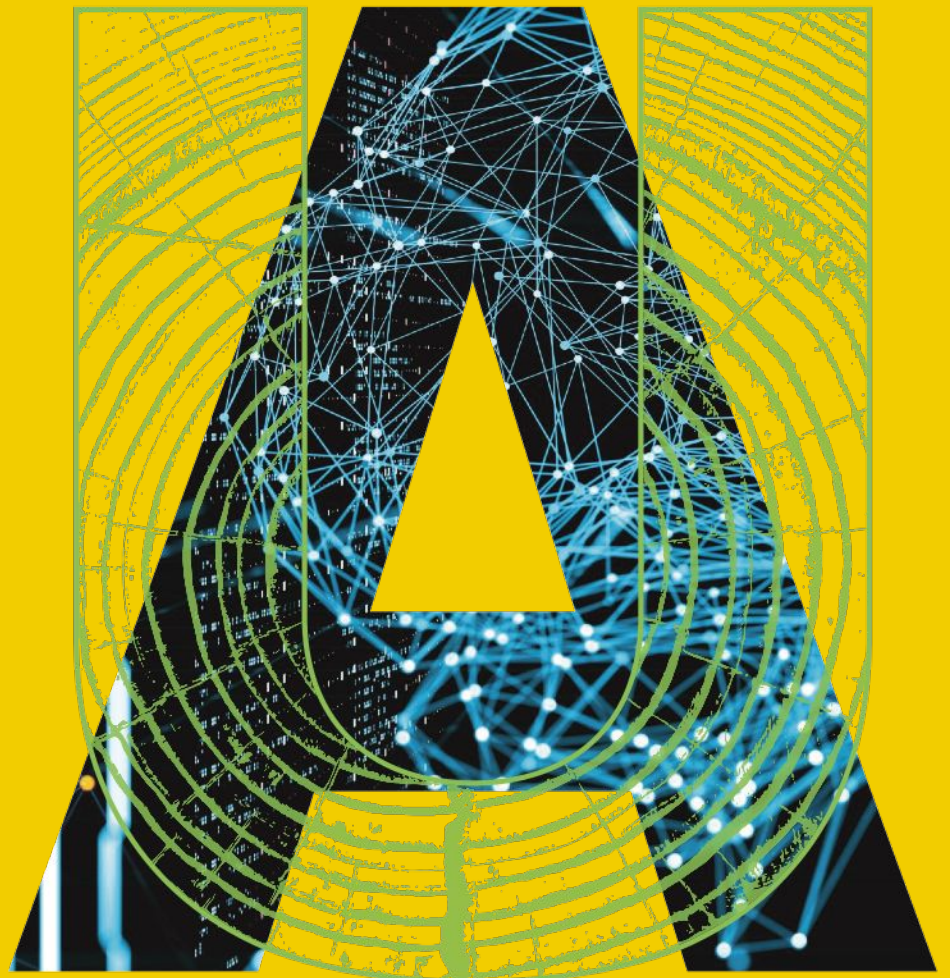
MACHINE LEARNING & THE BRAIN

Language Models & Language Neuroscience

Thursday 05 October 2023

Alex Murphy

UNIVERSITY OF ALBERTA



Today

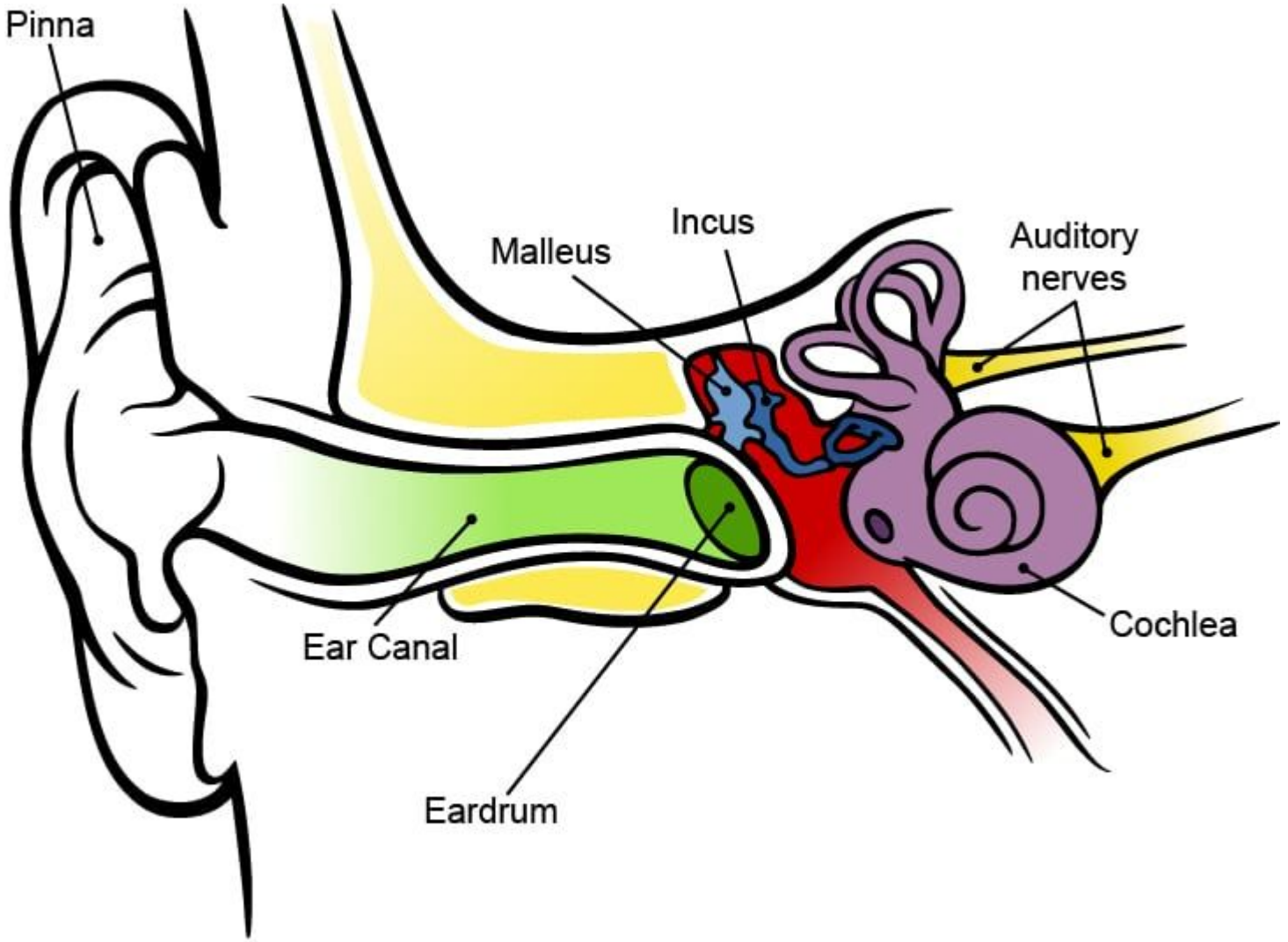
05 September 2023:	**Introduction to Neuroscience and Machine Learning**
12 September 2023:	**The Visual System & CNNs**
28 September 2023:	**Coding Workshop**
05 October 2023:	**Language Models & Language Neuroscience**
31 October 2023:	**Decision Making / Planning & Reinforcement Learning**

Language Input Streams

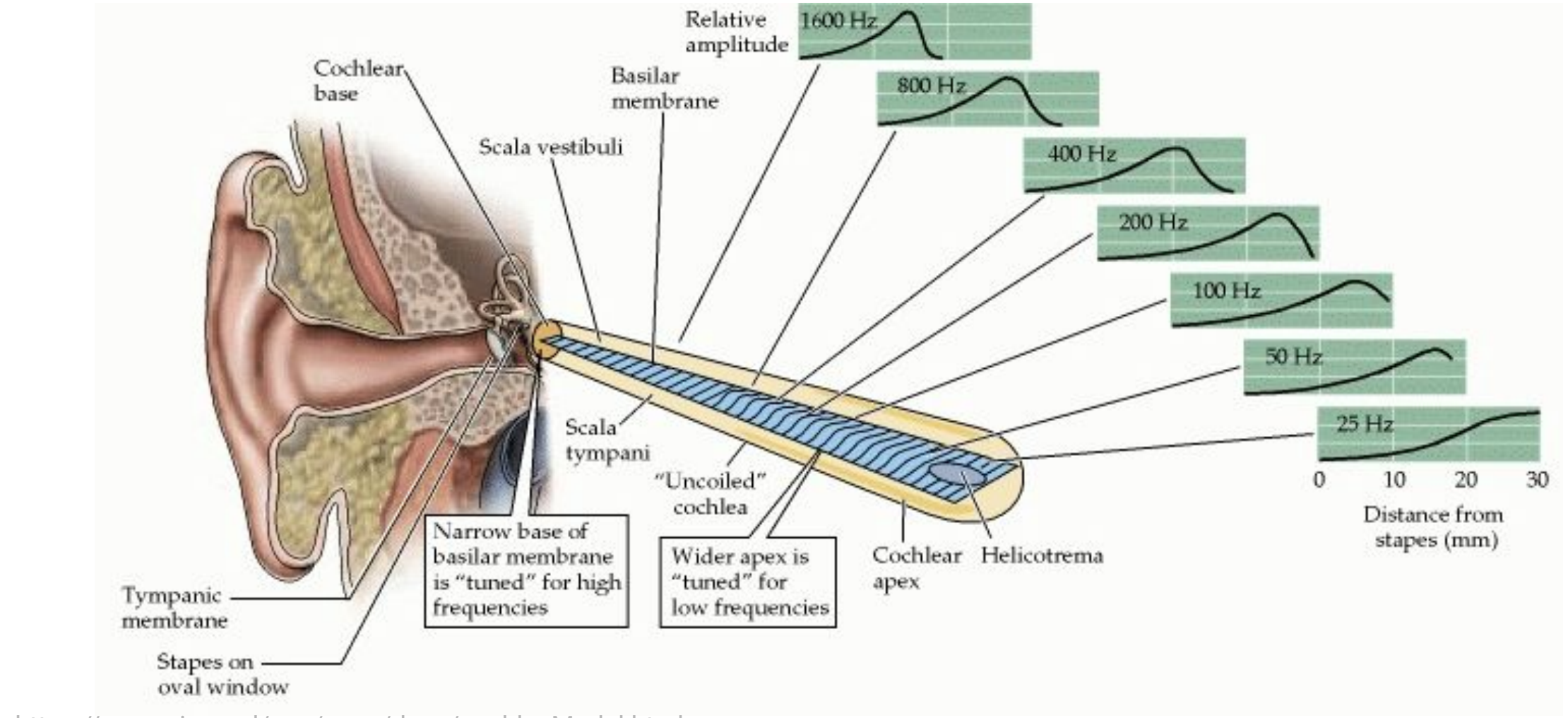
- Speech
- Reading
- Sign Language
- Touch (Braille)

Speech Processing

- Sound hits the ear
- Travels down ear canal
- Vibrations hit the eardrum
- Eardrum pushes the ossicles
- Ossicles disturb fluid in the cochlea
- Different frequencies activate different regions of the cochlea
- Hair cells disturbed in cochlea
- Nerves then send electrical info along the auditory nerve up to the brain

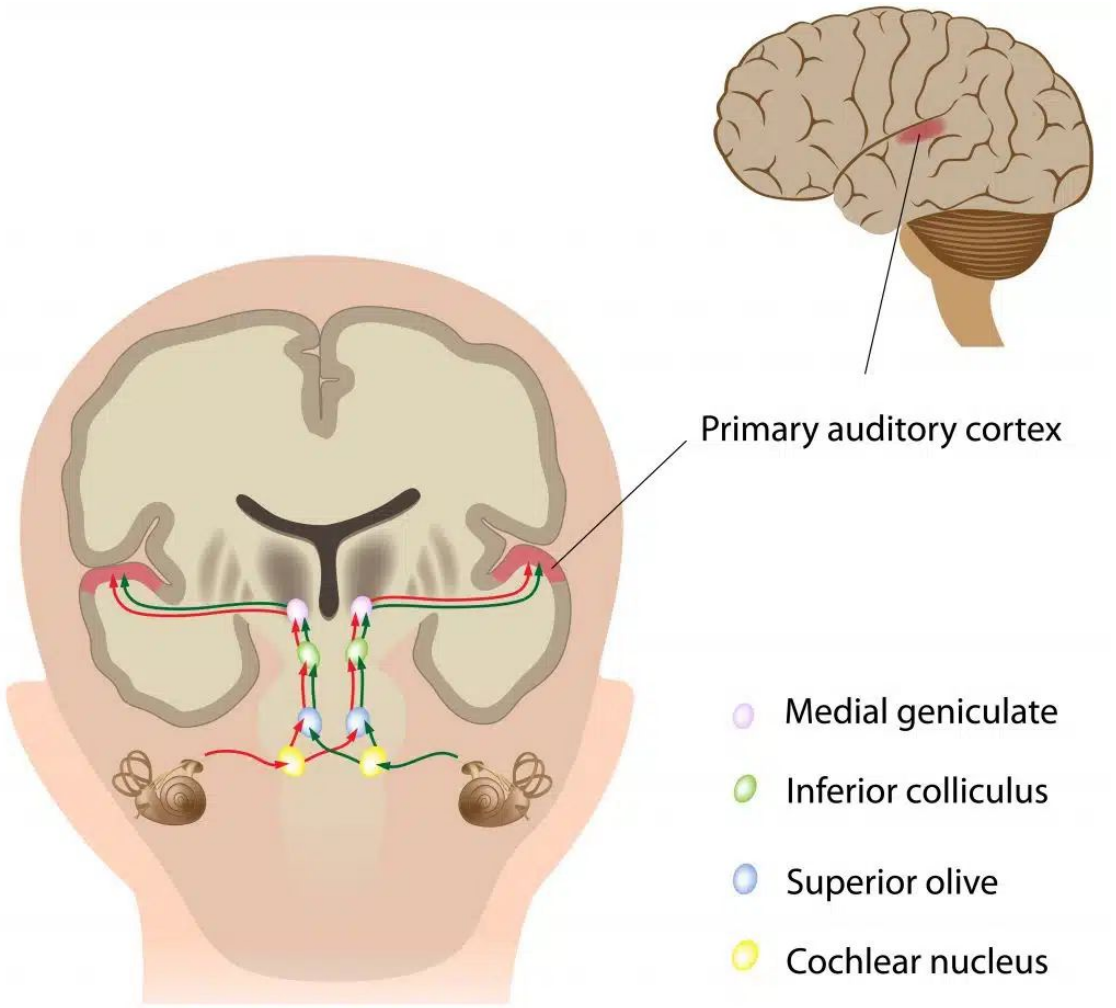


Speech Processing



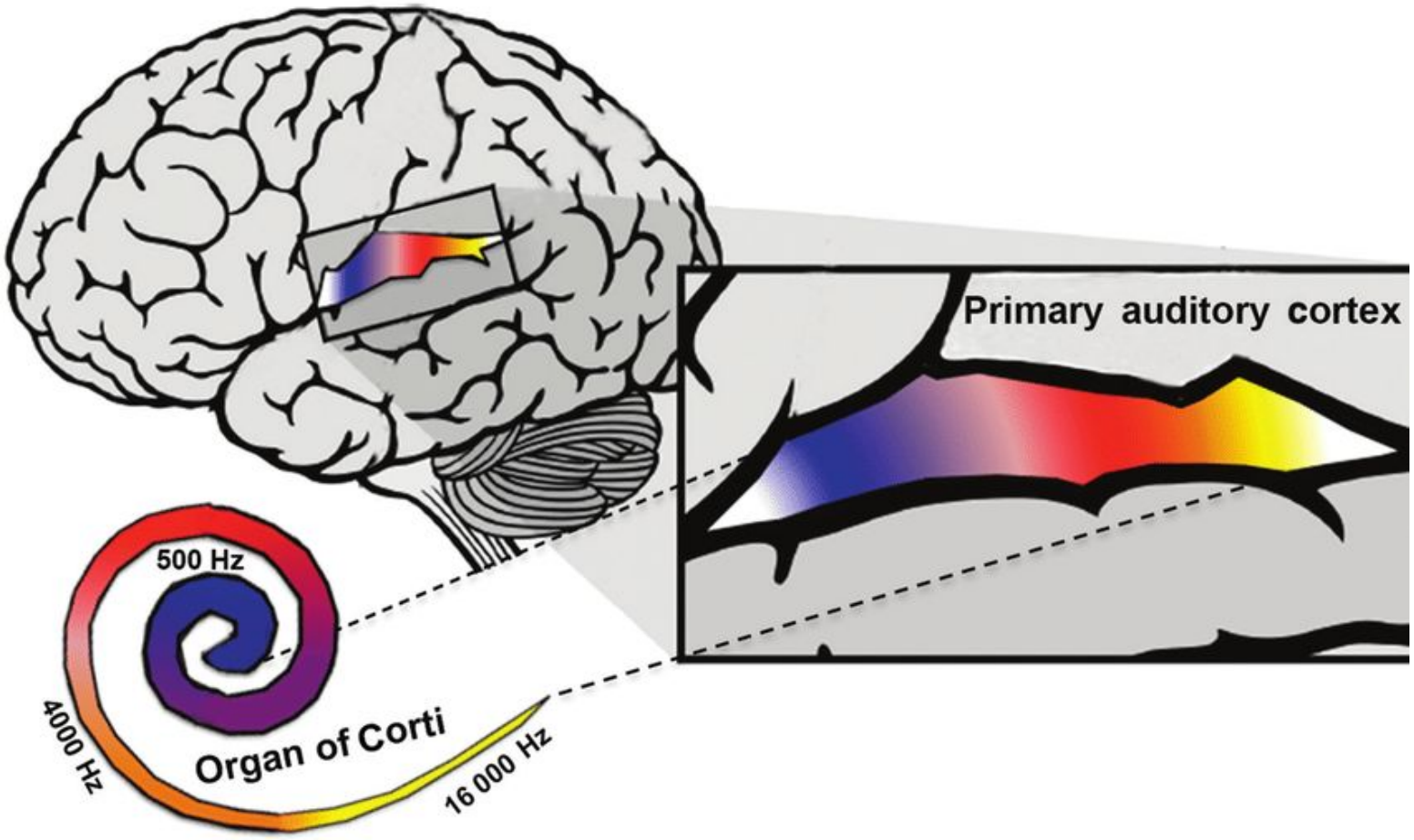
https://www.ai.rug.nl/acg/cpsp/docs/cochleaModel.html

Speech Processing



https://teachmeanatomy.info/neuroanatomy/pathways/auditory-pathway/

Speech Processing



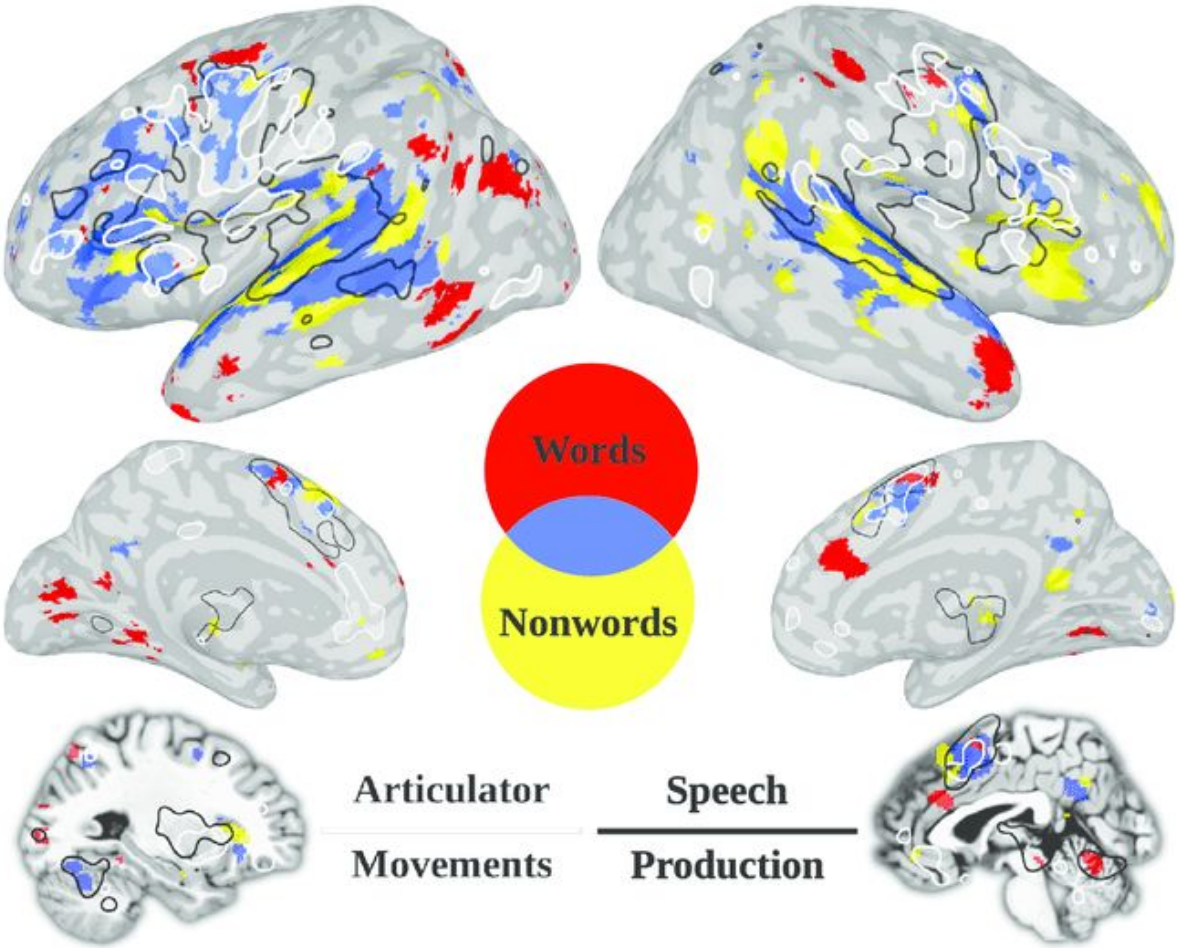
Perrone-Capano, Carla & Volpicelli, Floriana & di Porzio, Umberto. (2017). Biological bases of human musicality. Reviews in the neurosciences. 28.

