Word embeddings (I)

CS 485, Spring 2024
Applications of Natural Language Processing

Brendan O'Connor
College of Information and Computer Sciences
University of Massachusetts Amherst

[Slides from Laure Thompson]

- Proposal feedback should be accessible. Please meet your project mentor!
- Proposal revisions: due next week
- HW3 to be released tomorrow; will be due approx Monday 4/8

Word embeddings

- Today
    - 1. Question: how can we generally represent word meanings?
    - 2. Approach: train a language model with ***word embeddings*** to discover latent meanings of words!
        - ... which exploit the ***distributional hypothesis***

- Key idea: automatically discover aspects of language meaning, from raw textual corpora
    - Today / next week: word embeddings
    - Next: neural network language models & other hijinks

What is "asdfasdf"?

“ **<u>asdfasdf</u>**, Most Neglected American Fruit.” — NYTimes <u>1922</u>

“ **<u>asdfasdf</u>** Recommended by U.S. Food Experts, Along With Persimmon, as War Nutrition” — NYTimes <u>1942</u>

“ The **<u>asdfasdf</u>** is also pollinated by flies and other insects rather than by honeybees…”— NYTimes <u>2020</u>

“Many people also cook with ripe **<u>asdfasdf</u>**, making bread, beer, ice cream, or this **<u>asdfasdf</u>** pudding…” — NYTimes <u>2020</u>

What is a *pawpaw* ?

