Distributional Semantics and Word Embeddings - Part I

Adapted from Slides by Prof. Pawan Goyal

Somak Aditya

CSE, IIT Kharagpur

Lecture 15

Introduction

Why are we concerned about Semantics?

In IR, similarity between query and document ideally should take their "true meaning" into account.

- In the past: tf-idf based, VSM-based.
- Example: query "fall colors" or "colors of fall" should match documents about *Fall* season (not the verb *fall, rainfall*)

Introduction

Why are we concerned about Semantics?

In IR, similarity between query and document ideally should take their "true meaning" into account.

- In the past: tf-idf based, VSM-based.
- Example: query "fall colors" or "colors of fall" should match documents about *Fall* season (not the verb *fall*, *rainfall*)

What is Semantics?

The study of meaning: Relation between symbols and their groundings.

Introduction

Why are we concerned about Semantics?

In IR, similarity between query and document ideally should take their "true meaning" into account.

- In the past: tf-idf based, VSM-based.
- Example: query "fall colors" or "colors of fall" should match documents about *Fall* season (not the verb *fall, rainfall*)

What is Semantics?

The study of meaning: Relation between symbols and their groundings. John told Mary that the train moved out of the station at 3 o'clock.

Computational Semantics

Computational Semantics

How do you represent meaning of natural language words, phrases, sentences?
How do you reason with them?

Computational Semantics

Computational Semantics

How do you represent meaning of natural language words, phrases, sentences?
How do you reason with them?

Generally two categories of methods:

- **Formal Semantics:** Precise mathematical models of the relations between natural language and the world.

Computational Semantics

Computational Semantics

How do you represent meaning of natural language words, phrases, sentences?
How do you reason with them?

Generally two categories of methods:

- **Formal Semantics:** Precise mathematical models of the relations between natural language and the world.
  John chases a bat $\rightarrow \exists x[bat(x) \wedge chase(john, x)]$

Computational Semantics

Computational Semantics

How do you represent meaning of natural language words, phrases, sentences?
How do you reason with them?

Generally two categories of methods:

- **Formal Semantics:** Precise mathematical models of the relations between natural language and the world.
  John chases a bat $\rightarrow \exists x[bat(x) \wedge chase(john, x)]$
- **Distributional Semantics:** Using statistical patterns of human written documents/corpora to extract semantics.

Computational Semantics

Computational Semantics

How do you represent meaning of natural language words, phrases, sentences?
How do you reason with them?

Generally two categories of methods:

- **Formal Semantics:** Precise mathematical models of the relations between natural language and the world.
  John chases a bat $\rightarrow \exists x[bat(x) \wedge chase(john, x)]$
- **Distributional Semantics:** Using statistical patterns of human written documents/corpora to extract semantics.

Combining Logical and Distributional semantics - `https://aclanthology.org/J16-4007` (Beltagy et al. 2016)

Distributional Hypothesis

Distributional Hypothesis: Basic Intuition

"The meaning of a word is its use in language." (Wittgenstein, 1953)

"You know a word by the company it keeps." (Firth, 1957)

Distributional Hypothesis

Distributional Hypothesis: Basic Intuition

"The meaning of a word is its use in language." (Wittgenstein, 1953)

"You know a word by the company it keeps." (Firth, 1957)

$\rightarrow$ Semantically similar words tend to have similar distributional patterns.

Example

government debt problems turning into banking crises as has happened in

saying that Europe needs unified banking regulation to replace the hodgepodge

Distributional Hypothesis

Distributional Hypothesis: Basic Intuition

"The meaning of a word is its use in language." (Wittgenstein, 1953)

"You know a word by the company it keeps." (Firth, 1957)

$\rightarrow$ Semantically similar words tend to have similar distributional patterns.

Example

government debt problems turning into	banking	crises as has happened in
saying that Europe needs unified	banking	regulation to replace the hodgepodge

These *surrounding* words will represent banking

Distributional Semantic Models (DSMs)

- Computational models that build contextual semantic repesentations from corpus data

Distributional Semantic Models (DSMs)

- Computational models that build contextual semantic repesentations from corpus data
- DSMs are models for semantic representations
  - The semantic content is represented by a vector
  - Vectors are obtained through the statistical analysis of the linguistic contexts of a word
- Alternative names
  - corpus-based semantics
  - statistical semantics
  - geometrical models of meaning
  - vector semantics
  - word space models

Distributional Semantics: The general intuition

- **Distributions** are vectors in a multidimensional semantic space, that is, objects with a magnitude and a direction.
- The **semantic space** has dimensions which correspond to possible contexts, as gathered from a given corpus.