Speech Processing

In a similar way that complex shapes in vision are thought to arise from co-activation of neurons responding to sub-components of more complex shapes, the phonological processing builds up sound representations in the same way.

Skipper et al. (2017) revealed the networks that then filter linguistically meaningful frequency combinations from non-words.

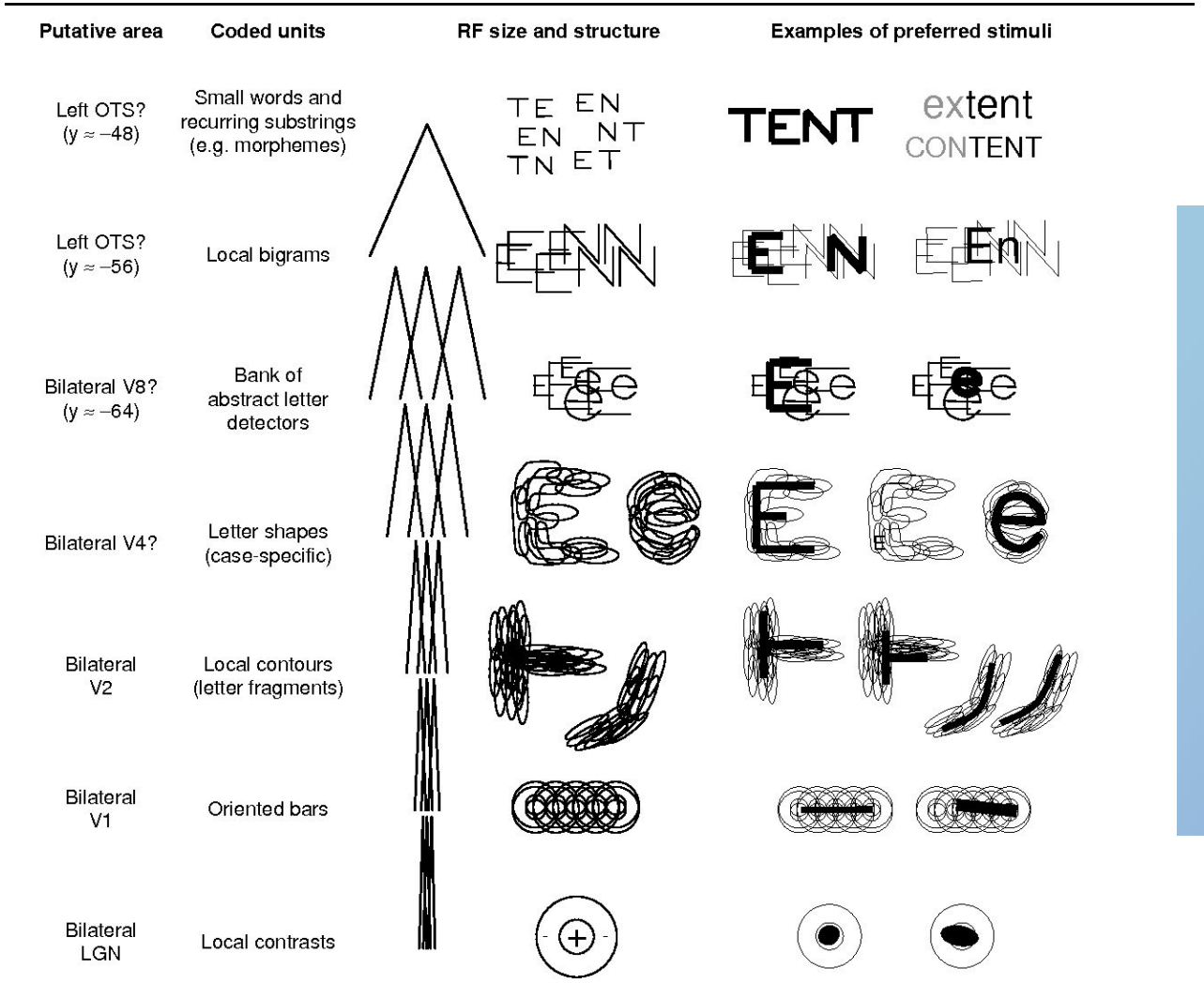
Much (much) more is known about this. Ask for resources if interested.

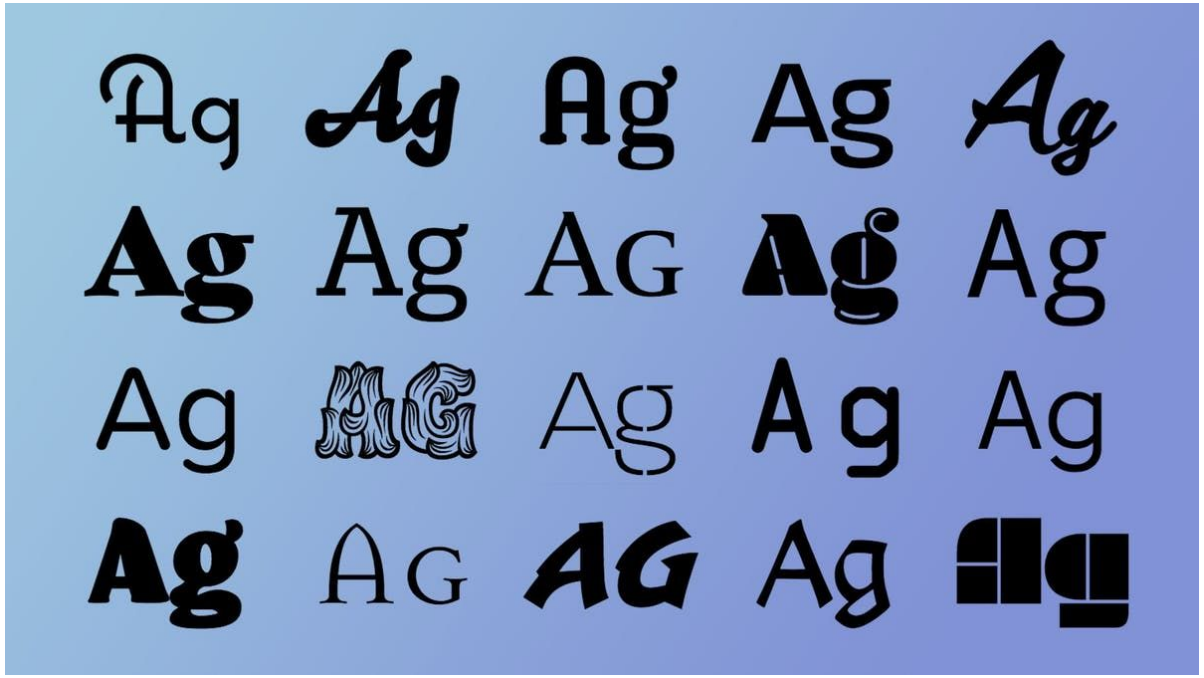


Reading

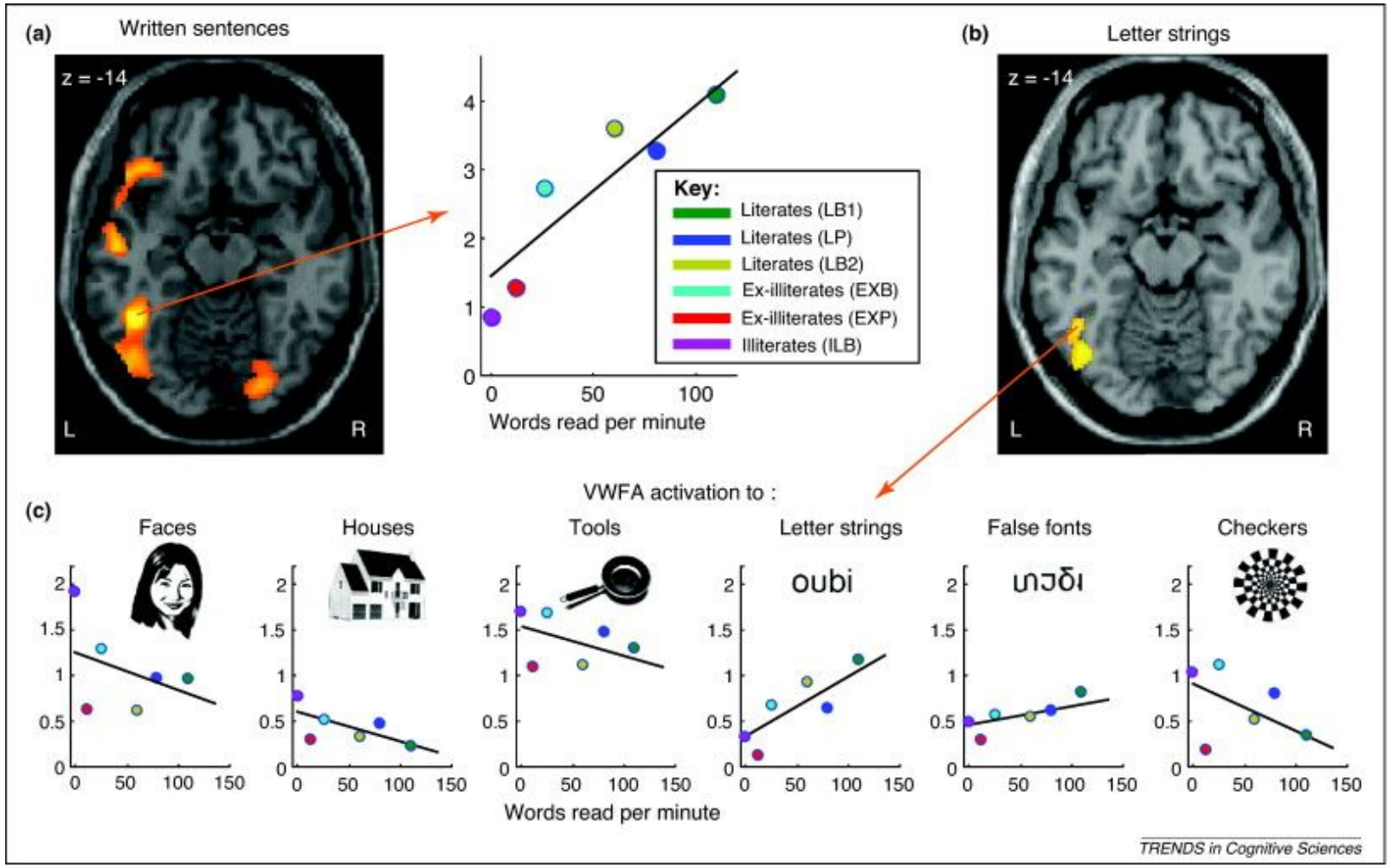
- We know(ish) how the visual system works
- We have seen various explanations for what drives the formation of representations in the ventral visual hierarchy
- How do we process written language?

Reading: Letter Processing





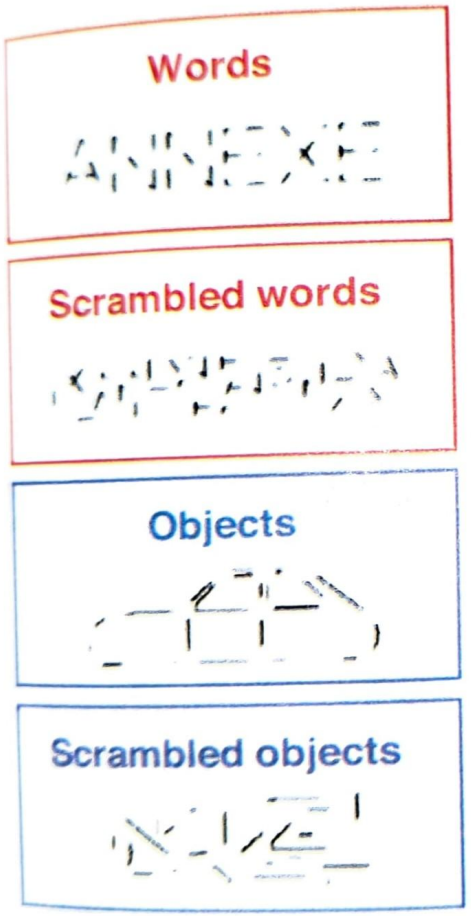
Reading: VWFA

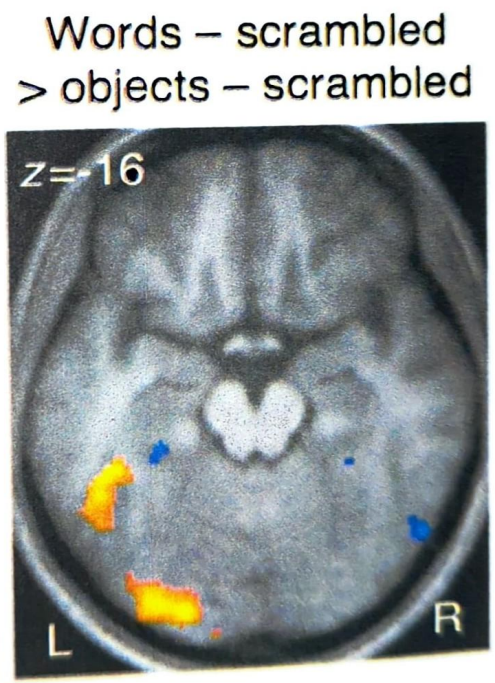


Dehaene & Cohen (2011)

Reading: A Specialised Area?

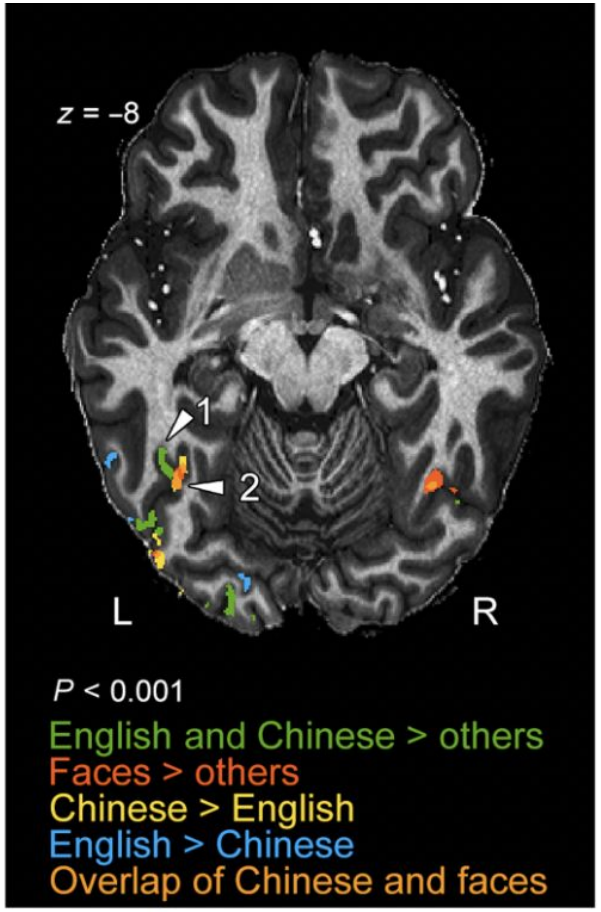
- Some challenges to the VWFA were raised, showing that this area also responded to line-drawings of non-words
- Critically, these challenges did not control for various factors such as quantity of line strokes
- Szwed et al. (2011) set out to solve this by carefully balancing non-word control conditions



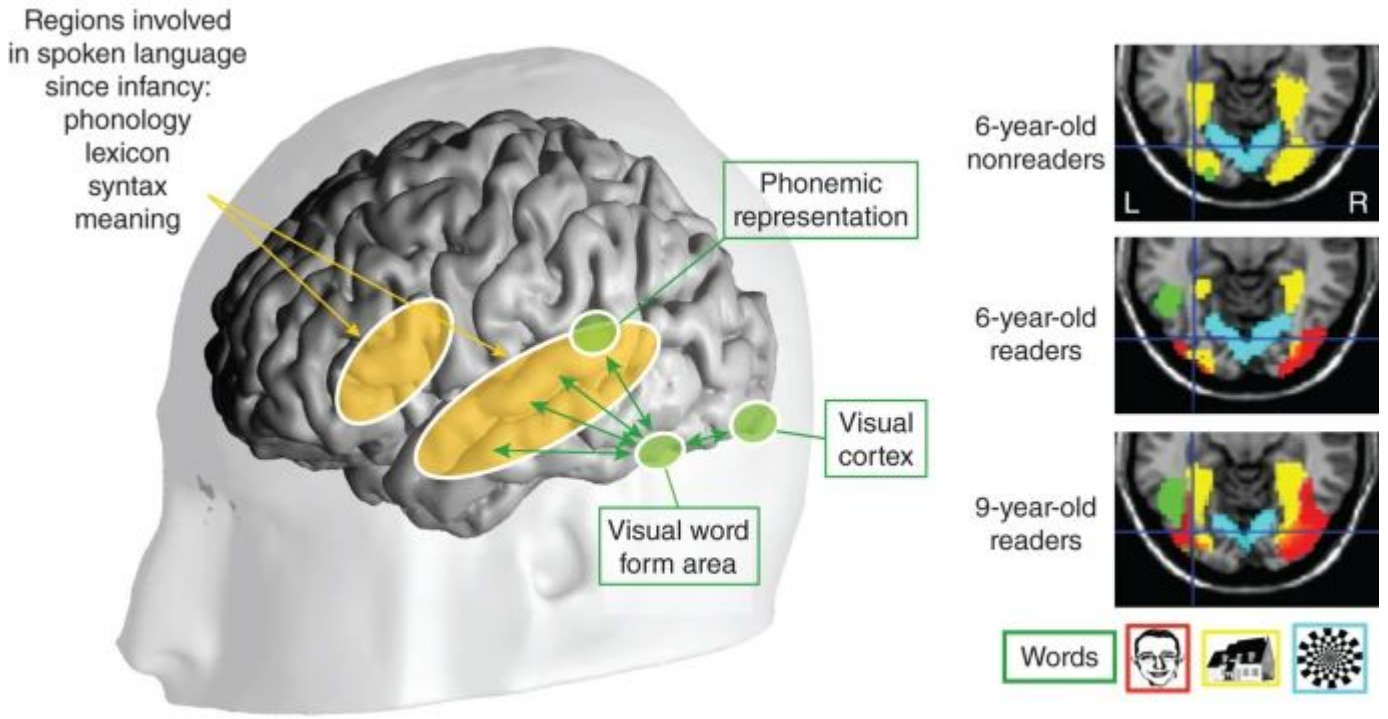


Szwed et al. (2011) via Kemmerer (2023)