I. Look it up in a dictionary

https://www.merriam- webster.com/

https://www.oed.com/

https://en.wiktionary.org/

pawpaw noun

Save Word

paw·paw

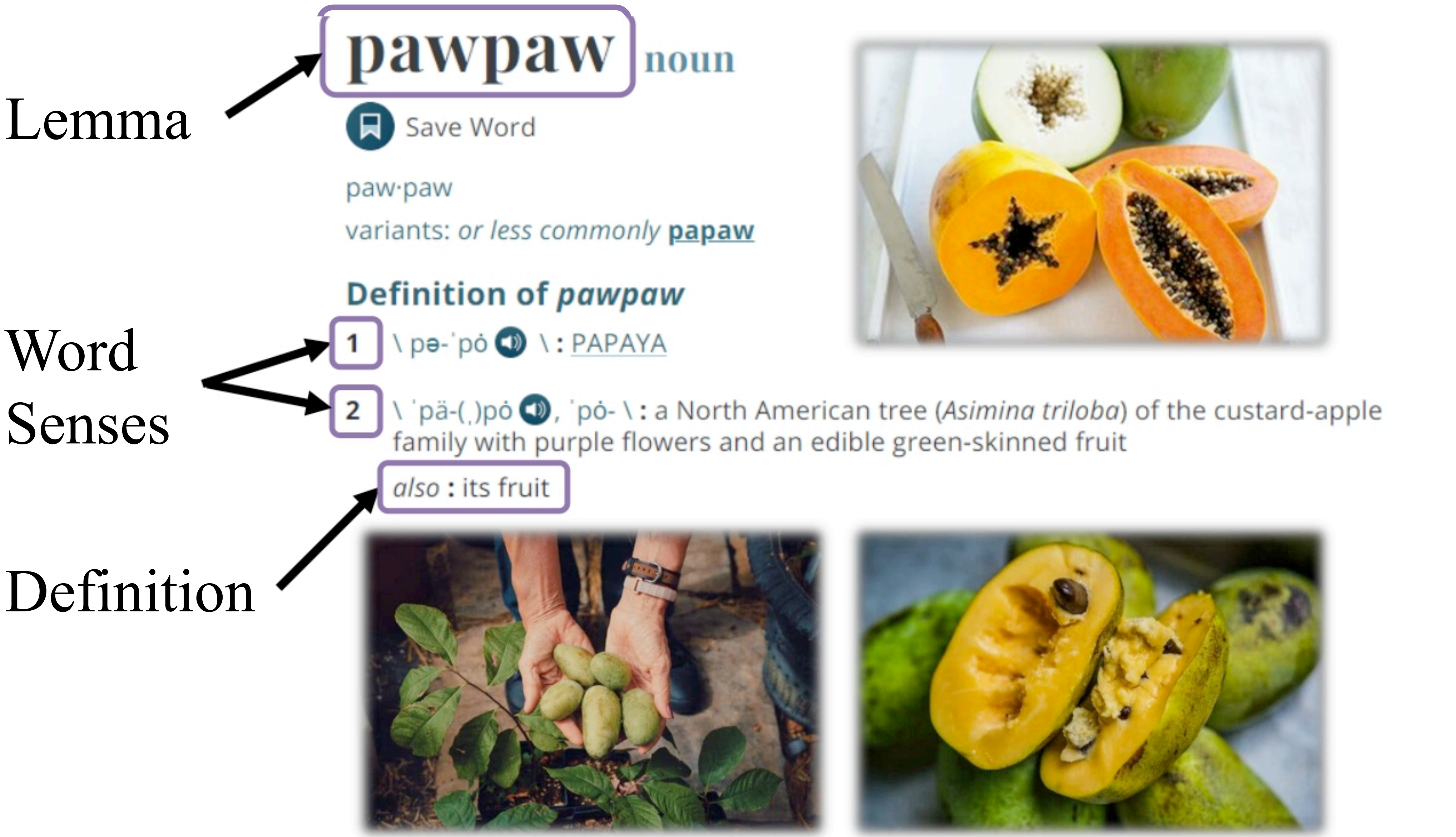
variants: *or less commonly* **papaw**

Definition of *pawpaw*

1 \ pə-ˈpȯ \ **:** PAPAYA

2 \ ˈpä-(ˌ)pȯ, ˈpȯ- \ **:** a North American tree (*Asimina triloba*) of the custard-apple family with purple flowers and an edible green-skinned fruit

also **:** its fruit



Lemma
pawpaw noun
Save Word
paw·paw
variants: or less commonly papaw
Definition of pawpaw
Word Senses
1 \ pə-ˈpȯ \ : PAPAYA
2 \ ˈpä-(ˌ)pȯ, ˈpȯ- \ : a North American tree (Asimina triloba) of the custard-apple family with purple flowers and an edible green-skinned fruit
also : its fruit
Definition

II. Look it at how its used

“ <u>**Pawpaw**</u>, Most Neglected American Fruit.” — NYTimes <u>1922</u>

“ <u>**Pawpaw**</u> Recommended by U.S. Food Experts, Along With Persimmon, as War Nutrition” — NYTimes <u>1942</u>

“ The <u>**pawpaw**</u> is also pollinated by flies and other insects rather than by honeybees…”— NYTimes <u>2020</u>

“Many people also cook with ripe <u>**pawpaws**</u>, making bread, beer, ice cream, or this <u>**pawpaw**</u> pudding…” — NYTimes <u>2020</u>

II. Look it at how its used

“ *Pawpaw*, Most Neglected **American Fruit** .” — NYTimes 1922

“ *Pawpaw* Recommended by U.S. Food Experts, Along With **Persimmon** , as War **Nutrition** ” — NYTimes 1942

“ The *pawpaw* is also **pollinated** by **flies** and other insects rather than by honeybees…”— NYTimes 2020

“Many people also **cook** with **ripe** *pawpaws* , making **bread** , **beer**, **ice cream** , or this *pawpaw* **pudding** …” — NYTimes 2020

Word Relations

Synonyms

· couch / sofa

· oculist / eye - doctor

· car / automobile

· water / H_2O

· draft / draught

Antonyms

· yes / no

· dark / light

· hot / cold

· up / down

· clip / clip

Word Relations

Similarity

· cat / dog

· cardiologist / pulmonologist

· car / bus

· sheep / goat

· glass / mug

Relatedness

· coffee / cup

· waiter / menu

· farm / cow

· house / roof

· theater / actor

Quantifying Similarity

Ask humans how ***similar*** two words are on a scale of 1 - 10

Word 1	Word 2	SimLex - 999
area	region	9.47
horse	mare	8.33
water	ice	6.7
hill	cliff	4.28
absence	presence	0.4
princess	island	0.3

Hill et al. 2015

…but what about computers?

Word Embeddings

Represent each word type as a **vector**

On Vectors:

