Language Modeling, Transformers

CSE 473: Introduction to Artificial Intelligence

James Weichert (jpw@cs)
cs.uw.edu/473
acknowledgements

Welcome!
What's your favorite word?
sli.do **#cse473**

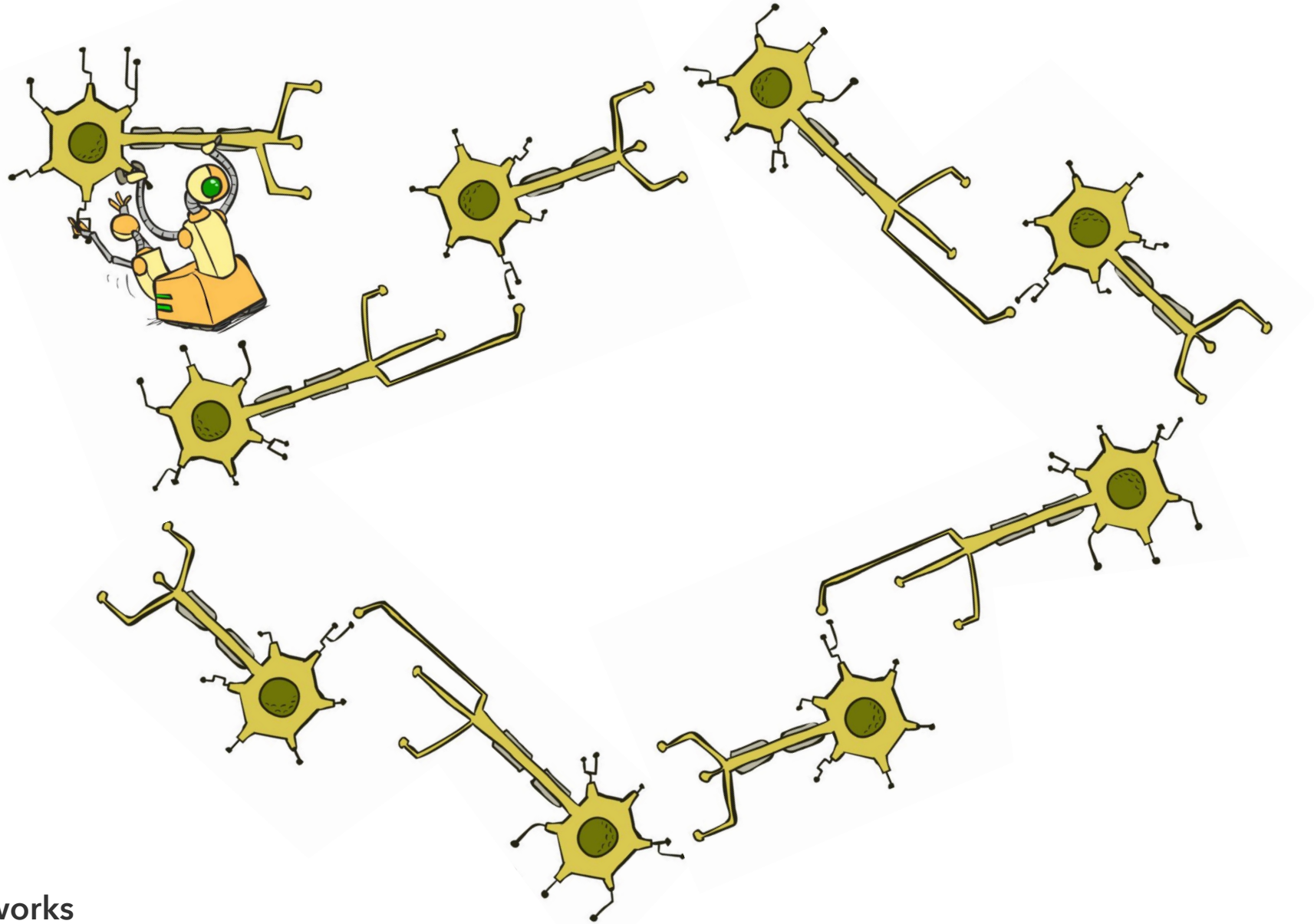
Administrivia

ANNOUNCEMENTS

- CSE 473 stickers!

ASSIGNMENTS

- **Homework 4 due** next week (8.20)
  - No late period – HW4 must be turned in by 8.21



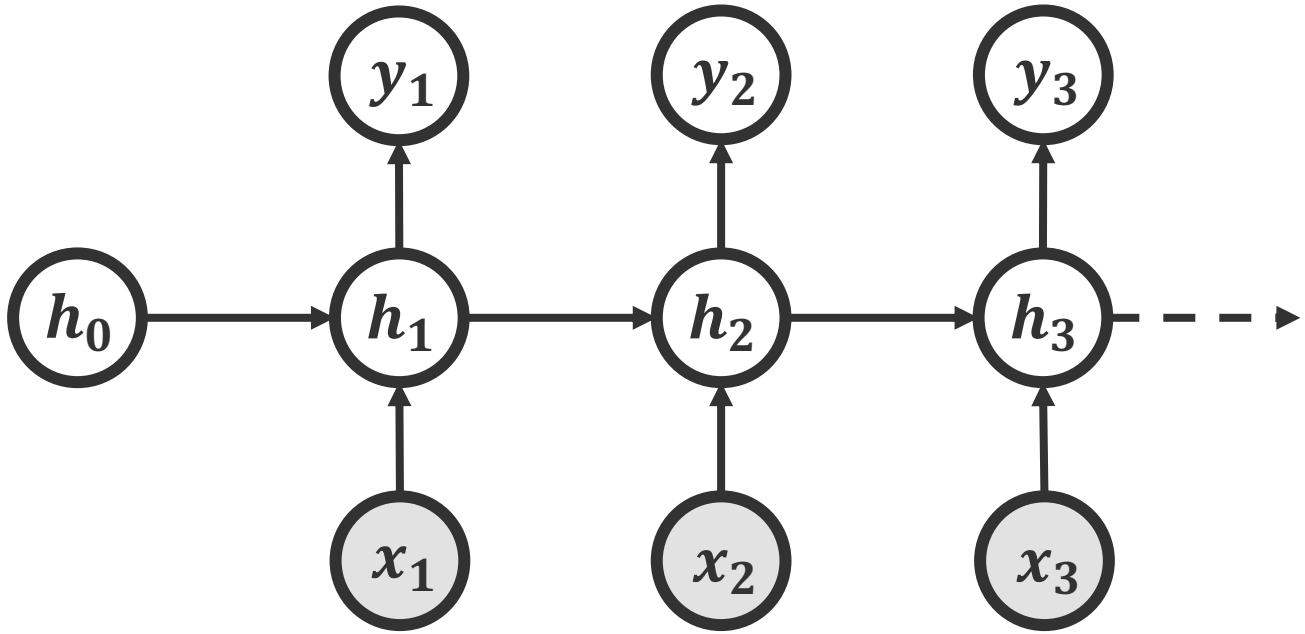
Recurrent Networks

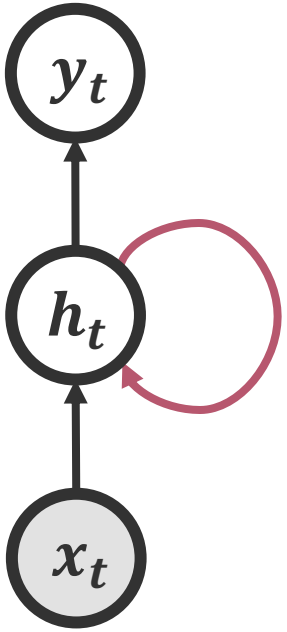
Dependent Sequences

CONTEXT

Prediction at time t depends on prediction at time $t-1$, etc.