Reading: VWFA Consistency



Zhan et al. (2023)



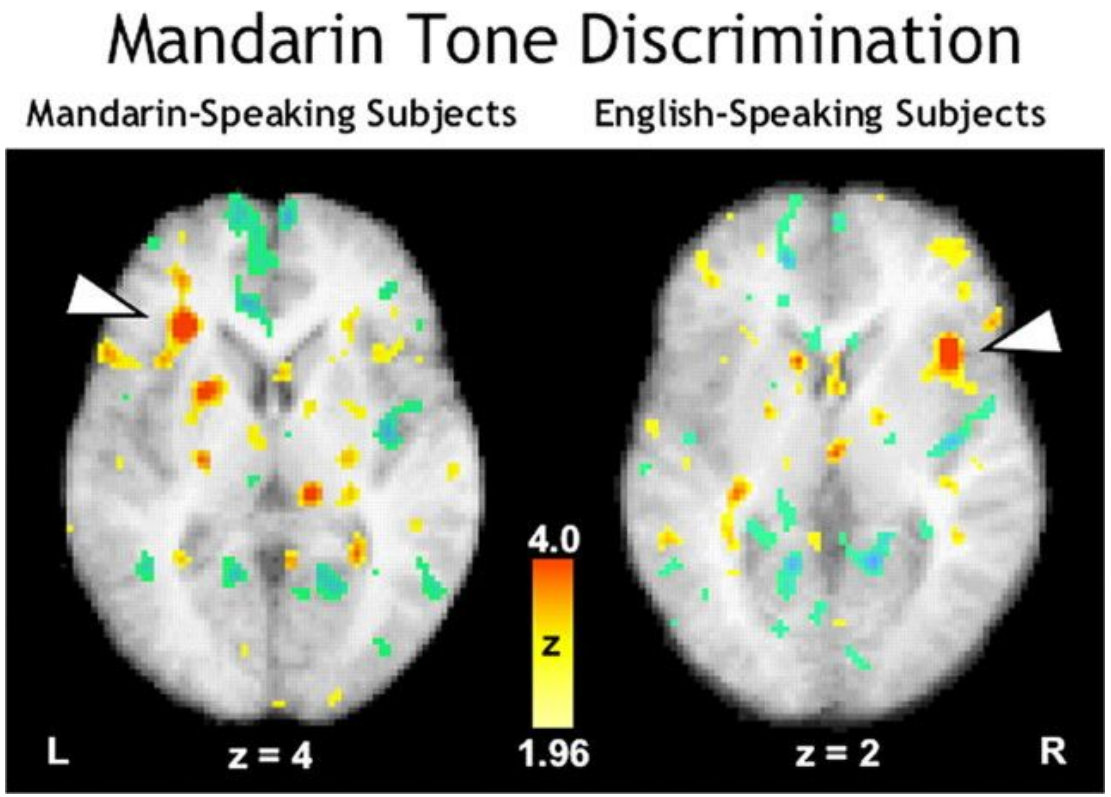
https://www.nature.com/articles/nn.4369

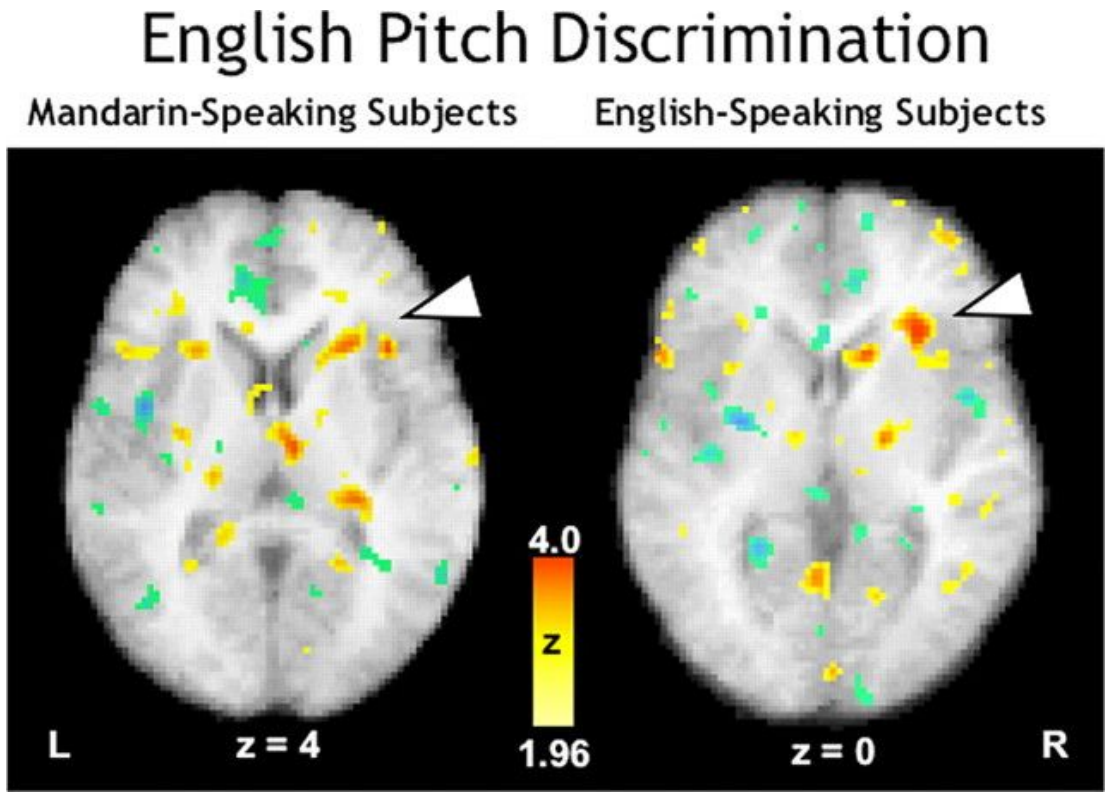
Neuronal Recycling Hypothesis

- Reading has only been culturally adopted in the last 5,000 years
- How could we have evolved a special area devoted to word reading?
- In 1820 only 12% of the world's population could read
- Can scan literate / illiterate participants and compare differences
- Can scan people as they learn to read
- Present a barrage of visual stimuli at many intervals and map which areas respond to which types of stimuli
- Once reading skill acquired, locate VWFA
- Go back to pre-reading results and ask what did this area used to respond to, but now does not?

https://ourworldindata.org/literacy

Tonal Languages



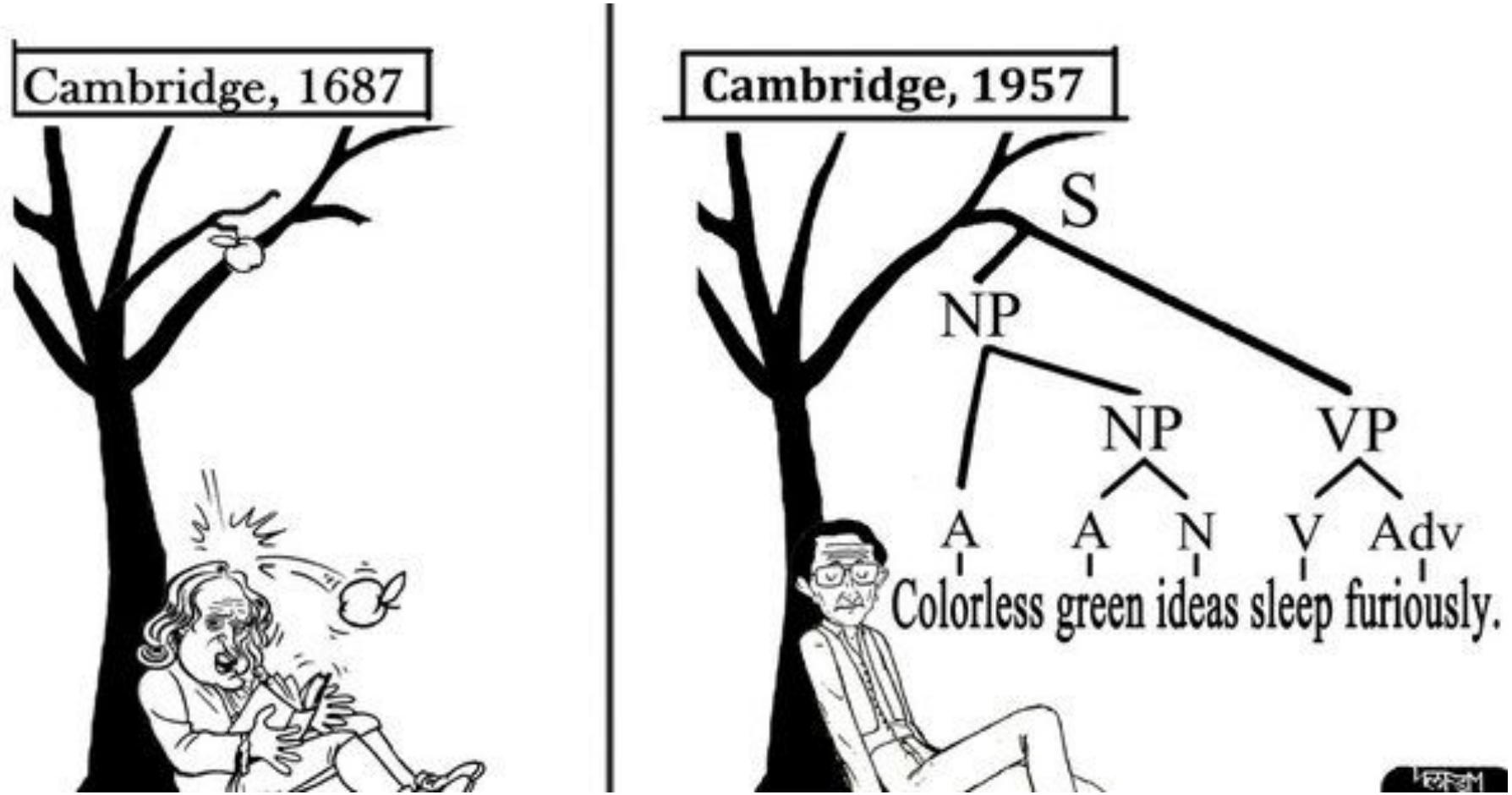


Syntax vs Semantics

Syntax can be defined as a set of rules, principles and processes that preside over the structuring of sentences in any given language.

Semantics can be defined as the science of the meanings of words, the central problem of the relationship between words and designata.

Syntax vs Semantics



Brain Signals

- Remember the EEG marker that responds to faces?
- Well, there are many more language-related ones

- Let’s review how we detect and categorise these effects
- Two conditions, well matched except for feature of interest
- Record responses to both conditions
- Look at the difference wave

Brain Signals: N400

- Net negative electrical brain response (ERP) when a sentence ends with a word that is semantically unexpected
- Central / parietal origin
- First major linguistic response in 1980s
- Font / lettering / grammatical oddities do not cause N400 (not superficial visual surprising input, but conceptual)
- We now know it's not purely linguistic
  - faces / actions / sounds / odors
- Occurs during:
  - spoken comprehension
  - reading
  - signing

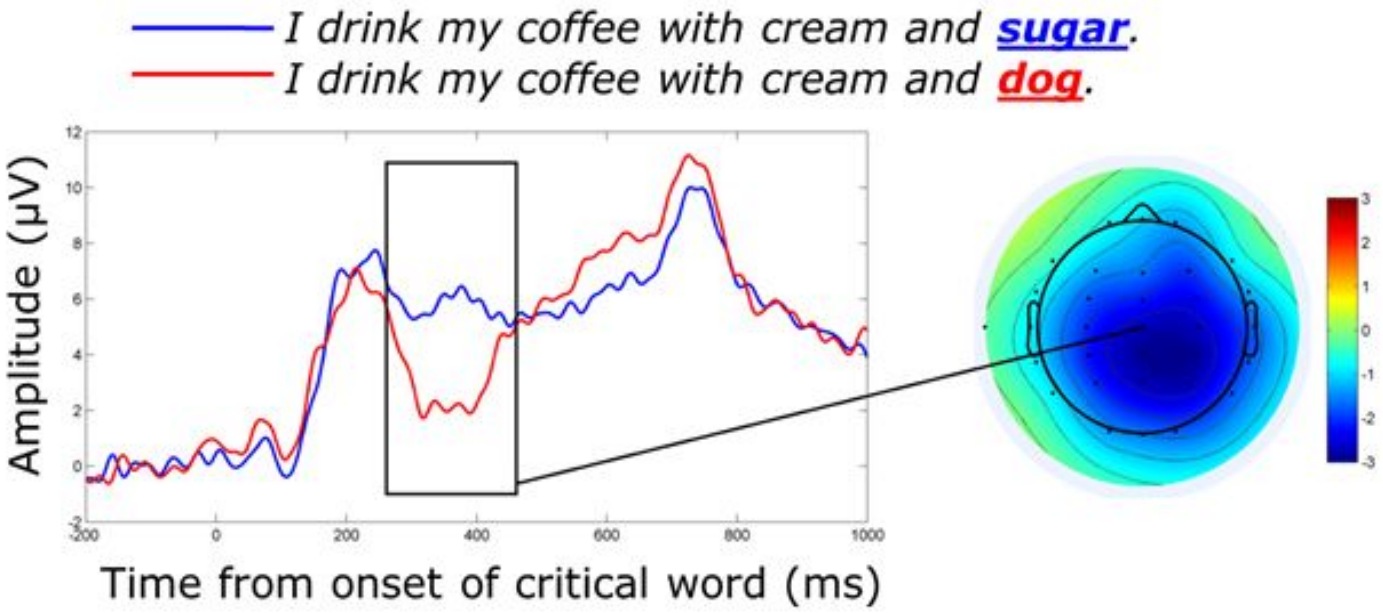


Figure adapted from Hunt, Politzer-Ahles, Gibson, Minai, & Fiorentino (2013)

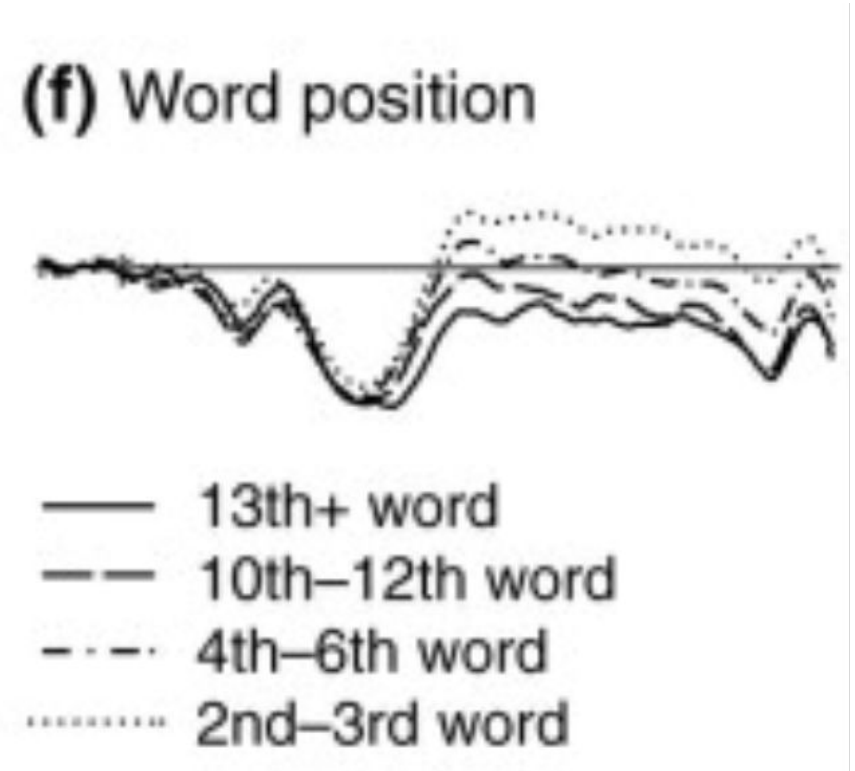
https://people.ku.edu/~sjpa/Classes/CBS592/Neurolinguistics/n400.html

Brain Signals: N400

Cloze Probability: the percentage of people who would continue a sentence fragment with a specific word.

The magnitude of the N400 can be manipulated by the cloze probability of a specific sentence continuation (the more unlikely -> the bigger the N400).

As sentences unfold, potential continuations become more restricted. This implies semantic oddities have less of an impact on the N400 (when cloze probabilities become restricted, N400 effects are diminished).



Kutas & Federmeier (2000)

Brain Signals: N400

Relationship to individual conceptual knowledge.

LANGUAGE, COGNITION AND NEUROSCIENCE
2020, VOL. 35, NO. 5, 641–657
https://doi.org/10.1080/23273798.2018.1503309

REGULAR ARTICLE

Harry Potter and the Chamber of *What*?: the impact of what individuals know on word processing during reading

Melissa Troyer[a] and Marta Kutas[a,b,c,d]

[a]Department of Cognitive Science, University of California, San Diego, CA, USA; [b]Center for Research in Language, University of California, San Diego, CA, USA; [c]Department of Neurosciences, University of California, San Diego, CA, USA; [d]Kavli Institute for Brain and Mind, University of California, San Diego, CA, USA

Brain Signals: N400

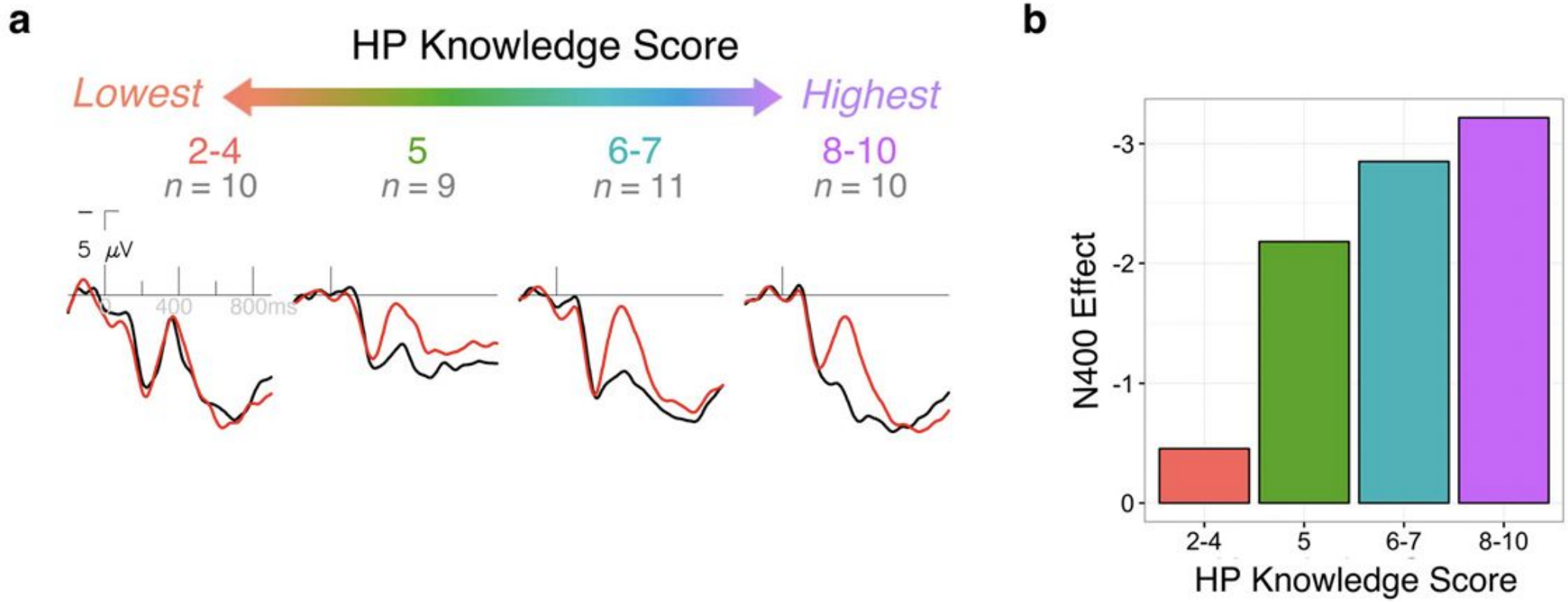
Relationship to individual conceptual knowledge.