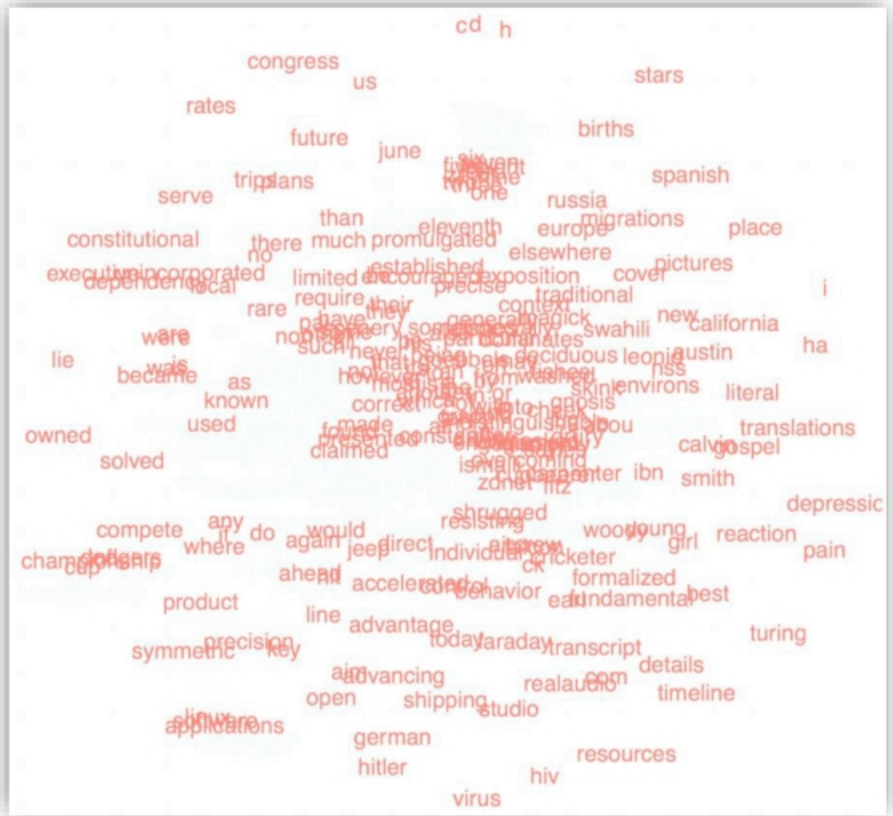
· A **vector** is a list of numbers

· A **vector** can also be considered a **point** in a *k*- dimensional space

Capturing Word Similarity

Operationalize word similarity by computationally **comparing** vectors

Distance reflects semantic relationships



Closer vectors represen t more similar words

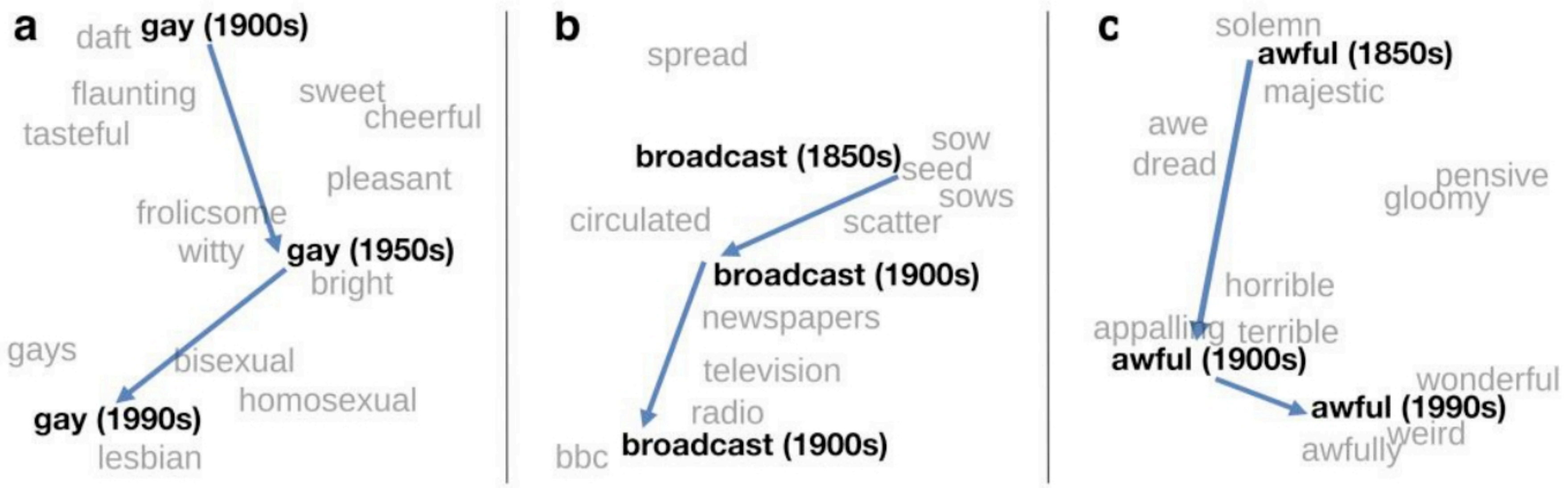
More distant vectors represent less similar words

Applications

Task-driven: e.g. use for improve text classification (next week)

... or ...

