Lecture 18:
NATURAL LANGUAGE REPRESENTATION

COURSE: BIOMEDICAL DATA SCIENCE

Parisa Rashidi

FALL 2019

Credit

- Slides are partially based on
  - CS276: Information Retrieval and Web Search Pandu Nayak and Prabhakar Raghavan
  - Natural Language Processing for Precision Medicine, Hoifung Poon, Chris Quirk, Kristina Toutanova, Scott Wen-tau Yih
  - Word2vec: From theory to practice. Hendrik Heuer. Stockholm NLP Meetup
  - From Word Embeddings to Sentence Meanings - Stanford NLP, Simon, Manning, 2017
  - Word2Vec and FastText Word Embedding with Gensim, Steeve Huang

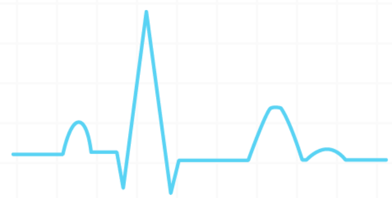
Machine Learning Input

We converted nominal/ordinal variables into numbers.

ID	WGT	HGT	Cholesterol	**Risk (Class)**
1	56	160	260	high
2	92	158	254	high
3	86	181	142	med

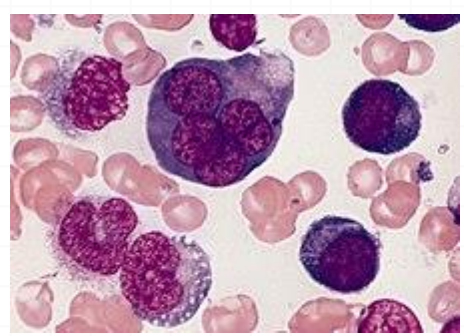
This is structured ☺

A series of numbers.



Semi-structured

Still represented as numbers (pixel intensity, RGB values).



This is unstructured.

How to represent text features??

1. **HISTORY OF PRESENT ILLNESS:** The patient is a (XX)-month-old child. Family notes that since last night, the patient was fussy and irritable, possibly pulling at her ears. No obvious pain or discomfort with urination. No chest pain, cough. No abdominal pain. No nausea, vomiting or diarrhea and is having usual stool and wet diapers. Child, however, has been much more fussy at times and now comes to the ER for further evaluation. This is the mother's first child. The child is consolable and not constantly crying in the ER and is able to be addressed and approached by me without being upset. Family notes the child has been fully immunized and is compliant with scheduled appointments.

2. **HISTORY OF PRESENT ILLNESS:** This (XX)-year-old very pleasant gentleman presents to the emergency room with a one day complaint of pain in his right ear. The patient states last evening he thought he had wax. He used a wax softener. He has had lots of drainage from his ear today. He is still having pain. He attributes it to his new hearing aid. He also is complaining of pain now in the right side of his head. There is no nausea, no vomiting, no tinnitus, visual, olfactory or auditory changes. He states the headache is behind the ear, and it is related directly to the pain in his ear. There is no chest pain, no shortness of breath, nausea, vomiting. No other complaints. He is very affable and in no apparent distress.

Unstructured!

Problem ?

HISTORY OF PRESENT ILLNESS: The patient is a (XX)-month-old child. Family notes that since last night, the patient was fussy and irritable, possibly pulling at her ears. No obvious pain or discomfort with urination. No chest pain, cough. No abdominal pain. No nausea, vomiting or diarrhea and is having usual stool and wet diapers. Child, however, has been much more fussy at times and now comes to the ER for further evaluation. This is the mother's first child. The child is consolable and not constantly crying in the ER and is able to be addressed and approached by me without being upset. Family notes the child has been fully immunized and is compliant with scheduled appointments.

[1, 2, 1, 1, 2, 0, 0, 0, 1, 1]

Text

Vectors

?

Binary Term-document matrices

- If a term is seen in a document:
  - Each document is a binary count vector
  - Issue 1: There are 1,025,109.8 words in English!
  - Issue 2: a word might 10 times, once, or 150 times. Still represented by 0/1.

