Week 14

Machine Learning and Big Data - DATA622

CUNY School of Professional Studies

Meaning of words

Words may have many meanings (polysemy). The meaning of a word depends on its context.

An Example: ***bow***

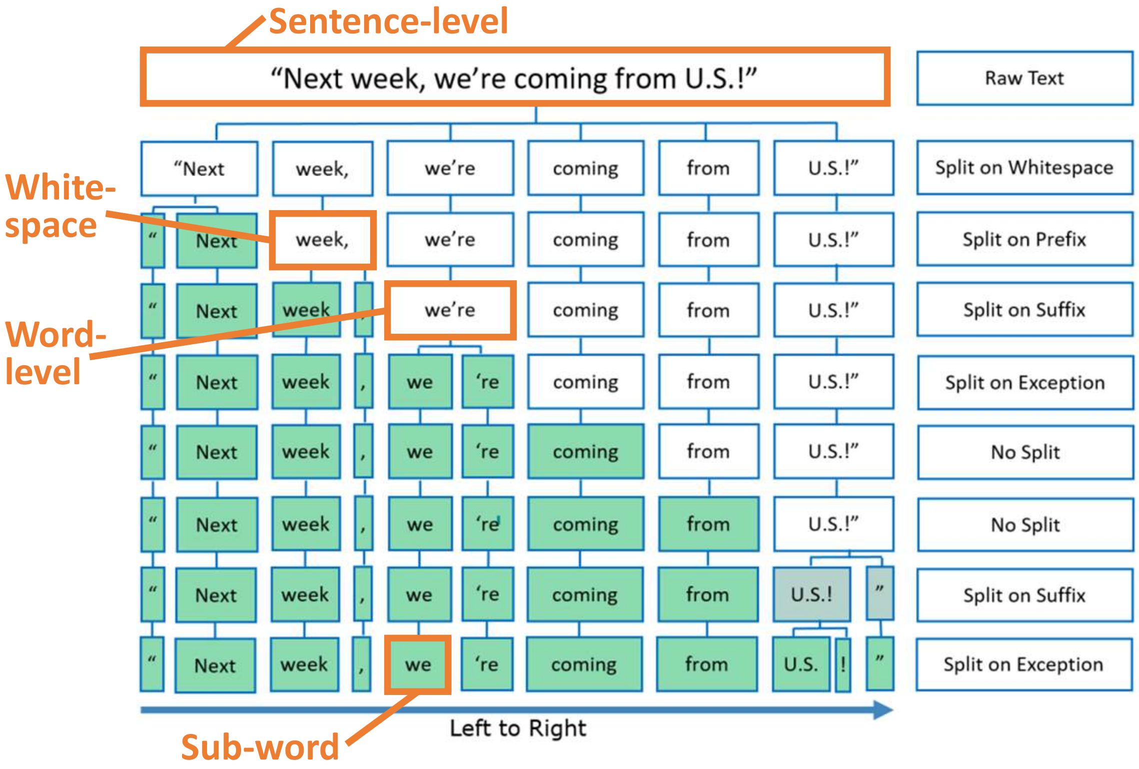
- the front of a ship
- to bend forward in respect
- a weapon that shoots arrows
- to bend outward

Tokenization

Tokenization is the processes of splitting text into manageable pievces: tokens

Types of tokenization:

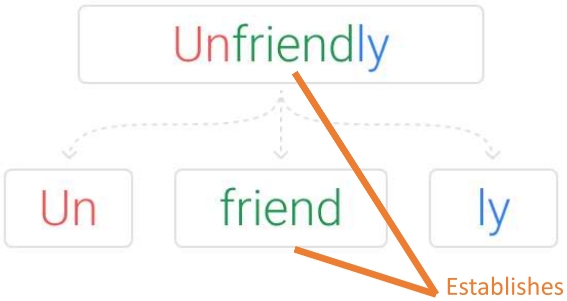
- Character-level
- Subword-level
- Word-level
- Whitespace-level
- Sentence-level



“Next week, we’re coming from U.S.!”										Raw Text
“Next	week,	we’re	coming	from	U.S.!”					Split on Whitespace
“	Next	week,	we’re	coming	from	U.S.!”				Split on Prefix
“	Next	week	,	we’re	coming	from	U.S.!”			Split on Suffix
“	Next	week	,	we	‘re	coming	from	U.S.!”		Split on Exception
“	Next	week	,	we	‘re	coming	from	U.S.!”		No Split
“	Next	week	,	we	‘re	coming	from	U.S.!”		No Split
“	Next	week	,	we	‘re	coming	from	U.S.!	”	Split on Suffix
“	Next	week	,	we	‘re	coming	from	U.S.	!	” Split on Exception