Exploratory / descriptive
Study word use over time
[Hamilton et al. 2016]



One – Hot Vectors

Each word is represented by a vector with a 1 in the word's index in the vocabulary and 0's elsewhere. (We've implicitly used these already...)

Term	Vector
i	<1, 0, 0, 0, 0, 0>
hate	<0, 1, 0, 0, 0, 0>
love	<0, 0, 1, 0, 0, 0>
the	<0, 0, 0, 1, 0, 0>
movie	<0, 0, 0, 0, 1, 0>
film	<0, 0, 0, 0, 0, 1>

Q: What are some issues with these representations?

Learning word vectors

- Let's learn learn a word vector ("word embedding") for each word type in the vocabulary
- Goal: general-purpose representation applicable to a wide variety of tasks

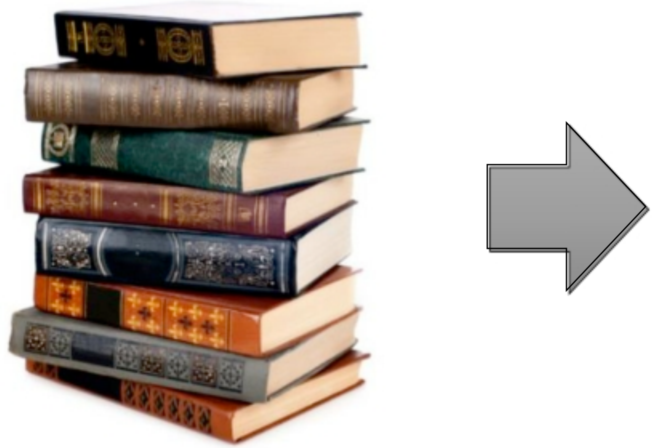
Distributional Semantics

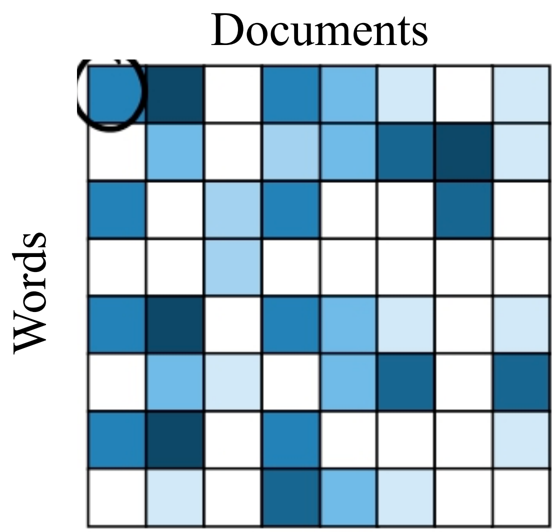
"You shall know a word by the company it keeps!" — Firth (1957)

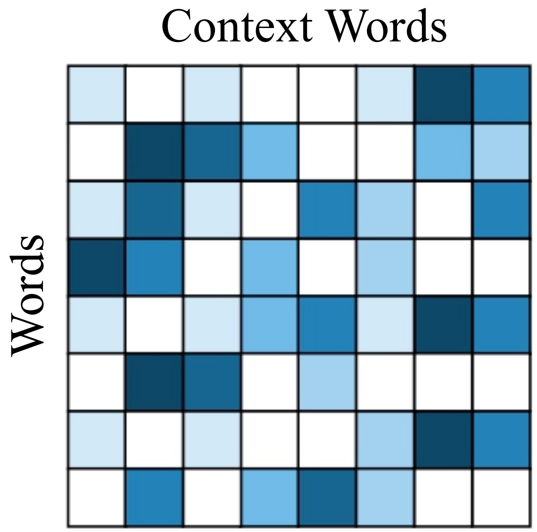
Intuitions: Harris (1954)

"If A and B have almost identical environments except chiefly sentences which contain both, we say they are synonyms: *oculist* and *eye- doctor* ."

