Language and General Structure Discovery

Unstructured Data

The University of Western Ontario

Language Structure

- What does it mean to recover structure?
- Word Embeddings
- Neural Networks
  - Autoencoders
  - General networks
- Thinking about General Representations

Creating Representations to Recover Relational Structure

Word Representations

- Map each word to a vector
- Relational structure captured by vector similarity (cosine, Euclidean distance, whatever.)
- One “relational structure” may be “similar meaning”
  - Does the word “boat” mean something similar to the word “ship?” If so, representations should be close.
  - Representation for “boat” and representation for “mountaineering” should probably be far apart.

Word Representation 1
One-hot Encoding

- Word vectors of length *m* (size of vocabulary)
- For word *i*, vector is all 0s except at location *i*, where it is 1.
- Easy to define, but pretty useless.
- Dot product between vectors for any 2 different words is 0.

first	1	0	0	0	0	0	0	0	…
hurlyburly	0	1	0	0	0	0	0	0	…
in	0	0	1	0	0	0	0	0	…
thunder	0	0	0	1	0	0	0	0	…
witch	0	0	0	0	1	0	0	0	…
witchcraft	0	0	0	0	0	1	0	0	…
witches	0	0	0	0	0	0	1	0	…
…	…	…	…	…	…	…	…	…	…

Word Representation 2
Term-Document Matrix

- Word vectors of length *n* (size of corpus)
- For word *i*, vector is the number of occurrences of the word in each document.
- Dot product between vectors for different words is positive only if they occur in the same document(s)

	D1	**D4**	**D5**	**D8**	**D9**	**D22**	**D37**	**…**
first	2	0	0	0	0	3	1	…
hurlyburly	0	0	0	0	1	1	0	…
in	2	0	0	0	0	2	1	…
thunder	1	0	1	0	0	2	1	…
witch	2	0	0	0	0	4	1	…
witchcraft	2	0	1	0	0	0	1	…
witches	0	2	0	0	0	2	0	…
witching	0	0	0	1	0	0	0	…
…	…	…	…	…	…	…	…	…

Word Representation 3: rows of U from LSA

- Word vectors of length *p*
- Each entry *j* of the vector corresponds to how much that word occurs in topic *j*.
- Words can be similar to each other even if they never occurred in the same document, as long as they show up in the same topics.

	T1	T2	T3
first	-0.5037	0.0484	0.1659
hurlyburly	-0.1121	-0.1912	0.2517
in	-0.3936	0.2149	0.007
thunder	-0.3349	0.0938	-0.1469
witch	-0.6138	-0.118	0.3248
witchcraft	-0.183	0.6434	-0.5782
witches	-0.2375	-0.6916	-0.6691
witching	0	0	0
…	…	…	…

The Meaning of Representations

- You often hear folks assert that a representation puts word vectors “nearby” if their words have similar “meaning.” (Often “meaning” is not defined.)
- This is not magic. Representations come from data.
  - Representation from TDM: Nearby if in same document
  - Representation from U: Nearby if in same topic
- We will explore some other definitions that result in different representations.

“Word Embeddings”

- Word representations where similarity (cosine) is high between words if they are used similarly

- “Used Similarly” is captured by “used nearby” in text.

- Corpus is *not divided into documents.*

Word Representations: Topic Models vs. Word Embeddings

- Topic Models
  - Similarity: Occur within same documents
  - Capture information about documents in the corpus
  - Representations supposed to be useful within the given corpus
  - Can work even on small-ish datasets
- Word Embeddings
  - Similarity: Occur nearby similar words within sentences
  - Capture information about language use in general
  - Representations supposed to be useful more generally
  - Work best when learned from very large datasets

(Simple) Word Embedding Uses

- Building representations of sentences/paragraphs/documents by summing the vectors of their words

- Used as input representations for sequences of words going into neural networks

- Building up more complex embeddings that take context into account

GloVE: Global Vectors for Word Representation

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

- Uses *co-occurrence matrix* $X_{m \times m}$

- Element $x_{i,j}$ tells how many times word *i* appears "near" (say within 5 words) of word *j*
  - Symmetric: $x_{i,j} = x_{j,i}$
  - Sparse

Co-occurrence Matrix

	pig	cow	fish	animal	apple	pear	tomato	fruit
pig	2	1	0	8	0	0	0	0
cow	1	13	0	1	0	0	0	0
fish	0	0	102	9	1	0	1	5
animal	8	1	9	72	0	0	0	2
apple	0	0	1	0	124	6	0	3
pear	0	0	0	0	6	2	0	2
tomato	0	0	1	0	0	0	0	1
fruit	0	0	5	2	3	2	1	27

Factored Approximate Co-occurrence Matrix

Same matrix, just transposed!

	pig	cow	fish	animal	apple	pear	tomato	fruit
F1	0.000	0.00	0.028	0.002	0.998	0.048	0.000	0.027
F2	-0.014	-0.002	-0.991	-0.122	0.03	0.001	-0.010	-0.054

Word reps

	F1	F2
pig	0.000	-0.014
cow	0.000	-0.002
fish	0.028	-0.991
animal	0.002	-0.122
apple	0.998	0.030
pear	0.048	0.001
tomato	0.000	-0.010
fruit	0.027	-0.054

Actual Factorization used by GloVe

- $\log(1 + X)_{m \times m} \approx W_{m \times p}\, W^{T}{}_{p \times m} + \mathbf{b}_{m \times 1}\mathbf{1}^{T}{}_{1 \times m} + \mathbf{1}^{T}{}_{m \times 1}\mathbf{b}_{1 \times m}$
- $\log(1 + X)_{i,j} \approx \mathbf{w}_i^{T}\mathbf{w}_j + b_i + b_j$
- *b* is for "*bias*" – accounts for some terms being overall more common than other terms
- There is no "closed form" solution to find W and **b**
  - (Can't use SVD.)
- Rows of W are the word representations

Demo