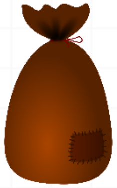
	Diabetes	Pneumonia	Joint Pain	Arthritis	Back pain
Report 1	1	1	0	0	0
Report 2	0	1	0	1	0
Report 3	1	0	0	1	1
Report 4	0	1	0	0	0
Report 5	1	0	0	0	0
Report 6	0	0	1	1	1
Report 7	0	0	1	1	1

Term-document count matrices

- Consider the number of occurrences of a term in a document:
  - Each document is a count vector

	Diabetes	Pneumonia	Joint Pain	Arthritis	Back pain
Report 1	1	2	0	0	0
Report 2	0	2	0	1	0
Report 3	1	0	0	2	1
Report 4	0	3	0	0	0
Report 5	2	0	0	0	0
Report 6	0	0	3	5	5
Report 7	0	0	1	1	1

Bag of words model (BOW)



- Represent text as the bag of its words, in terms of count frequency of each word (term)

(1) John likes to watch movies. Mary likes movies too.

(2) John also likes to watch football games.

Count Word Frequency

(1) [1, 2, 1, 1, 2, 0, 0, 0, 1, 1]

(2) [1, 1, 1, 1, 0, 1, 1, 1, 0, 0]

Bag of words model (BOW)

- *John is quicker than Mary*
- *Mary is quicker than John*

[1, 1, 1, 1, 1]

[1, 1, 1, 1, 1]

Disregarding grammar and word order:

These two sentences have the same vector representation.
Also context is ignored.
Also, .. see next slide

Raw frequency

- Raw term frequency is not what we want:
  - Two documents with 10 occurrences of the term "of" are more relevant than two documents with 2 occurrence of the term "Pneumonia".
- Some words might be frequent, but are not helpful.
  - Relevance does not increase proportionally with term frequency.
- A long document is not equal to a short document
  - Longer = Higher frequency of terms

Term frequency (TF)

- The term frequency TF(*t,d*)of term *t* in document *d* is defined as the number of times that *t* occurs in *d*.
- This addresses the problem of long vs. short documents.

$$\mathrm{TF}(t, d) = \frac{\text{Number of times term } t \text{ appears in a document } d}{\text{Total number of terms in the document } d}$$

HISTORY OF PRESENT ILLNESS: The patient is a (XX)-month-old child. Family notes that since last night, the patient was fussy and irritable, possibly pulling at her ears. No obvious pain or discomfort with urination. No chest pain, cough. No abdominal pain. No nausea, vomiting or diarrhea and is having usual stool and wet diapers. Child, however, has been much more fussy at times and now comes to the ER for further evaluation. This is the mother's first child. The child is consolable and not constantly crying in the ER and is able to be addressed and approached by me without being upset. Family notes the child has been fully immunized and is compliant with scheduled appointments…….

Count Word Frequency

$$\mathrm{TF}(\textit{“pain”, Report 1}) = \frac{5}{320}$$

$$\mathrm{TF}(\textit{“the”, Report 1}) = \frac{17}{320}$$

Word Frequency

- Rare terms are more informative than frequent terms

  Recall stop words which are not so useful:

  [the, a, an, another, for, an, nor, but, or, yet, so, ….]

  - In clinical NLP, a different set of stop words might be needed.
    - [mcg, dr, patient ,….]

  - Generally, we want to emphasize words that occur frequently in a given document, while at the same time de-emphasizing words that occur frequently in many documents.

Inverse Document Frequency (IDF)

- DF(t) is the document frequency of t: the number of documents that contain t
  - DF(t) is an inverse measure of the informativeness of t
- We define the IDF (inverse document frequency) of t by

$$\text{IDF(t)} = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents with term t in it}}\right)$$

- This addresses the problem of rare words.

TF-IDF weighting

- The tf-idf weight of a term is the product of its TF weight and its IDF weight for a collection of documents D

$$\text{TF-IDF}(t,d,D) = Tf(t,d) * IDF(t,D)$$

There are many variants of the above formula.

Example

1. Consider a document containing 100 words wherein the word Tumor appears 3 times.
   - The term frequency (i.e., tf) for Tumor is then (3 / 100) = 0.03.

2. Now, assume we have 100,000 documents and the word Tumor appears in 10 of these.
   - Then, the inverse document frequency (i.e., idf) is calculated as log(100,000 / 10) = 4.