- Time series data, moving images, *language*

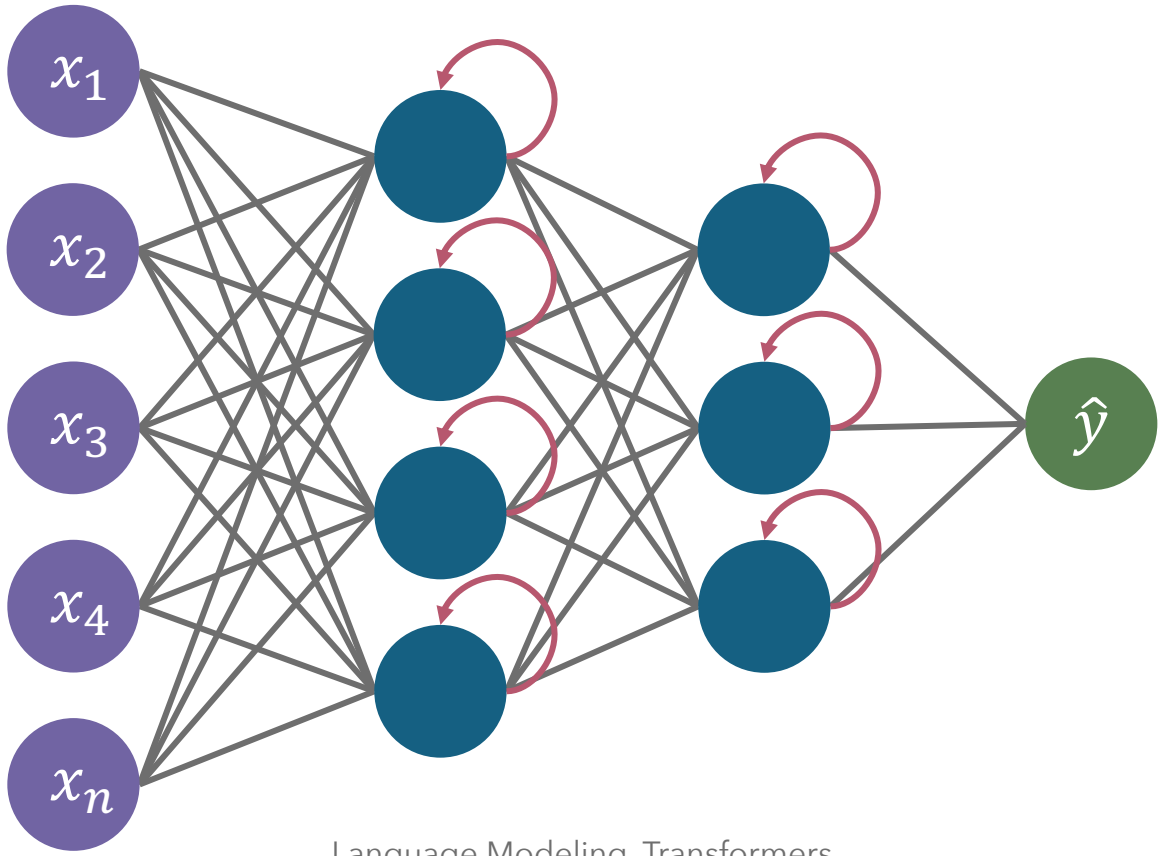




Recurrent Neural Networks

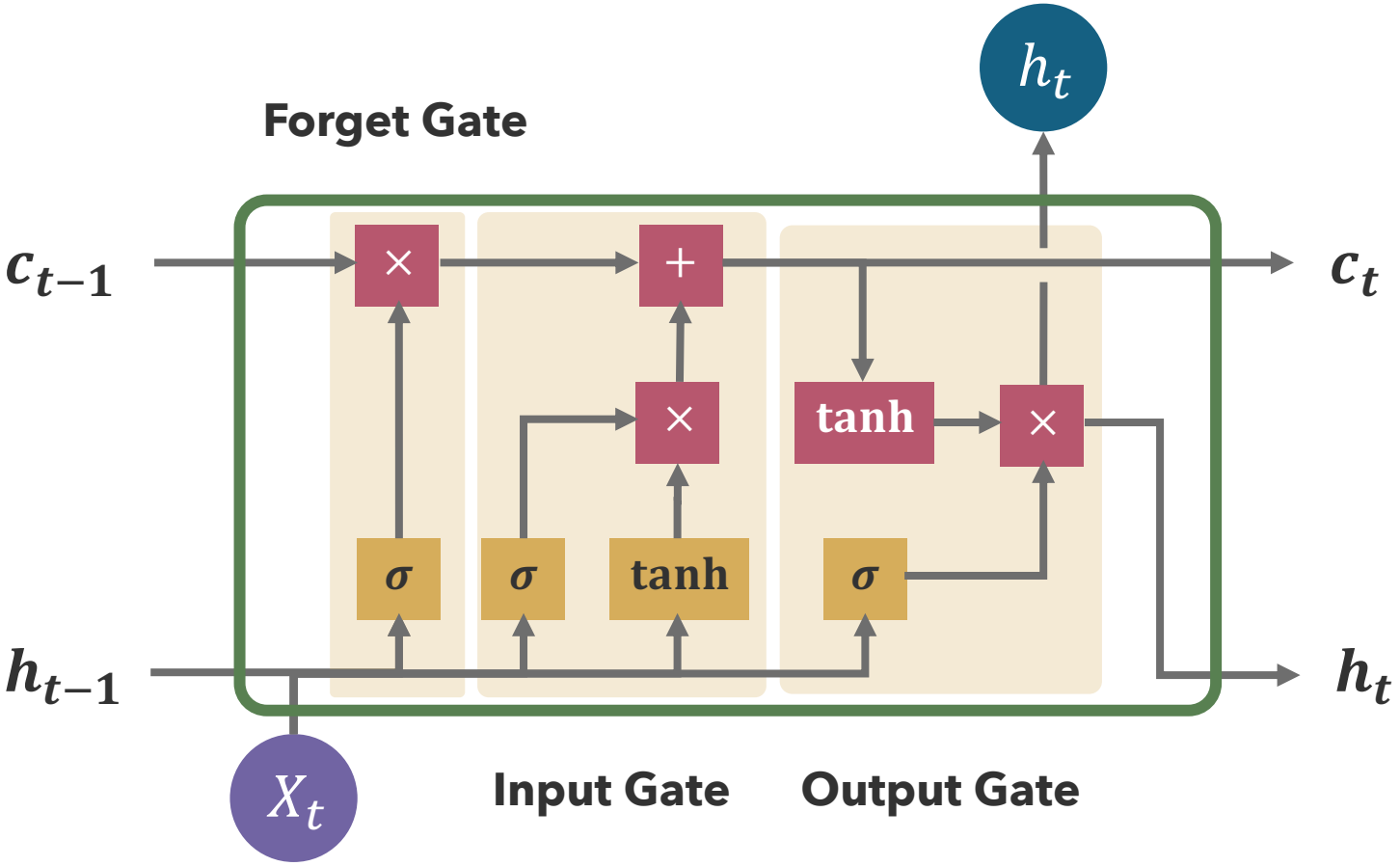
FOUNDATIONS

Recurrent Neural Networks include backwards (recurrent) connections



Long Short Term Memory (LSTM)

DEFINITION



LSTM Text Generation

EXAMPLE

"[dogs are] the companion animal most frequently reported for exposure to support earlier findings that pet ownership is each year, a common breeding practice for pet dogs are dispersed less likely to be found within the hunter of every dogs."