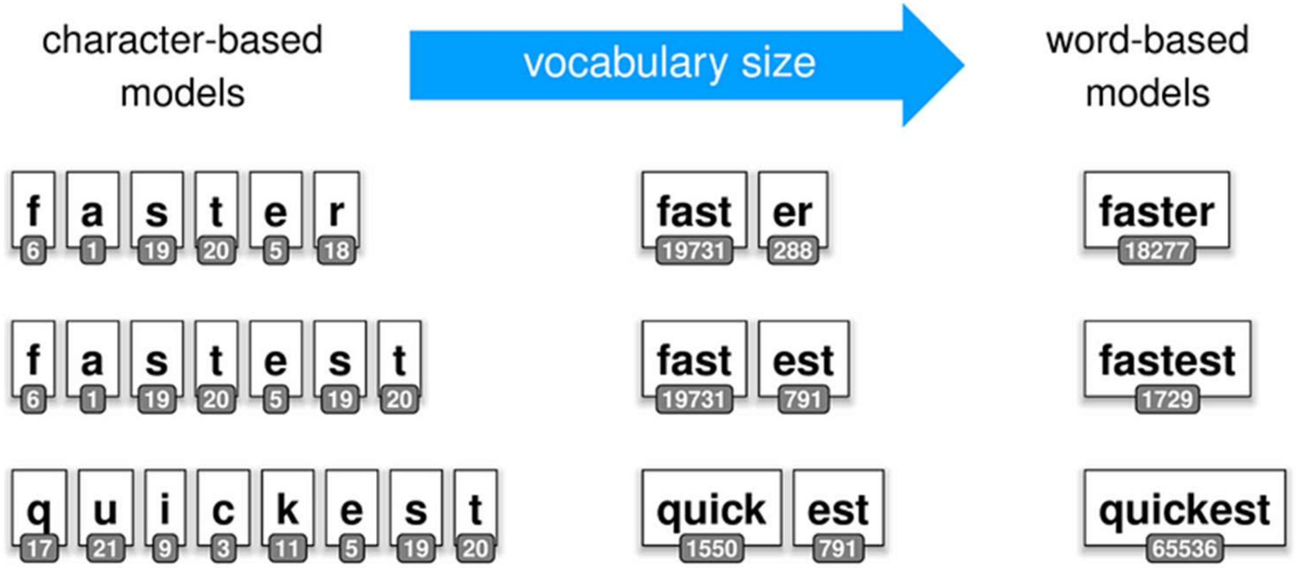
Tokenization

Sub-word is most popular, best balance
of vocabulary size and retention of meaning

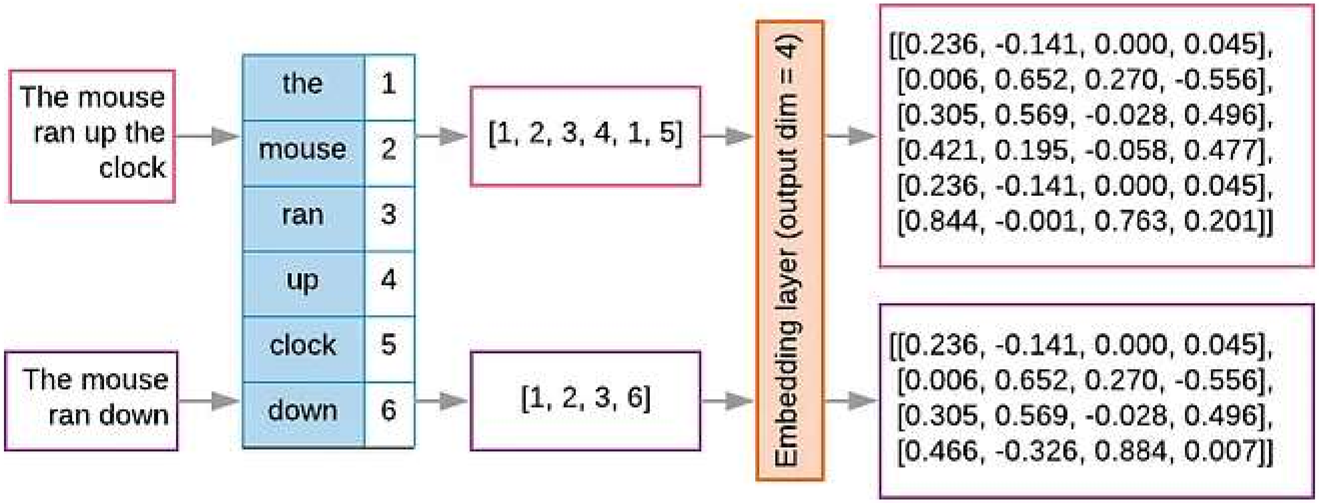


Establishes link between *unfriendly* and an existing token *friend*, and not a completely different concept (token)

Tokenization



Embeddings



The mouse ran up the clock

The mouse ran down

the	1
mouse	2
ran	3
up	4
clock	5
down	6

[1, 2, 3, 4, 1, 5]

