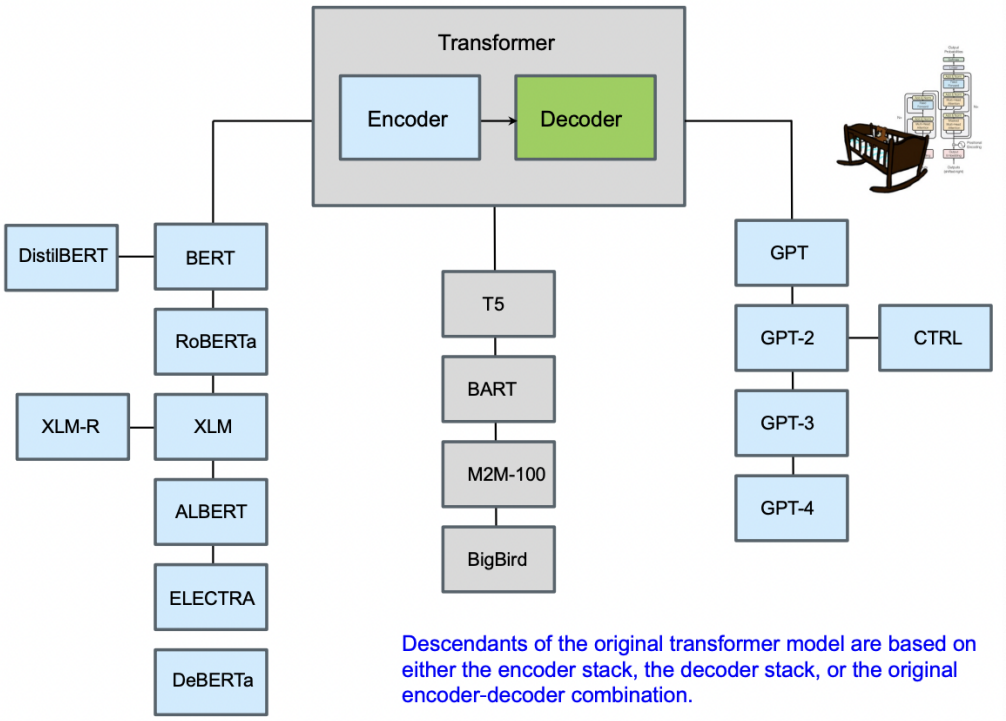
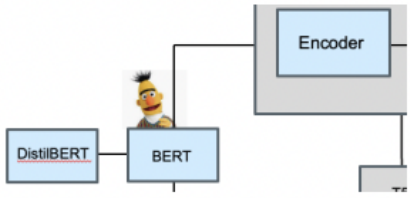
CAS CS 505
Lec 20