Table 1. Sample experimental stimuli.

Sentence frame	Supported	Unsupported
Control Sentences		
We had been watching the blue jay for days. The bird laid her eggs in the	nest	yard
The vampire moved in. He bit his victim on the	neck	shoulder
Alicia's first client was a failure. But her second was a	success	triumph
Harry Potter Sentences		
The character Peter Pettigrew changes his shape at times. He takes the form of a	rat	dog
There are two Beaters on every Quidditch team. Their job is to protect their team from	Bludgers	Spellotape
Wizards are able to conjure the Dark Mark. They can use a spell called	Morsmordre	Stupefy

Troyer & Kutas (2020)

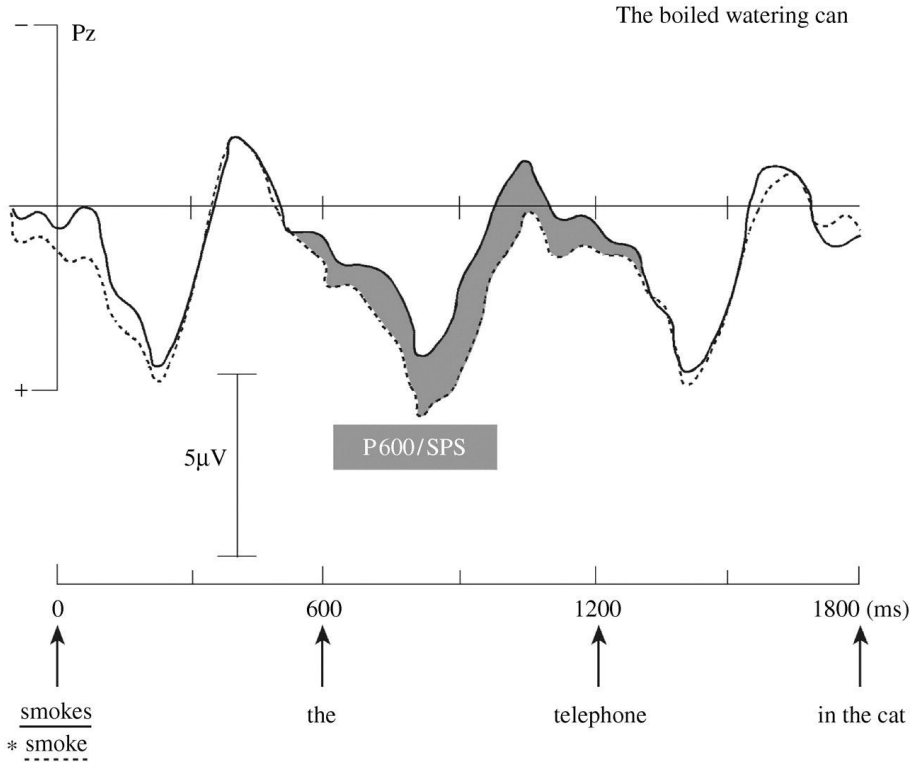
Brain Signals: N400



Troyer & Kutas (2020)

Brain Signals: P600

- A syntactically-modulated effect
- Appears from 500 ms after critical word in sentence
- Structure building
- Suggestive evidence it's also seen in music interpretation
- Can be seen in semantically meaningless sentences that have grammatical errors



Hagoort (2007)

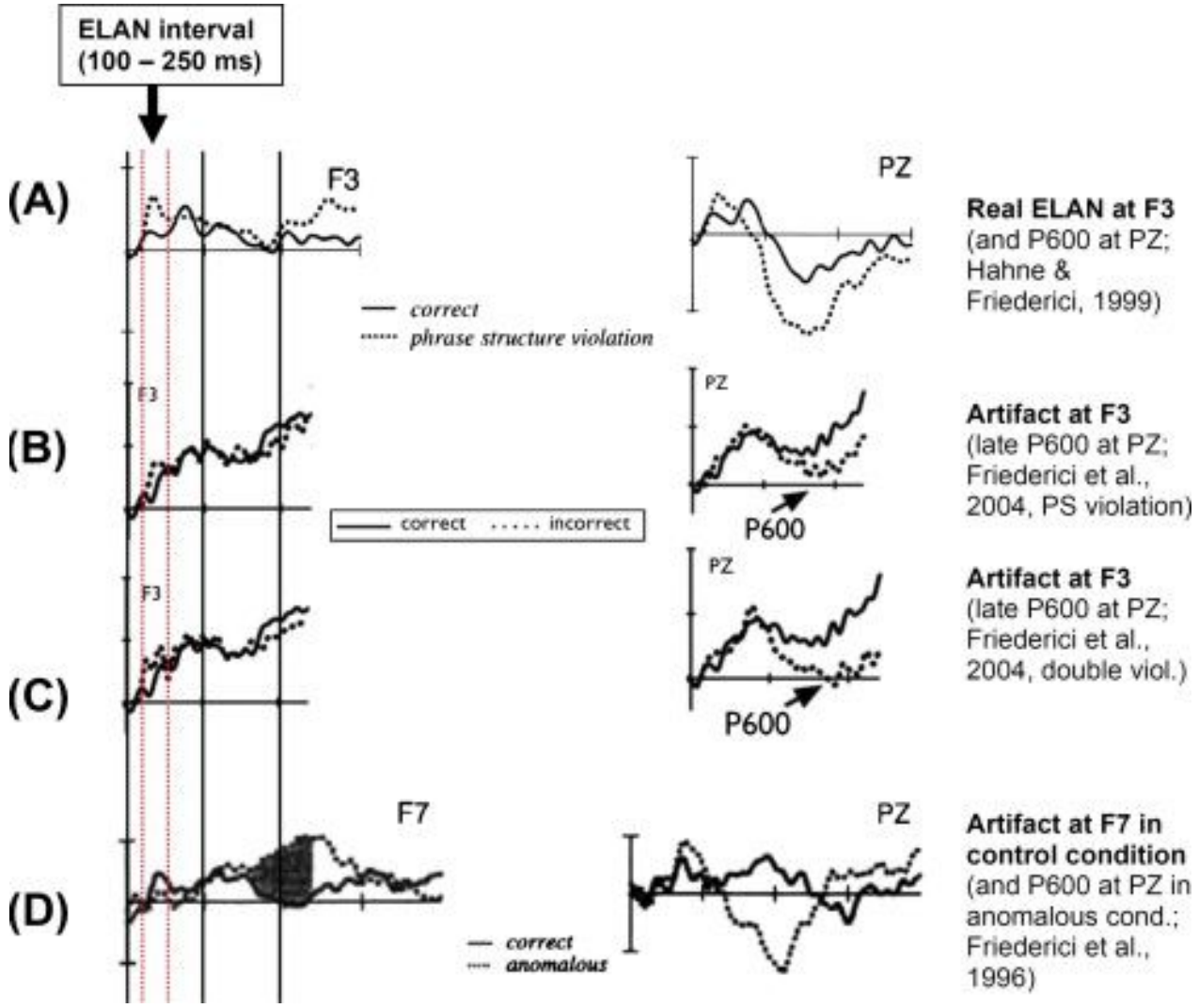
Brain Signals: P600

Sentence re-analysis triggers the P600 (most common in **garden-path sentences**)

- 1. The old man the boat.
- 2. The girl told the story cried.
- 3. After the student moved the chair broke.
- 4. Fat people eat accumulates.
- 5. The horse raced down the barn fell.

Brain Signals: (E)LAN

- (Early) Left Anterior Negativity
- Discovered by Angela Friederici
- Many neurolinguistic models of sentence processing suggest that syntax is the first stage of processing (prior to semantics) and syntactic violations should therefore be seen prior to N400
- ELAN is an effect that has been shown to occur early (100-200 ms) after a syntactic violation

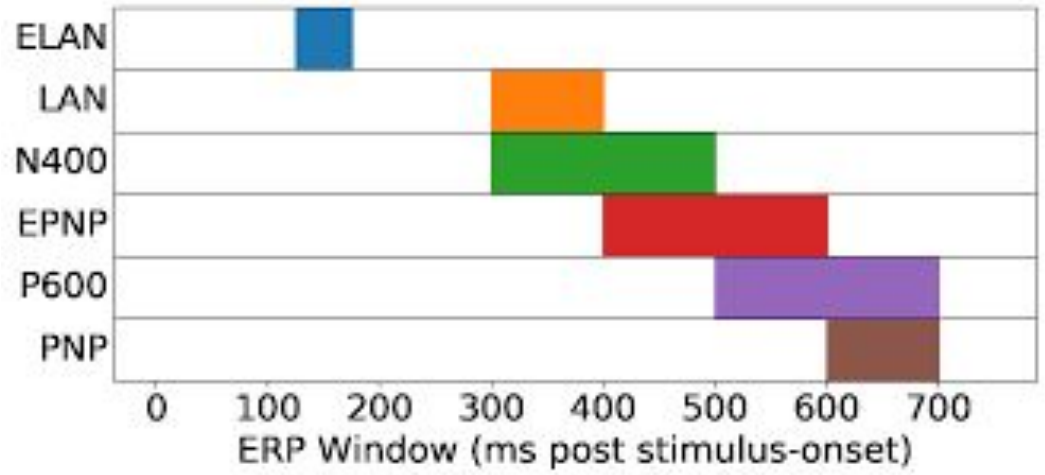


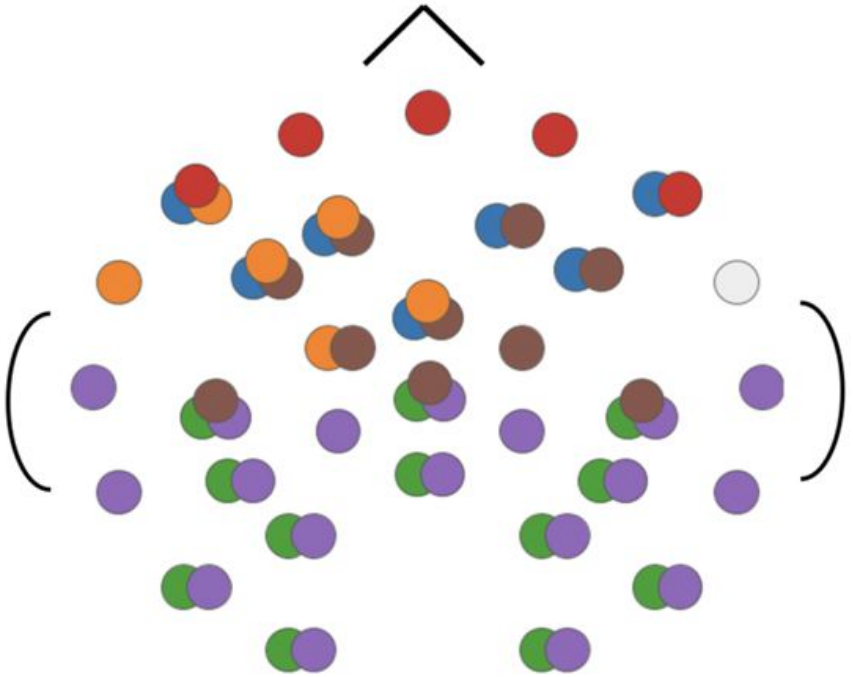
Brain Signals: (E)LAN

Steinhauer & Drury (2012):

"We hypothesise that syntactic violations in auditory studies always elicit sustained negativities ***and no local ELANs****. Whenever an ELAN looks like a local (transient) effect, this is* ***likely to be due to a concurrent P600 component cancelling out the later part of the negativity****. If our hypothesis is true, there may be no need to account for any local ELAN effects between 100-300 ms (as suggested by Friederici's model) but* ***there is a need to explain sustained negativities with a remarkably early onset.****"*

Brain Signals





Schwarz & Mitchell (2019)

Language Models

What is a Language Model?

A probability distribution over strings according to a model

```
P(string | model)
```

Corpus (plural ***corpora***): a collection of written texts

String:

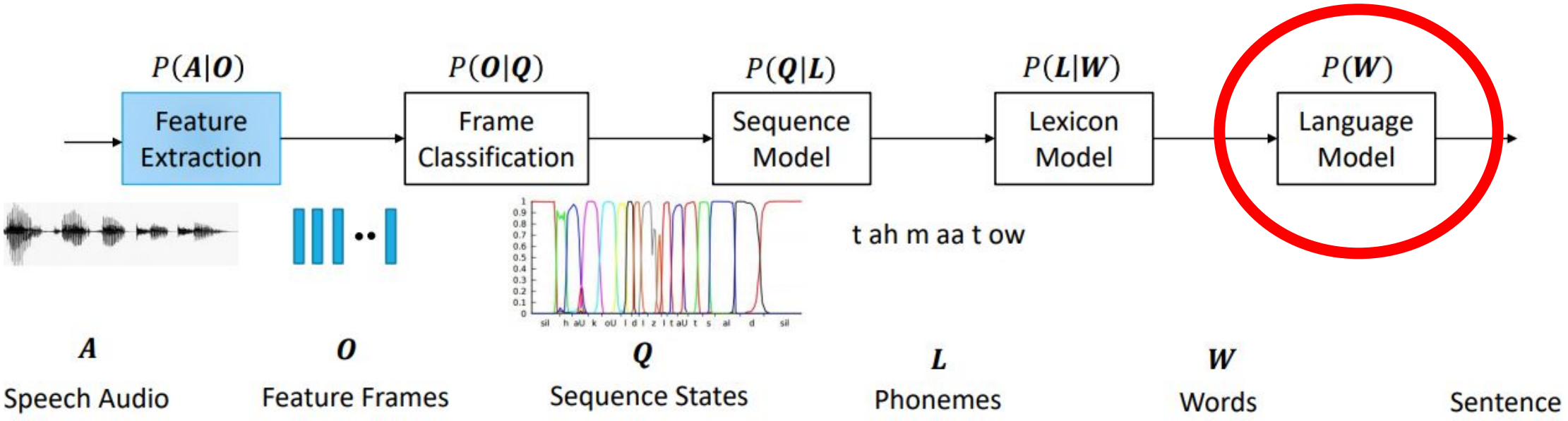
- Word sequence
- Character sequence
- Letter triplets
- Can be most things!

```
P("Edmonton winters are enjoyable" | model)
P("Enjoyable Edmonton are winters" | model)
P("Les hivers d'Edmonton sont plaisants" | model)
```

Language models are defined according to their model and associated model parameters. Calculation of this value depends on the model, here it's not specified.

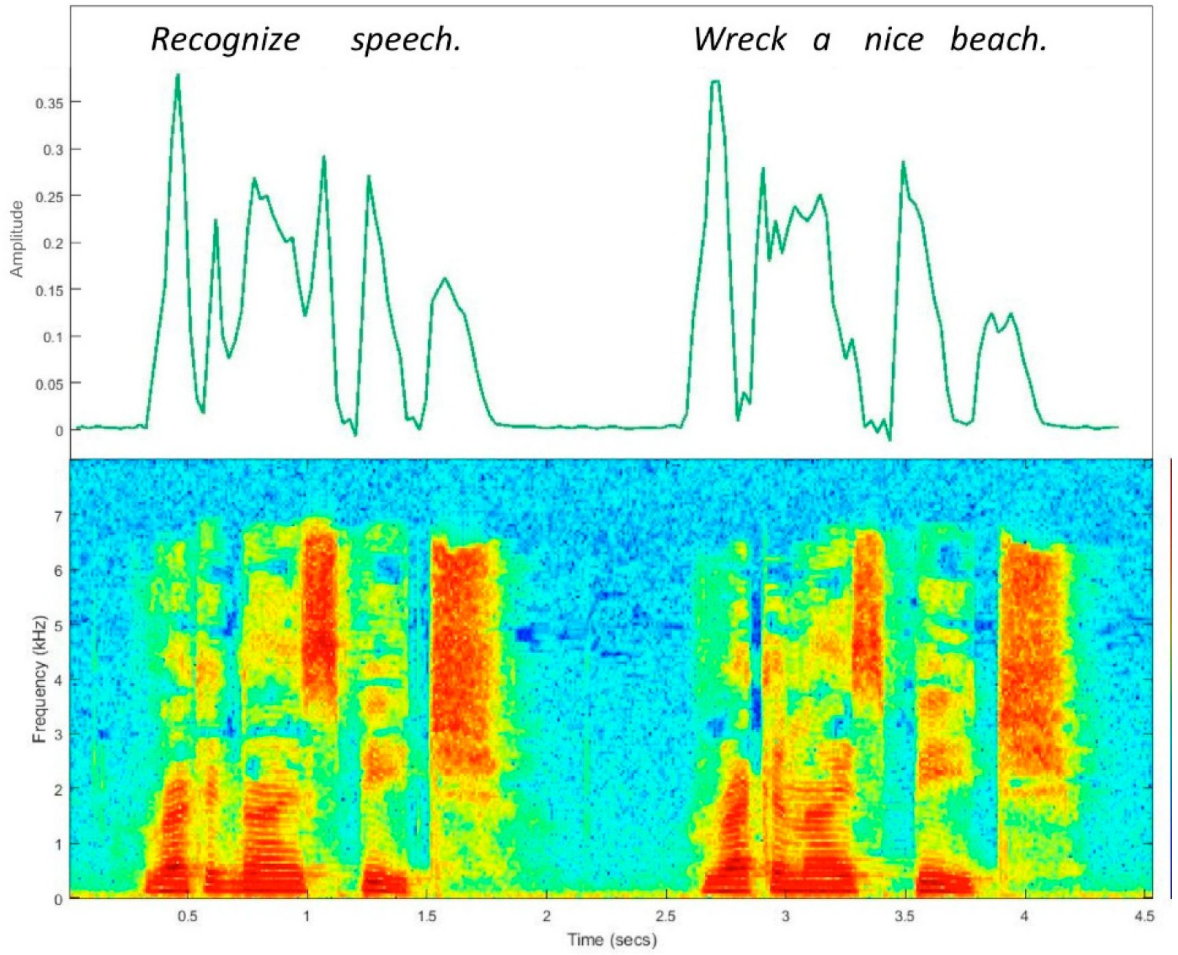
Why did we need LMs?

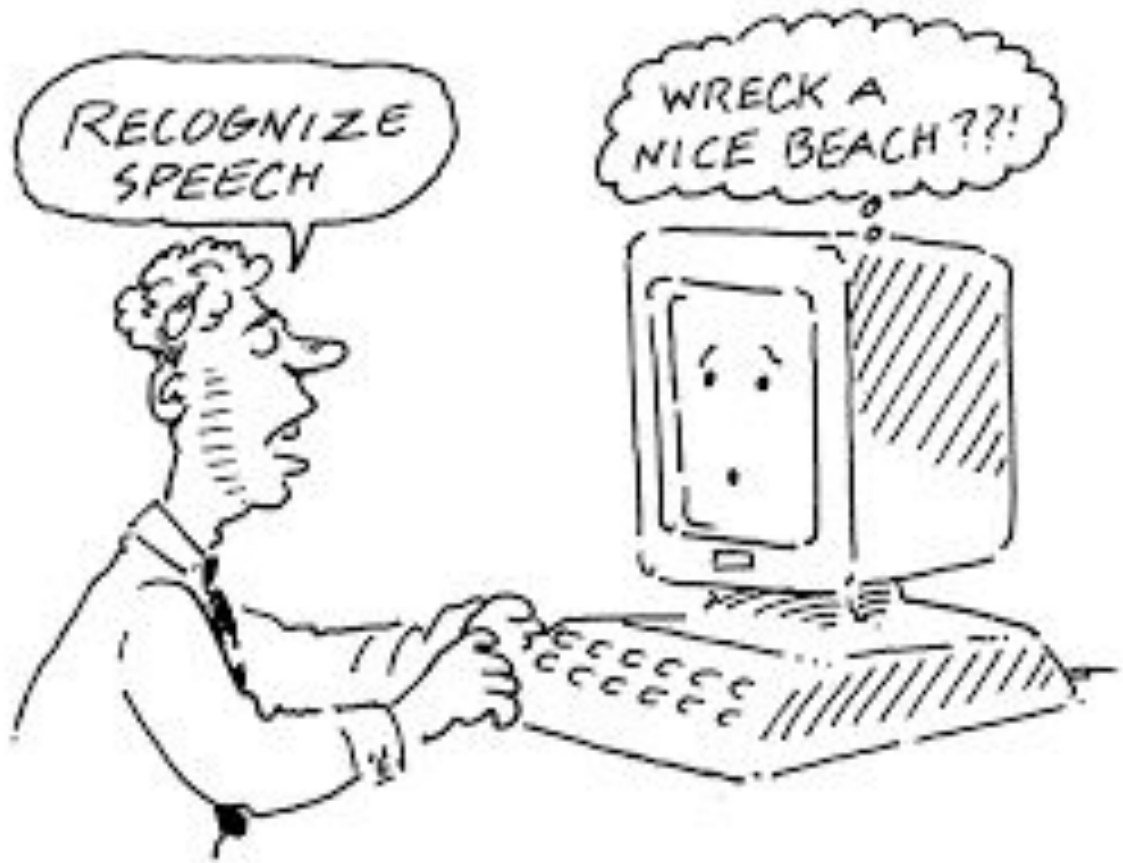
$$\widehat{W} = \underset{W}{\operatorname{argmax}}\, P(W|O) = \underset{W}{\operatorname{argmax}}\, P(A|O)P(O|Q)P(Q|L)P(L|W)P(W)$$



http://slazebni.cs.illinois.edu/spring17/lec26_audio.pdf

Why did we need LMs?





