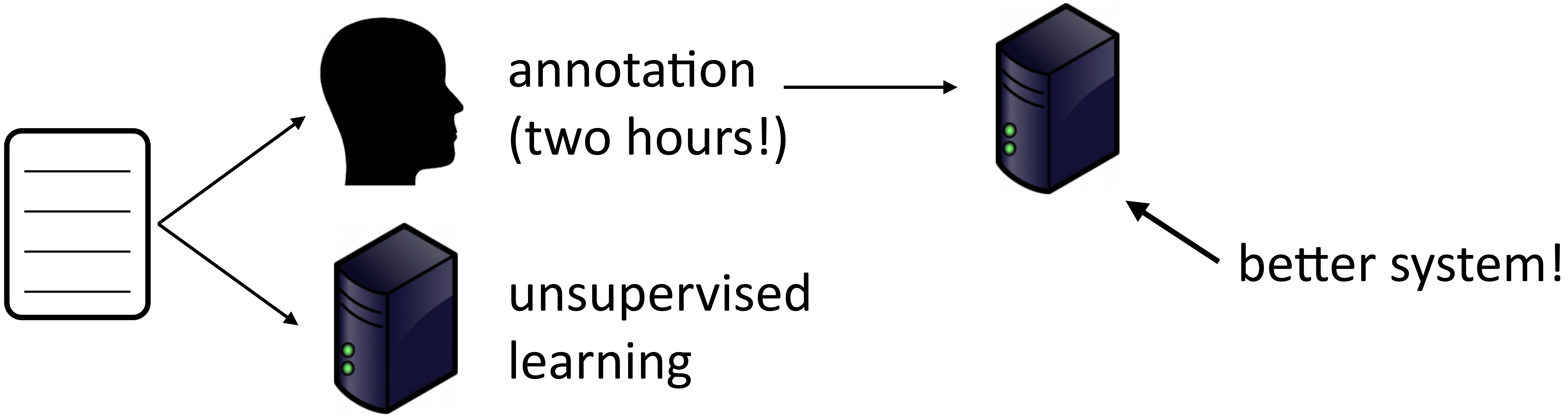
Neural



1980 1990 2000 2010 2020

How Much Training Data do we Need?

- All of these techniques are data-driven! Some data is naturally occurring, but may need to label
- Supervised techniques work well on very little data



- Even neural nets can do pretty well!

"Learning a Part-of-Speech Tagger from Two Hours of Annotation"
Garrette and Baldridge (2013)

Pretraining

- Language modeling: predict the next word in a text $P(w_i|w_1, \ldots, w_{i-1})$