3. Thus, the Tf-idf weight is the product of these quantities: 0.03 * 4 = 0.12.

count → weight matrix

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	5.25	3.18	0	0	0	0.35
Brutus	1.21	6.1	0	1	0	0
Caesar	8.59	2.54	0	1.51	0.25	0
Calpurnia	0	1.54	0	0	0	0
Cleopatra	2.85	0	0	0	0	0
mercy	1.51	0	1.9	0.12	5.25	0.88

Each document is now represented by a real-valued vector of tf-idf weights $\in \mathbb{R}^{|V|}$

BOW/TF-IDF Issues

- *John is quicker than Mary*
- *Mary is quicker than John*

[1, 1, 1, 1, 1]

[1, 1, 1, 1, 1]

Disregarding grammar and word order:

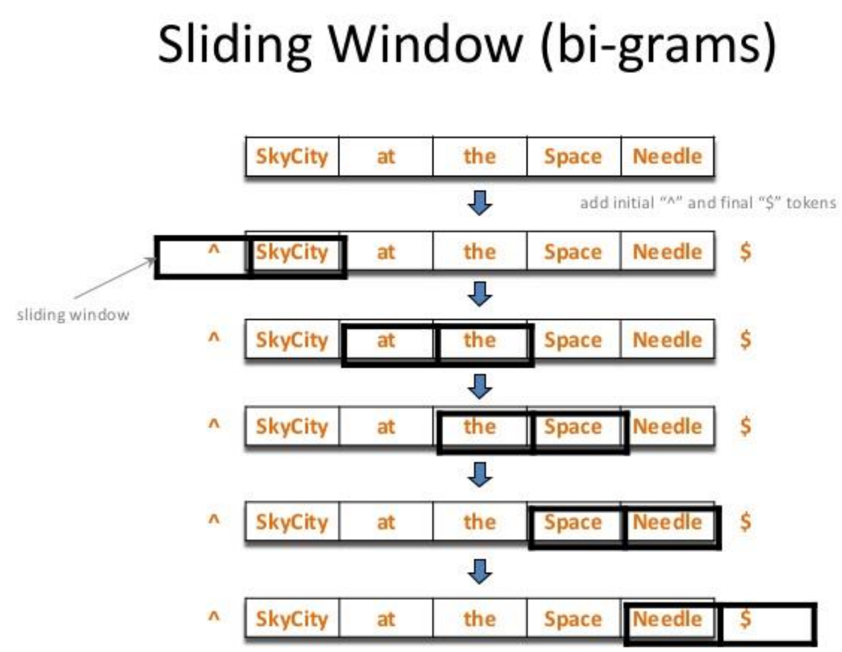
These two sentences have the same vector representation.

And … context is ignored.

N-Gram Models of Language

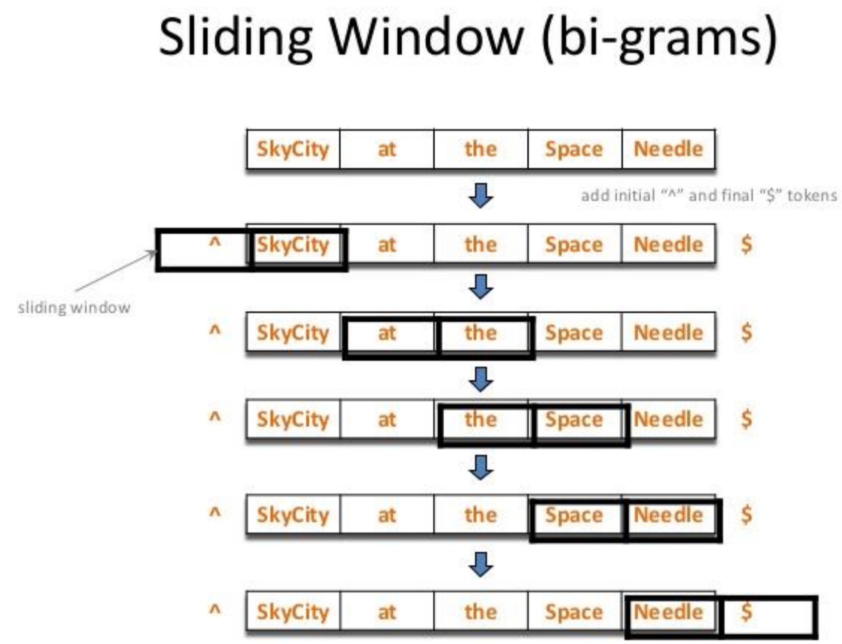
- Use the previous N-1 words in a sequence to capture some context
  - unigrams, bigrams, trigrams,…

$$P(W_n|W_{n-1}) = \frac{P(W_{n-1}, W_n)}{P(W_{n-1})}$$



Building N-Gram Models

- How do we build these models?
  - Very large corpora
    - Wall Street Journal
    - AP newswire
    - Google …



Google NGrams

All Our N-gram are Belong to You

By Peter Norvig - 8/03/2006 11:26:00 AM

Posted by Alex Franz and Thorsten Brants, Google Machine Translation Team

Here at Google Research we have been using word n-gram models for a variety of R&D projects, such as statistical machine translation, speech recognition, spelling correction, entity detection, information extraction, and others. While such models have usually been estimated from training

to share this enormous dataset with everyone. We processed 1,024,908,267,229 words of running text and are publishing the counts for all 1,176,470,663 five-word sequences that appear at least 40 times. There are 13,588,391 unique words, after discarding words that appear less than 200 times.