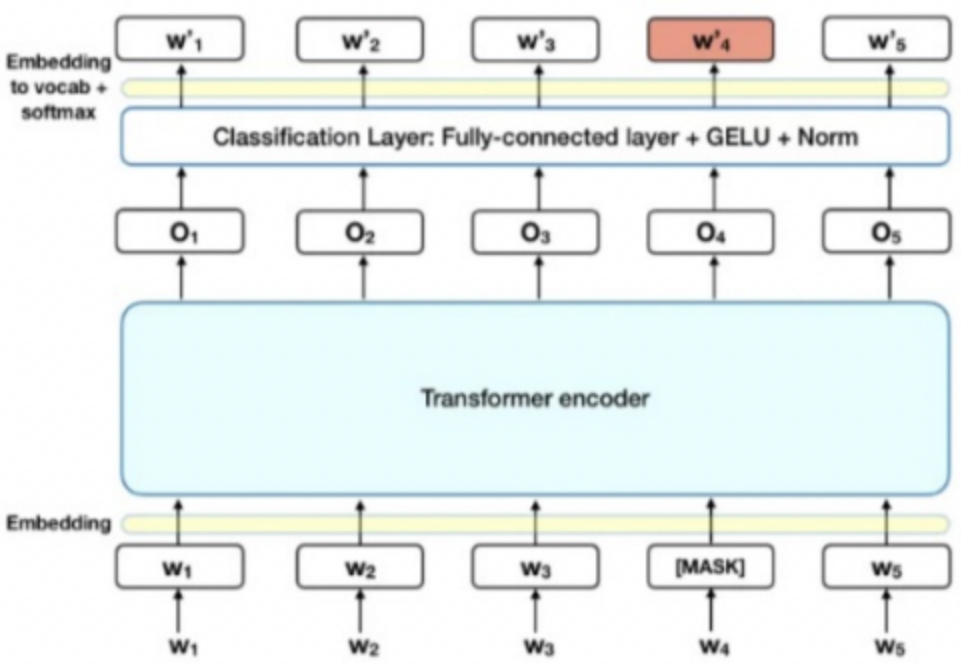
The Transformer Family

1. The Transformer Family
   a. 

   b. Descendants of the original transformer model are based on either the encoder stack, the decoder stack, or the original encoder-decoder combination.
2. BERT: Bidirectional Encoder Representations from Transformers
   a. The most significant difference in the models is how they process the input sequence:
   b. BERT consists of:
      i. 

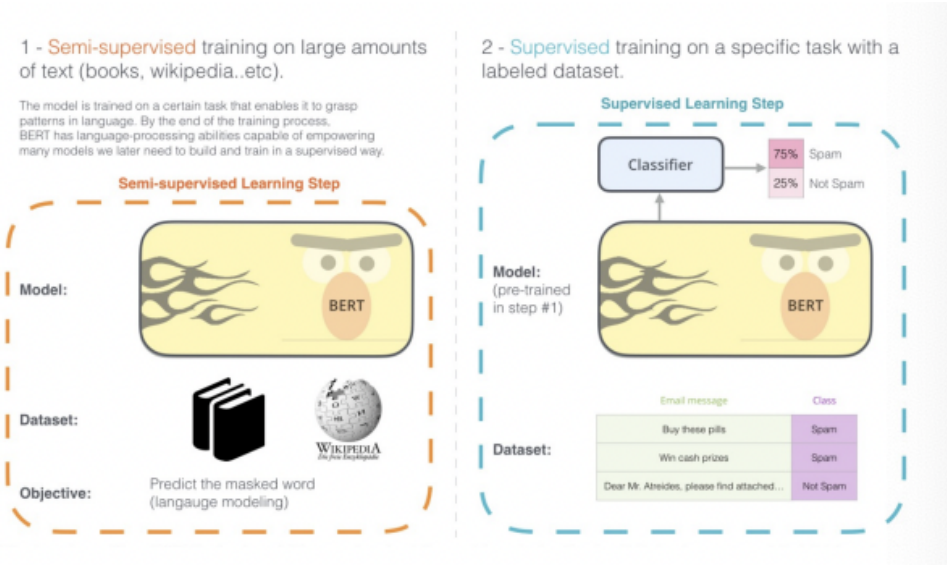
      ii. The stacked-encoder part of the full transformer model, with
      iii. A single linear layer on top, acting as a classifier (depending on the task);
      iv. Pretraining on a Masked Language Model (give a sequence of tokens and mask to predict a word) and Next-Sentence Prediction; and
      v. Transfer learning to adapt to a new task.

c.



i. This is bidirectional

d.

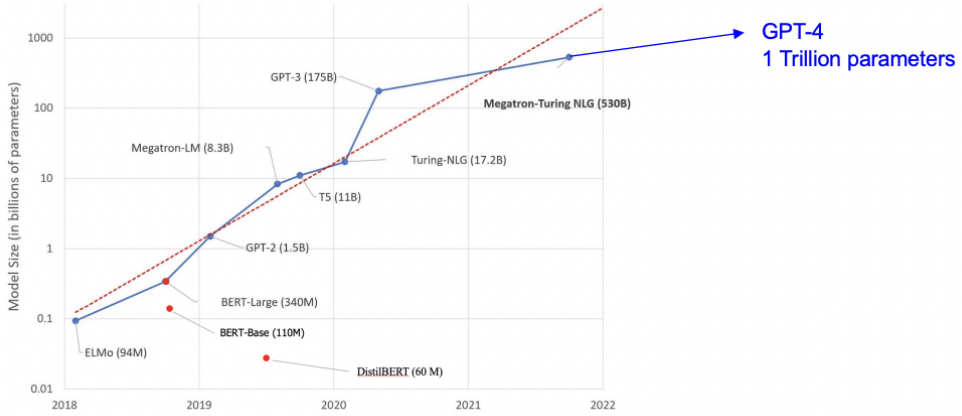


3. NLP Tasks for BERT
    a. Training the model is difficult and the amount of data is critical (how wide, how big, how many layers)
    b. The way various training has developed
    c. BERT understands how words work together (knows the language)
    d. One thing left out is the start token
    e. For classification, you can ignore all the 512 outputs and choose only one to see what category it is (happy, sad, etc.)