Building a DSM step-by-step

The "linguistic" steps

Pre-process a corpus (to define targets and contexts)

$\Downarrow$

Select the targets and the contexts

	against	age	agent	ages	ago	agree	ahead	ain.t	air	aka	al
against	2003	90	39	20	88	57	33	15	58	22	24
age	90	1492	14	39	71	38	12	4	18	4	39
agent	39	14	507	2	21	5	10	3	9	8	25
ages	20	39	2	290	32	5	4	3	6	1	6
ago	88	71	21	32	1164	37	25	11	34	11	38
agree	57	38	5	5	37	627	12	2	16	19	14
ahead	33	12	10	4	25	12	429	4	12	10	7
ain't	15	4	3	3	11	2	4	166	0	3	3
air	58	18	9	6	34	16	12	0	746	5	11
aka	22	4	8	1	11	19	10	3	5	261	9
al	24	39	25	6	38	14	7	3	11	9	861

(a) Word × Word

	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
against	0	0	0	1	0	0	3	2	3	0
age	0	0	0	1	0	3	1	0	4	0
agent	0	0	0	0	0	0	0	0	0	0
ages	0	0	0	0	0	2	0	0	0	0
ago	0	0	0	2	0	0	0	0	3	0
agree	0	1	0	0	0	0	0	0	0	0
ahead	0	0	0	1	0	0	0	0	0	0
ain't	0	0	0	0	0	0	0	0	0	0
air	0	0	0	0	0	0	0	0	0	0
aka	0	0	0	1	0	0	0	0	0	0

(b) Word × Document

Building a DSM step-by-step

The "linguistic" steps

Pre-process a corpus (to define targets and contexts)

$\Downarrow$

Select the targets and the contexts

The "mathematical" steps

Count the target-context co-occurrences

$\Downarrow$

Build the distributional matrix

$\Downarrow$

Reduce the matrix dimensions (optional)

$\Downarrow$

Compute the vector distances on the (reduced) matrix

Many design choices

Matrix type		Weighting		Dimensionality reduction		Vector comparison
word × document		probabilities		LSA		Euclidean
word × word		length normalization		PLSA		Cosine
word × search proximity	×	TF-IDF	×	LDA	×	Dice
adj. × modified noun		PMI		PCA		Jaccard
word × dependency rel.		Positive PMI		IS		KL
verb × arguments		PPMI with discounting		DCA		KL with skew
⋮		⋮		⋮		⋮

Many design choices

Matrix type		Weighting		Dimensionality reduction		Vector comparison
word $\times$ document		probabilities		LSA		Euclidean
word $\times$ word		length normalization		PLSA		Cosine
word $\times$ search proximity	$\times$	TF-IDF	$\times$	LDA	$\times$	Dice
adj. $\times$ modified noun		PMI		PCA		Jaccard
word $\times$ dependency rel.		Positive PMI		IS		KL
verb $\times$ arguments		PPMI with discounting		DCA		KL with skew
$\vdots$		$\vdots$		$\vdots$		$\vdots$

General Questions

- How do the rows (words, ...) relate to each other?
- How do the columns (contexts, documents, ...) relate to each other?

Word Vectors

- At one level, it is simply a vector of weights.

Word Vectors

- At one level, it is simply a vector of weights.
- In a simple 1-of-N (or 'one-hot') encoding every element in the vector is associated with a word in the vocabulary.

Word Vectors

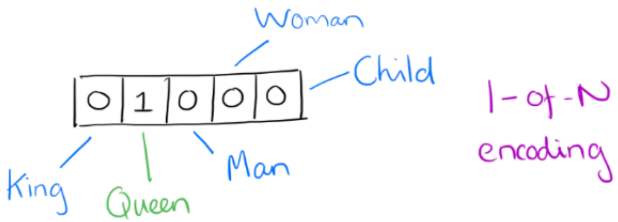
- At one level, it is simply a vector of weights.
- In a simple 1-of-N (or 'one-hot') encoding every element in the vector is associated with a word in the vocabulary.
- The encoding of a given word is the vector in which the corresponding element is set to one, and all other elements are zero.