Google Ngram Viewer

Programs

- Scikit-learn

```
from sklearn.feature_extraction.text import CountVectorizer

vectorizer = CountVectorizer()
```

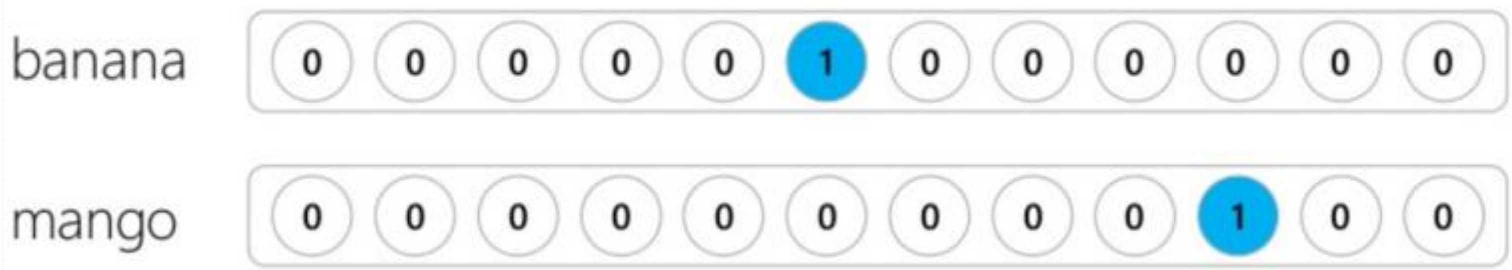
```
tf_idf_matrix = tfidf.transform(freq_term_matrix)
```

- You can also use NLTK

Neural Embeddings

‘One-hot’ Representation

- Bag of words and bi-gram vectors are fine, but:
  - Each word is a sparse vector with one 1 and a lot of zeros (or in case of tf-idf, with a real number)



https://towardsdatascience.com/beyond-word-embeddings-part-2-word-vectors-nlp-modeling-from-bow-to-bert-4ebd4711d0ec

‘One-hot’ Representation

- Dimensionality of the vector will be the size of vocabulary.
    - We can choose the most frequent features, or
    - Features with higher tf-idf weight

    - Fine, but … there is still one more problem!

What about Context?

government debt problems turning into **banking** crises as has happened in

saying that Europe needs unified **banking** regulation to replace the hodgepodge

↖ These words will represent *banking* ↗

Quoted after Socher

Word Similarity

- Similar words have very different representations

size [0 0 0 0 0 0 0 0 0 1 0 0 0 0]

capacity [0 0 0 0 0 0 1 0 0 0 0 0 0 0]

Neural Embedding Models

- Embeddings are low-dimensional, learned continuous vector representations of discrete variables.
- Word2Vec is a neural embedding model that learns relationships between words using a neural network.

Basic idea of learning neural network word embeddings (Predict!)

We define a model that predicts between a center word w_t and context words in terms of word vectors, e.g.,