LSTM model code generated by Claude Code

Generated using text from the Wikipedia article on "Dog"

Problems with Sequential Processing

CONTEXT

Vanishing Gradients

Gradients shrink over long sequences

Mitigated with LSTMs

Memory Bottleneck

Hidden state (memory) has fixed size

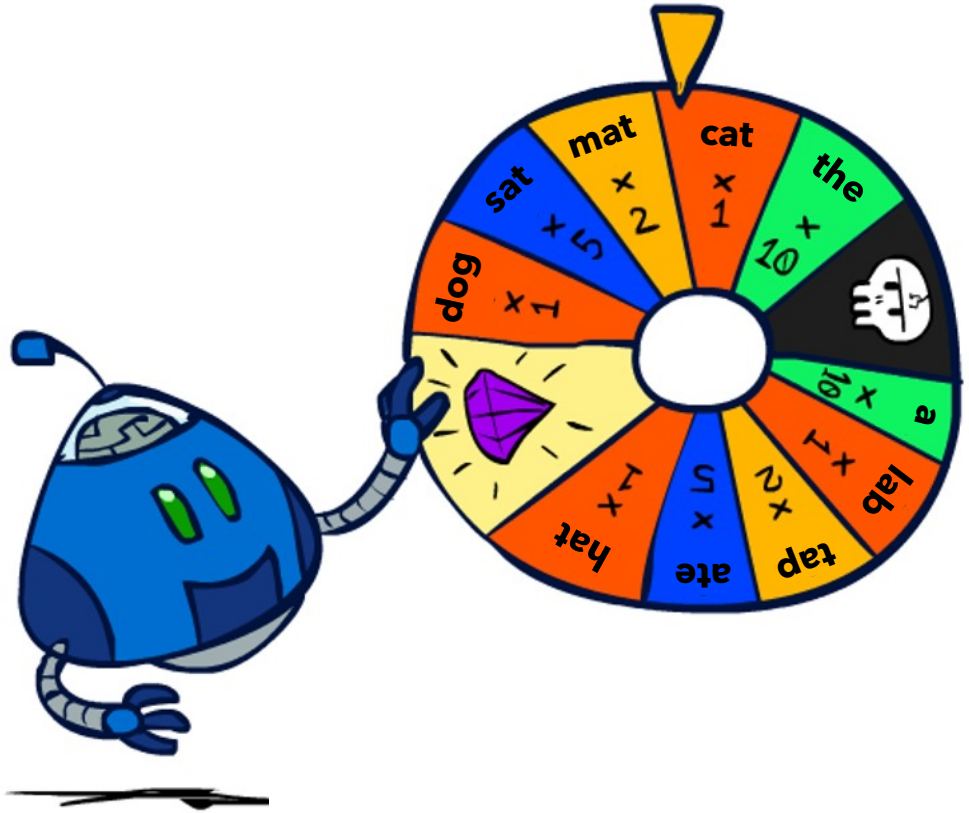
Training Speed

Cannot parallelize training across time steps

QUESTIONS

Questions?

live and on sli.do #cse473



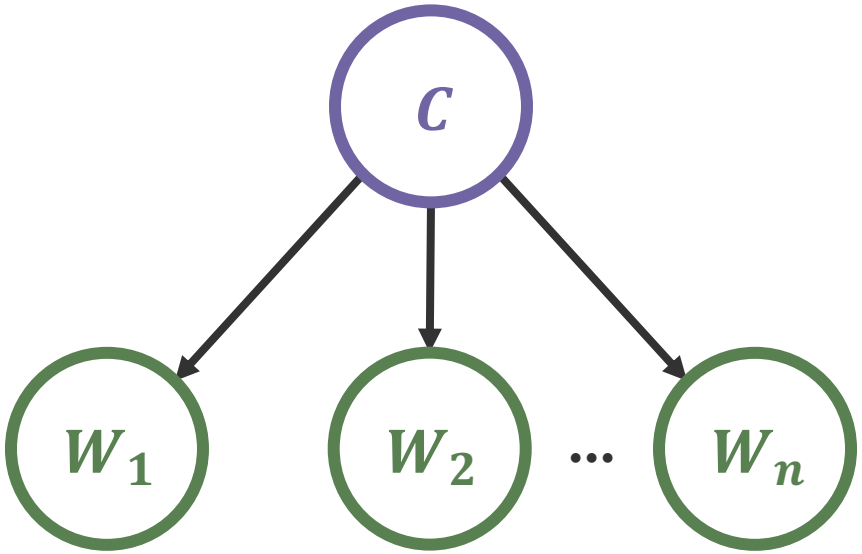
Modeling Language

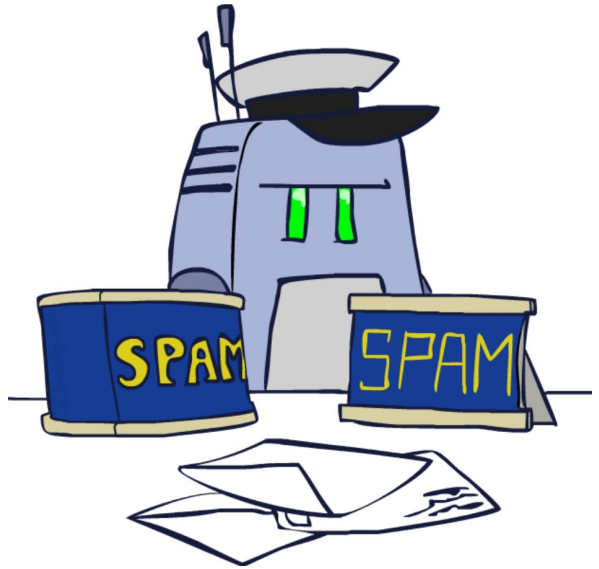
Bag of Words Modeling

CONTEXT

Model language generation as a **naïve Bayes network** where words are conditionally independent of each other.