Build vectors based on context

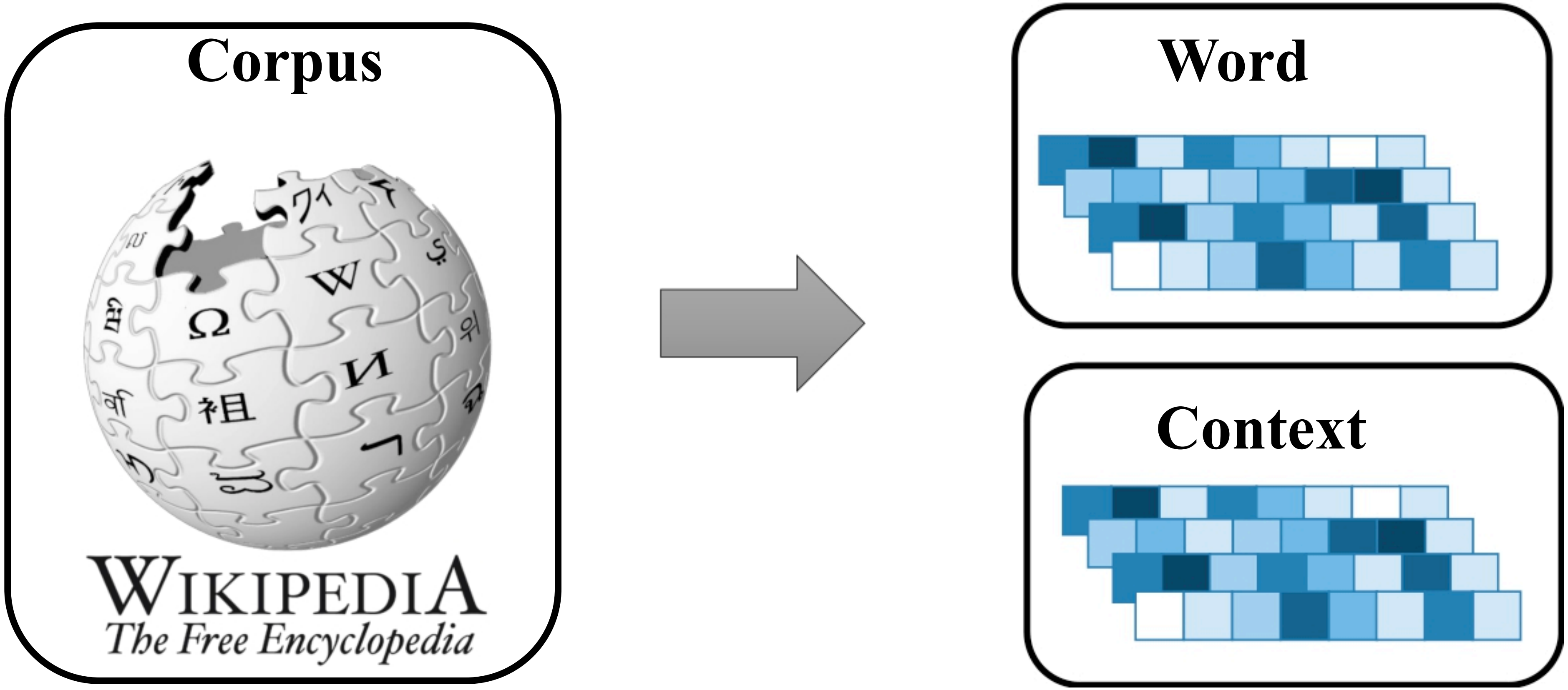