Simple Language Models

"n-gram models"

1: unigram

2: bigram

3: trigram

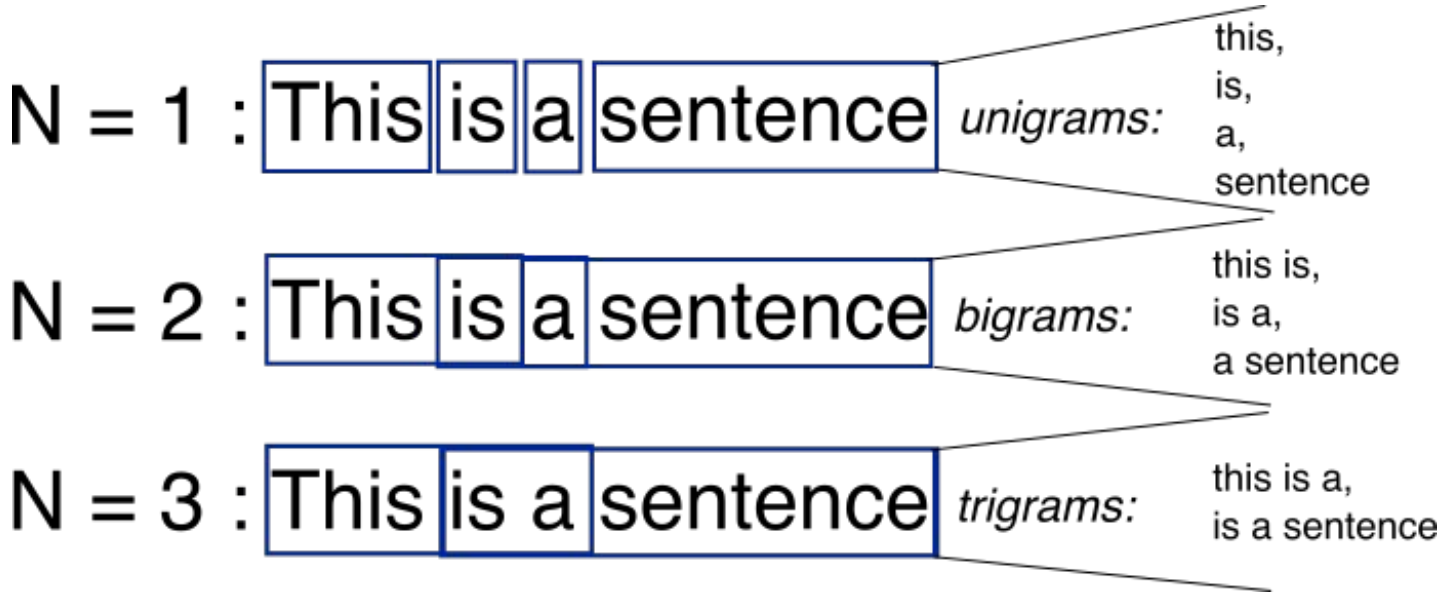
4: four-gram

5: five-gram

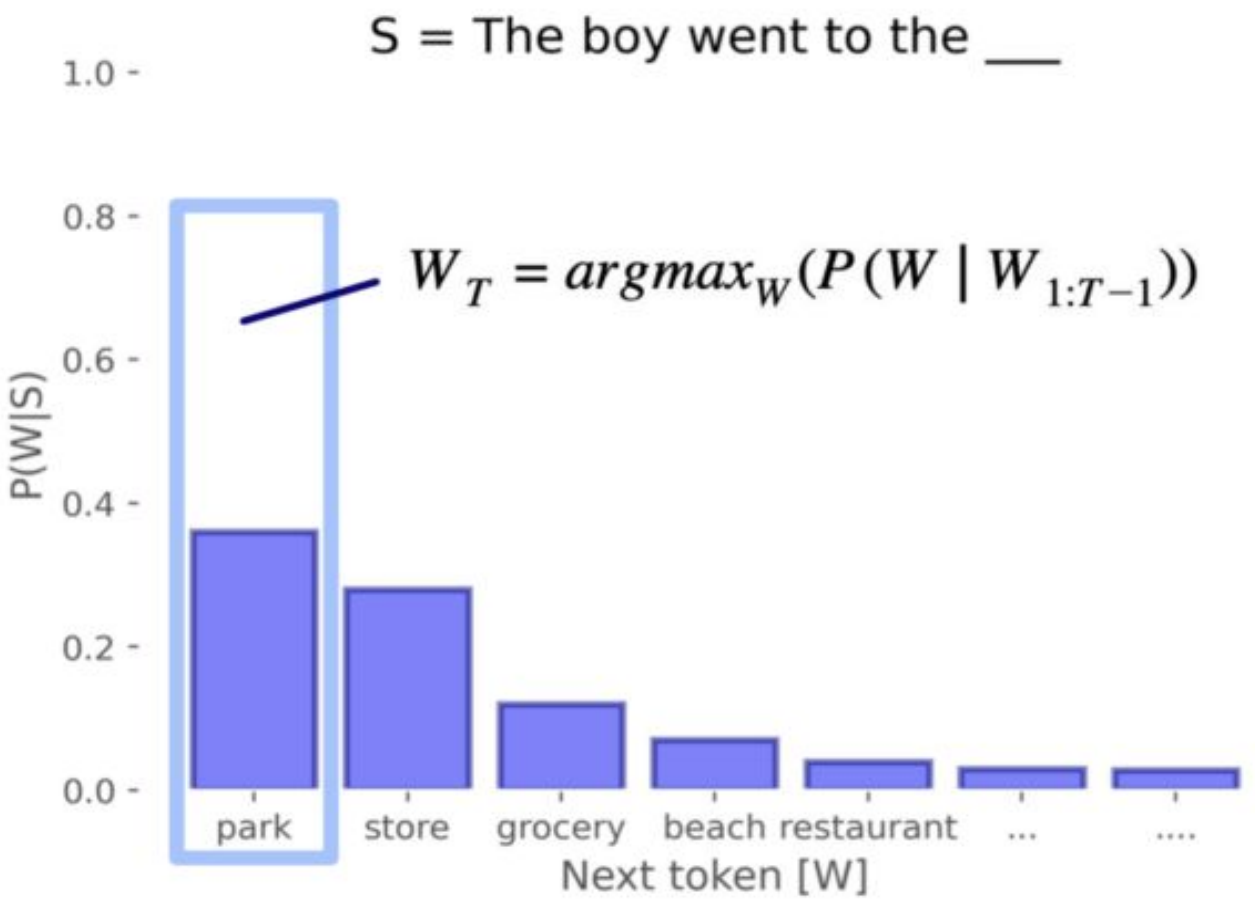
etc…

$$\begin{aligned}\mathrm{P}(X_4, X_3, X_2, X_1) &= \mathrm{P}(X_4 \mid X_3, X_2, X_1) \cdot \mathrm{P}(X_3, X_2, X_1) \\ &= \mathrm{P}(X_4 \mid X_3, X_2, X_1) \cdot \mathrm{P}(X_3 \mid X_2, X_1) \cdot \mathrm{P}(X_2, X_1) \\ &= \mathrm{P}(X_4 \mid X_3, X_2, X_1) \cdot \mathrm{P}(X_3 \mid X_2, X_1) \cdot \mathrm{P}(X_2 \mid X_1) \cdot \mathrm{P}(X_1)\end{aligned}$$

The Chain Rule of probability



Simple Language Models



https://blog.allenai.org/a-guide-to-language-model-sampling-in-allennlp-3b1239274bc3

Shall I compare thee to a Corpus?

1 gram	–To him swallowed confess hear both. Which. Of save on trail for are ay device and rote life have –Hill he late speaks; or! a more to leg less first you enter
2 gram	–Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow. –What means, sir. I confess she? then all sorts, he is trim, captain.
3 gram	–Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done. –This shall forbid it should be branded, if renown made it empty.
4 gram	–King Henry. What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in; –It cannot be but so.

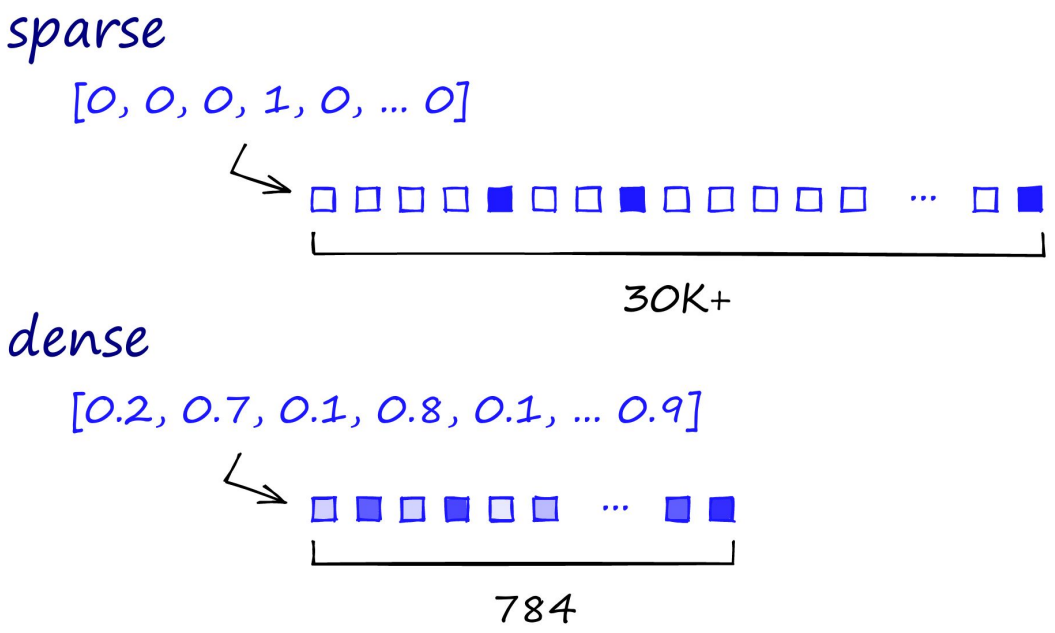
Problems with *n-gram* models

What are some issues you can think of relating to n-gram models?

- Unknown words
- Spelling mistakes
- Long distance dependencies
- Synonyms
- Data sparsity

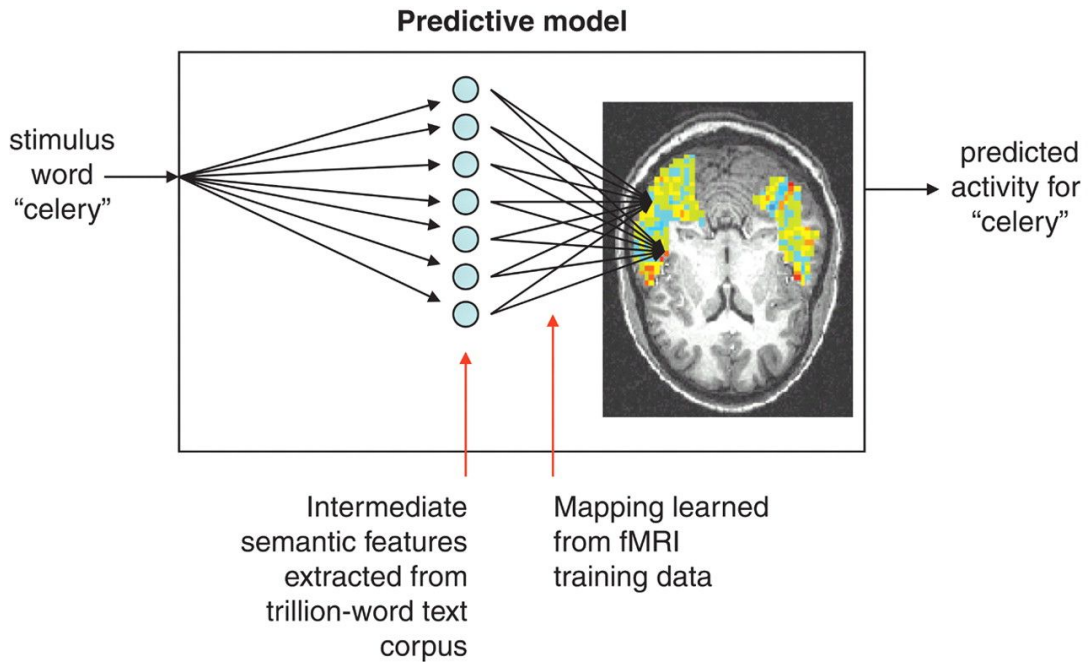
Dense Representations / Embeddings

- We need to be able to capture word meaning in a better way
- Three main methods were developed to achieve this
  - Co-occurrence statistics
  - Matrix factorisation methods
  - Neural network methods
- These (latter two methods) serve as key components in modern (L)LMs



Cooccurence Matrices

- A word's meaning is a function of the words it appears with
- This is known as the "**Distributional Hypothesis**"
- Word co-occurrences with common words not so helpful
- Co-occurrences with a specific subset of words is better



> "You will know a word by the company it keeps"
>
> J.R. Firth (1957)

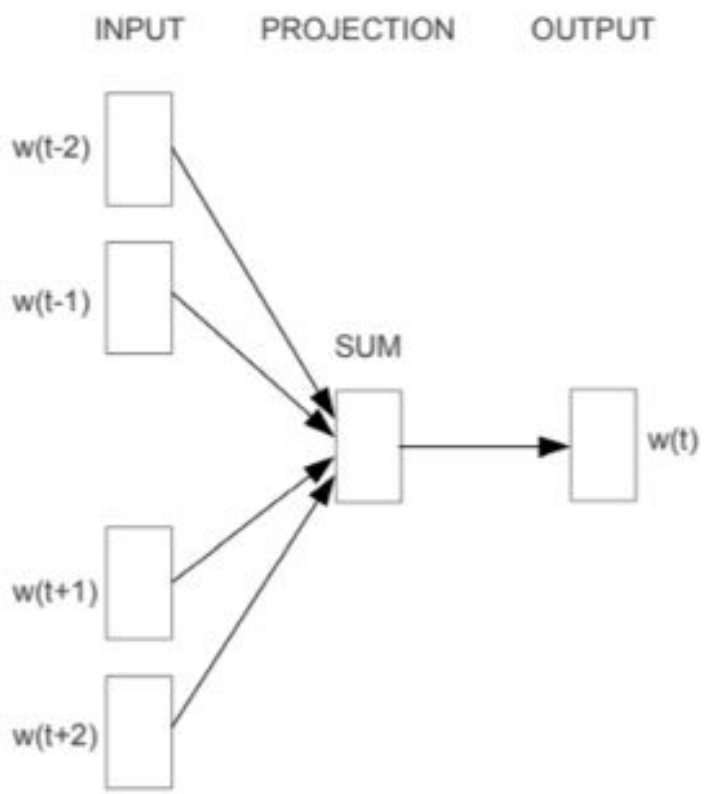
Word2Vec

- Scan a fixed-size window over a corpus of text
- For each window, pick the either the central word or all the context words and remove from, predict from what is remaining

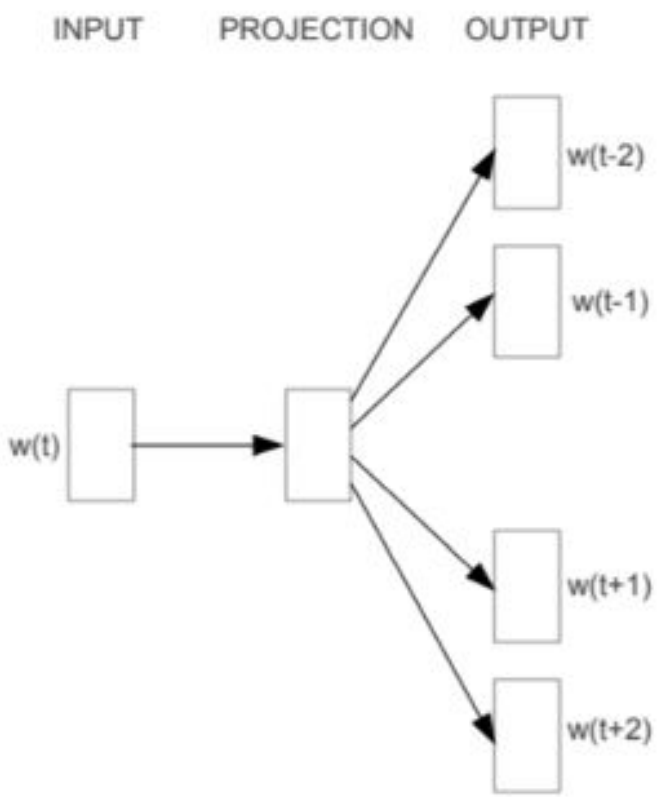
    - The choice of each leads to a slight variation of the Word2Vec algorithm
    - If you predict central word from context -> **CBOW**
    - If you predict context from central word -> **Skipgram**

http://jalammar.github.io/illustrated-word2vec/

Word2Vec



CBOW



Skip-gram

Word2Vec (CBOW)

Jay was hit by a _____ bus in...

by	a	red	bus	in

input 1	input 2	input 3	input 4	output
by	a	bus	in	red

Word2Vec (Skipgram)

Jay was hit by a red bus in...

by	a	red	bus	in

input	output
red	by
red	a
red	bus
red	in

Thou shalt not make a machine in the likeness of a human mind

thou	shalt	not	make	a	machine	in	the	...