Maybe good enough for **classification**, not effective for **generation**

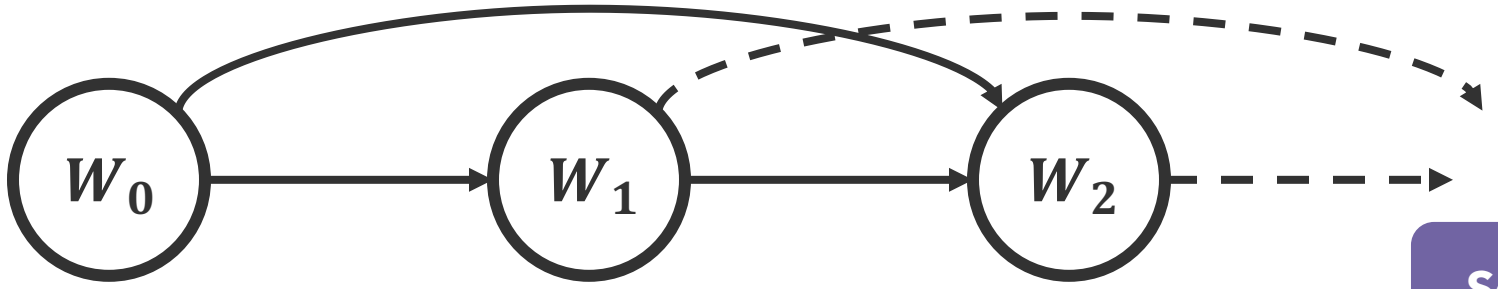




N-Gram Models

FOUNDATIONS

Leverage **Markov assumption** to model words as dependent only on a *subset* of prior words.



Search Problems!

Unigram	$P(W_0)$	$P(W_1)$	$P(W_2)$
Bigram	$P(W_0)$	$P(W_1\|W_0)$	$P(W_2\|W_1)$
Trigram	$P(W_0)$	$P(W_1\|W_0)$	$P(W_2\|W_1, W_0)$

$$\text{argmax}_{\boldsymbol{w}} \prod_{w_i} P(w_i)$$

$$\text{argmax}_{\boldsymbol{w}} \prod_{w_i} P(w_i|w_{i-1})$$

$$\text{argmax}_{\boldsymbol{w}} \prod_{w_i} P(w_i|w_{i-1}, w_{i-2})$$

N-Grams in Action

EXAMPLE

Bigram

"[dogs] are the dog population is dominated by the dog among its closest extant relatives"

Trigram

"[dogs are] a source of zoonoses for humans"

Generated using text from the Wikipedia article on "Dog"

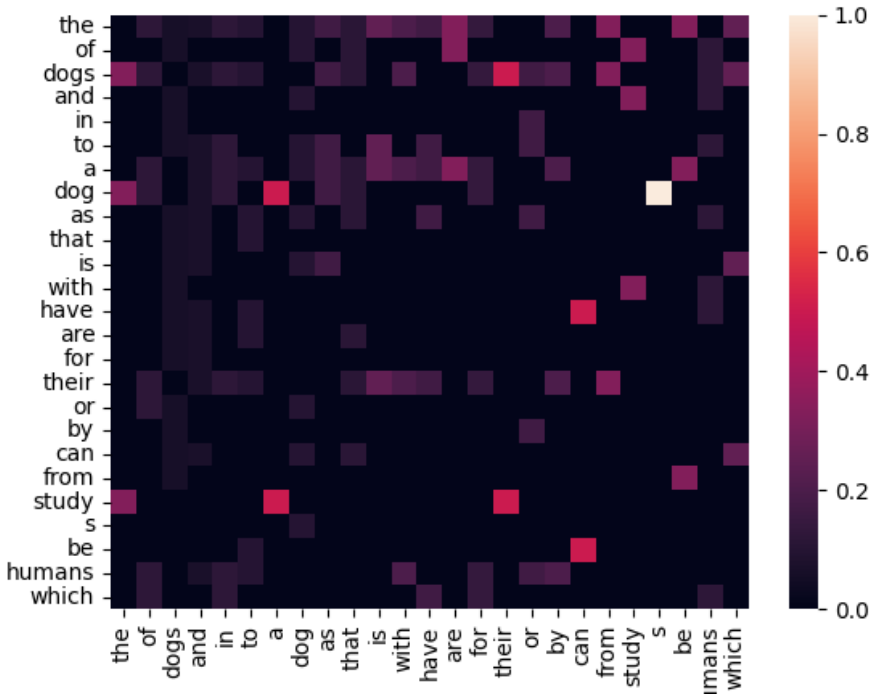
Problems with N-Grams

FOUNDATIONS

Tracking, Storing N-Grams

Sparse N-Grams

Poor Long-Term Context



Heatmap of bigrams for top-25 words in Wikipedia article on "Dog"

QUESTIONS

Questions?

live and on sli.do #cse473

Sentence Context

CONTEXT

vs.

Word Embeddings

FOUNDATIONS

For neural language models, we need a **numerical representation** for words:

- One-hot encoding
  - Each word has its own vector position

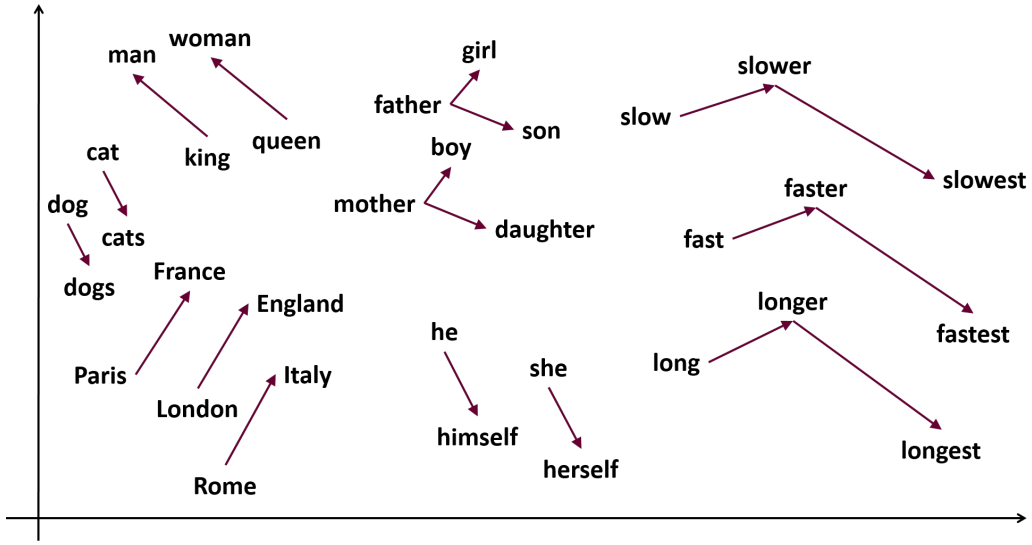
  $\text{cat} = [1 \quad 0 \quad 0], \text{dog} = [0 \quad 1 \quad 0], \text{bird} = [0 \quad 0 \quad 1]$

- Vector embeddings
  - Map words to d-dimensional embedding space

  $\text{cat} = [0.25 \quad \cdots \quad -1.3], \text{dog} = [0.8 \quad \cdots \quad -1.2]$

  - Semantic meaning encoded in spatial relationships

  $$\text{dog} + (\text{kitten} - \text{cat}) = \text{puppy}$$



Embeddings Embed Societal Bias

CONTEXT

RESEARCH

REPORT

COGNITIVE SCIENCE

Semantics derived automatically from language corpora contain human-like biases

Aylin Caliskan,[1]* Joanna J. Bryson,[1,2]* Arvind Narayanan[1]*

Machine learning is a means to derive artificial intelligence by discovering patterns in existing data. Here, we show that applying machine learning to ordinary human language results in human-like semantic biases. We replicated a spectrum of known biases, as measured by the Implicit Association Test, using a widely used, purely statistical machine-learning model trained on a standard corpus of text from the World Wide Web. Our results indicate that text corpora contain recoverable and accurate imprints of our historic biases, whether morally neutral as toward insects or flowers, problematic as toward race or gender, or even simply veridical, reflecting the status quo distribution of gender with respect to careers or first names. Our methods hold promise for identifying and addressing sources of bias in culture, including technology.