f. BERT's training in Masked Language Modeling makes it useful for NLP tasks that involve understanding words in bidirectional context:
    i. Named Entity Recognition
    ii. POS Tagging
    iii. Sentiment Analysis
        1. If the text is positive, the some parts missing will not be negative
    iv. Classification
    v. Coreference Resolution (Connect pronouns with the nouns to which they refer)
g. BERT's training in Next-Sentence Prediction makes it useful for short-range inference about the relationship of sentences:
    i. Question Answering
    ii. Language Inference: Does one sentence imply the other? Are they similar?
h. With fine-tuning, and in conjunction with other models, BERT can be used for more generative tasks, such as Language Generation, Translation, and Summarization.

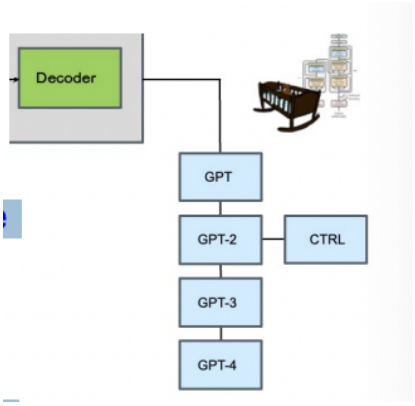
4. BERT Punches Above Its Weight!

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$BERT_{BASE}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$BERT_{LARGE}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**



a.
    i. Exponential growth

5. GPT and Friends
    a. The decoder side of the transformer family tree includes the Generative Pre-Trained Transformer, and similar models (only takes the decoder).
    b. Attempts to simulate the left-to-right context language generation

c. 

d. GPT-1 (2018)
    i. Developed in 2018 based on a generative, decoder-only architecture; o 117M parameters;
    ii. 12 decoder layers;
    iii. 12 masked-attention heads (to simulate auto-regressive language generation) per layer
    iv. Sequence length of 512 tokens
    v. Trained on “predict the next word” task;
    vi. Data set was BookCorpus, 7K unpublished books, 800M words in a variety of genre, including romance, fantasy, scifi, etc.
    vii. Can be fine-tuned for language generation tasks.
e. GPT-2 (2019)
    i. Similar architecture to GPT-1, but:
    ii. 48 layers (from 12 layers)
    iii. 32 masked-attention heads per layer
    iv. 1.5B parameters (from 117M parameters)
    v. Trained on the “predict next word” task on 40GB of text from 8M web pages
        1. Started to worry about the data (how good was the data)
    vi. Because of the large data set, GPT-2 became an “unsupervised multitask learner”: no task-specific data, objectives, or instructions were provided, but GPT-2 was able to perform Q&A, summarization, and translation without explicitly being trained on these tasks.
        1. Predict the next word is so general and dataset was very large, which allowed to do tasks that it was not trained on
    vii. This is sometimes called “zero-shot learning” (do things that were never trained upon) since zero examples were used in training.
    viii. Example: No training in question answering was performed, but because Q & As occurred in the data set, GPT-2 had SOTA performance in Q&A tasks:

ix.

Question	Generated Answer	Correct	Probability
Who wrote the book the origin of species?	Charles Darwin	✓	83.4%
Who is the founder of the ubuntu project?	Mark Shuttleworth	✓	82.0%
Who is the quarterback for the green bay packers?	Aaron Rodgers	✓	81.1%
Panda is a national animal of which country?	China	✓	76.8%
Who came up with the theory of relativity?	Albert Einstein	✓	76.4%
When was the first star wars film released?	1977	✓	71.4%
What is the most common blood type in sweden?	A	✗	70.6%
Who is regarded as the founder of psychoanalysis?	Sigmund Freud	✓	69.3%
Who took the first steps on the moon in 1969?	Neil Armstrong	✓	66.8%
Who is the largest supermarket chain in the uk?	Tesco	✓	65.3%
What is the meaning of shalom in english?	peace	✓	64.0%
Who was the author of the art of war?	Sun Tzu	✓	59.6%

f. GPT-2 was the first time that LLMs showed "emergent properties" wherein its large size created behaviors that were not intended, or desirable (large enough to do strange things).

g. Here are some examples from an early blog post from OpenAI, showing GPT-2 abilities, and which led OpenAI to release it "in stages" (made things up):

System Prompt (human-written)

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

Model Completion (machine-written, 10 tries)