$$p(\text{context}|w_t) = \ldots$$

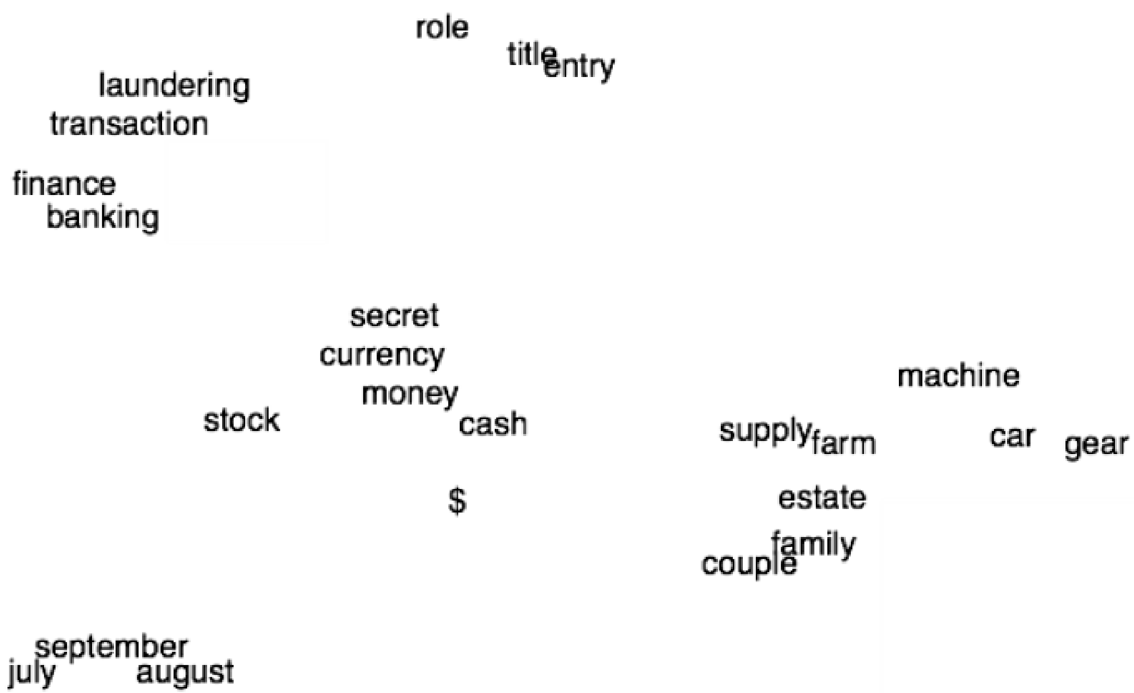
We look at **many** positions t in a big language corpus

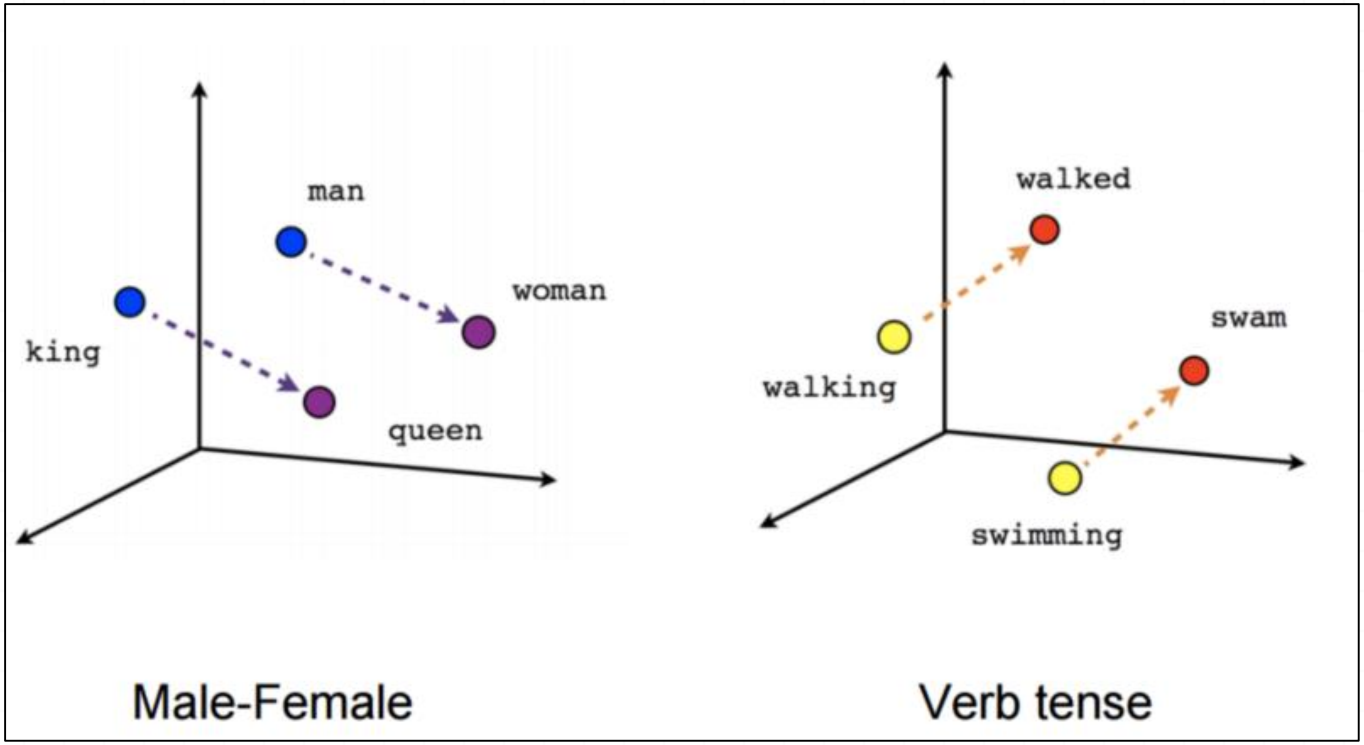
Word meaning as a vector

The result is a dense vector for each word type, chosen so that it is good at predicting other words appearing in its context