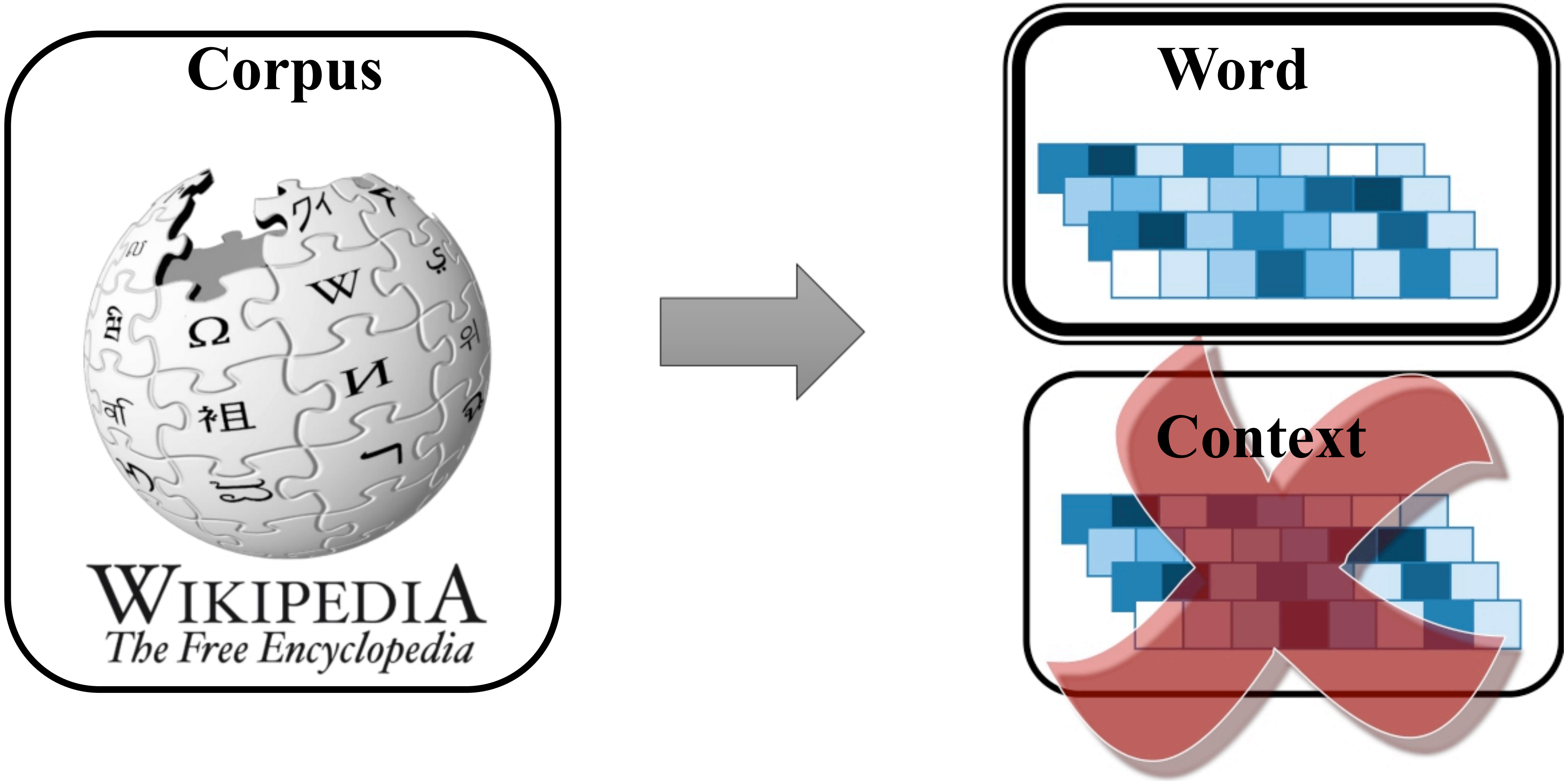


Q: What are some issues with these representations?