We show that standard machine learning can acquire stereotyped biases from textual data that reflect everyday human culture. The general idea that text corpora capture semantics, including cultural stereotypes and empirical associations, has long been known in corpus linguistics (*1*, *2*), but our findings add to this knowledge in three ways. First, we used word embeddings (*3*), a powerful tool to extract associations captured in text corpora; this method substantially amplifies the signal found in raw statistics. Second, our replication of documented human biases may yield tools and insights for studying prejudicial attitudes and behavior in humans. Third, since we performed our experiments on off-the-shelf machine learning components [primarily the Global Vectors for Word Representation (GloVe) word embedding], we show that cultural stereotypes propagate to artificial intelligence (AI) technologies in widespread use.

Before presenting our results, we discuss key terms and describe the tools we use. Terminology varies by discipline; these definitions are intended for clarity of the present article. In AI and machine learning, bias refers generally to prior information, a necessary prerequisite for intelligent action (*4*). Yet bias can be problematic where such information is derived from aspects of human culture known to lead to harmful behavior. Here, we will call such biases "stereotyped" and actions taken on their basis "prejudiced."

We used the Implicit Association Test (IAT) as our primary source of documented human biases (*5*). The IAT demonstrates enormous differences in response times when subjects are asked to pair two concepts they find similar, in contrast to two concepts they find different. We developed our first method, the Word-Embedding Association Test (WEAT), a statistical test analogous to the IAT, and applied it to a widely used semantic representation of words in AI, termed word embeddings. Word embeddings represent each word as a vector in a vector space of about 300 dimensions, based on the textual context in which the word is found. We used the distance between a pair of vectors (more precisely, their cosine similarity score, a measure of correlation) as analogous to reaction time in the IAT. The WEAT compares these vectors for the same set of words used by the IAT. We describe the WEAT in more detail below.

Most closely related to this paper is concurrent work by Bolukbasi *et al.* (*6*), who propose a method to "debias" word embeddings. Our work is complementary, as we focus instead on rigorously demonstrating human-like biases in word embeddings. Further, our methods do not require an algebraic formulation of bias, which may not be possible for all types of bias. Additionally, we studied the relationship between stereotyped associations and empirical data concerning contemporary society.

Using the measure of semantic association described above, we have been able to replicate every stereotype that we tested. We selected IATs that studied general societal attitudes, rather than those of subpopulations, and for which lists of target and attribute words (rather than images) were available. The results are summarized in Table 1.

Greenwald *et al.* introduced and validated the IAT by studying biases that they consider nearly universal in humans and about which there is no social concern (*5*). We began by replicating these inoffensive results for the same purposes. Specifically, they demonstrated that flowers are significantly more pleasant than insects, based on the reaction latencies of four pairings (flowers + pleasant, insects + unpleasant, flowers + unpleasant, and insects + pleasant). Greenwald *et al.* measured effect size in terms of Cohen's *d*, which is the difference between two means of log-transformed latencies in milliseconds, divided by the standard deviation. Conventional small, medium, and large values of *d* are 0.2, 0.5, and 0.8, respectively. With 32 participants, the IAT comparing flowers and insects resulted in an effect size of 1.35 ($P < 10^{-8}$). Applying our method, we observed the same expected association with an effect size of 1.50 ($P < 10^{-7}$). Similarly, we replicated Greenwald *et al.*'s finding (*5*) that musical instruments are significantly more pleasant than weapons (see Table 1).

Notice that the word embeddings "know" these properties of flowers, insects, musical instruments, and weapons with no direct experience of the world and no representation of semantics other than the implicit metrics of words' co-occurrence statistics with other nearby words.

We then used the same technique to demonstrate that machine learning absorbs stereotyped biases as easily as any other. Greenwald *et al.* (*5*) found extreme effects of race as indicated simply by name. A bundle of names associated with being European American was found to be significantly more easily associated with pleasant than unpleasant terms, compared with a bundle of African-American names.

In replicating this result, we were forced to slightly alter the stimuli because some of the original African-American names did not occur in the corpus with sufficient frequency to be included. We therefore also deleted the same number of European-American names, chosen at random, to balance the number of elements in the sets of two concepts. Omissions and deletions are indicated in our list of keywords (see the supplementary materials).

In another widely publicized study, Bertrand and Mullainathan (*7*) sent nearly 5000 identical résumés in response to 1300 job advertisements, varying only the names of the candidates. They found that European-American candidates were 50% more likely to be offered an opportunity to be interviewed. In follow-up work, they argued that implicit biases help account for these effects (*8*).

We provide additional evidence for this hypothesis using word embeddings. We tested the names in their study for pleasantness associations. As before, we had to delete some low-frequency names. We confirmed the association using two different sets of "pleasant/unpleasant" stimuli: those from the original IAT paper and also a shorter, revised set published later (*9*).