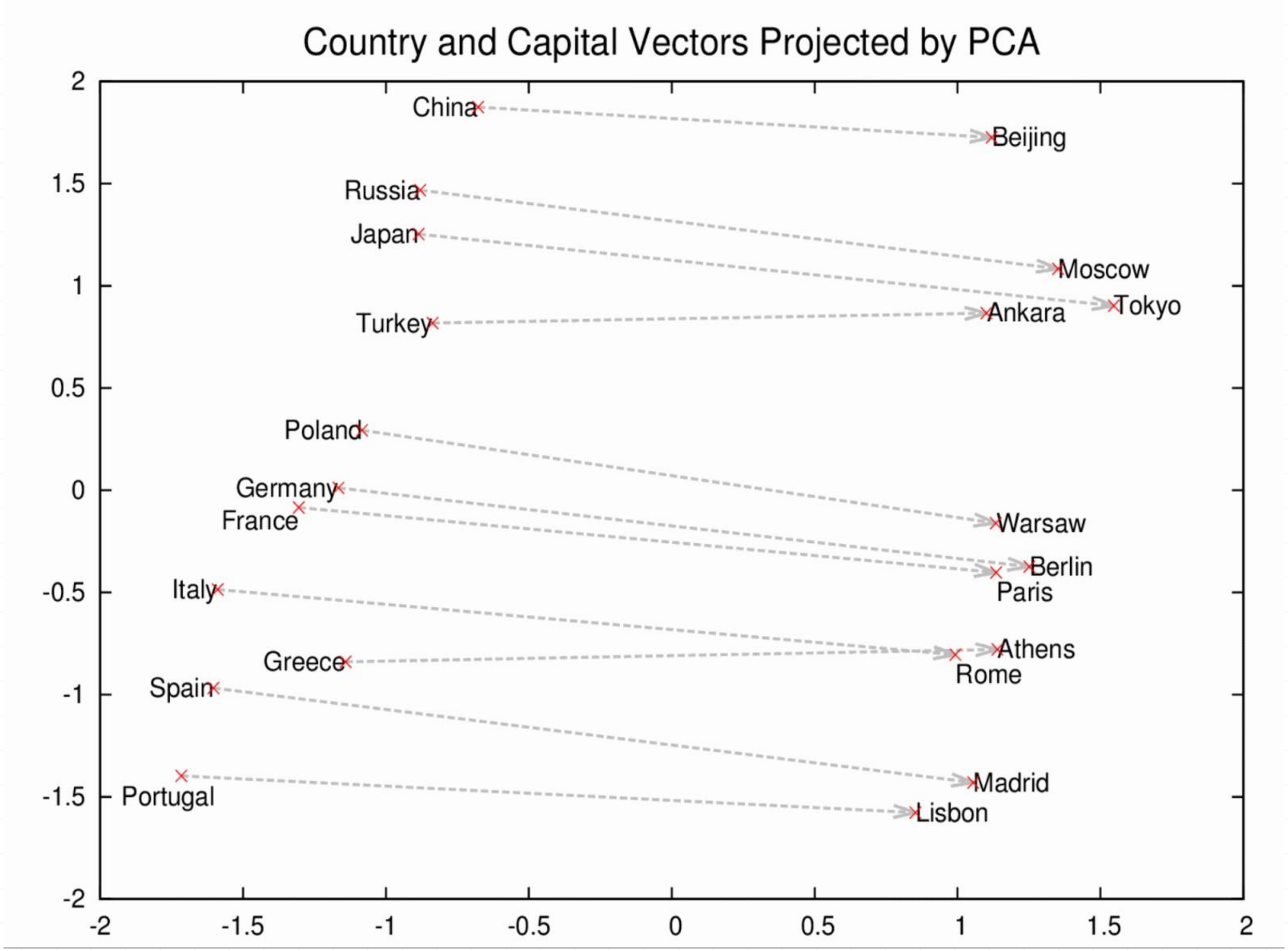
… those other words also being represented by vectors