Neural Word Embeddings

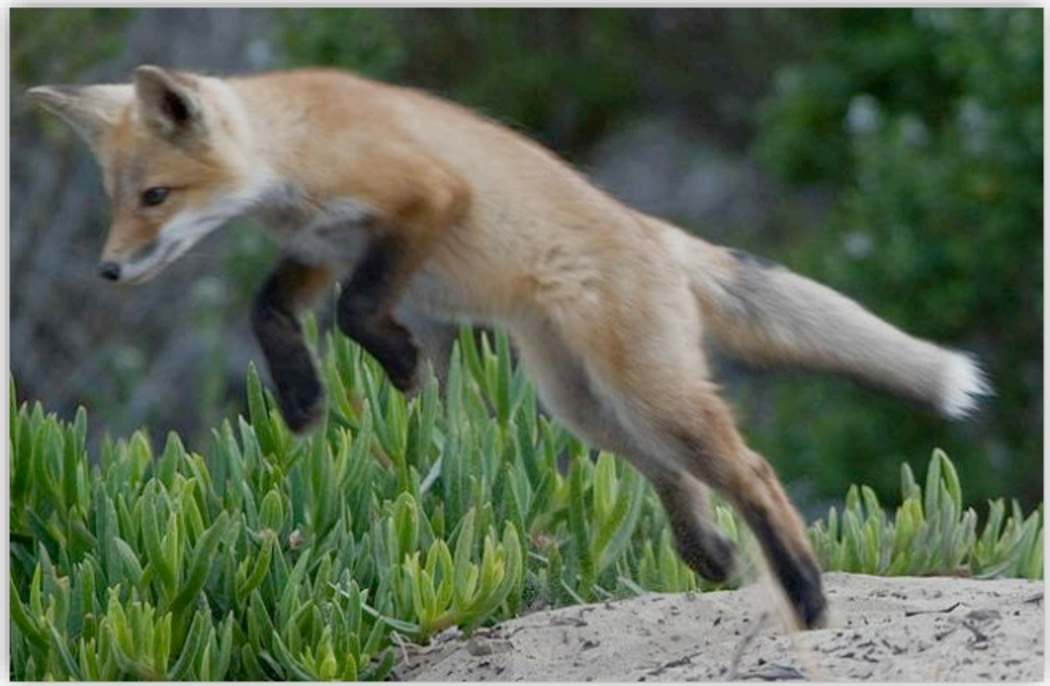


Neural Word Embeddings



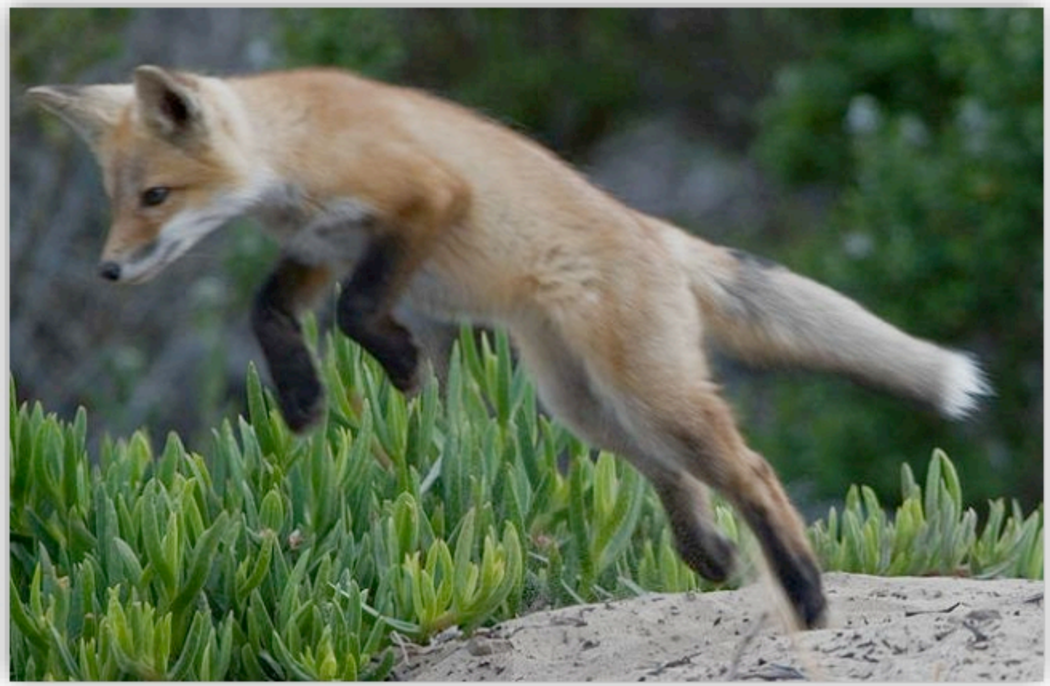
Skip- Gram with Negative Sampling (SGNS)

The brown fox jumps over the lazy dog.



<u>**SG**</u> NS: Skip- Gram Model

The brown fox **jumps** over the lazy dog.



<u>SG</u> NS: Skip- Gram Model

The brown fox **jumps** over the lazy dog.

Context Window Size = 2

<u>**SG**</u> NS: Skip- Gram Model

The **brown fox jumps over the** lazy dog.

Context Window Size = 2

jumps → { brown, fox, over, the }

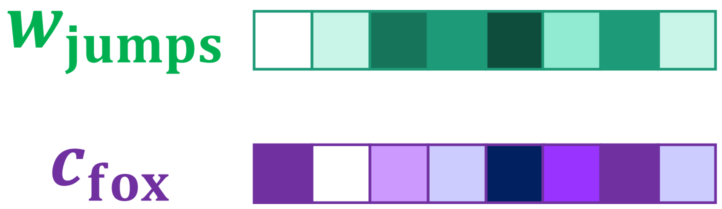
Simple idea: from a word, predict its context words! (A funny type of language model.)
Learn a vector that's good at that. Similar words should get similar vectors.