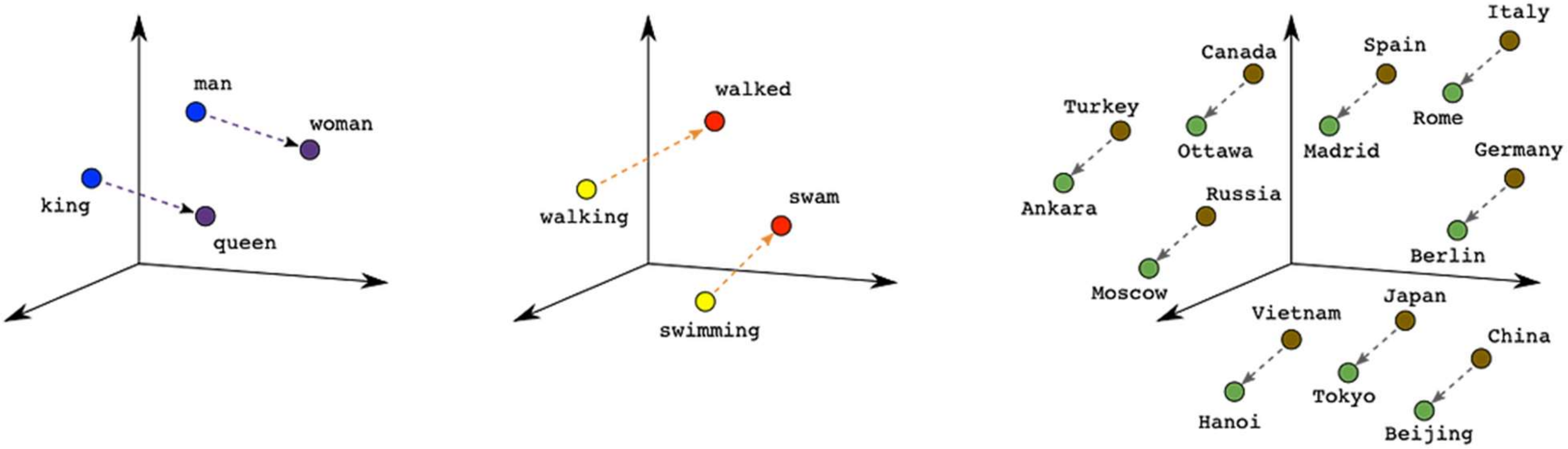
[1, 2, 3, 6]

Embedding layer (output dim = 4)

```
[[0.236, -0.141, 0.000, 0.045],
 [0.006, 0.652, 0.270, -0.556],
 [0.305, 0.569, -0.028, 0.496],
 [0.421, 0.195, -0.058, 0.477],
 [0.236, -0.141, 0.000, 0.045],
 [0.844, -0.001, 0.763, 0.201]]
```

```
[[0.236, -0.141, 0.000, 0.045],
 [0.006, 0.652, 0.270, -0.556],
 [0.305, 0.569, -0.028, 0.496],
 [0.466, -0.326, 0.884, 0.007]]
```