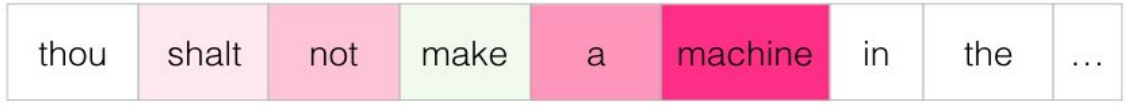
input word	target word

Word2Vec (Skipgram)

Thou shalt not make a machine in the likeness of a human mind

input word	target word

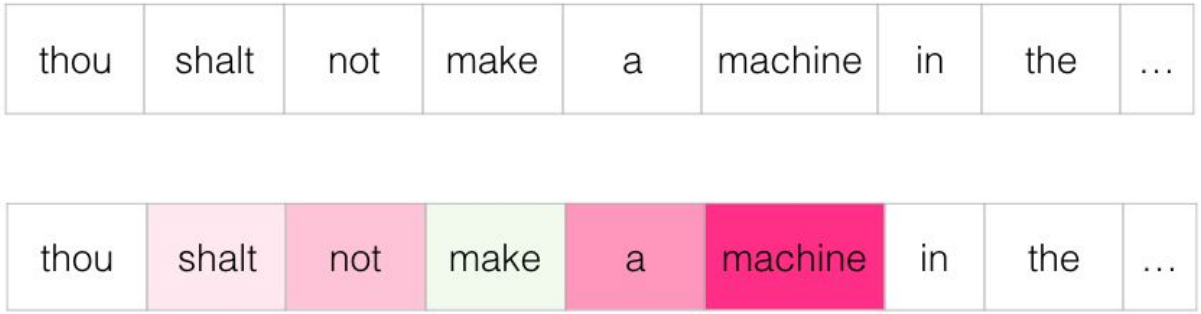
Thou shalt not make a machine in the likeness of a human mind



input word	target word
not	thou
not	shalt
not	make
not	a

Word2Vec

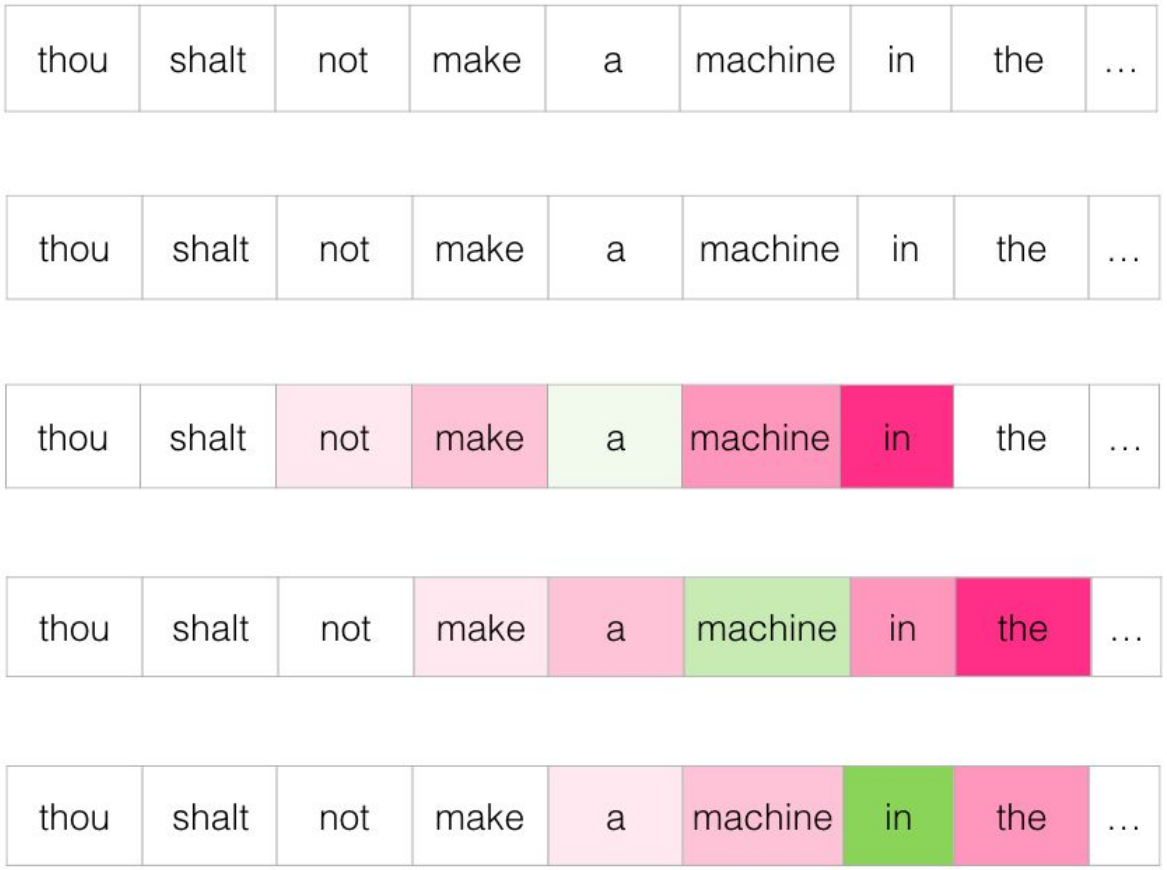
Thou shalt not make a machine n the likeness of a human mind



input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine

Word2Vec

Thou shalt not make a machine in the likeness of a human mind

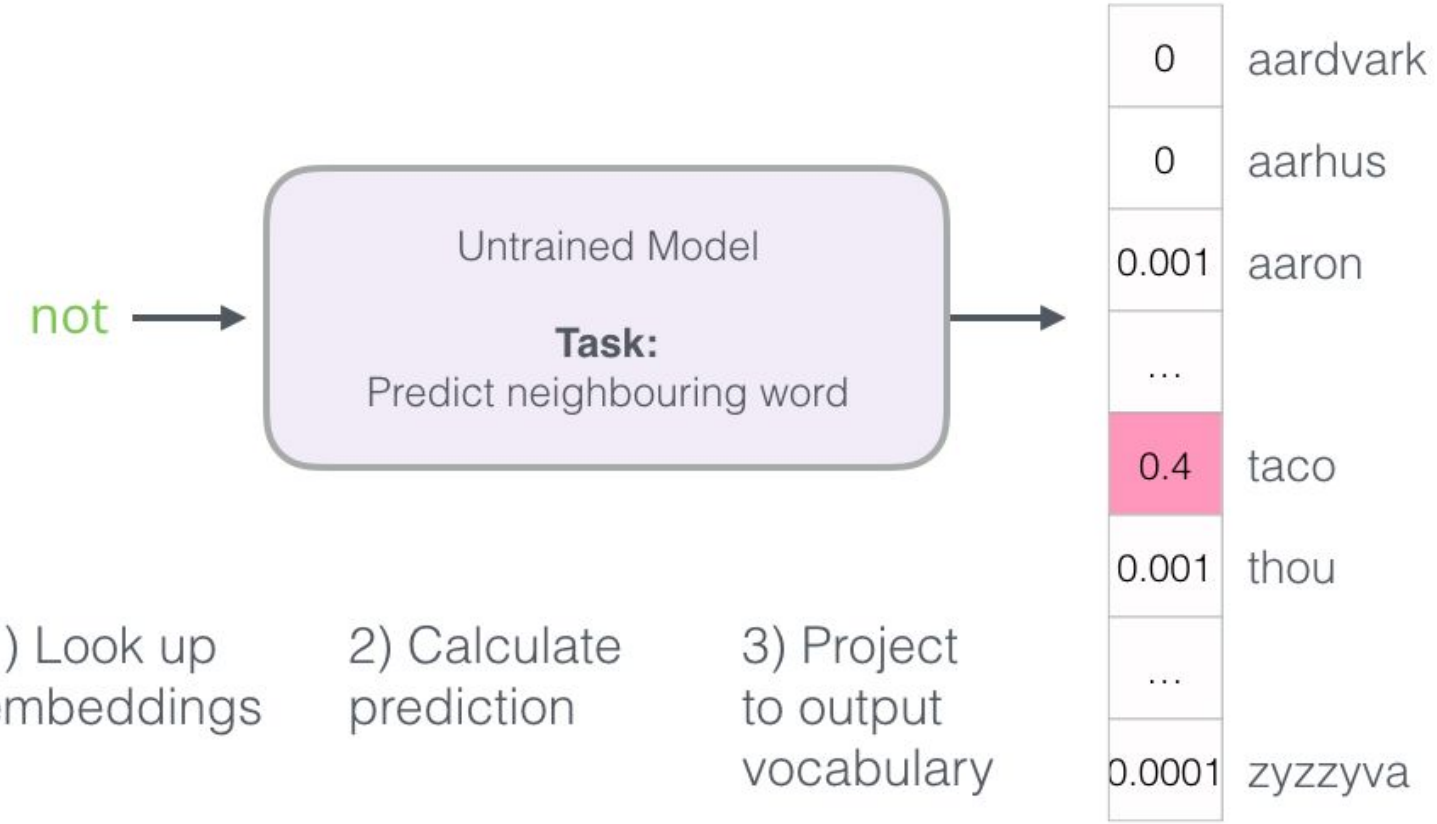


input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine
a	not
a	make
a	machine
a	in
machine	make
machine	a
machine	in
machine	the
in	a
in	machine
in	the
in	likeness

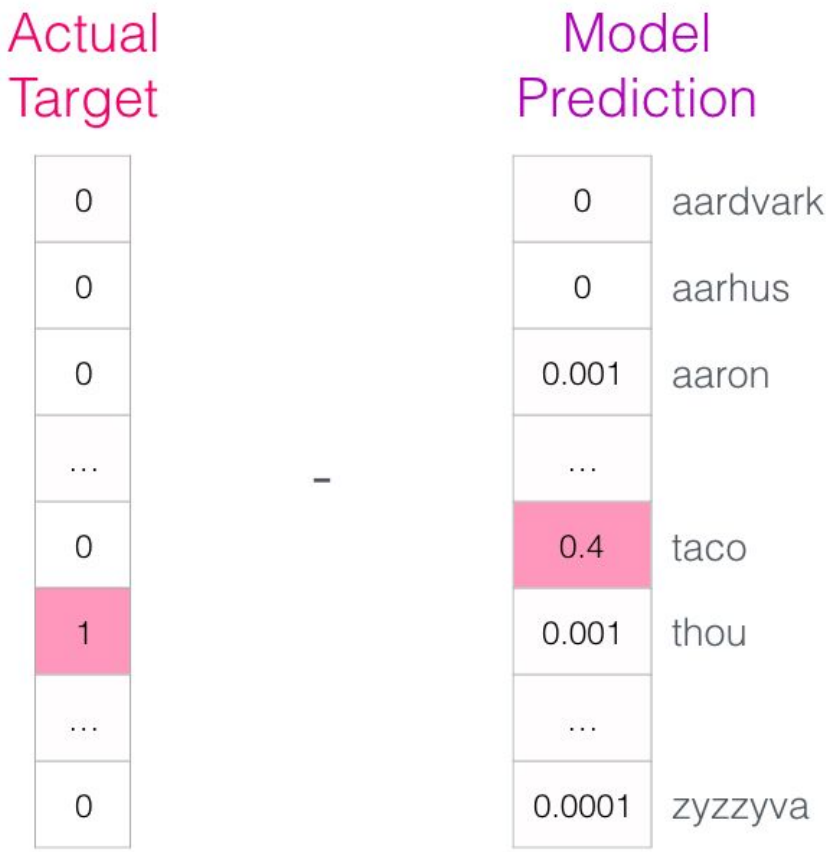
Word2Vec

input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine
a	not
a	make
a	machine
a	in
machine	make
machine	a
machine	in
machine	the
in	a
in	machine
in	the
in	likeness

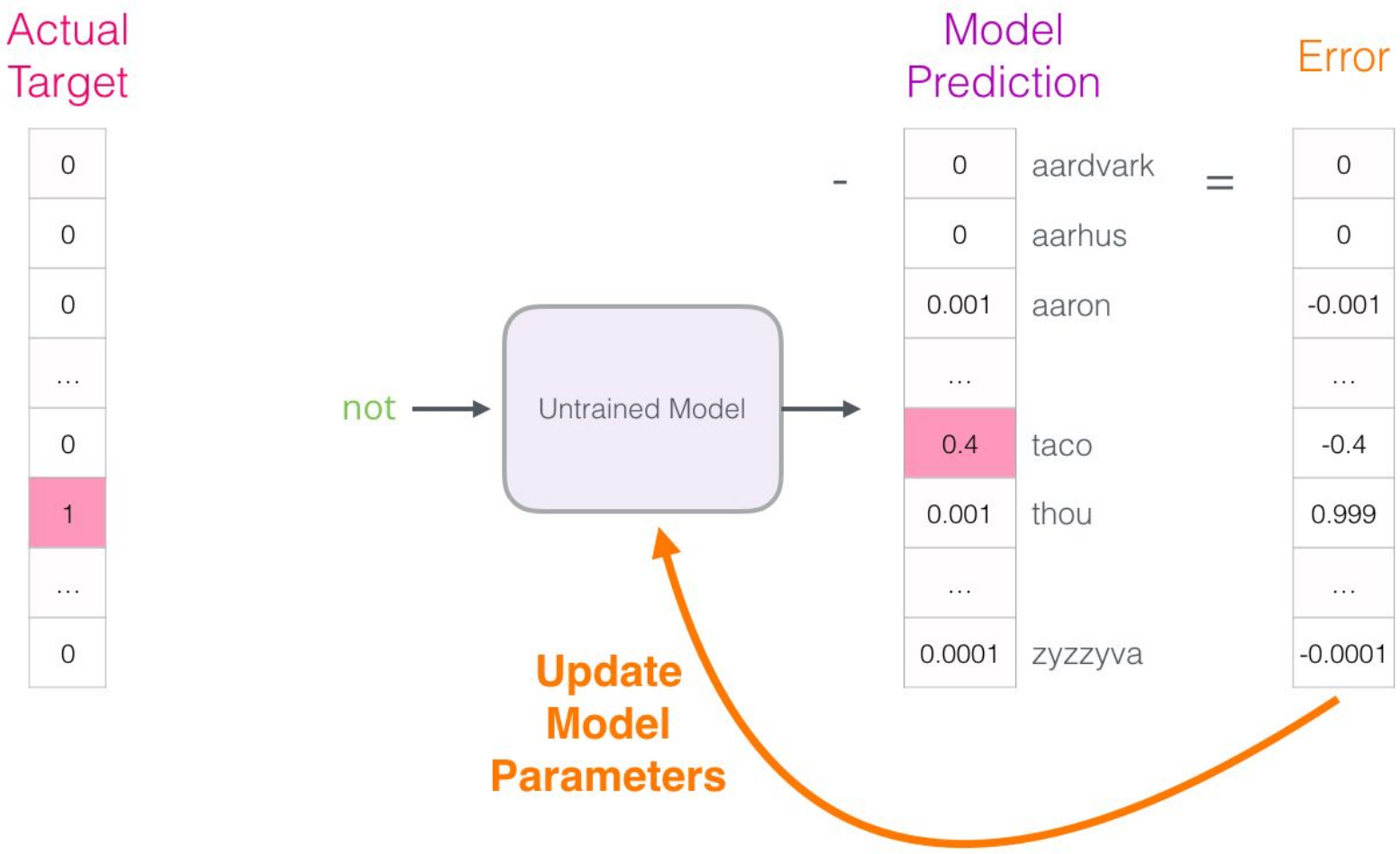
Word2Vec


not
Untrained Model
Task:
Predict neighbouring word
0
aardvark
0
aarhus
0.001
aaron
...
0.4
taco
0.001
thou
...
0.0001
zyzzyva
1) Look up embeddings
2) Calculate prediction
3) Project to output vocabulary

Word2Vec


Actual Target
Model Prediction
0
0
0
…
0
1
…
0
-
0
aardvark
0
aarhus
0.001
aaron
…
0.4
taco
0.001
thou
…
0.0001
zyzzyva

Word2Vec


Actual Target
0
0
0
…
0
1
…
0
not
Untrained Model
-
Model Prediction
0 aardvark
0 aarhus
0.001 aaron
…
0.4 taco
0.001 thou
…
0.0001 zyzzyva
=
Error
0
0
-0.001
…
-0.4
0.999
…
-0.0001
Update Model Parameters

Word2Vec

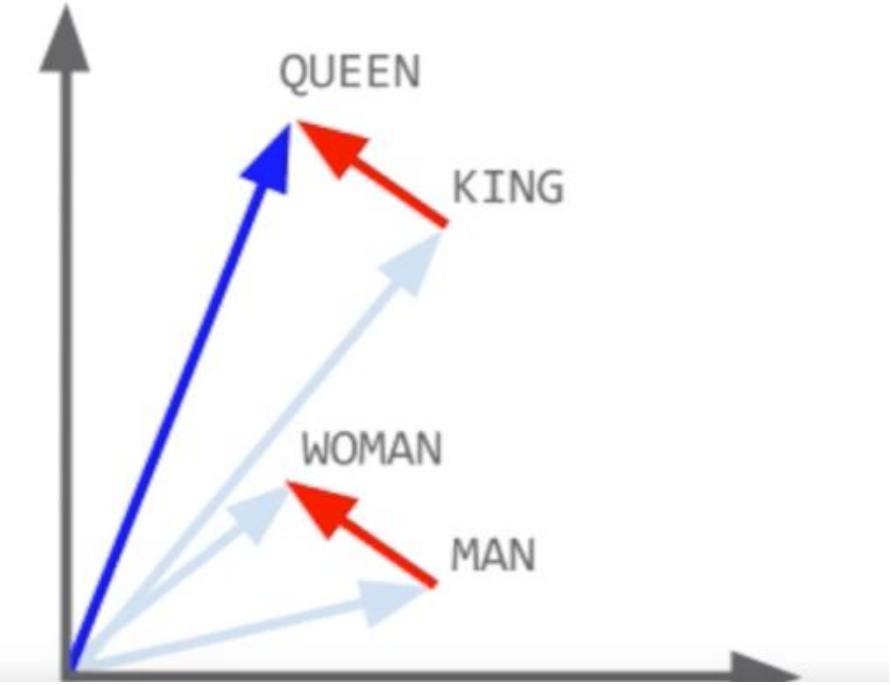
What size should the context window be?

Small window sizes → representations that are more interchangeable
Larger window sizes →representations that are semantically related

The idea is that **larger contexts** contain **more of a semantic domain** to influence a target word's meaning

Word2Vec

- These word representations turned out to be interpretable
- They obey a certain representational geometry where subtraction of vectors encoded a semantic meaning that could be added to other words
- The closest vectors in the model to these dimensions were very often coherent



Word2Vec - An Aside

- The representational geometry was not something we could have expected in advance
- This is absolutely not an "**obvious**" thing to happen (though in hindsight we pretend it was)
- The creator of word2vec, **Tomáš Mikolov,** recounts a funny story about this

> "*I wanted to convince my mentor at Microsoft that there was something interesting going on with the vector algebra, that* `king - man + woman = queen`*. I mentioned the idea and he didn't seem convinced at all that something like this could work. So, I asked him if he thought you could do plus / minus on the vectors and see correct results. He said "Of course not. That's completely stupid," and he was looking at me as if I had gone crazy. So, I took him to the computer and told him to look for the nearest neighbour and check the result. He was very surprised, but then got very excited and started thinking about how we could evaluate this in a more rigorous way.* " (My paraphrasing)

https://youtu.be/p-_0lOMfMYI?t=1275 (the interview)

Matrix Factorisation

GloVe (Global Vector for Word Representation)

- Word-word co-occurrence matrix derived from a corpus
- Uses statistics from entire corpus (*not just local context windows*, hence: **global**)
- Factorisation of this matrix results in numerical word vectors

Intuition:

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	high	low	high	low
$P(k\|steam)$	low	high	high	low
$P(k\|ice)/P(k\|steam)$	> 1	< 1	~ 1	~ 1

https://nlp.stanford.edu/projects/glove/

LMs without “Context”

- The term “**context**” is used a lot for the previous algorithms

- However, those word representations are highly ***non-contextual***
- Words with one written form and multiple meanings (= **homonyms)** are processed the same
  - i.e. the same vector for ‘row’ will be updated in the following cases:

    - A huge **row** erupted after the results were revealed (argument)
    - Everybody line up in a **row** (a line / ordering)
    - The team were able to **row** for hours at a time (propel a boat with an oar)

- Words with opposite meanings (= **antonyms**) often appear in the same linguistic contexts
  - this means the representations are similar even though the meanings are opposite
  - is this always a bad thing? A relationship does exist between them ….

LM Evaluation

- How can we **evaluate** how good / bad a language model is?
- Depending on the goal:
  - **Extrinsic Evaluation**
  - **Intrinsic Evaluation**

Extrinsic Evaluation

- LM is a component in an NLP system
- You have different LM versions
- Switch them in the system and measure the change in performance
- Often computationally expensive

Intrinsic Evaluation

- More of a quick & easy method
- Requires independent test set
- Compare LMs on how well they predict the independent test set
- Common metric is **perplexity**

LM Evaluation

Perplexity

- Perplexity is the inverse probability of test set, normalised by the number of words in the test set

$$\begin{aligned} \text{PP}(W) &= P(w_1 w_2 \ldots w_N)^{-\frac{1}{N}} \\ &= \sqrt[N]{\frac{1}{P(w_1 w_2 \ldots w_N)}} \end{aligned}$$

- The probability of the test set is rearranged according to the chain rule of probability

$$\text{PP}(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i | w_1 \ldots w_{i-1})}}$$

- In this example, consider a bigram language model

$$\text{PP}(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i | w_{i-1})}}$$

See Jurafsky & Martin, Section 3.2.1 for more details

LM Evaluation

Perplexity

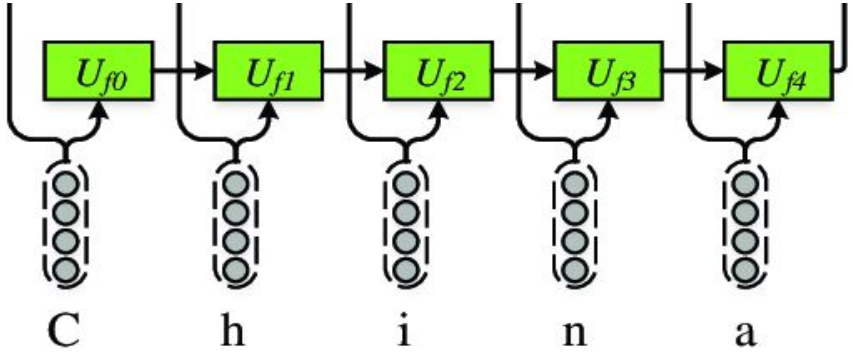
Finally, let's look at an example of how perplexity can be used to compare different n-gram models. We trained unigram, bigram, and trigram grammars on 38 million words (including start-of-sentence tokens) from the *Wall Street Journal*, using a 19,979 word vocabulary. We then computed the perplexity of each of these models on a test set of 1.5 million words with Eq. 3.16. The table below shows the perplexity of a 1.5 million word WSJ test set according to each of these grammars.