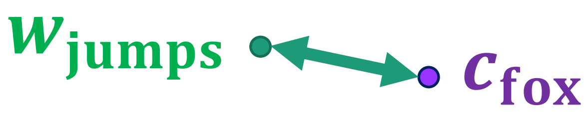
Key idea: use unlabeled text as *implicitly supervised data*

- A word *s* near *apricot*
  - Acts as gold 'correct answer' to the question
  - "Is word *w* likely to show up near *apricot*?"
- No need for hand-labeled supervision
- The idea comes from **neural language modeling**
  - Bengio et al. (2003)
  - Collobert et al. (2011)

SG<u>NS</u> : Negative Sampling

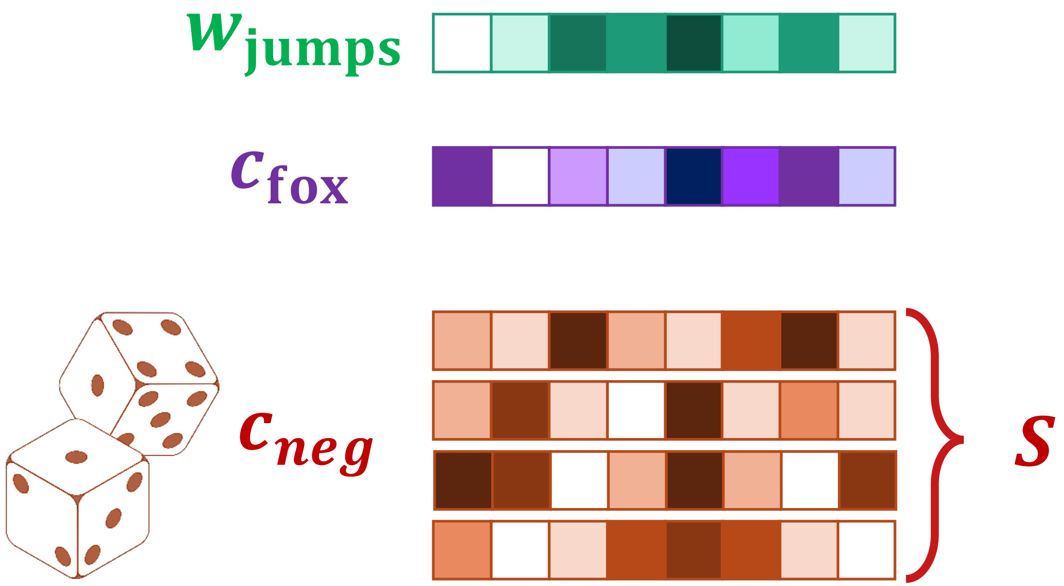
Co-occurrence **jumps** , **fox**:

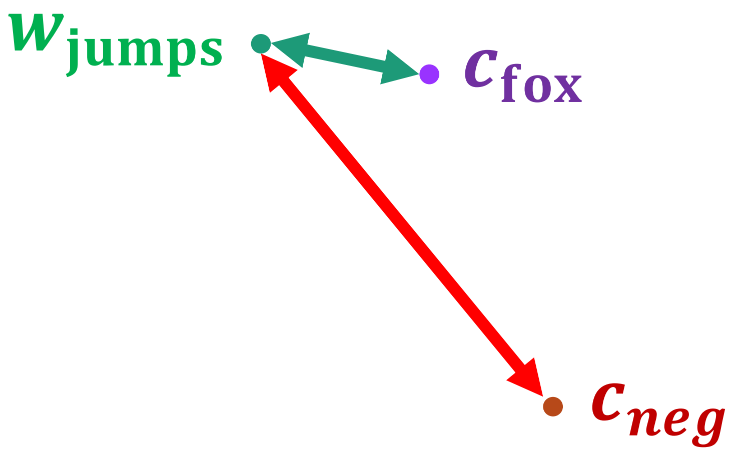




SGNS : Negative Sampling

Co-occurrence **jumps** , **fox**:





Modeling goal

- Given a (target, context) tuple
  - [+] (apricot, jam)
  - [–] (apricot, aardvark)
- Want binary probability
  - P(c | t) for a real context [+])
  - 1-P(c | t) for a “fake”, unseen context [–])
- Let u_t and v_c be their vectors.
- $P(c \mid t) = \sigma(u_t' v_c)$: logistic in their *affinity/ similarity*

How do we compare vectors?

- Similarity measurements
  - Larger values → similar vectors → similar words
  - Smaller values → dissimilar vectors → dissimilar words

- Distance / dissimilarity measurements
  - Note: distance metric requires triangle inequality
  - Larger values → dissimilar vectors → dissimilar words
  - Smaller values → similar vectors → similar words

Euclidean Distance

$$d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$$

Issue: Vector length depends on frequency. More frequent words will have longer vectors.

Cosine Similarity

$$s(x, y) = \frac{x \cdot y}{|x||y|}$$

Only depends on vector angle

Range:

Non- negative vectors & cosine similarity

If all vectors have non - negative values, then their cosine similarity will be between 0 and 1