

# Understanding Semantics and Context in Learned Word Representations

EECS 273

Akshay Bhatia

MS EECS @ UC Merced  
[akshaybhatia10.github.io](https://akshaybhatia10.github.io)

# Table of contents

- Introduction
- Models
  - Leabra Hebbian Word Semantics Model
  - Co-occurrence Matrix
  - Bag of Words
  - Word Embeddings - word2vec
  - Transformer Language Models - BERT
- Experiments and Results
- Conclusion

# Introduction

# Introduction

“After stealing money from the **bank vault**, the **bank robber** was seen fishing on the Mississippi **river bank**.”

*“You shall know a word by the company it keeps”* (J. R. Firth 1957: 11)

# Goal

**What is meaning of a word?** → The idea that a person wants to express

**Problem:** How to represent that idea?

**Solution:** Linguistic way of thinking about meaning

signifier (symbol)  $\Leftrightarrow$  signified (idea or thing)

= denotational semantics

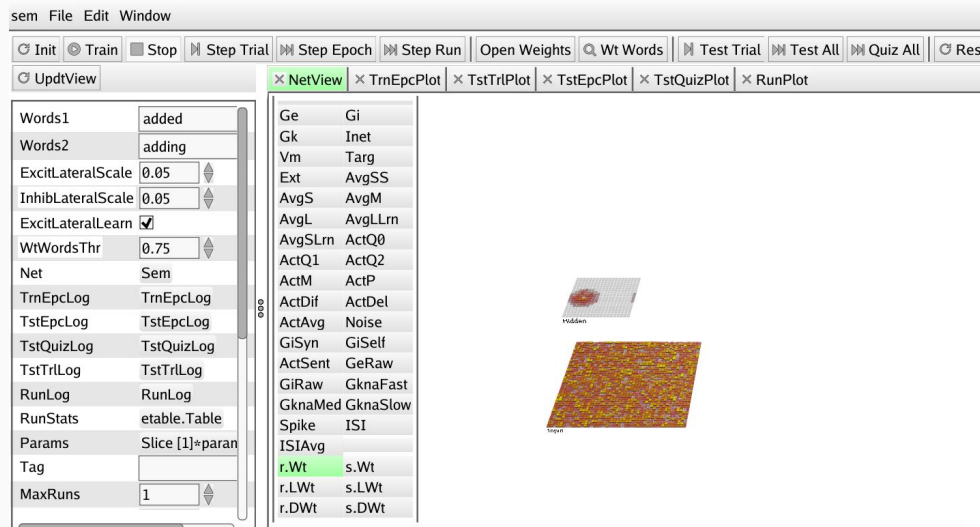
tree  $\Leftrightarrow$  {, , , ...}

# Models

- Leabra Hebbian Word Semantics Model
- Co-occurrence Matrix
- Bag of Words
- Word Embeddings - word2vec
- Transformer Language Models - BERT

# Leabra Hebbian Word Semantics Model

- Understand word meaning by company it keeps - statistics of word co-occurrence
- Uses BCM-style learning mechanism
- Includes recurrent lateral excitatory and inhibitory connections
- As in the '*sem*' simulation Ch. 9



# Co-occurrence Matrix

“After stealing money from the **bank vault**, the **bank robber** was seen fishing on the Mississippi river bank.”

- Start with a  $n \times n$  matrix where  $n$  is vocab size
- Record co-occurrence for each word in a specific window size, say 5
- No learning required