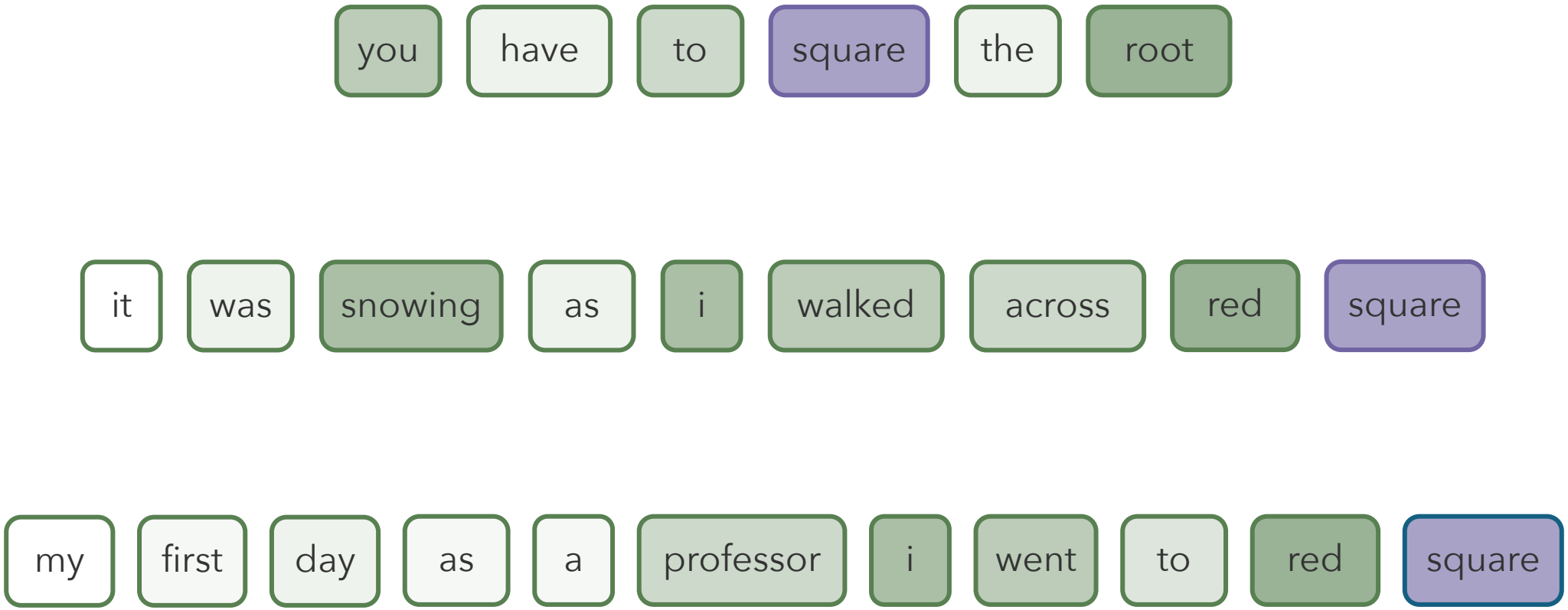
Turning to gender biases, we replicated a finding that female names are more associated with family than career words, compared with male names (*9*). This IAT was conducted online and thus has a vastly larger subject pool but far fewer keywords. We replicated the IAT results even with these reduced keyword sets. We also replicated an online IAT finding that female words (e.g., "woman" and "girl") are more associated than male words with the arts than with mathematics (*9*). Finally, we replicated a laboratory study showing that

[1]Center for Information Technology Policy, Princeton University, Princeton, NJ, USA. [2]Department of Computer Science, University of Bath, Bath BA2 7AY, UK.
*Corresponding author. Email: aylinc@princeton.edu (A.C.); jjb@alum.mit.edu (J.J.B.); arvindn@cs.princeton.edu (A.N.)

Caliskan et al. (2017)

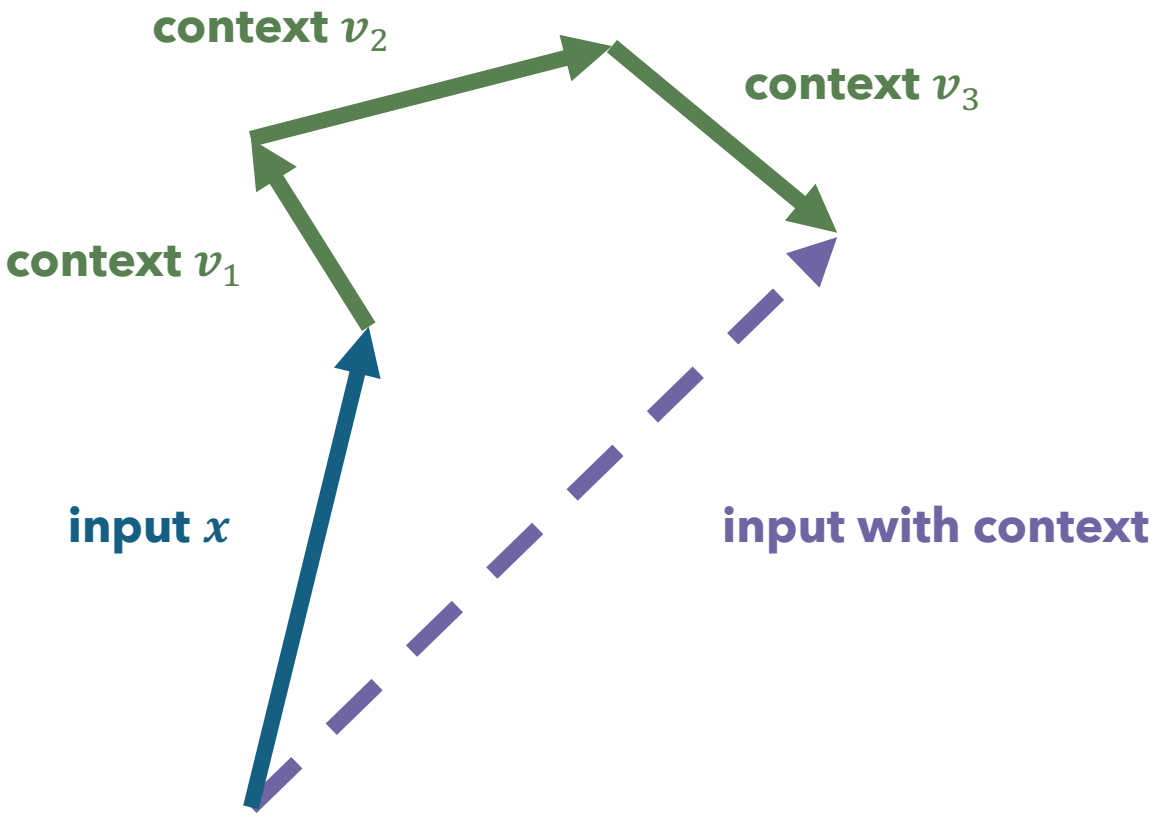
Adaptive Context

CONTEXT



Embeddings + Context

CONTEXT



QUESTIONS

Questions?

live and on sli.do #cse473

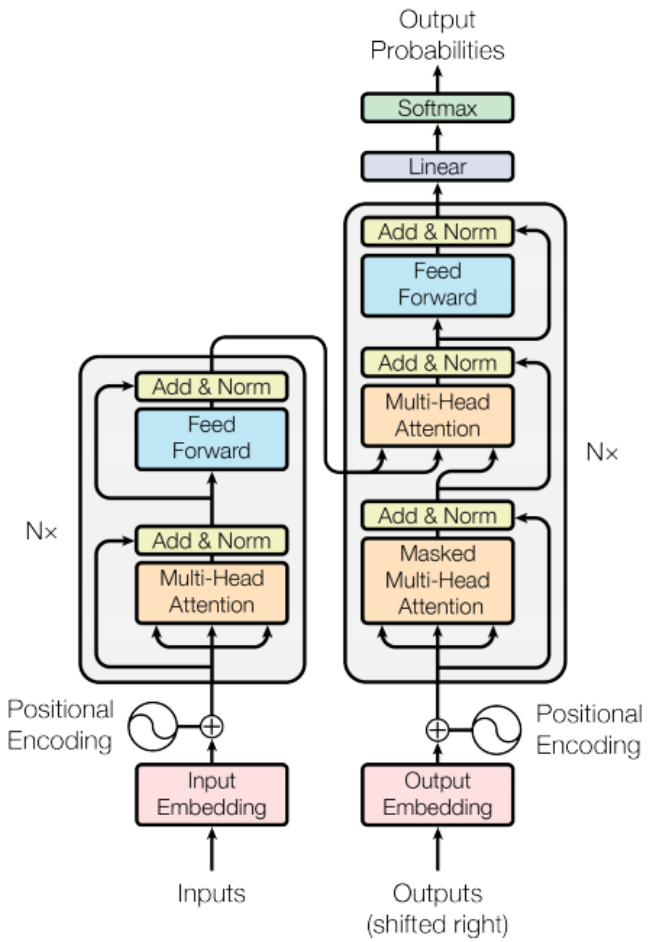


Figure 1: The Transformer - model architecture.