P(*w* | I want to go to) = 0.01 Hawai'i

0.005 LA

0.0001 class

 : use this model for other purposes

P(*w* | the acting was horrible, I think the movie was) = 0.1 bad

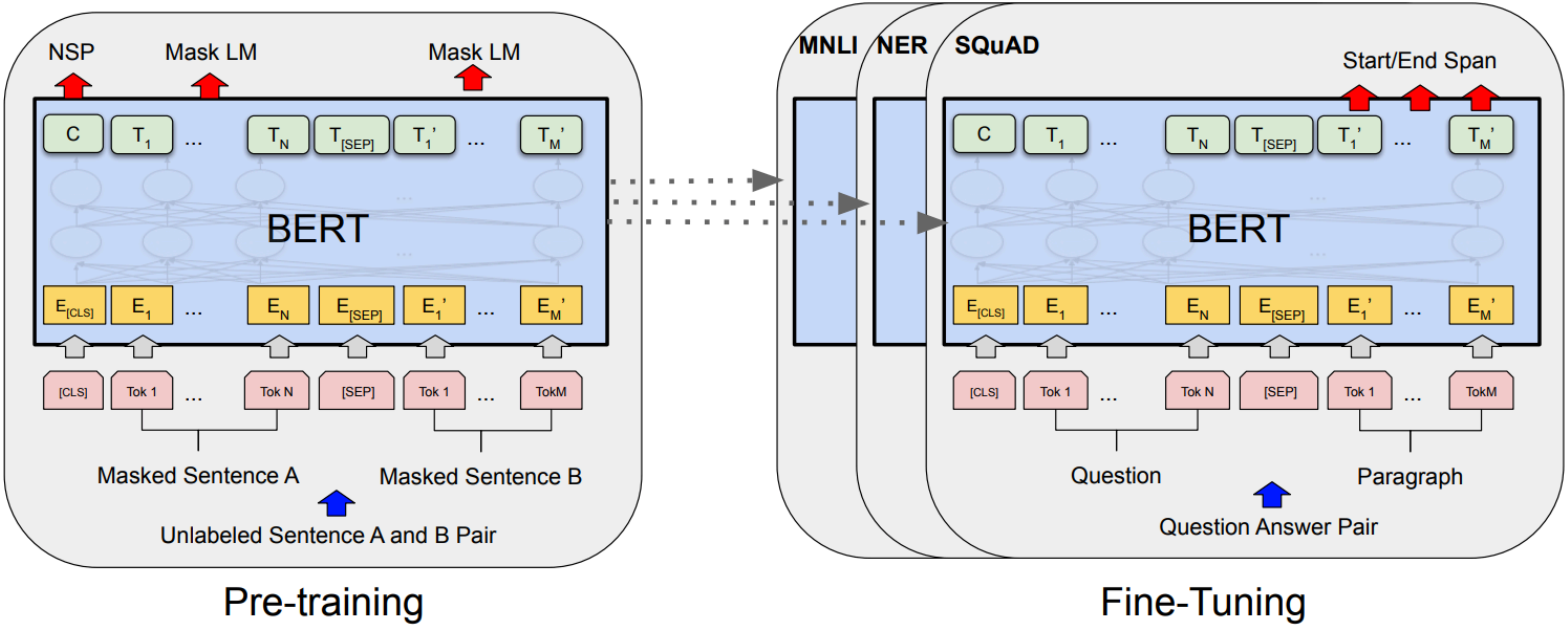
0.001 good

- Model understands some sentiment?

- Train a neural network to do language modeling on massive unlabeled text, fine-tune it to do {tagging, sentiment, question answering, …}

Peters et al. (2018), Devlin et al. (2019)

BERT



- Key parts which we will study: (1) Transformer architecture; (2) what data is used (both for pre-training and fine-tuning)

Devlin et al. (2019)