$$\textit{currency} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix}$$

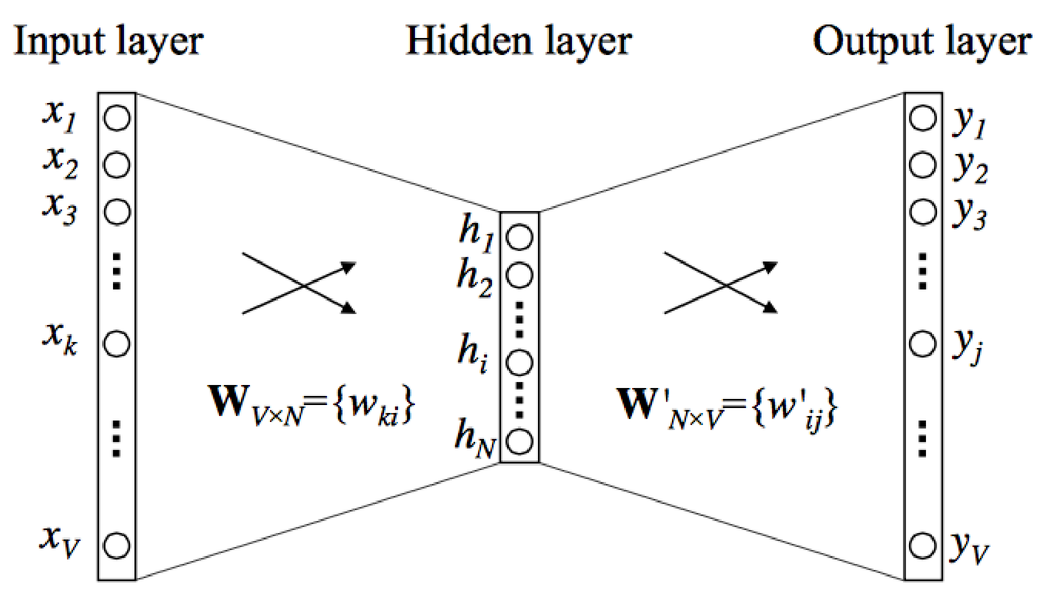




The vectors obtained by subtracting two related words sometimes express a meaningful concept such as gender or verb tense.


Country and Capital Vectors Projected by PCA
China
Beijing
Russia
Japan
Moscow
Tokyo
Turkey
Ankara
Poland
Germany
France
Warsaw
Berlin
Paris
Italy
Greece
Spain
Athens
Rome
Madrid
Portugal
Lisbon
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

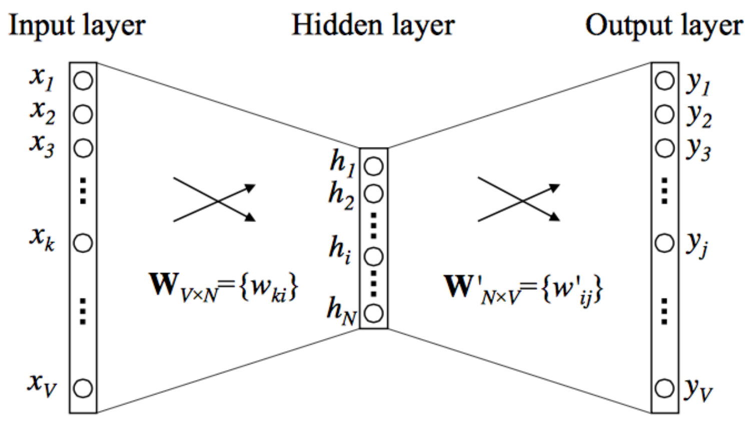
Overall Architecture (shallow neural network)



http://www.folgertkarsdorp.nl/word2vec-an-introduction/

Intuition

- Word2Vec creates word projections in an N-dimensional latent space (N is the size of the word vectors that are obtained).
  - Typical N values: 100, 200, 300