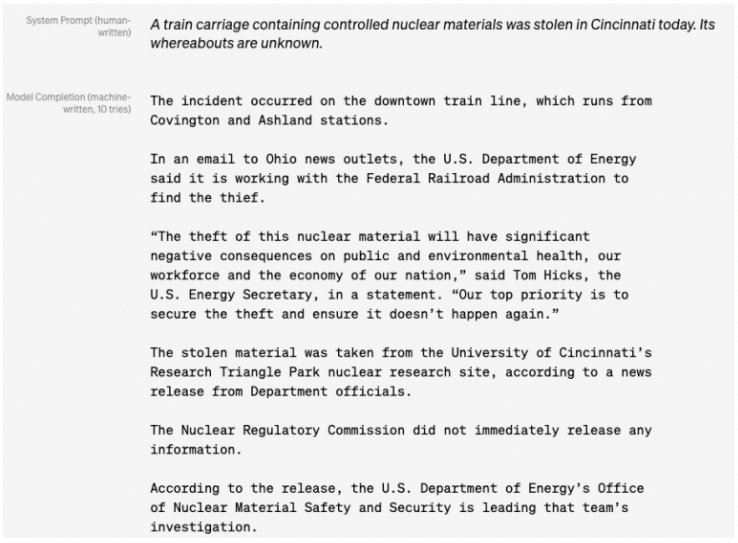
```
The scientist named the population, after their distinctive horn,
Ovid's Unicorn. These four-horned, silver-white unicorns were
previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this
odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of
La Paz, and several companions, were exploring the Andes
Mountains when they found a small valley, with no other animals
or humans. Pérez noticed that the valley had what appeared to be
a natural fountain, surrounded by two peaks of rock and silver
snow.

Pérez and the others then ventured further into the valley. "By
the time we reached the top of one peak, the water looked blue,
with some crystals on top," said Pérez.
```

h. Here are some examples from an early blog post from OpenAI, showing GPT-2 abilities:



System Prompt (human-written)

A train carriage containing controlled nuclear materials was stolen in Cincinnati today. Its whereabouts are unknown.

Model Completion (machine-written, 10 tries)

The incident occurred on the downtown train line, which runs from Covington and Ashland stations.

In an email to Ohio news outlets, the U.S. Department of Energy said it is working with the Federal Railroad Administration to find the thief.

"The theft of this nuclear material will have significant negative consequences on public and environmental health, our workforce and the economy of our nation," said Tom Hicks, the U.S. Energy Secretary, in a statement. "Our top priority is to secure the theft and ensure it doesn't happen again."

The stolen material was taken from the University of Cincinnati's Research Triangle Park nuclear research site, according to a news release from Department officials.

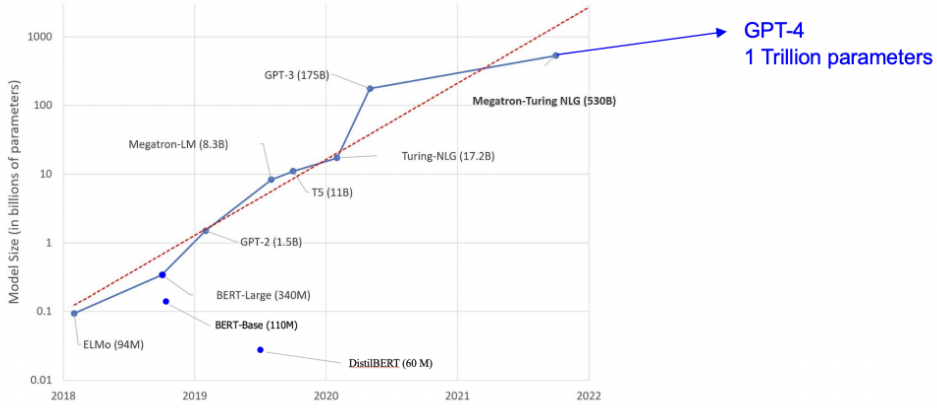
The Nuclear Regulatory Commission did not immediately release any information.

According to the release, the U.S. Department of Energy's Office of Nuclear Material Safety and Security is leading that team's investigation.