	Unigram	Bigram	Trigram
Perplexity	962	170	109

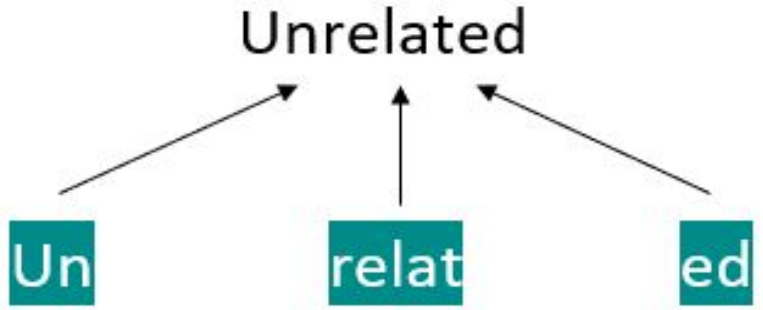
See Jurafsky & Martin, Section 3.2.1 for more details

Beyond Word Vectors

- Character-based models
  - Easily solve the “unknown word” problem
  - Lose a lot of information at higher levels, such as common morphemes in a language

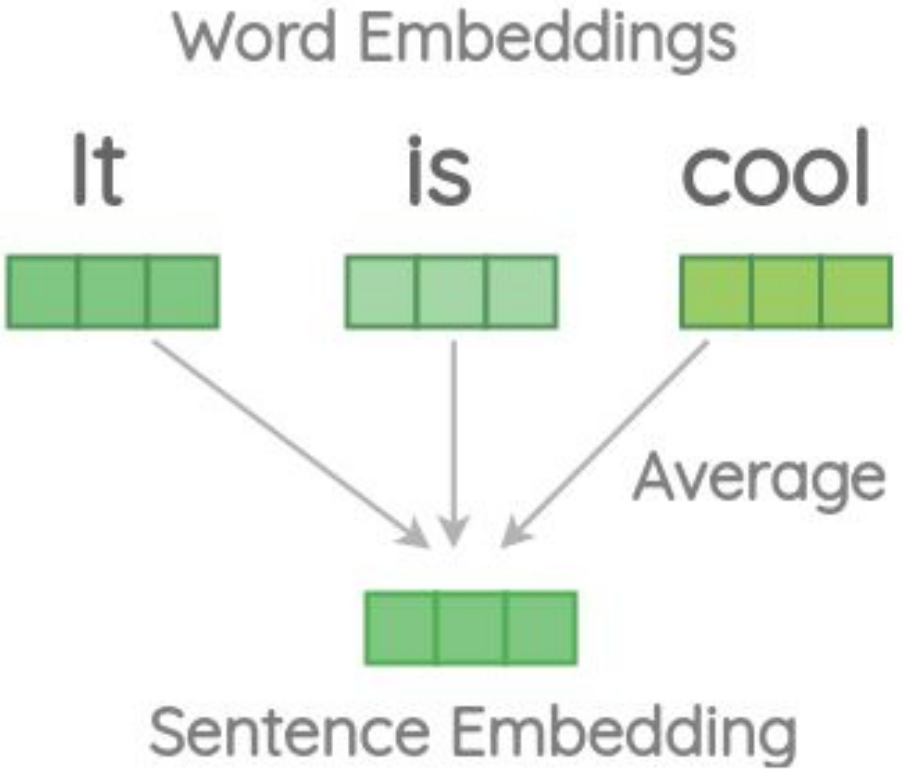


- Subword-based models
  - A compromise between word and character level models
  - Capture common morphemes and linguistically salient units, while also able to deal with novel vocabulary



Beyond Word Vectors

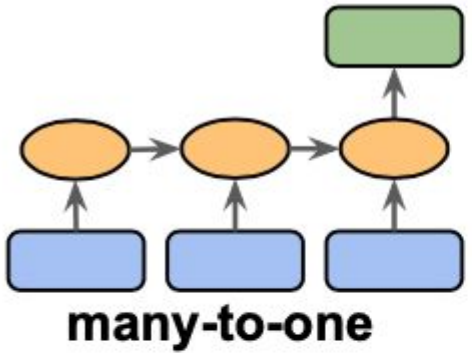
- How should sentences be represented?

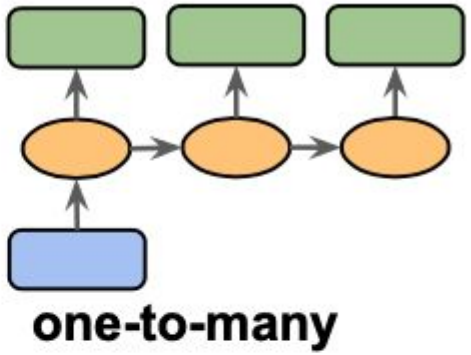


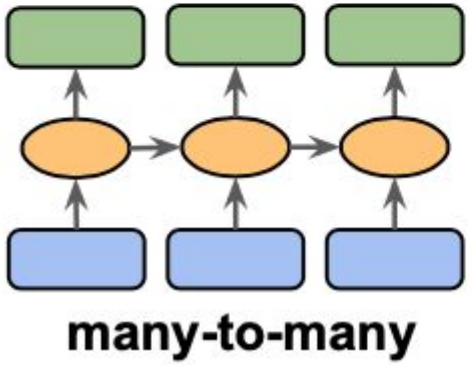
Language as a Sequence

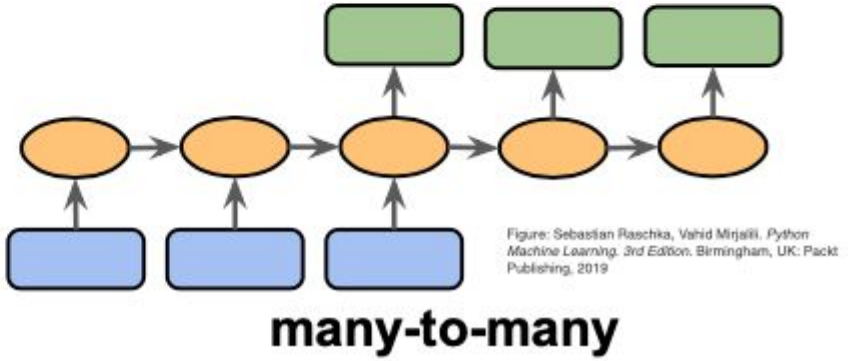
Language is easily framed as a sequential decoding problem

- *Letters in a word*
- *Words in a sentence*
- *Sentences in a document*
- *The aural equivalent, decoding ongoing speech* etc.



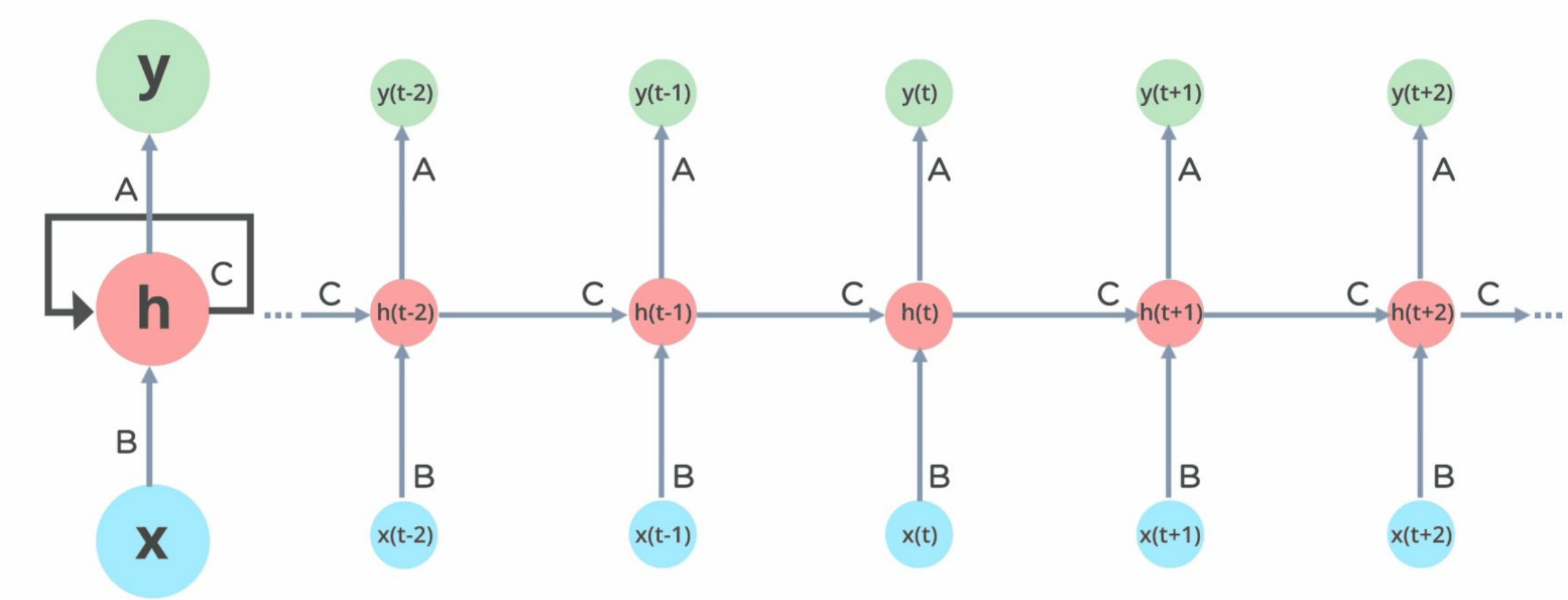






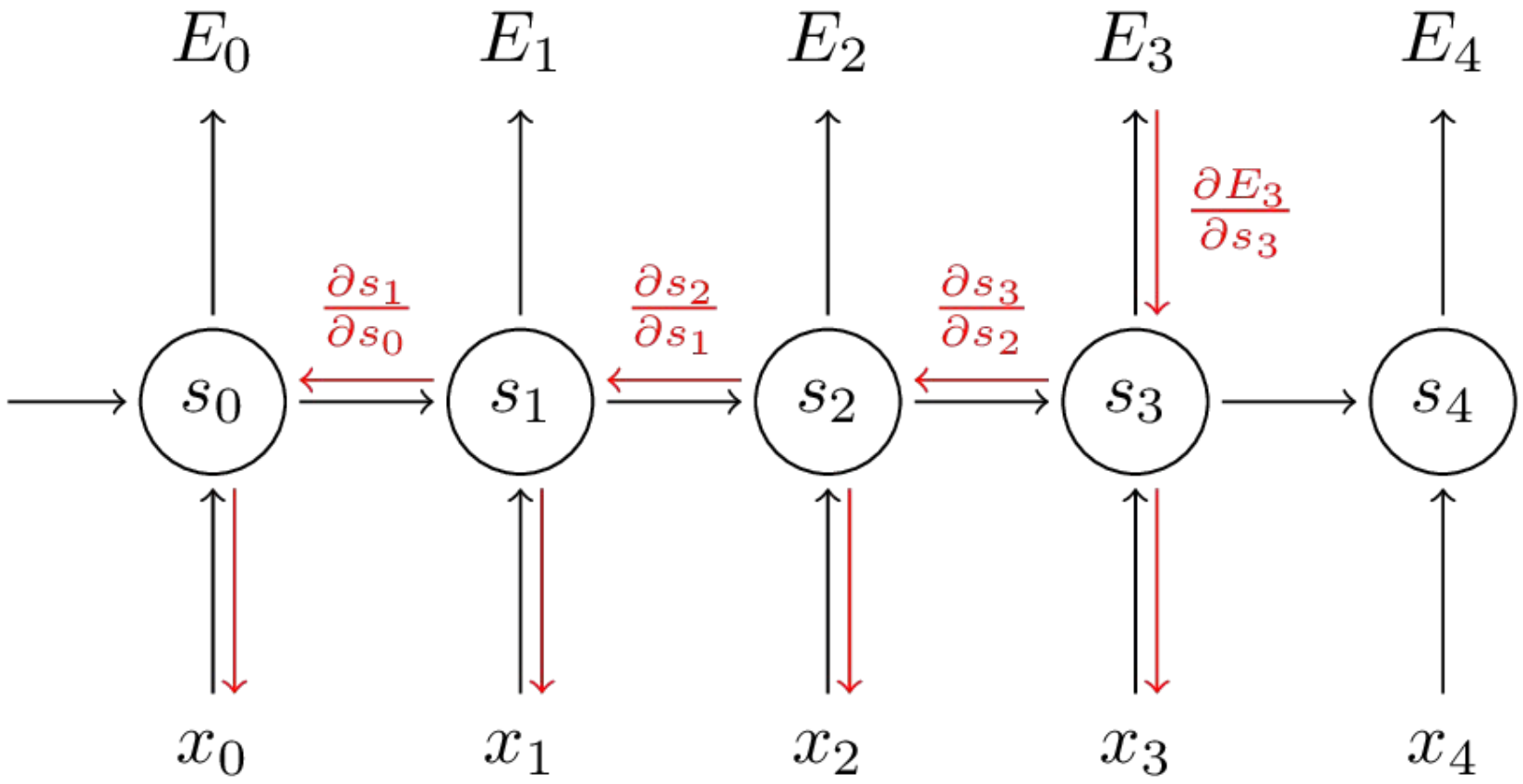
https://deeplearningmath.org/sequence-models

Recurrent Neural Networks



https://www.analyticsvidhya.com/blog/2022/03/a-brief-overview-of-recurrent-neural-networks-rnn/

RNNs: Vanishing Gradients



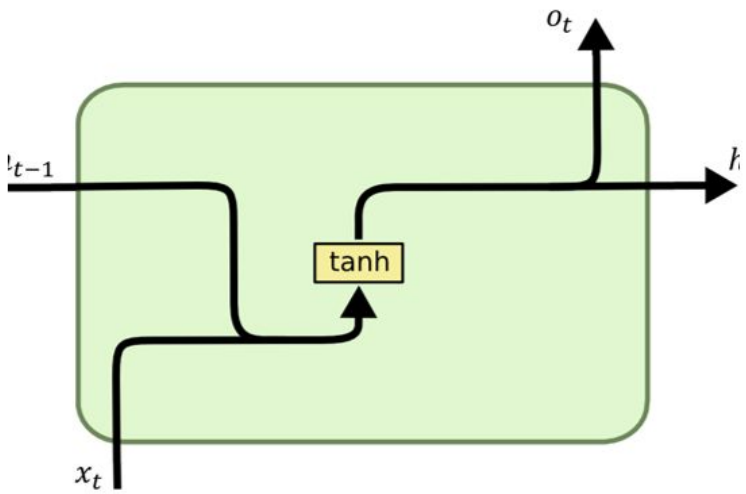
https://www.analyticsvidhya.com/blog/2021/07/lets-understand-the-problems-with-recurrent-neural-networks/

RNNs: Solutions?

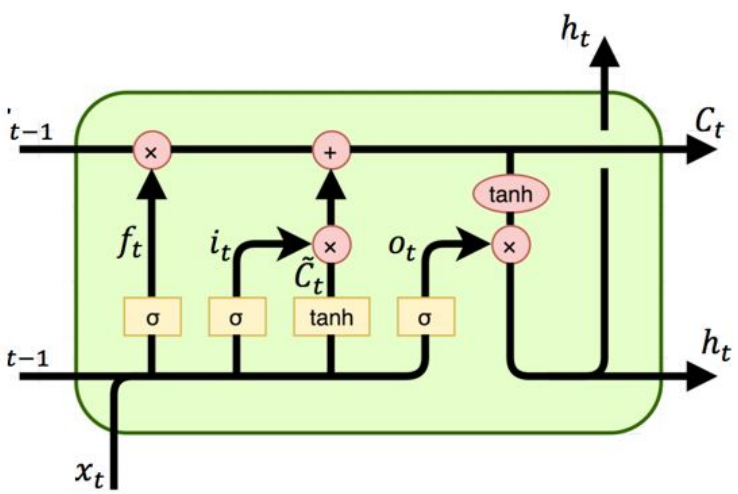
- Gated Recurrent Units (GRU)
- Long Short-Term Memory (LSTM)
- Ongoing “state” passed forward over time

RNNs: Solutions

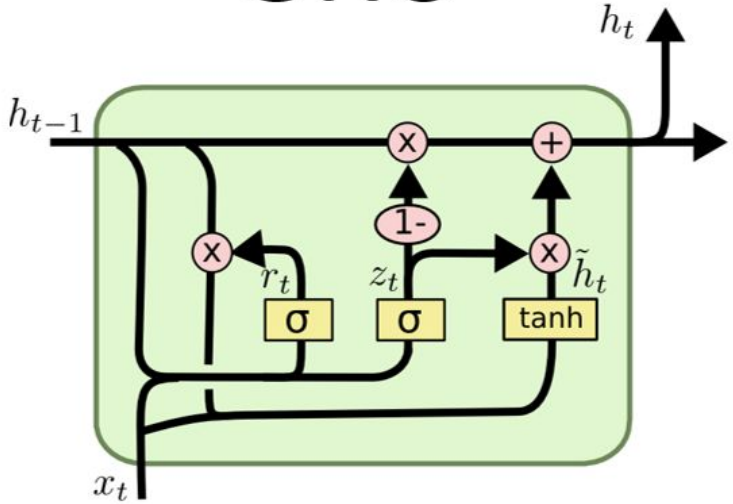
RNN



LSTM

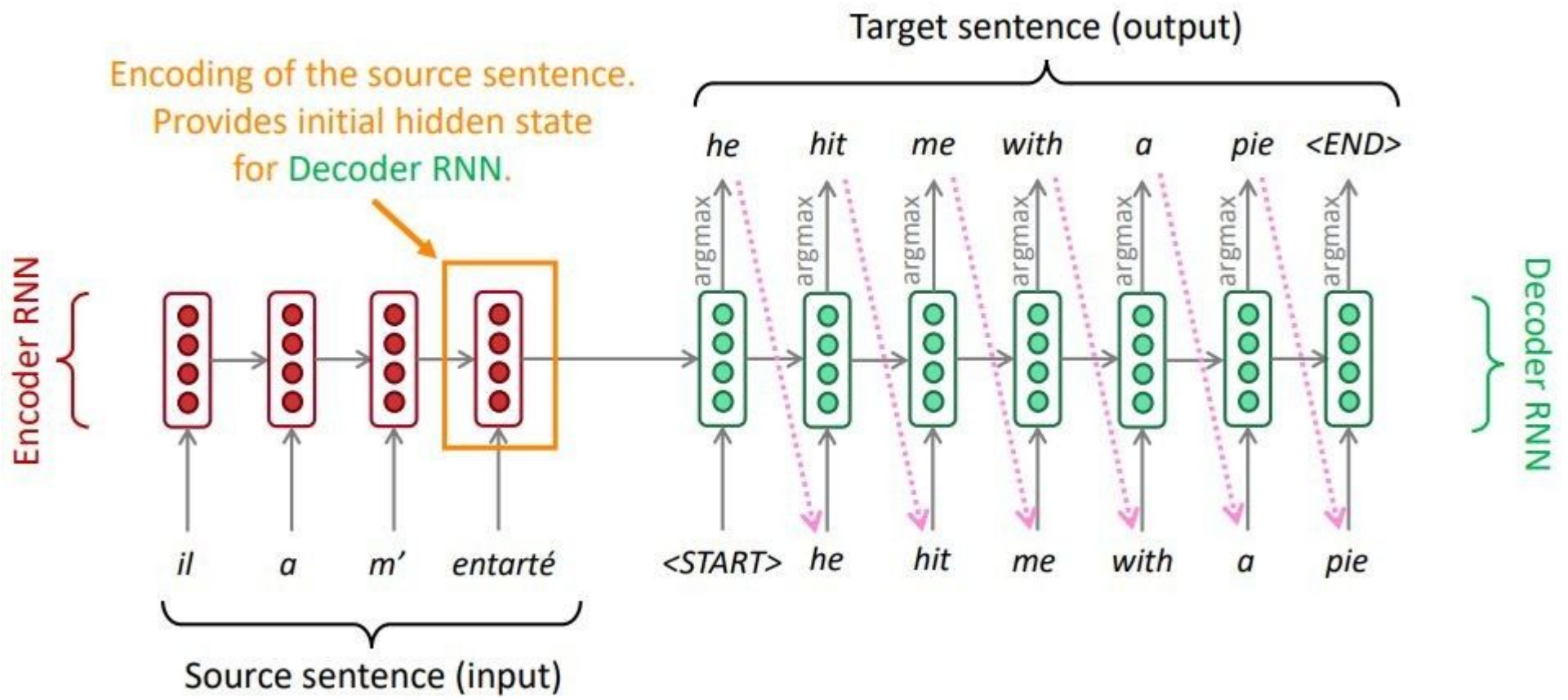


GRU



http://dprogrammer.org/rnn-lstm-gru

Bottleneck Problem



https://www.linkedin.com/pulse/attention-neural-machine-translation-nmt-without-pain-sobh-phd/

Bottleneck Problem

- The full representation of the input (to be encoded) needs to be well “represented” by the final state of the encoder
- The decoder is passed this final state of the encoder and needs to start iteratively producing sequential outputs
- Context lengths differ among tasks
- Can encode simple sentences? Sure
- Can encode long documents with lots of information? Hmm?