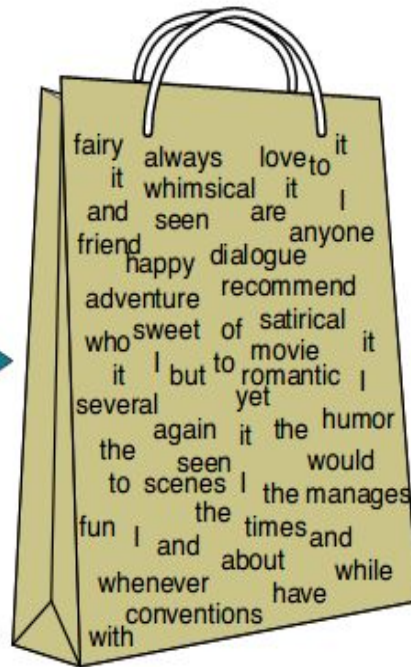
	after	bank	fishing	from	mississippi	money	on	river	robber	seen	stealing	the	vault	was
after	0	1	0	1	0	1	0	0	0	0	1	1	0	0
bank	1	1	2	2	1	1	2	1	2	1	1	5	2	2
fishing	0	2	0	0	1	0	1	1	1	1	0	2	0	1
from	1	2	0	0	0	1	0	0	0	0	1	2	1	0
mississippi	0	1	1	0	0	0	1	1	0	1	0	1	0	1
money	1	1	0	1	0	0	0	0	0	0	1	2	1	0
on	0	2	1	0	1	0	0	1	1	1	0	1	0	1
river	0	1	1	0	1	0	1	0	0	1	0	1	0	0
robber	0	2	1	0	0	0	1	0	0	1	0	3	1	1
seen	0	1	1	0	1	0	1	1	1	0	0	2	1	1
stealing	1	1	0	1	0	1	0	0	0	0	0	1	1	0
the	1	5	2	2	1	2	1	1	3	2	1	1	2	2
vault	0	2	0	1	0	1	0	0	1	1	1	2	0	1
was	0	2	1	0	1	0	1	0	1	1	0	2	1	0



# Bag of Words

- Represent a sentence as bag(multiset) of its words
- Disregards grammar and word order while keeping multiplicity
- No learning required

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



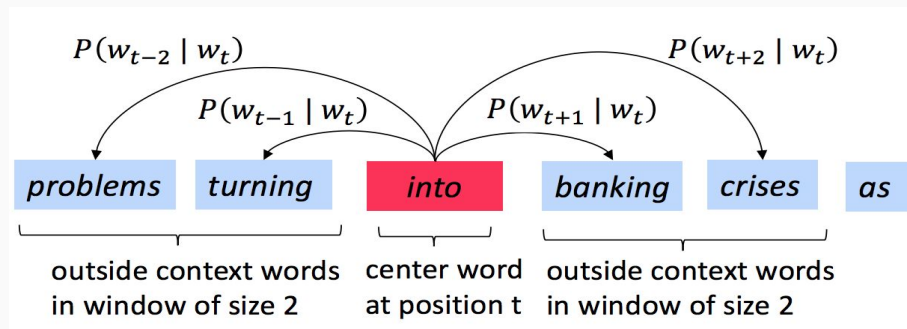
it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
...	...

## Word Embeddings - word2vec

- Represent each word in some  $n$ -dimensional space
- **Idea: A word's meaning is given by the words that frequently appear close-by**
- Trained on large text corpus from co occurrence statistics

Football

0.2 1.2 0.4 2.9 1.8 0.3



Source: CS224n Stanford

### Problems solved:

- Encode general semantic relationship
- Incorporate similarity between words
- Fixed vector size for every word (50, 100, 300) instead of vocab size
- Word order remains intact

# Transformer Language Models - BERT

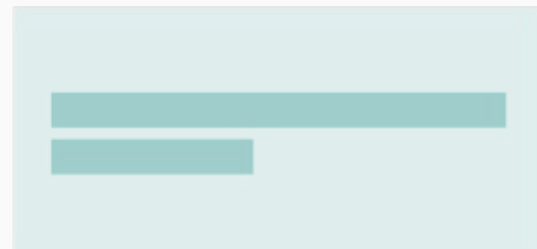
Text input



Language  
Model



Text output

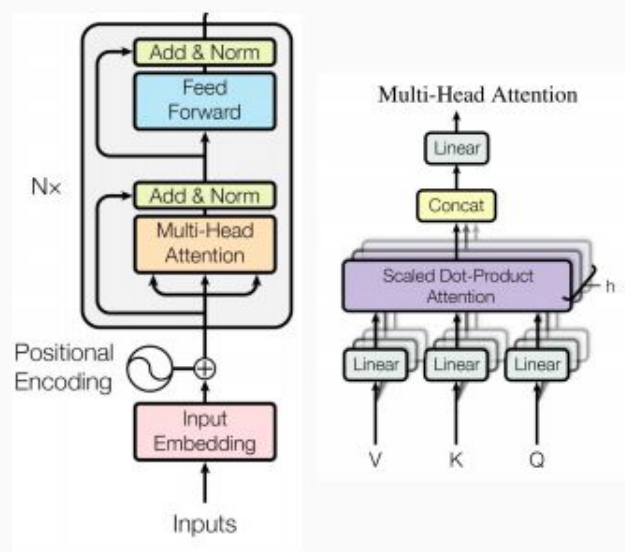


Numeric representation of text  
useful for other systems

# Transformer Language Models - BERT

- BERT(Bidirectional Encoder Representations from Transformers) by Devlin et. al 2018
- Large Neural Network with billions of parameters
- Trains 2 *auxiliary* tasks:
  - Masked Language Model
  - Next sentence prediction
- Outperforms all other models on many NLP benchmarks