Vaswani et al. (2017)

Transformer Models

Attention

DEFINITION

The attention mechanism scales outputs V by relevance of key K to query Q

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) \cdot V$$

Query Q

"What am I looking for?"

Current token's information needs

Key K

"What do I contain?"

Summarizes information offered by token

Value V

"What do I provide?"

Actual information provided by token

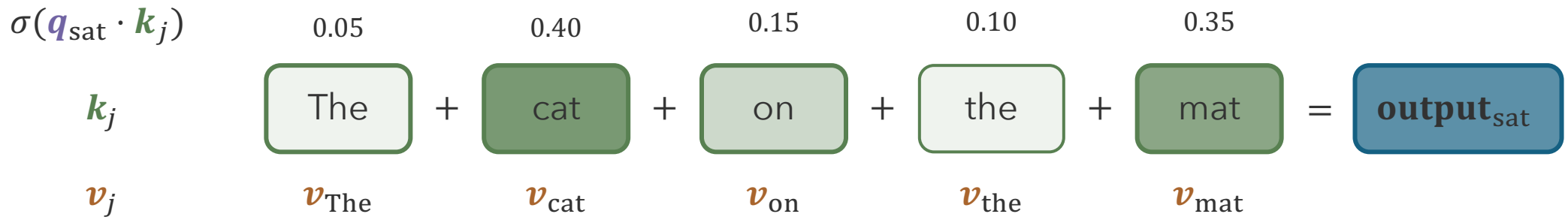
Attention Outputs

EXAMPLE

The output is a weighted sum of input embeddings

$$\mathbf{output}_i = \sum_j \text{softmax}\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j}{\sqrt{d_k}}\right)\boldsymbol{v}_j$$

"The cat sat on the mat"



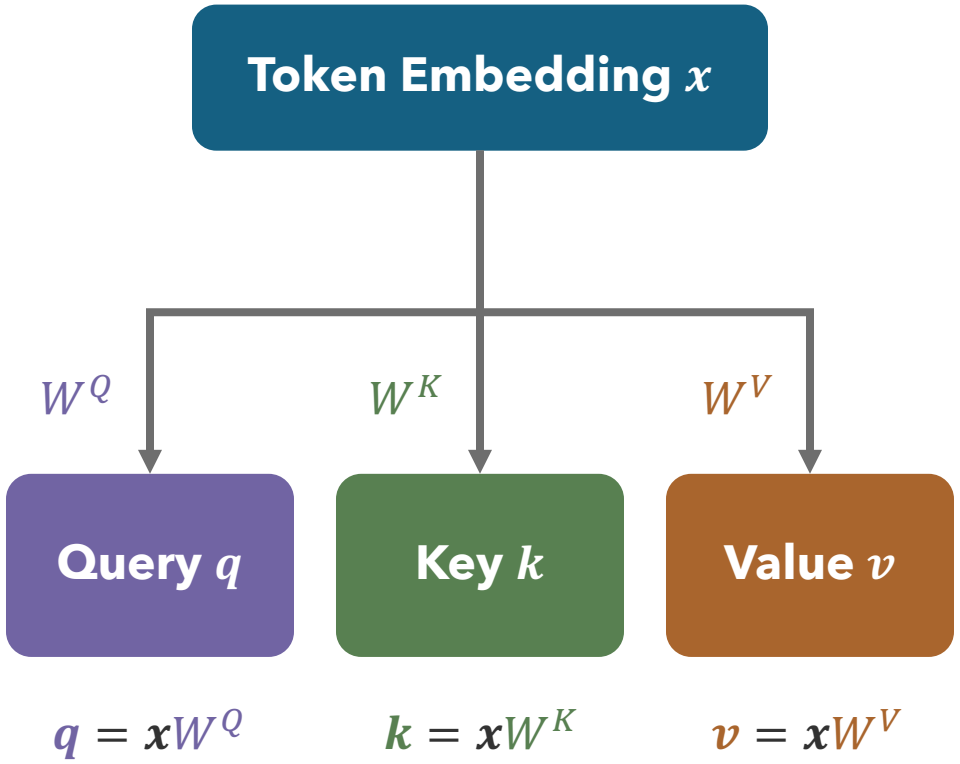
Self-Attention

DEFINITION

Self-attention is **bidirectional**, allowing each token to attend to every other token

- Learn projection matrices W^Q, W^K, W^V
- Project each token $\boldsymbol{x}$ to query $\boldsymbol{q}$, key $\boldsymbol{k}$, and value $\boldsymbol{v}$
- Encoding of X captures full context

	The	cat	sat	on	the	mat
The	1.00	0.10	0.05	0.05	0.00	0.10
cat	0.10	1.00	0.40	0.10	0.05	0.30
sat	0.05	0.40	1.00	0.15	0.05	0.35
on	0.05	0.10	0.15	1.00	0.40	0.50
the	0.00	0.05	0.05	0.40	1.00	0.30
mat	0.10	0.30	0.35	0.50	0.30	1.00



Masked Attention

DEFINITION

Problem: When predicting tokens, we don't want the network to attend to *future* tokens

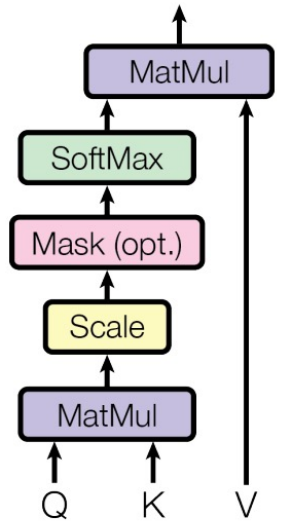
Solution: Mask out future tokens

$$\text{Attention}_M(Q, K, V) = \text{softmax}\left(\frac{QK^\top + M}{\sqrt{d_k}}\right) \cdot V$$

- $M = \begin{cases} 0, & j \le i \\ -\infty, & j > i \end{cases}$

j

i	The	cat	sat	on	the	mat
The	1.00	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
cat	0.10	1.00	$-\infty$	$-\infty$	$-\infty$	$-\infty$
sat	0.05	0.40	1.00	$-\infty$	$-\infty$	$-\infty$
on	0.05	0.10	0.15	1.00	$-\infty$	$-\infty$
the	0.00	0.05	0.05	0.40	1.00	$-\infty$
mat	0.10	0.30	0.35	0.50	0.30	1.00



Vaswani et al. (2017)