Embeddings

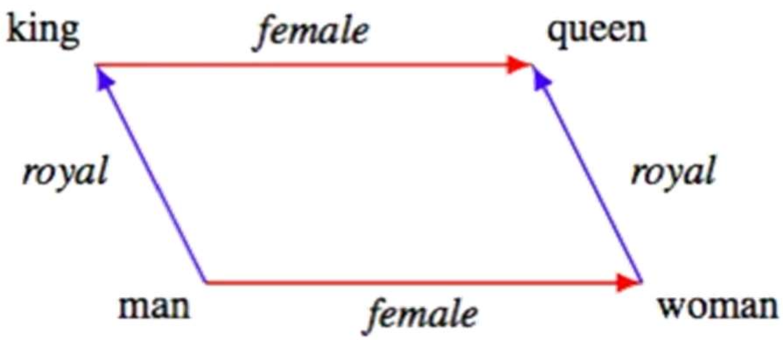


Male-Female

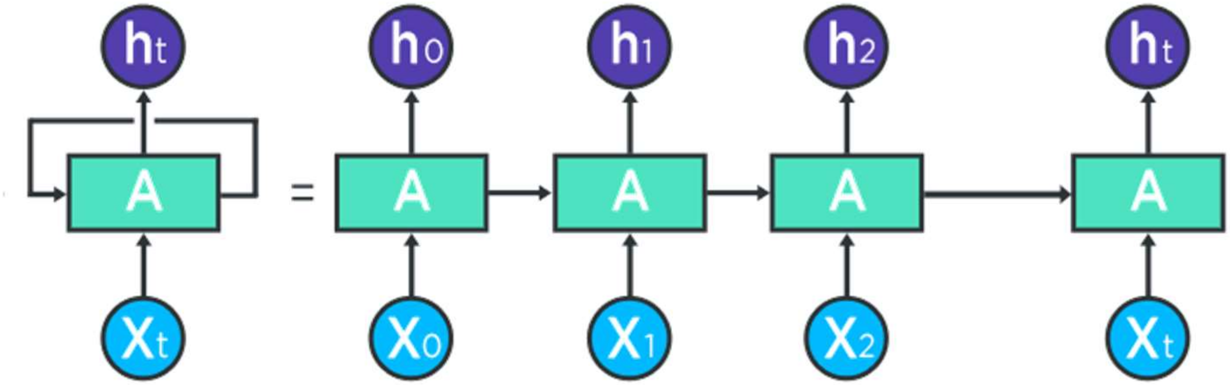
Verb Tense

Country-Capital

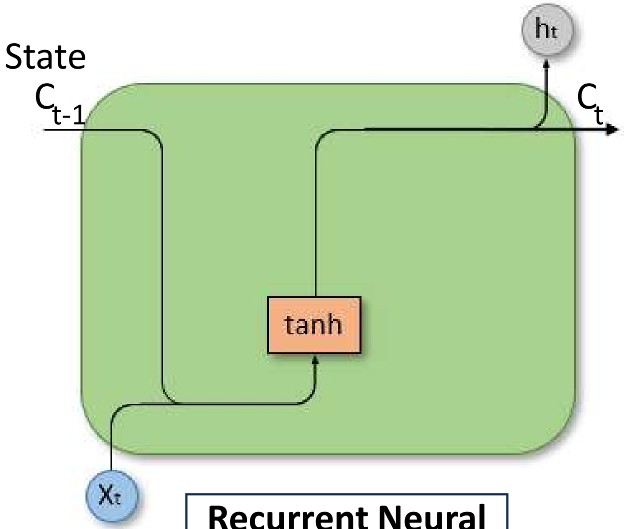
Embeddings



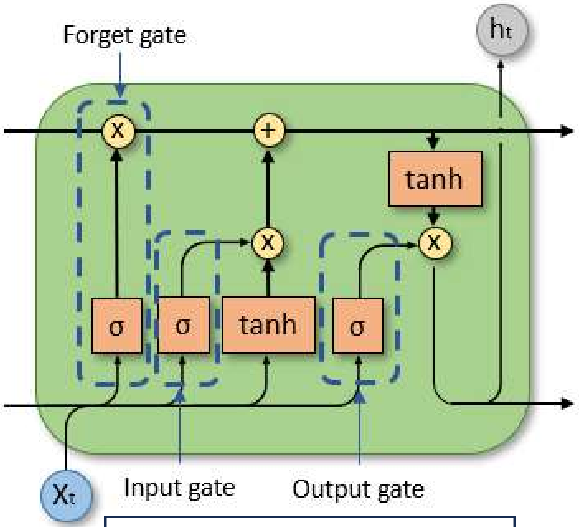
RNN



Evolution: RNN – LSTM - GRU

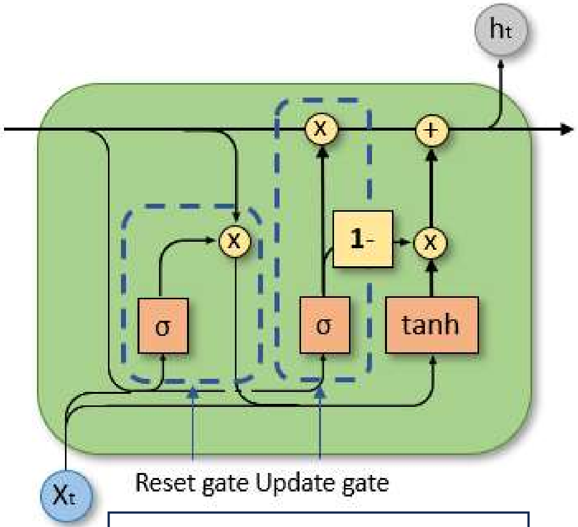


Recurrent Neural Network (RNN)



Long Short-Term Memory (LSTM)

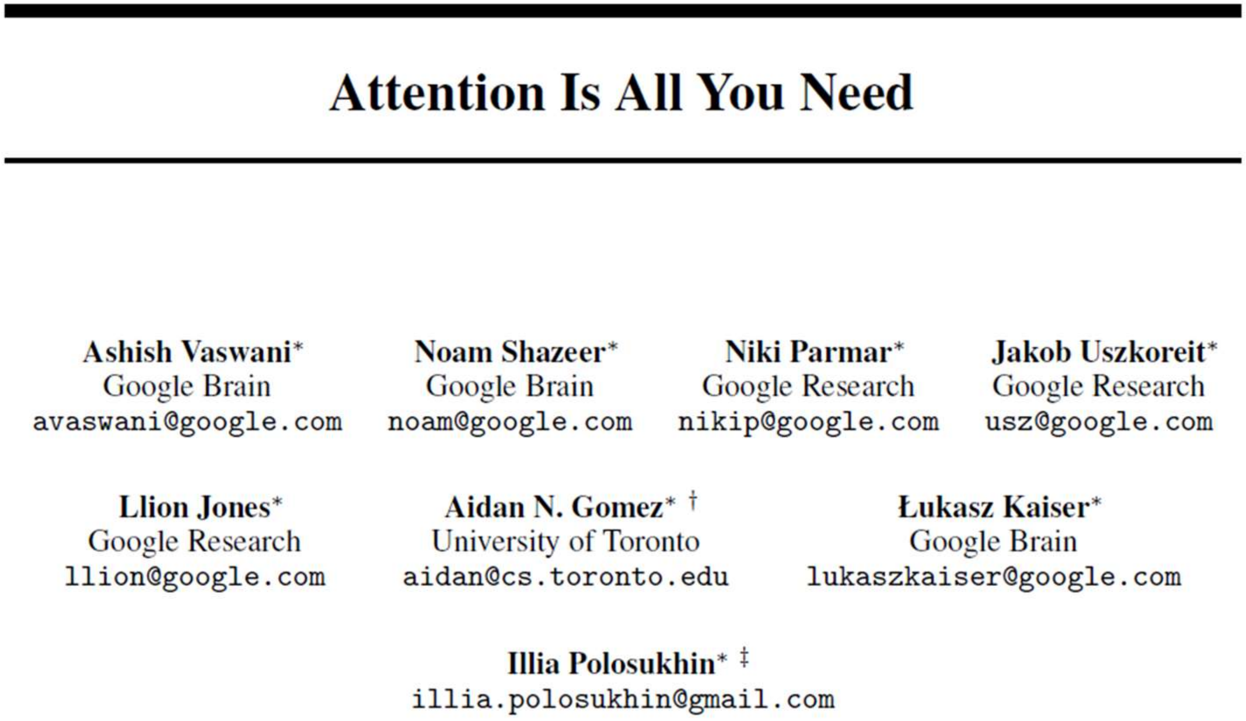
The LSTM adds three gates (forget gate, input gate, and output gate) to the RNN