Transformer Architecture



Masked Language Modelling Task

the man went to the [MASK] to buy a [MASK] of milk

store                      gallon

Next Sentence Prediction Task

**Sentence A** = The man went to the store.  
**Sentence B** = He bought a gallon of milk.  
**Label** = IsNextSentence

**Sentence A** = The man went to the store.  
**Sentence B** = Penguins are flightless.  
**Label** = NotNextSentence

# Experiments and Results

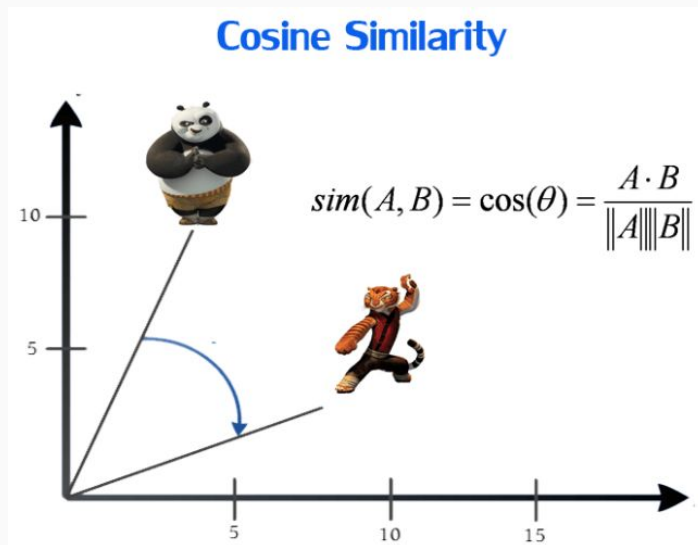
# Experiments

Hebbian Learning

v/s

Co-Occurrence  
BoW  
word2vec  
BERT

Added vs Adding



Dyslexia vs reading  
problem

## Results - Same Root(Lemma) Words

Word Pair	Leabra	Co-Occurrence	BoW	Word Emb	Word Emb(Gen)	BERT
Added vs Adding	0.0350	<b>0.3185</b>	0.0	-0.1311	0.7415	<b>0.9247</b>
Summary vs Summarize	-0.0199	<b>0.0435</b>	0.0	-0.0611	0.5143	<b>0.5922</b>
Introducing vs Introduction	<b>-0.0194</b>	0.0	0.0	-0.0607	0.6485	<b>0.7805</b>
Continuous vs Continuum	-0.0086	<b>0.0757</b>	0.0	-0.0309	0.2267	<b>0.6488</b>
Receptive vs Receptors	-0.027	0.0370	0.0	<b>0.0727</b>	0.0700	<b>0.5904</b>

## Results - General Topics

Word Pair	Leabra	Co-Occurrence	BoW	Word Emb	Word Emb(Gen)	BERT
Dyslexia vs Reading problem	0.128	<b>0.1412</b>	0.0216	0.0038	0.3448	<b>0.5202</b>
Dyslexia vs Speech Problem	<b>0.109</b>	0.0954	0.0114	-0.0610	0.2674	<b>0.6410</b>
Dyslexia vs Speaking Problem	-0.040	<b>0.0597</b>	0.0122	-0.0736	0.2450	<b>0.3625</b>
Dyslexia vs Reading	<b>0.561</b>	0.2367	0.0217	0.0510	0.2974	0.3129
Dyslexia vs Speech	<b>0.479</b>	0.1301	0.0	-0.0382	0.1965	0.3852
Dyslexia vs Speaking	-0.021	<b>0.0279</b>	0.0	-0.0627	0.1710	<b>0.2603</b>



# Results - Multiple-Choice Quiz

Based on your knowledge of the textbook, which of the options following each "question" provides the best match to the meaning?