RNNs: Enter Attention

- What if we don’t need to “cram” everything into the final state of the decoder
- What if we store the outputs at each encoding step
- We learn a linear weighting of these encoding vectors
- Important: at different decoding steps, different parts of the input become more / less relevant

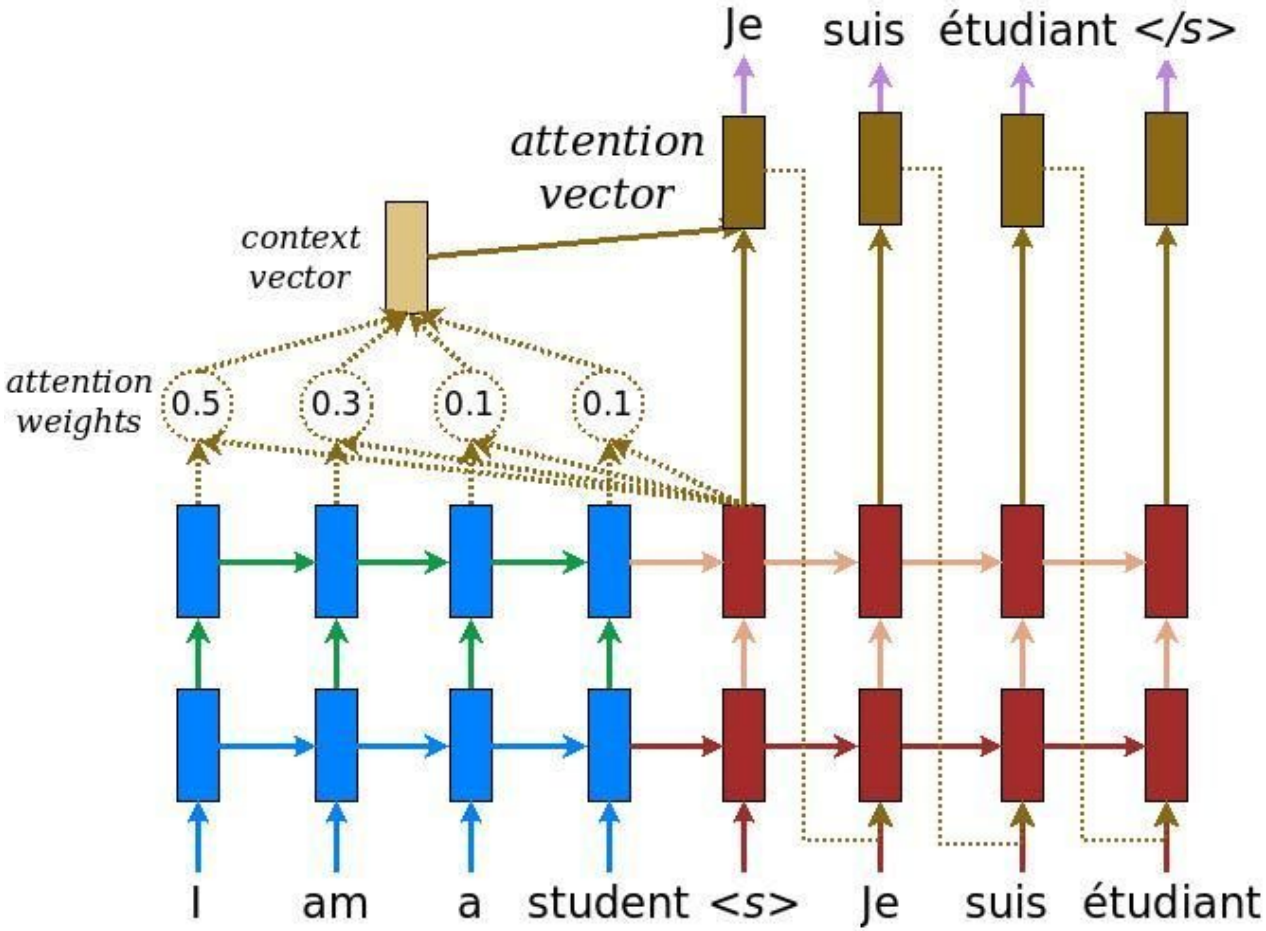
RNNs: Enter Attention

We take the output from the final state of the encoder and also weighted sum of all previous encoder states.

When decoding “Je”, you see highest attention weight on the “I” embedding.

As you step through decoding steps (gradually translating), different attention weights are mixed that help to decode at that specific position.

How does a model know how to weight the encoder embeddings during translation? The weights are learned and predicted as a function of the current “state” of the RNN.



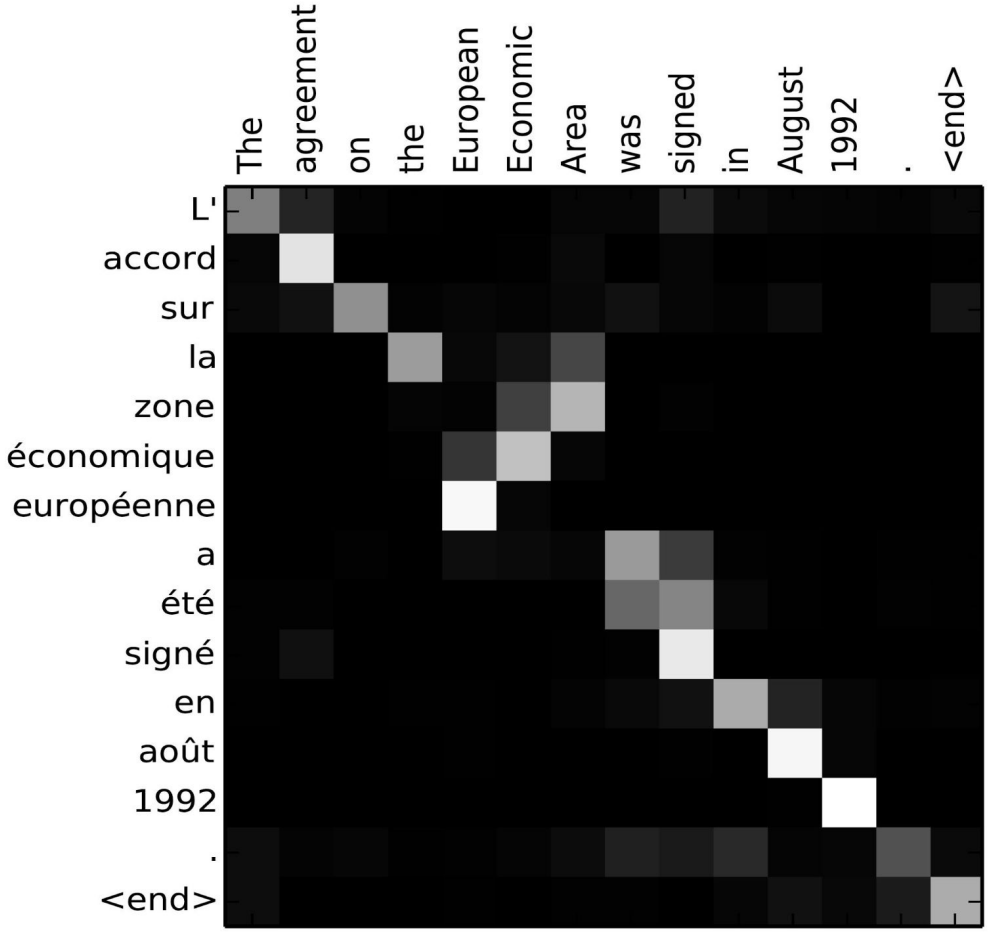
https://medium.com/syncedreview/a-brief-overview-of-attention-mechanism-13c578ba9129

RNNs: Enter Attention

English & French differ in respect of noun + adjective order:

Black cat
Chat noir

During translation, from seeing the pattern repeated many times during training, attention weights know to focus on the correct order (not just taking attention at relative position in both sequences).



https://lilianweng.github.io/posts/2018-06-24-attention/

Transformers

RNNs have to be trained sequentially.

But if you encode token position via “positional encodings” in a sequence, then you can do away with the recurrence requirement.

Iteratively decoding based on attention alone is much faster, allows for big gains in scaling / computational efficiency -> **huge boost in learning**

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

Transformers

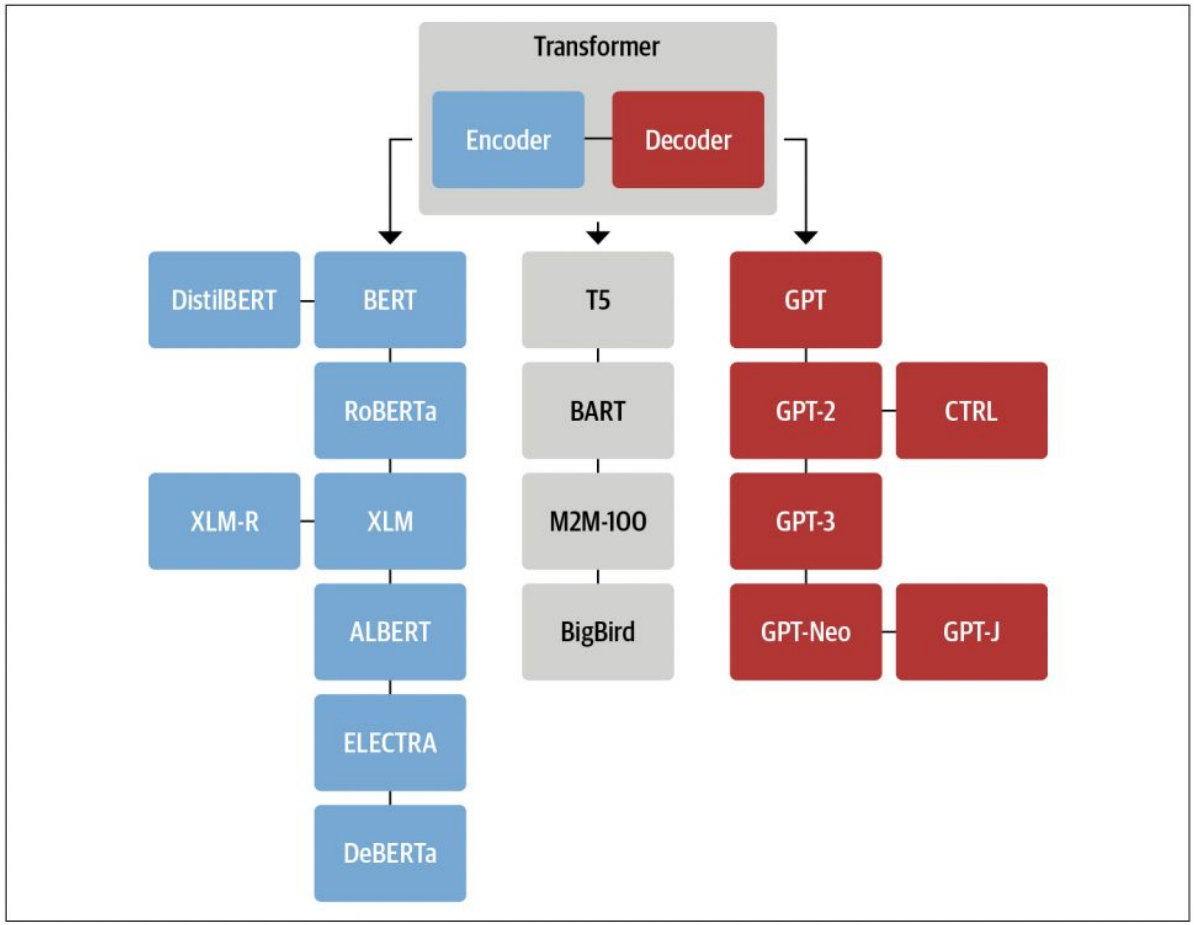


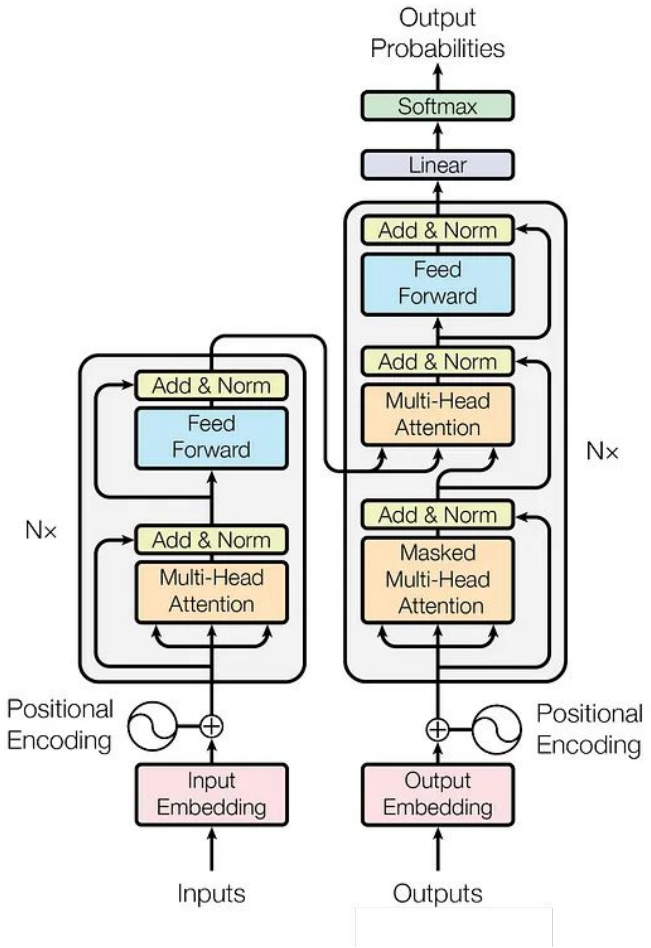
Figure 3-8. An overview of some of the most prominent transformer architectures

From “Natural Language Processing with Transformers”

Transformers

BERT

Encoder



GPT

Decoder

LLMs & Brain Responses

- We saw in vision that CNNs were capable of modelling the ventral visual stream
- As models got *better at their tasks*, their predictions of visual responses got better

Is the same true for language?

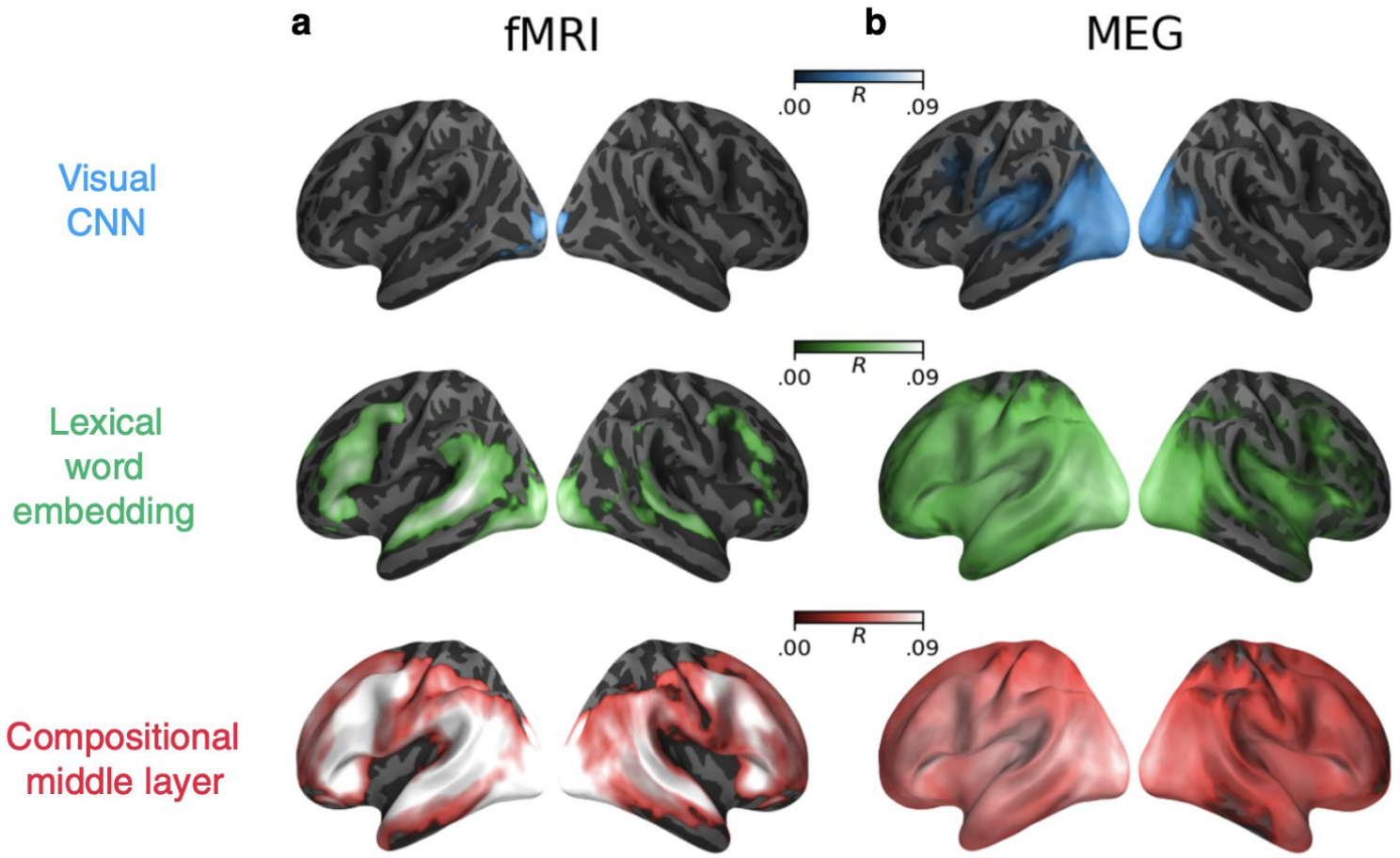
LLMs & Brain Responses

- We saw in vision that CNNs were capable of modelling the ventral visual stream
- As models got *better at their tasks*, their predictions of visual responses got better

Is the same true for language?

It appears so!

LLMs & Brain Responses



Caucheteux & King (2022)

LLMs & Brain Responses

In some of the upcoming paper presentations, we're going to see that models that get better at modelling language can progressively model more of the cortical language network.

Beyond just modelling …

Very recent work has shown that LLMs can generate stimuli to specifically result in stronger / weaker responses from fMRI in a way not dissimilar from Bashivan et al. (our last paper presentation).

We're converging on a theme here …

Questions?

UNIVERSITY
OF ALBERTA