Gated recurrent unit (GRU)

Variant of the LSTM that synthesizes forget and input gate into a single update gate

Attention



Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

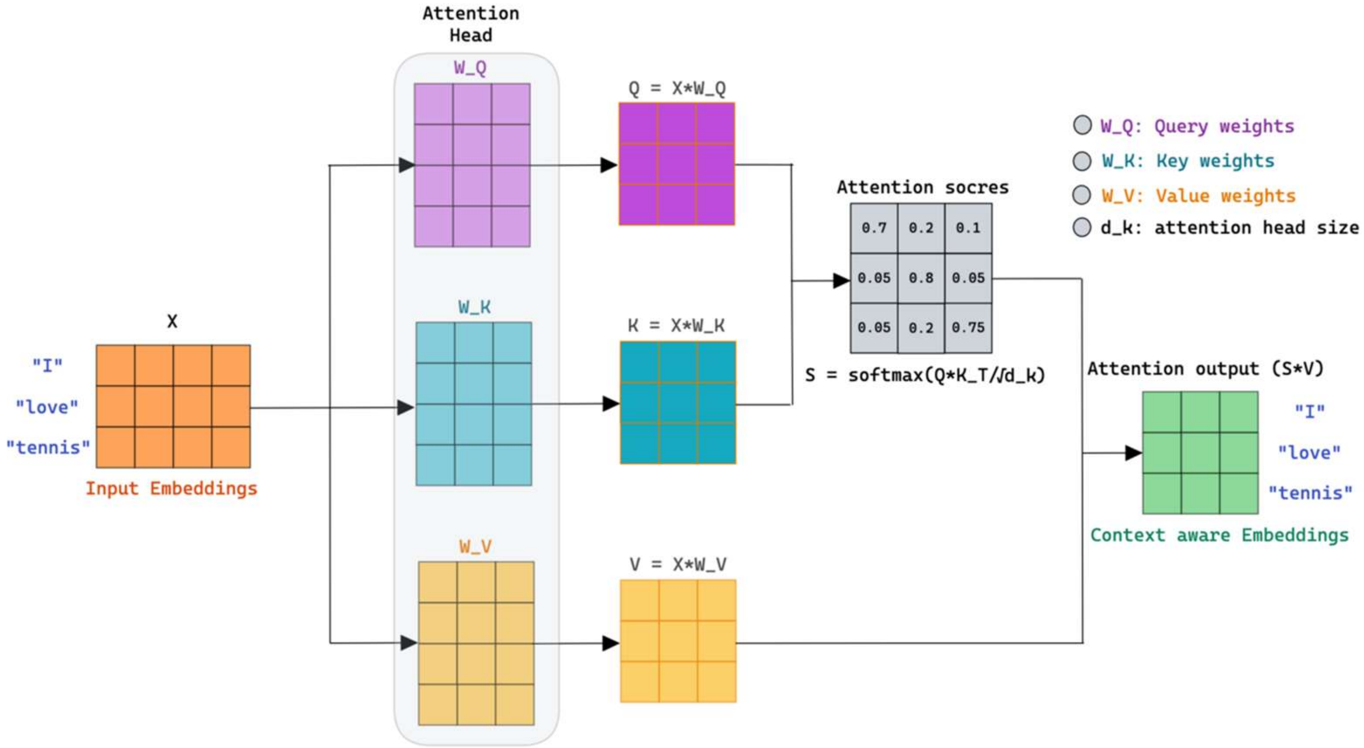
Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser*
Google Brain
lukaszkaiser@google.com