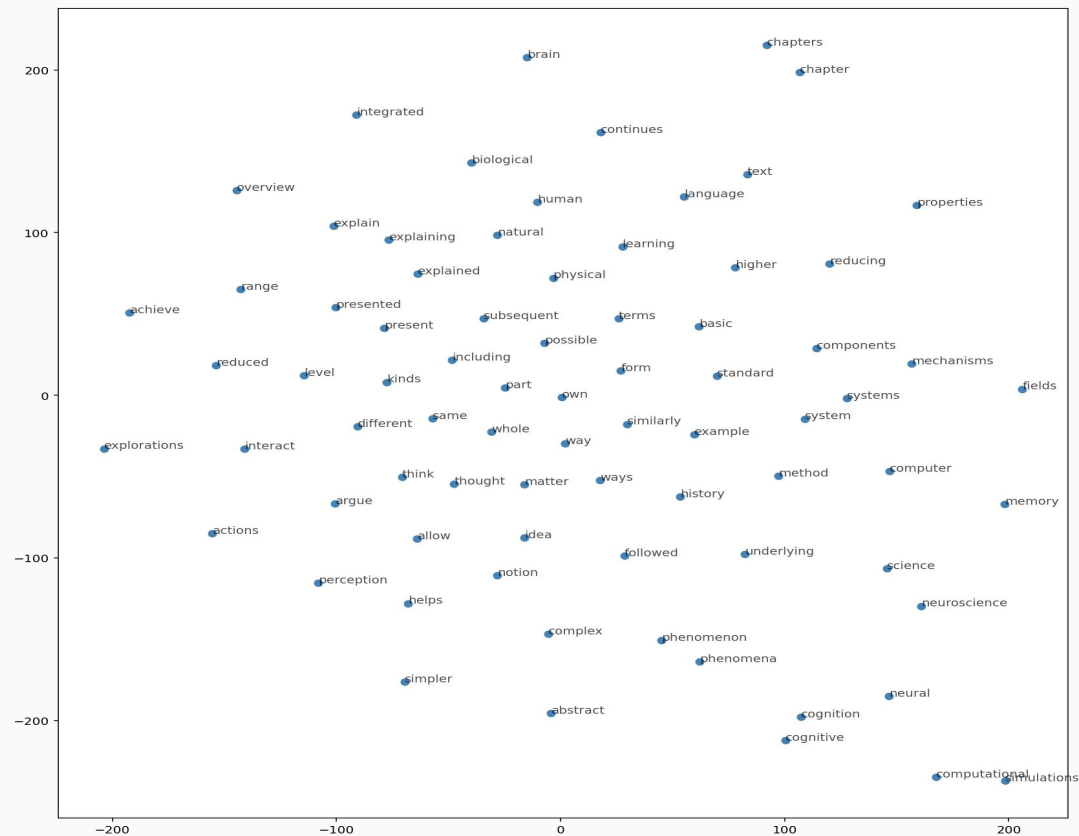
## Dyslexia

- A. surface deep phonological reading problem damage
- B. speech output hearing language nonwords
- C. competition inhibition selection binding

Model	Accuracy
Leabra	<b>0.8</b>
Co-occurrence Matrix	0.6
BoW	<b>0.8</b>
Word Embedding	0.5
Word Embedding(Gen)	0.3
BERT	0.5

# Conclusion

## Conclusion - Model Explainability



## Visualization of word embeddings in 2D space



Attention(Hidden Layer) weights of BERT for sentiment analysis

# Code

The screenshot shows the GitHub interface for the repository 'akshaybhatia10 / EECS273-Term-Project'. The repository is public and has 0 stars, 0 forks, and 1 watcher. The main branch is 'main' with 1 branch and 0 tags. The repository contains a file tree with the following files and folders:

File/Folder	Description	Last Commit
checkpoints	word2vec custom trained checkpoints	7 minutes ago
EECS 273 - Models.ipynb	Models notebook	now
README.md	Initial commit	41 minutes ago
ccnbook_ed4.txt	CCN text file	38 minutes ago
utils.py	Add utils.py	38 minutes ago

The README.md file is displayed below the file tree, showing the title 'EECS273-Term-Project' and the description 'Term project for EECS 273 UC Merced - Understanding Semantics and Context in Learned Word Representations'.

On the right side of the repository page, there are sections for 'About', 'Releases', and 'Packages'. The 'About' section contains the repository description and statistics. The 'Releases' section shows that no releases have been published, and the 'Packages' section shows that no packages have been published.

<https://github.com/akshaybhatia10/EECS273-Term-Project>

# Thank You!

Questions?

Akshay Bhatia

**Email:** [abhatia8@ucmerced.edu](mailto:abhatia8@ucmerced.edu)

**Website:** [akshaybhatia10.github.io](https://akshaybhatia10.github.io)