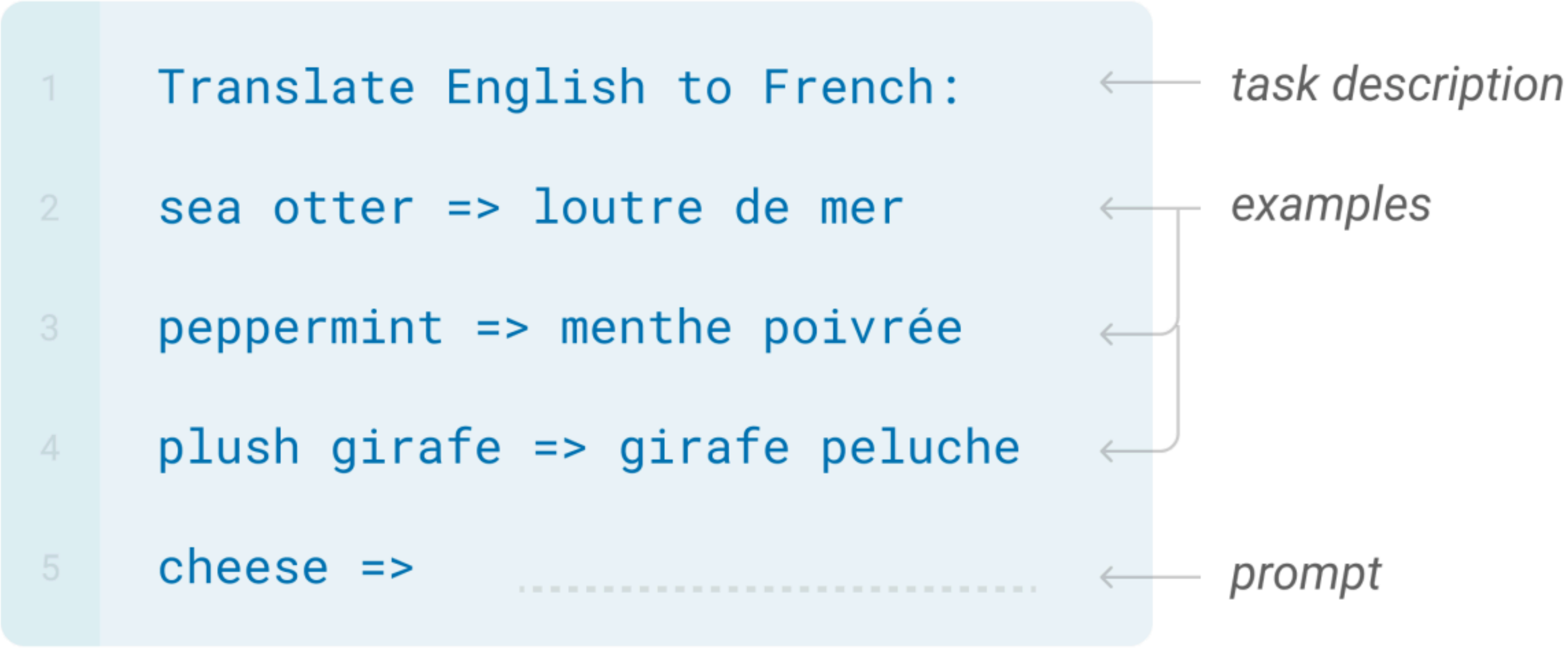
GPT and In-Context Learning

- Even more "extreme" setting: no gradient updates to model, instead large language models "learn" from examples in their context

- Many papers studying why this works. We will read some!

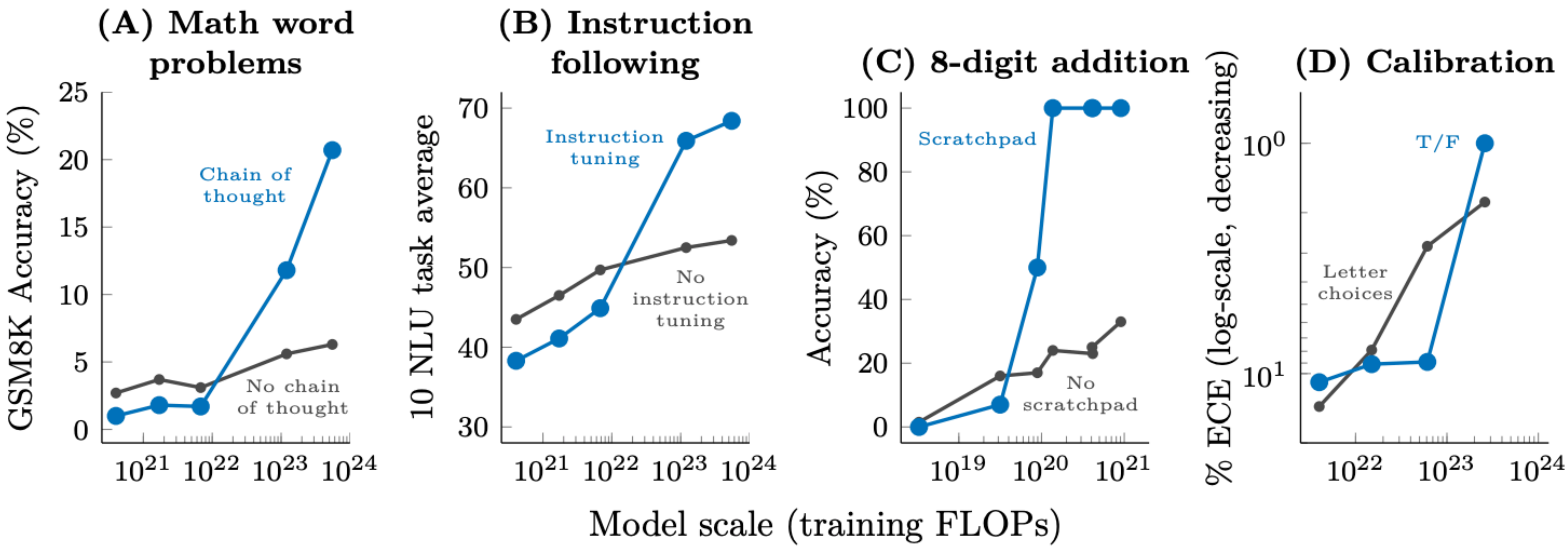
Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



Brown et al. (2020)