Multi-Head Attention

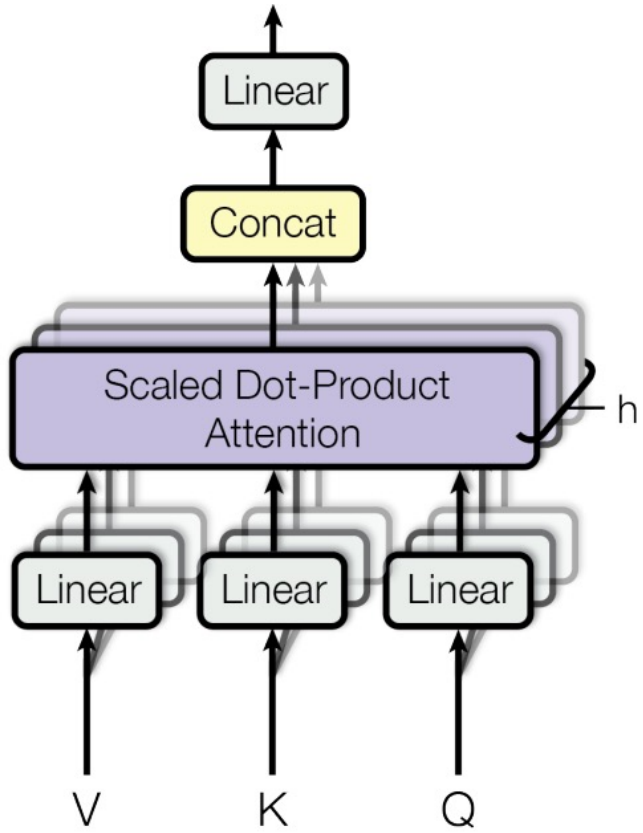
DEFINITION

Train multiple attention blocks in parallel to learn different relationships

$$\text{head}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right)$$

$$\text{MultiHead}(Q, K, V) = \text{concat}(\text{head}_1, \ldots, \text{head}_h)W^O$$

- Attention heads learn to specialize in specific relationship types
  - e.g. subject-verb, sentence clauses, topics, etc.



Vaswani et al. (2017)

QUESTIONS

Questions?

live and on sli.do #cse473

Transformers

FOUNDATIONS

The Transformer model uses an **encoder-decoder architecture**, useful when input and output sequences have different lengths

- Encoder
  - **Multi-head self-attention** to learn input context representation
- Decoder
  - Autoregressive model – uses previous output as input
  - **Masked multi-head attention** to generate output
  - **Multi-head cross-attention** combines encoder (keys and values) with decoder (query)

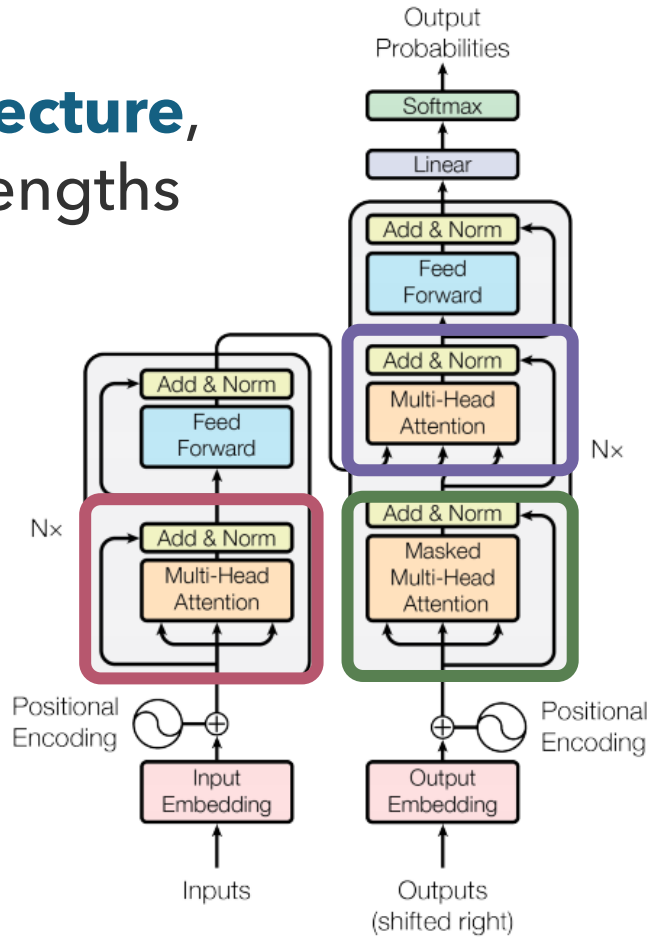


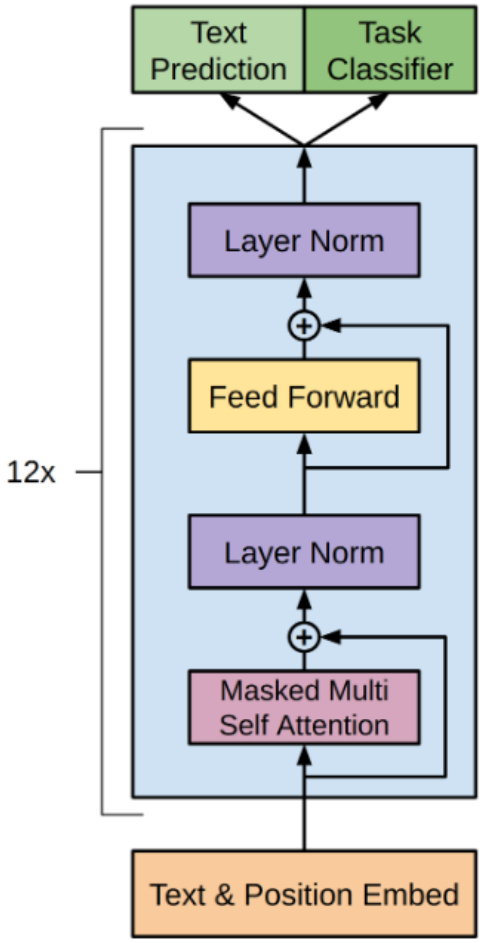
Figure 1: The Transformer - model architecture.

Vaswani et al. (2017)

GPT

EXAMPLE

Decoder-only model suffices for text generation



Radford et al. (2018)

GPT-3
(Generative Pre-trained Transformer)

- $n_{\text{params}} = 175B$
- $n_{\text{layers}} = 96$
- $n_{\text{heads}} = 96$
- $d_{\text{embedding}} = 12{,}288$
- $d_{\text{head}} = 128$
- $|V| \approx 50{,}000$

Brown et al. (2020)

"[My first day as a professor, I was walking across Red Square] in Moscow when I realized I was about to enter a world I had spent years preparing for—and absolutely no amount of preparation could tell me what would happen next."

ChatGPT…

QUESTIONS

Questions?

live and on sli.do #cse473

Bigram Model?

that's it for today!

SUMMARY

- Language is complex, requiring flexible 'focus'
- N-gram models encode conditional distributions over short subsequences
- The attention mechanism solves the context problem by weighting prior tokens according to their relevance

UPCOMING

- Large Language Models
- The Frontier of AI

REMINDERS

- Complete **Practice Problem 20**
- Do **Homework 4** (due 8.20)