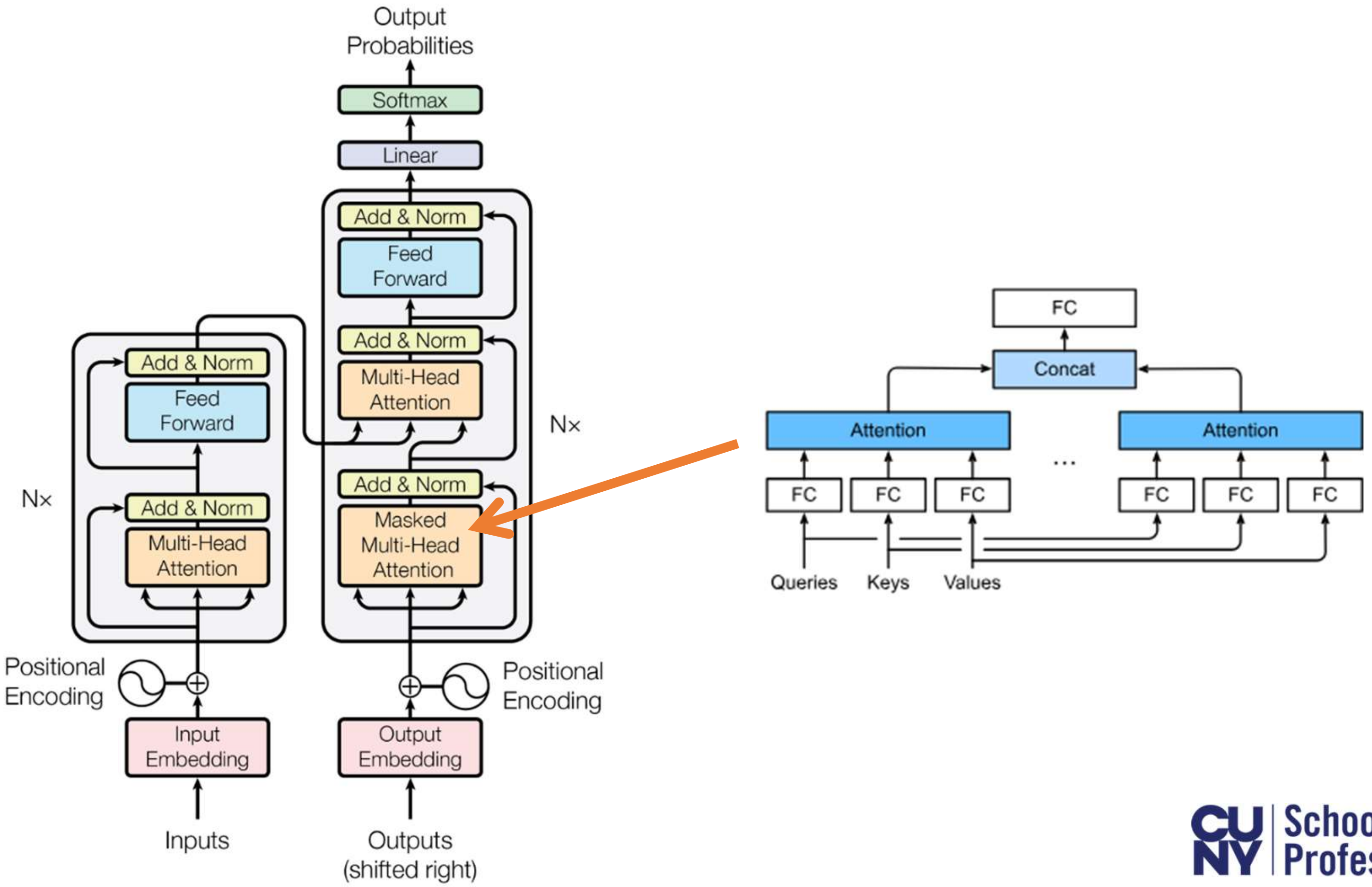
Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Attention



0.7	0.2	0.1
0.05	0.8	0.05
0.05	0.2	0.75

Transformers



Large Language Models

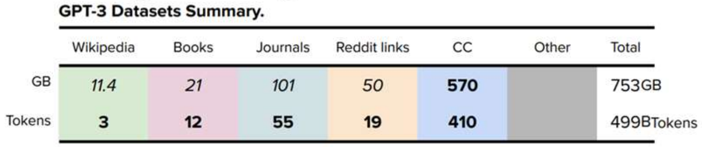
CUNY School of Professional Studies

Large Language Models

Large Language Models (LLMs)

- Pre-trained with extremely large datasets – architected to scale
- Can be adopted to a wide range of downstream tasks
- LLMs are Foundation Models.

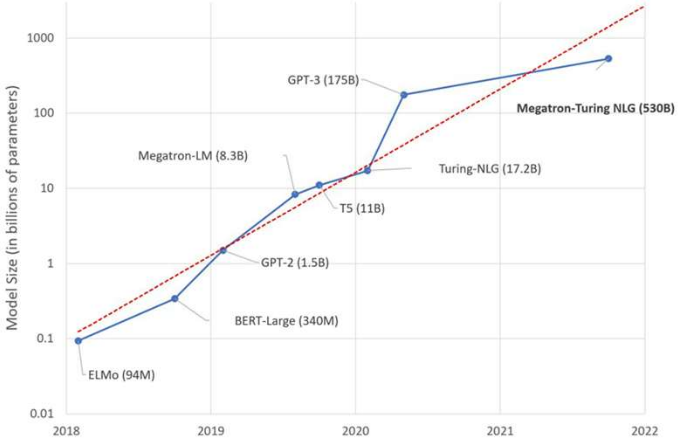
Very Large training datasets



GPT-3 Datasets Summary.