Scaling Laws

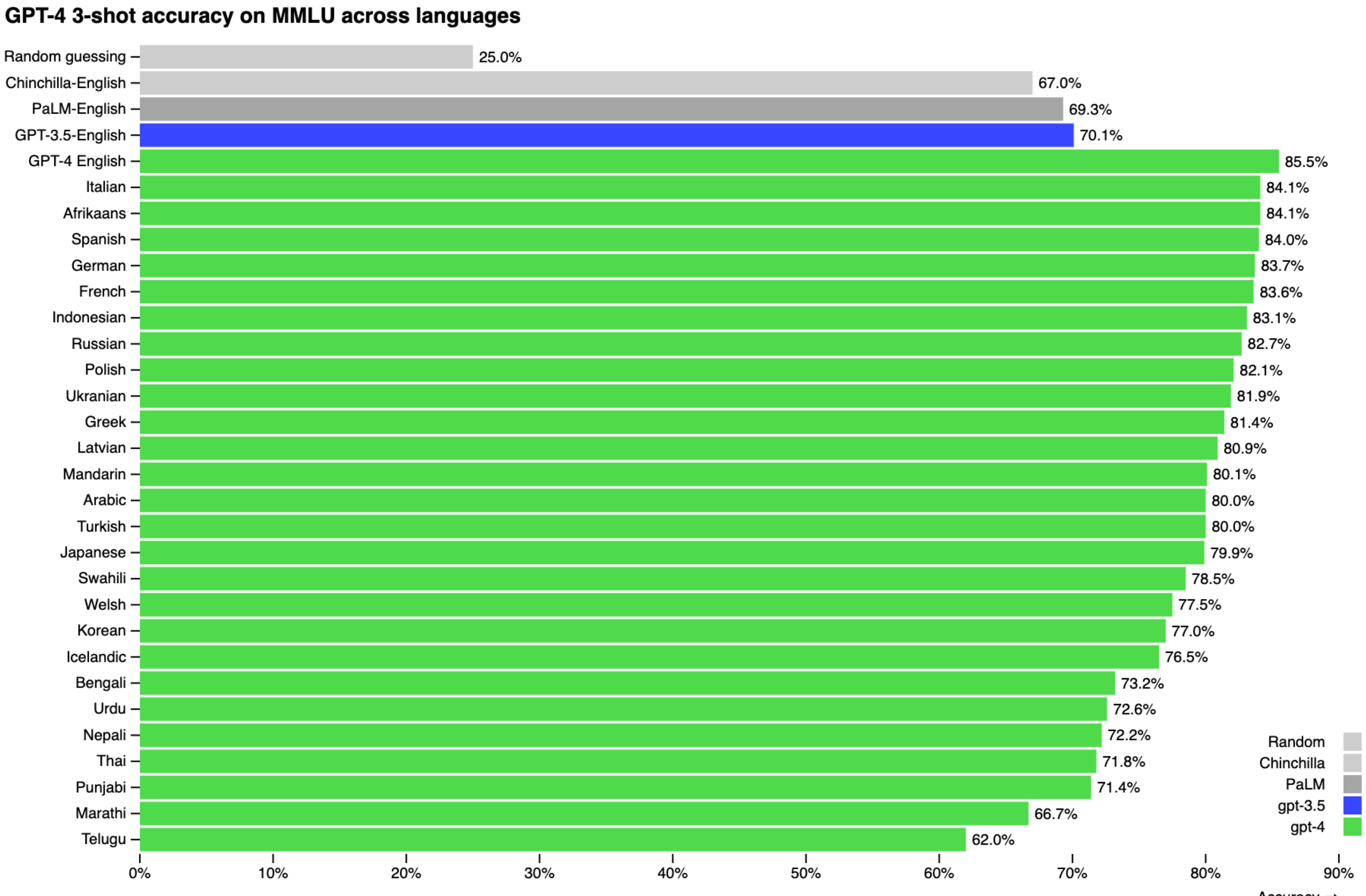


- Many of the ideas that are big in 2023 only make sense and only work because the models are so big!

Kaplan et al. (2020), Jason Wei et al. (2022)

GPT-4

- Tested on 26 languages, MMLU - Multiple-choice questions in 57 subjects



Where are we?

- NLP consists of: analyzing and building representations for text, solving problems involving text

- These problems are hard because language is ambiguous, requires drawing on data, knowledge, and linguistics to solve

- Knowing which techniques to use requires understanding dataset size, problem complexity, and a lot of tricks!

- NLP encompasses all of these things

QA Time