Facts all made up by GPT-2!

i. Start to worry people since it made up stuff whether is true or false

6. LLM Power Law Scaling
   a. In further developing the GPT models, explicit reference was made to the observations that LLMs obey power-law (exponential) scaled growth in performance relative to parameter and data set size, and computing resources used in training.



b.

c. Remarkably smooth correlations exist between performance (measured by test loss) and each of these three components of the design and training of LLMs:

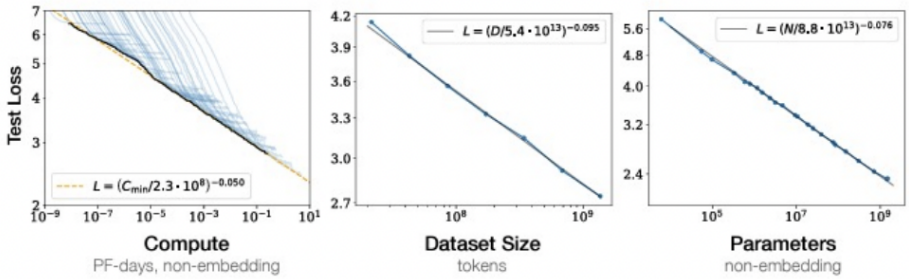


Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

i. If you measure the loss, it became a straight line

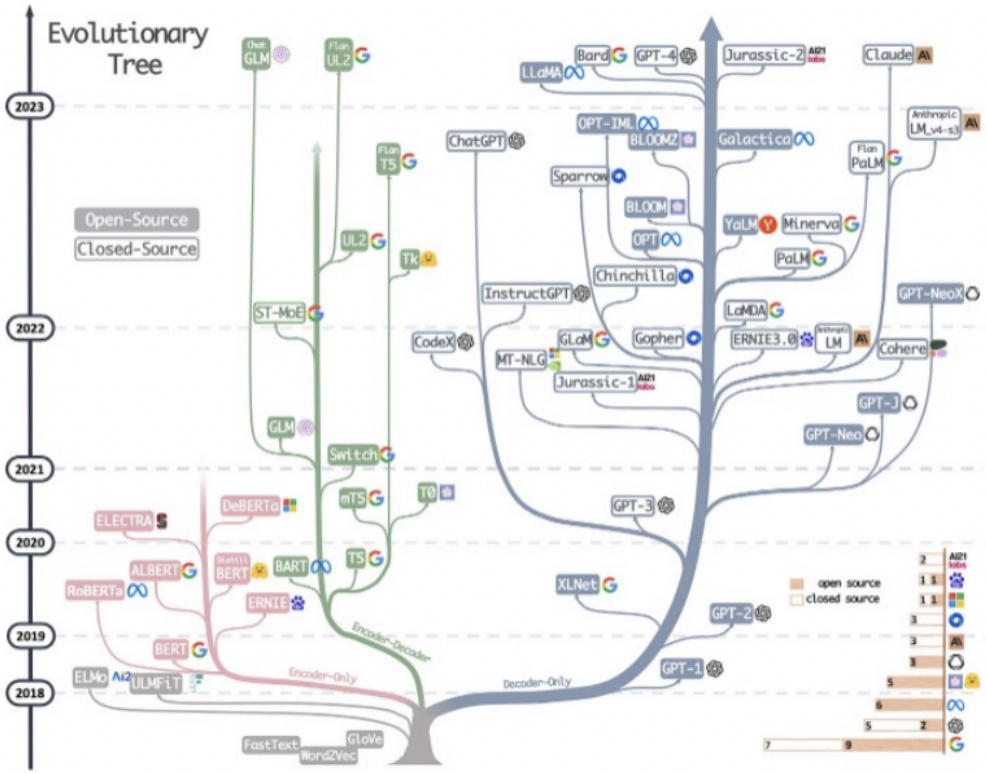
7. GPT and Friends (cont.)
   a. GPT-3 (2020)
      i. 175B parameters
      ii. 96 layers
      iii. 96 masked-attention heads in each layer; used a more efficient alternation of dense and sparse attention patterns
      iv. Input width: 2048 tokens
      v. Used softmax/temperature and Top-p sampling
      vi. Training for GPT-3 was a scaled-up version of GPT-2, including a larger text corpus:
         1. Common Crawl of Web
         2. Entire English Wikipedia
         3. WebText2 (continuation of WebText from GPT-2)
         4. Lots of miscellaneous books, technical manuals, encyclopedias, etc. etc. etc.
8. Encoder-Decoder Models: T5
   a. T5 (2019 – Google)
      i. Uses original encoder-decoder architecture
      ii. Several sizes, largest was 11B, 128 attention heads, 512 input length (everything was text to text)
      iii. Used the "Colossal Clean Crawled Corpus (C4), a curated and extensively cleaned up version of Common Crawl (e.g., removing HTML)
      iv. Training was based on a "denoising" text-to-text task:
      v. Model was trained to reconstruct the original text after random spans of text had been masked (replaced by [MASK]);
      vi. Finetuning was on multiple NLP tasks using "text prefixes:
      vii. "Translate English to German: Natural Language Processing is fun"

viii. Performed almost (88.9) at human level (89.8) on the SuperGLUE benchmark set:

SuperGLUE Tasks

Name	Identifier	Download	More Info	Metric
Broadcoverage Diagnostics	AX-b			Matthew's Corr
CommitmentBank	CB			Avg. F1 / Accuracy
Choice of Plausible Alternatives	COPA			Accuracy
Multi-Sentence Reading Comprehension	MultiRC			F1a / EM
Recognizing Textual Entailment	RTE			Accuracy
Words in Context	WiC			Accuracy
The Winograd Schema Challenge	WSC			Accuracy
BoolQ	BoolQ			Accuracy
Reading Comprehension with Commonsense Reasoning	ReCoRD			F1 / Accuracy
Winogender Schema Diagnostics	AX-g			Gender Parity / Accuracy

9. The Transformer Family Tree



The evolutionary tree of modern LLMs via https://arxiv.org/abs/2304.13712.

a.

10. Most Important NLP Tasks: Classification
    a. Sentiment Analysis: identifying the position of a piece of text in some scale of sentiment.
    b. Position may be categorical (2 stars out of 5) or continuous in some range (2.3 on a scale 0 .. 10)
    c. Types of sentiment:
        i. Positive – Negative
        ii. Aspect or point of view or bias (e.g., political)
        iii. Intent detection
        iv. Emotion Detection
            1. Happiness
            2. Excited/enthusiastic
            3. Frustration or Anger
        v. Friendship, affection, love or sexual attraction
        vi. Humorous
        vii. Irony
        viii. Hate speech and Fake News detection (next slide)
11. Fake News and Hate Speech Detection
    a. Fake News Detection: detecting and filtering out texts containing false and misleading information.
    b. Stance Detection: determining an individual's reaction to a primary actor's claim. It is a core part of a set of approaches to fake news assessment.
    c. Hate Speech Detection: detecting if a piece of text contains hate speech.
12. Information retrieval
    a. Resource Retrieval from text queries/questions
        i. Resource could be
            1. Highly structured (relational database, code)
            2. Semi-structured (Markup Languages (XML), labeled documents)
            3. Unstructured (documents)
        ii. Database search from keywords
        iii. Google search
        iv. Backend to Speech to Text systems (siri)
        v. Question Answering (next slide)
    b. Sentence/document similarity: determining how "similar" two texts are
        i. Notion of "similar" is variable (similar topic, similar sentiment, ...)
        ii. Relationship to IR:
            1. How similar is text query to a document?
            2. "Retrieve more documents similar to this one"
        iii. Create a map/graph of documents similar to given sentence/document
        iv. Plagiarism/copyright infringement

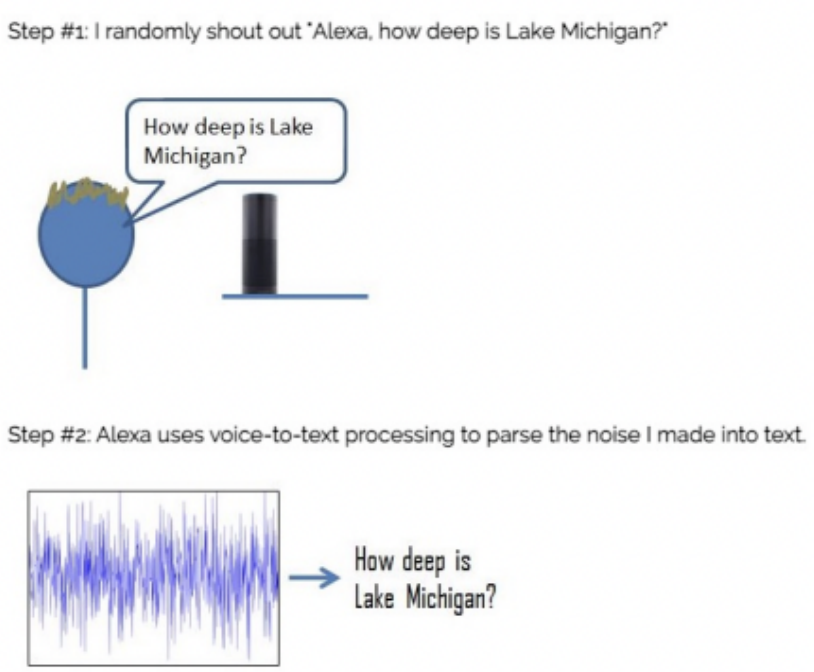
c. Document Ranking: Rank documents as to some criterion (e.g., PageRank)
    i. How well does this document satisfy my query?
    ii. How important/authoritative is this document?