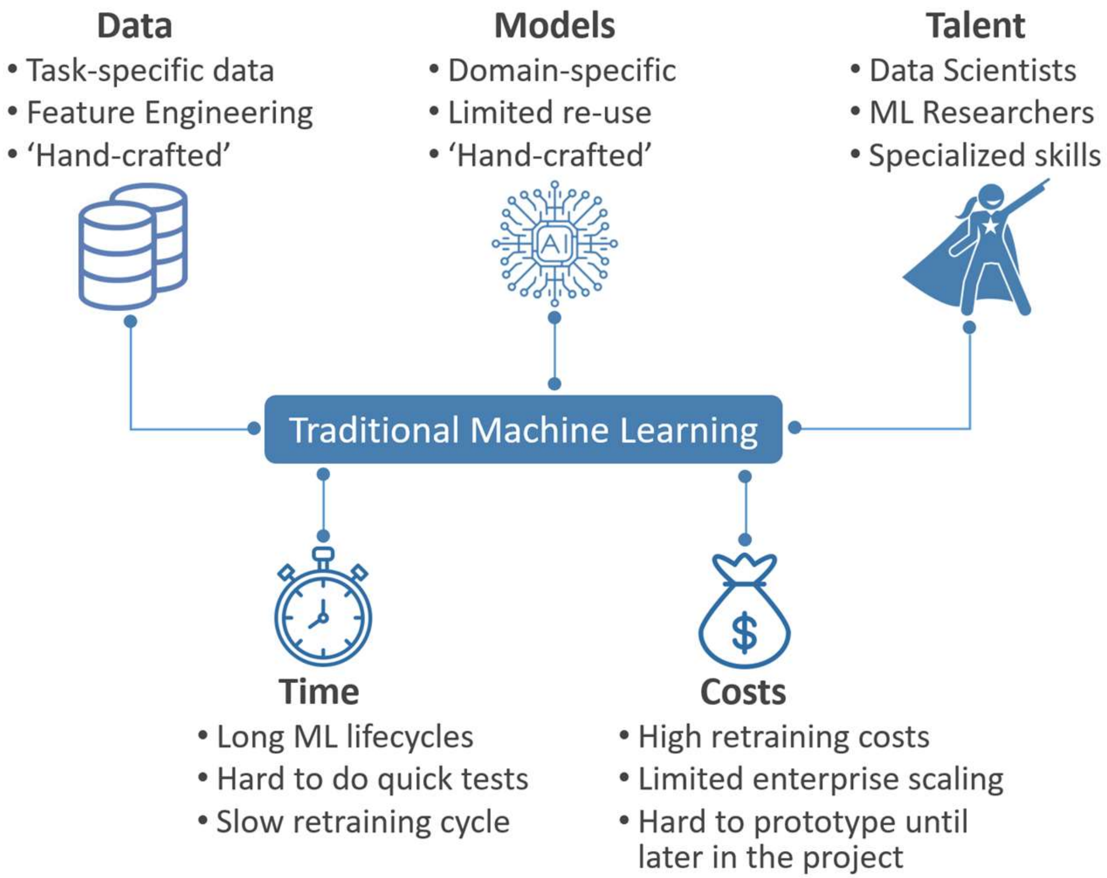
	Wikipedia	Books	Journals	Reddit links	CC	Other	Total
GB	*11.4*	*21*	*101*	*50*	**570**		753GB
Tokens	**3**	**12**	**55**	**19**	**410**		499BTokens

LLM Scaling: a new Moore's Law



CUNY School of Professional Studies

Challenges of ML Today



Data

- Task-specific data
- Feature Engineering
- ‘Hand-crafted’

Models

- Domain-specific
- Limited re-use
- ‘Hand-crafted’

Talent

- Data Scientists
- ML Researchers
- Specialized skills

Traditional Machine Learning

Time

- Long ML lifecycles
- Hard to do quick tests
- Slow retraining cycle

Costs

- High retraining costs
- Limited enterprise scaling
- Hard to prototype until later in the project

Benefits of LLMs

Increased Velocity

- Focus shifts from training models from scratch, to fine-tune models
- Faster time to market

Increased Opportunity for AI/ML involvement

- Potential to scale to a wider pool of users to perform AI/ML
- Simple text interface and natural language instruction

Cost effective

- Scale to multiple use-cases per LLM
- Faster / lower-cost prototyping

Tapping into state-of-the-art AI

- Few-shot learners (and ‘Surprisingly good without fine tuning’)
- Perform tasks not explicitly trained on

Emergent Capabilities

- Emergent capabilities that surface with LLM size
  i.e. capabilities not present in smaller models but emerge in larger models
- LLM scale highly correlated with downstream performance[1]