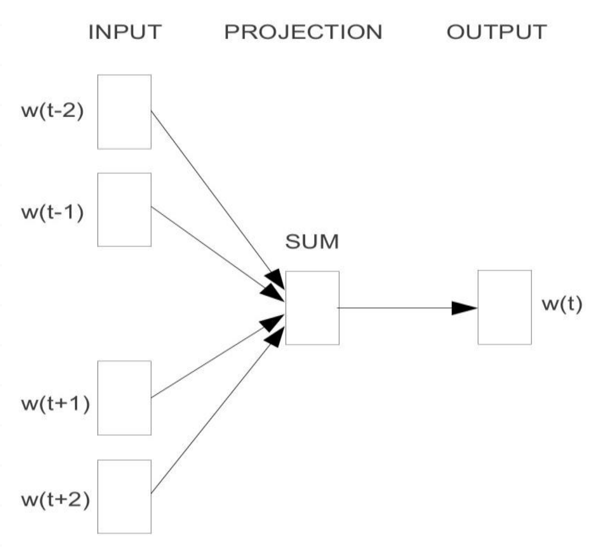
Word2Vec Details

- Two main Word2Vec algorithms
  - **continuous bag-of-words**
  - **continuous skip-gram**

continuous bag-of-words

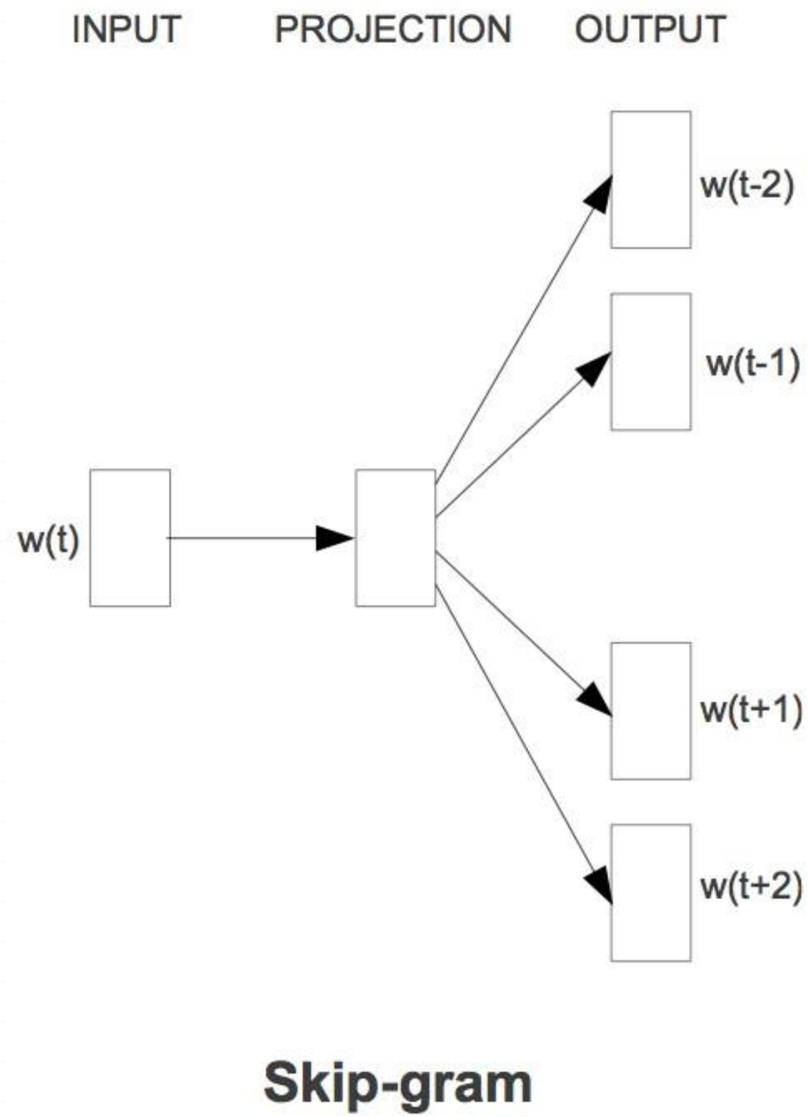
- Predicting the current word **based on the context.**

- All the input and output data are of the same dimension and one-hot encoded.



continuous skip-gram

- maximize classification of a word **based on another word in the same sentence**
- better word vectors for **frequent words**, but slower to train



Mikolov et al.

Other Methods

- FastText
  - An extension to Word2Vec proposed by Facebook in 2016
  - Instead of feeding individual words into the Neural Network, FastText breaks words into several n-grams (sub-words).
- GloVe
  - Similar to Word2Vec
  - GloVe is a "count-based" model

Recent Trends

- Moving beyond simple word embedding to language modeling
  - Compare to computer vision: only transferring edge/orientation information from a first layer will not be enough!

- Recent methods
  - ELMO (2018)
    - Embeddings are computed from the internal states of a two-layers bidirectional Language Model (LM), with characters as input
    - Outputs are the concatenations of the activations on several layers of the biLMs
  - BERT (2018)
    - bidirectional training of Transformer, a popular attention model