13. Text-to-Text Generation
    a. Machine Translation: translating from one language to another.
        i. Covered in lecture – Transformer technology transformed this task
    b. Text Generation: creating text from a prompt or subject phrase that appears indistinguishable from human-written text.
        i. Covered in lecture – Use language models, Large Language Models (GPT) have transformed this task
    c. Lexical Normalization: translating/transforming a non-standard text to a standard register.
    d. Paraphrase Generation: creating an output sentence that preserves the meaning of input but includes variations in word choice and grammar.
    e. Text Simplification: making a text easier to read and understand, while preserving its main ideas and approximate meaning.
    f. Text Summarization
14. Topics and Keywords; Text Summarization
    a. Topic Modeling: identifying abstract "topics" underlying a collection of documents.
    b. Keyword Extraction: identifying the most relevant terms to describe the subject of a document
    c. Text Summarization: Reducing size of document while preserving the most important information
        i. Extractive:
            1. Identify the most important sentences in a document and construct the summary from these exact sentences
            2. TextRank, LexRank (implements PageRank on sentences in a document)
            3. Latent Semantic Analysis (Singular Value Decomposition on a wordsentence matrix)
        ii. Abstractive:
            1. Create new text summarizing main points
            2. Use of Large Language Models: GPT, BERT, etc...
        iii. Use cases for Text Summarization:
            1. Summaries for busy executives or (students!)
            2. Summaries of articles, books, chapters
            3. Automatic Table of Contents or Indices

4. Downstream from Speech-to-Text systems
    a. Notetaking of meetings, lectures
    b. Abstracts of podcasts, YouTube videos
    c. Automatic summary of customer phone calls

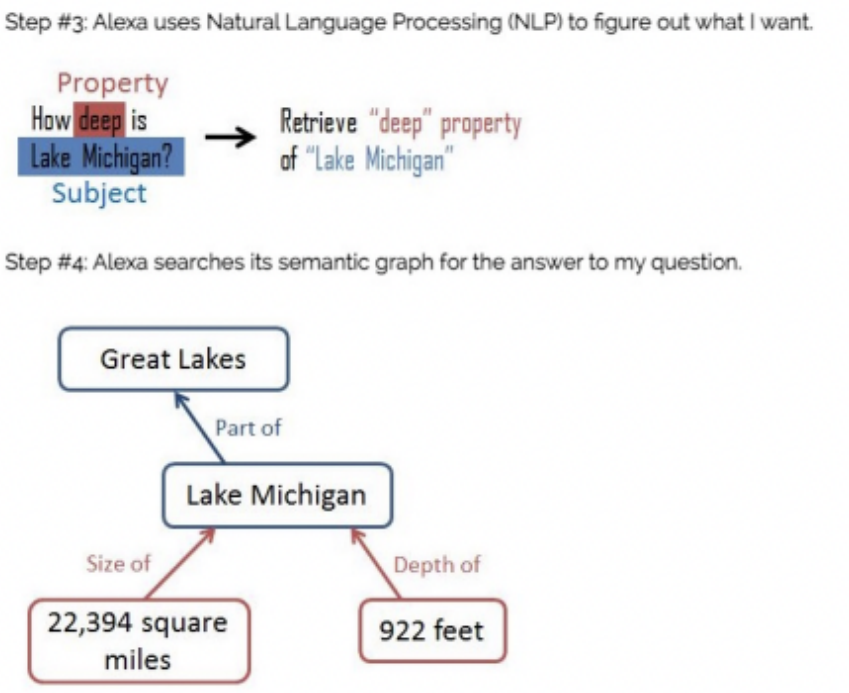
15. Chatbots and Question Answering
    a. Slot Filling or Cloze Task: aims to extract the values of certain types of attributes (or slots, such as cities or dates) for a given entity from texts.
    b. Chatbots: Conversation agents (started with Eliza in early 1060's!)
    c. Dialog Management: managing of state and flow of conversations.
    d. Question Answering: Responding to textual queries with textual answers
        i. Extractive QA: The model extracts the answer from a knowledge source, such as a knowledge graph, database, or document (next slide).
        ii. Open Generative QA: The model generates free text directly based on the (global) context.
        iii. Closed Generative QA: The model generates free text directly based only on the question.