Benefits of LLMs

Fine Tuning

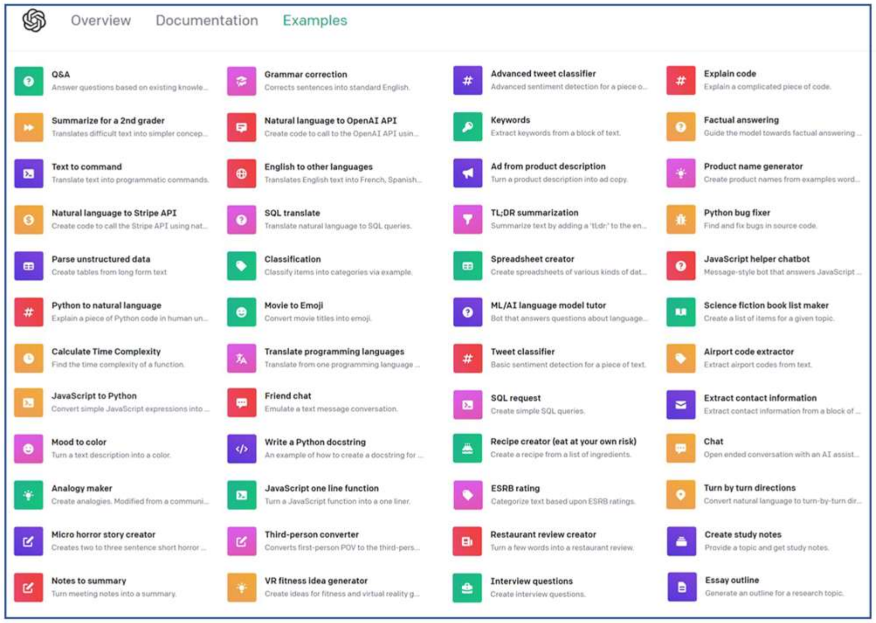
- Load foundational model
- Add task-specific prompts
- Minimal data, compute, time
- SOTA results

Embeddings

- Encode Content into dense vector for downstream use
- Use in downstream models or similarity search

In-context Learning

- “Ask” the model to perform a task as part of the input
- Provide examples to help



Q&A Answer questions based on existing knowle...	**Grammar correction** Corrects sentences into standard English.	**Advanced tweet classifier** Advanced sentiment detection for a piece o...	**Explain code** Explain a complicated piece of code.
Summarize for a 2nd grader Translates difficult text into simpler concep...	**Natural language to OpenAI API** Create code to call to the OpenAI API usin...	**Keywords** Extract keywords from a block of text.	**Factual answering** Guide the model towards factual answering ...
Text to command Translate text into programmatic commands.	**English to other languages** Translates English text into French, Spanish...	**Ad from product description** Turn a product description into ad copy.	**Product name generator** Create product names from examples word...
Natural language to Stripe API Create code to call the Stripe API using nat...	**SQL translate** Translate natural language to SQL queries.	**TL;DR summarization** Summarize text by adding a 'tl;dr:' to the en...	**Python bug fixer** Find and fix bugs in source code.
Parse unstructured data Create tables from long form text	**Classification** Classify items into categories via example.	**Spreadsheet creator** Create spreadsheets of various kinds of dat...	**JavaScript helper chatbot** Message-style bot that answers JavaScript ...
Python to natural language Explain a piece of Python code in human un...	**Movie to Emoji** Convert movie titles into emoji.	**ML/AI language model tutor** Bot that answers questions about language...	**Science fiction book list maker** Create a list of items for a given topic.
Calculate Time Complexity Find the time complexity of a function.	**Translate programming languages** Translate from one programming language ...	**Tweet classifier** Basic sentiment detection for a piece of text.	**Airport code extractor** Extract airport codes from text.
JavaScript to Python Convert simple JavaScript expressions into ...	**Friend chat** Emulate a text message conversation.	**SQL request** Create simple SQL queries.	**Extract contact information** Extract contact information from a block of ...
Mood to color Turn a text description into a color.	**Write a Python docstring** An example of how to create a docstring for ...	**Recipe creator (eat at your own risk)** Create a recipe from a list of ingredients.	**Chat** Open ended conversation with an AI assist...
Analogy maker Create analogies. Modified from a communi...	**JavaScript one line function** Turn a JavaScript function into a one liner.	**ESRB rating** Categorize text based upon ESRB ratings.	**Turn by turn directions** Convert natural language to turn-by-turn dir...
Micro horror story creator Creates two to three sentence short horror ...	**Third-person converter** Converts first-person POV to the third-pers...	**Restaurant review creator** Turn a few words into a restaurant review.	**Create study notes** Provide a topic and get study notes.
Notes to summary Turn meeting notes into a summary.	**VR fitness idea generator** Create ideas for fitness and virtual reality g...	**Interview questions** Create interview questions.	**Essay outline** Generate an outline for a research topic.

Risks of LLMs

Bias Propagation

- Potential bias / toxic output
- Responsible AI is critical

LLMs fail in subtle ways

- Hallucinations
- Evaluation / safeguards required

Increasing model scale

- Exponential growth in size
- Complexity in training / deployment

LLM costs

- Cost / Latency trade-off
- Understand & manage costs

Closed-Source models

- Legal restrictions to some models