One-hot representation

motel [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0] AND
hotel [0 0 0 0 0 0 0 1 0 0 0 0 0 0 0] = 0

Word Vectors - One-hot Encoding

- Suppose our vocabulary has only five words: King, Queen, Man, Woman, and Child.
- We could encode the word 'Queen' as:



Limitations of One-hot encoding

Limitations of One-hot encoding

Word vectors are not comparable

Using such an encoding, there is no meaningful comparison we can make between word vectors other than equality testing.

Word2Vec – A distributed representation

Distributional representation – word embedding?

Any word w_i in the corpus is given a distributional representation by an embedding

$$w_i \in R^d$$

i.e., a $d-$dimensional vector, which is mostly learnt!

Word2Vec – A distributed representation

Distributional representation – word embedding?

Any word w_i in the corpus is given a distributional representation by an embedding

$$w_i \in R^d$$

i.e., a $d-$dimensional vector, which is mostly learnt!

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

Distributional Representation

- Take a vector with several hundred dimensions (say 1000).
- Each word is represented by a distribution of weights across those elements.
- So instead of a 1-to-1 mapping between an element in the vector and a word, the representation of a word is <u>spread across all elements</u> of the vector, and
- Each element in the vector contributes to the definition of many words.

Distributional Representation: Illustration

If we label the dimensions in a hypothetical word vector (there are no such pre-assigned labels in the algorithm of course), it might look a bit like this:

Such a vector comes to represent in some abstract way the 'meaning' of a word

Word Embeddings

- d typically in the range 50 to 1000
- Similar words should have similar embeddings

Word Embeddings

- d typically in the range 50 to 1000
- Similar words should have similar embeddings

SVD can also be thought of as an embedding method

Reasoning with Word Vectors

- It has been found that the learned word representations in fact capture meaningful syntactic and semantic regularities in a very simple way.

Reasoning with Word Vectors

- It has been found that the learned word representations in fact capture meaningful syntactic and semantic regularities in a very simple way.
- Specifically, the regularities are observed as constant vector offsets between pairs of words sharing a particular relationship.

Reasoning with Word Vectors

- It has been found that the learned word representations in fact capture meaningful syntactic and semantic regularities in a very simple way.
- Specifically, the regularities are observed as constant vector offsets between pairs of words sharing a particular relationship.

Case of Singular-Plural Relations

If we denote the vector for word i as x_i, and focus on the singular/plural relation, we observe that

Reasoning with Word Vectors

- It has been found that the learned word representations in fact capture meaningful syntactic and semantic regularities in a very simple way.
- Specifically, the regularities are observed as constant vector offsets between pairs of words sharing a particular relationship.

Case of Singular-Plural Relations

If we denote the vector for word i as x_i, and focus on the singular/plural relation, we observe that

$$x_{apple} - x_{apples} \approx x_{car} - x_{cars} \approx x_{family} - x_{families} \approx x_{car} - x_{cars}$$

and so on.

Reasoning with Word Vectors

Perhaps more surprisingly, we find that this is also the case for a variety of semantic relations.

Good at answering analogy questions

a is to b, as c is to ?

man is to *woman* as *uncle* is to ? (*aunt*)

Reasoning with Word Vectors

Perhaps more surprisingly, we find that this is also the case for a variety of semantic relations.

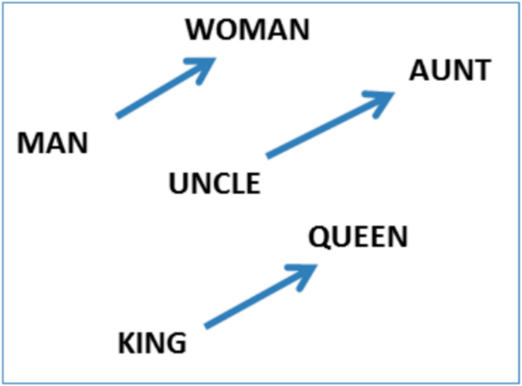
Good at answering analogy questions

a is to b, as c is to ?