16. Question Answering Using Knowledge Graphs

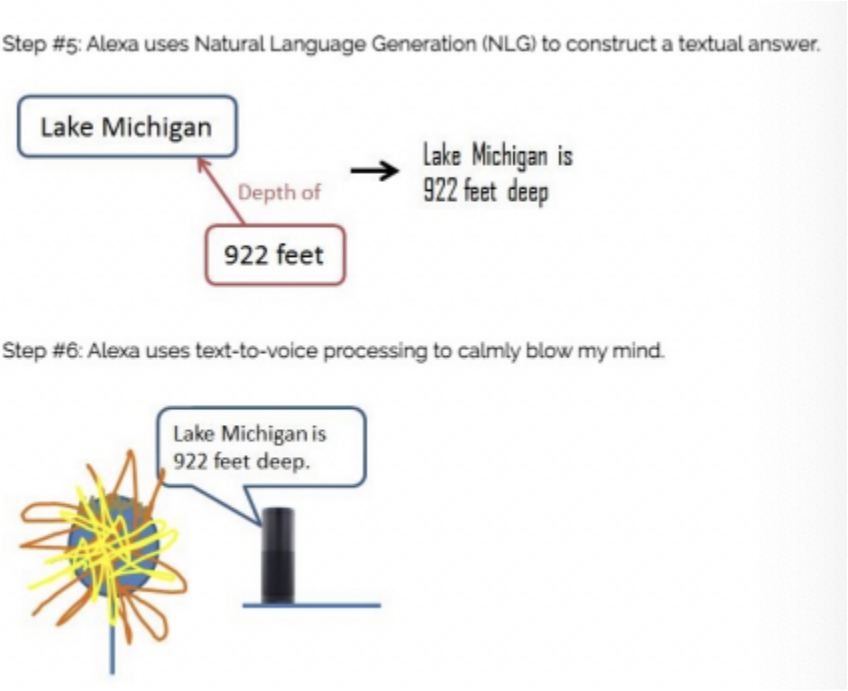


    a.

b.



c.



17. Reasoning with Text
    a. Logical Relationship of two sentences/documents:
        i. Entailment
        ii. Temporal sequence
        iii. Specialization
    b. Subsystem of text generation at scale

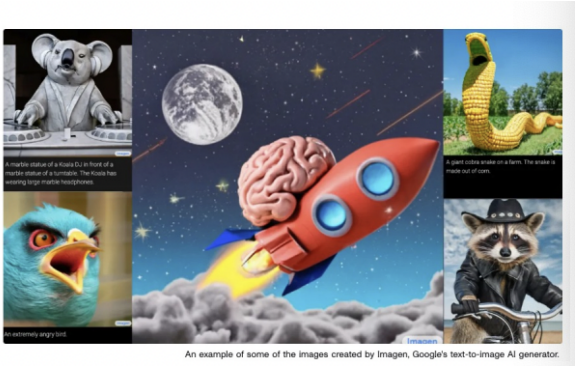
c. Text-to/from-First Order Logic: Translate between text and expressions in first-order logic:

No student failed Chemistry, but at least one student failed History.
$\neg\exists x\,(\text{Student}(x) \wedge \text{Failed}(x,\text{Chemistry})) \wedge \exists x\,(\text{Student}(x) \wedge \text{Failed}(x,\text{History}))$

10. Logic Puzzle: A farmer wants to cross a river and take with him a wolf, a goat and a cabbage. He has a boat, but it can only fit himself plus either the wolf, the goat or the cabbage. If the wolf and the goat are alone on one shore, the wolf will eat the goat. If the goat and the cabbage are alone on the shore, the goat will eat the cabbage. How can the farmer bring the wolf, the goat and the cabbage across the river without anything being eaten?

d. Use cases:
   i. Teaching logic
   ii. Game/puzzle solving
   iii. Interface to automated theorem prover
      1. Prolog
      2. Planner
      3. Wolfram Alpha

18. Text-to-Data and Data-to-Text
   a. Text-to-Image: generating photo-realistic images which are semantically consistent with the text descriptions.
   b. Image captioning: Generate captions for input images
   c. Video-to-Text: Generating text describing a sequence of images
   d. Text-to-Speech: Human-like reading of input text.
   e. Speech-to-Text: transcribing speech to text
   f.


An example of some of the images created by Imagen, Google's text-to-image AI generator.