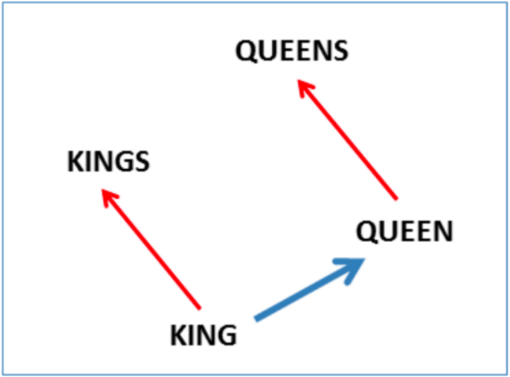
man is to *woman* as *uncle* is to ? (*aunt*)

A simple vector offset method based on cosine distance shows the relation.

Vcctor Offset for Gender Relation



Vcctor Offset for Singular-Plural Relation



Encoding Other Dimensions of Similarity

Analogy Testing

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

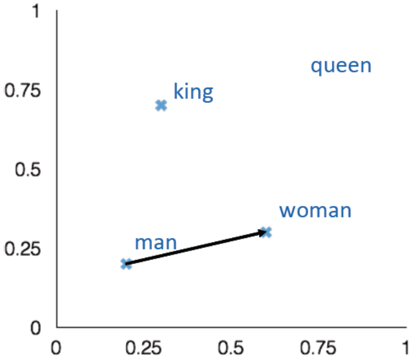
Analogy Testing

a:b :: c:?

$$d = \arg\max_{x} \frac{(w_b - w_a + w_c)^T w_x}{||w_b - w_a + w_c||}$$

man:woman :: king:?

+	king	[0.30 0.70]
-	man	[0.20 0.20]
+	woman	[0.60 0.30]
	queen	[0.70 0.80]



Country-capital city relationships

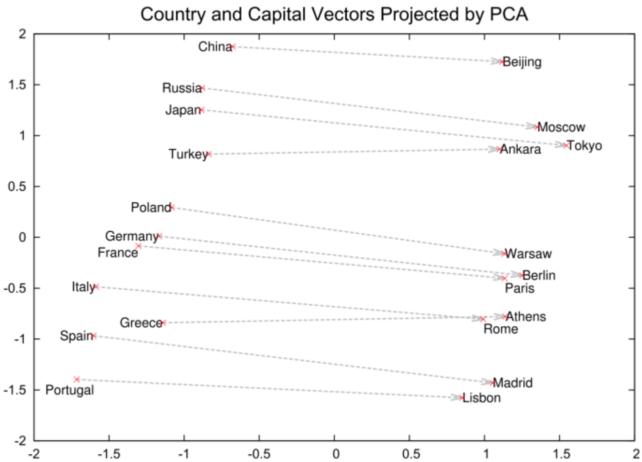


Figure 2: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors of countries and their capital cities. The figure illustrates ability of the model to automatically organize concepts and learn implicitly the relationships between them, as during the training we did not provide any supervised information about what a capital city means.

More Analogy Questions

Newspapers			
New York	New York Times	Baltimore	Baltimore Sun
San Jose	San Jose Mercury News	Cincinnati	Cincinnati Enquirer
NHL Teams			
Boston	Boston Bruins	Montreal	Montreal Canadiens
Phoenix	Phoenix Coyotes	Nashville	Nashville Predators
NBA Teams			
Detroit	Detroit Pistons	Toronto	Toronto Raptors
Oakland	Golden State Warriors	Memphis	Memphis Grizzlies
Airlines			
Austria	Austrian Airlines	Spain	Spainair
Belgium	Brussels Airlines	Greece	Aegean Airlines
Company executives			
Steve Ballmer	Microsoft	Larry Page	Google
Samuel J. Palmisano	IBM	Werner Vogels	Amazon

Table 2: Examples of the analogical reasoning task for phrases (the full test set has 3218 examples). The goal is to compute the fourth phrase using the first three. Our best model achieved an accuracy of 72% on this dataset.

Element Wise Addition

We can also use element-wise addition of vector elements to ask questions such as 'German + airlines' and by looking at the closest tokens to the composite vector come up with impressive answers:

Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

Table 5: Vector compositionality using element-wise addition. Four closest tokens to the sum of two vectors are shown, using the best Skip-gram model.

Basic Idea

Instead of capturing co-occurrence counts directly, predict (using) surrounding words of every word.

Code as well as word-vectors: *https://code.google.com/p/word2vec/*

Two Variations: CBOW and Skip-grams