Compositionality in DS

Raffaella Bernardi

University of Trento

November, 2017
Sample exam

Do you want to have the sample exam on the 22nd of November, or do you prefer having it later (on the 30 of November)?
Acknowledgments

**Credits:** Some of the slides of today lecture are based on earlier DS courses taught by Marco Baroni and Aurelie Herbelot.
Distributional Semantics

Recall

The main questions have been:

1. What is the sense of a given word?
2. How can it be induced and represented?
3. How do we relate word senses (synonyms, antonyms, hyperonym etc.)?

Well established answers:

1. The sense of a word can be given by its use, viz. by the contexts in which it occurs;
2. It can be induced from (either raw or parsed) corpora and can be represented by vectors.
3. Cosine similarity captures synonyms (as well as other semantic relations).
Compositional Distributional Semantics: motivation

- Formal semantics gives an elaborate and elegant account of the productive and systematic nature of language.
- The formal account of compositionality relies on:
  - *words* (the minimal parts of language, with an assigned meaning)
  - *syntax* (the theory which explains how to make complex expressions out of words)
  - *semantics* (the theory which explains how meanings are combined in the process of particular syntactic compositions).
But formal semantics does not actually say anything about lexical semantics (the meaning of *president, president*, is the set of all presidents in particular world).

Who is to say that being a president is being important, and that being ‘president of the United States is being super-important? How will we know that it is equivalent to ‘POTUS’ on social media?

Distributions a potential solution. But if we make the approximation that distributions are ‘meaning’, then we need a way to account for *compositionality* in a distributional setting.
Why not just look at the distribution of phrases?

- The distribution of phrases – even sentences – can be obtained from corpora, but...
  - those distributions are very sparse;
  - observing them does not account for productivity in language.

- Some models assume that corpus-extracted phrasal distributions are irrelevant data.

- Some models assume that, given enough data, corpus-extracted phrasal distributions have the status of gold standard.
Compositionality in FS and DS
Syntax and semantics

“gingerbread gnomes dance under the red moon”

“gingerbread gnomes” “dance under the red moon”
gingerbread gnomes dance “under the red moon”
der under “the red moon”
the “red moon”
red moon
Do all words live in the same space?
What about compositionality of word sense?
How do we “infer” some piece of information out of another?
From one space to multiple spaces, and from only vectors to vectors and matrices.

Several Compositional DS models have been tested so far.

New “similarity measures” have been defined to capture lexical entailment and tested on phrasal entailment too.
All the expressions of the same syntactic category live in the same semantic space.
For instance, ADJ N ("special collection") live in the same space of N ("archives").

<table>
<thead>
<tr>
<th><strong>important route</strong></th>
<th><strong>nice girl</strong></th>
<th><strong>little war</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>important transport</td>
<td>good girl</td>
<td>great war</td>
</tr>
<tr>
<td>important road</td>
<td>big girl</td>
<td>major war</td>
</tr>
<tr>
<td>major road</td>
<td>guy</td>
<td>small war</td>
</tr>
<tr>
<td><strong>red cover</strong></td>
<td><strong>special collection</strong></td>
<td><strong>young husband</strong></td>
</tr>
<tr>
<td>black cover</td>
<td>general collection</td>
<td>small son</td>
</tr>
<tr>
<td>hardback</td>
<td>small collection</td>
<td>small daughter</td>
</tr>
<tr>
<td>red label</td>
<td>archives</td>
<td>mistress</td>
</tr>
</tbody>
</table>
Multiple semantics spaces

Problem of one semantic space model

<table>
<thead>
<tr>
<th></th>
<th>and</th>
<th>of</th>
<th>the</th>
<th>valley</th>
<th>moon</th>
</tr>
</thead>
<tbody>
<tr>
<td>planet</td>
<td>&gt; 1K</td>
<td>&gt; 1K</td>
<td>&gt; 1K</td>
<td>20.3</td>
<td>24.3</td>
</tr>
<tr>
<td>night</td>
<td>&gt; 1K</td>
<td>&gt; 1K</td>
<td>&gt; 1K</td>
<td>10.3</td>
<td>15.2</td>
</tr>
<tr>
<td>space</td>
<td>&gt; 1K</td>
<td>&gt; 1K</td>
<td>&gt; 1K</td>
<td>11.1</td>
<td>20.1</td>
</tr>
</tbody>
</table>

“and”, “of”, “the” have similar distribution but a very different meaning:

“the valley of the moon” vs. “the valley and the moon”

the semantic space of these words must be different from those of eg. nouns (“valley’, “moon”).
Compositionality in DS: Expectation

Disambiguation

Compositionality in DS: Expectation

Semantic deviance

\[
\cos(N, AN)
\]

- steak

- remarkable steak

- residential steak
Kintsch (2001): The meaning of a predicate varies depending on the argument it operates upon:

*The horse run vs. the color run*

Hence, take “gallop” and “dissolve” as landmarks of the semantic space,

- “the horse run” should be closer to “gallop” than to “dissolve”.
- “the color run” should be closer to “dissolve” than to “gallop”

(or put it differently, the verb acts differently on different nouns.)
Compositionality: ADJ N
Pustejovsky (1995)

- red Ferrari [the outside]
- red watermelon [the inside]
- red traffic light [only the signal]
- ..

Similarly, “red” will reinforce the concrete dimensions of a concrete noun and the abstract ones of an abstract noun.
Some distributional compositionality models

- Pointwise models: word-based model, task-evaluated.
- Lexical function model: word-based, evaluated against phrasal distributions.
- Pregroup grammar model: CCG-based model, task-evaluated. [not covered here*]
- Neural Network [not covered here. ML for NLP]

Background: Vector and Matrix

Operations on vectors

Vector addition:

$$\vec{v} + \vec{w} = (v_1 + w_1, \ldots, v_n + w_n)$$

similarly for the −.

Scalar multiplication: 

$$c\vec{v} = (cv_1, \ldots, cv_n)$$ where $c$ is a “scalar”.
Vector visualization

Vectors are visualized by arrows. They correspond to points (the point where the arrow ends.)

\[ \mathbf{v} = (4, 2) \]
\[ \mathbf{w} = (-1, 2) \]
\[ \mathbf{v} + \mathbf{w} = (3, 4) \]
\[ \mathbf{v} - \mathbf{w} = (5, 0) \]

Vector addition produces the diagonal of a parallelogram.
Background: Matrix

Matrices multiplication

A matrix is represented by [nr-rows x nr-columns]. Eg. for a 2 x 3 matrix, the notation is:

\[
A = \begin{bmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23}
\end{bmatrix}
\]

\(a_{ij}\) i stands for the row nr, and j stands for the column nr.

The multiplication of two matrices is obtained by

*Rows of the 1st matrix x columns of the 2nd.*

A matrix with m-columns can be multiplied only by a matrix of m-rows:

\([n \times m] \times [m \times k] = [n \times k]\).
Background: Vector and Matrix

A matrix acts on a vector

Example of 2 x 2 matrix multiplied by a 2 x 1 matrix (viz. a vector). Take $A$ and $\vec{x}$ to be as below.

$$A \vec{x} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} (1, 0) \cdot (x_1, x_2) \\ (-1, 1) \cdot (x_1, x_2) \end{bmatrix} = \begin{bmatrix} 1(x_1) + 0(x_2) \\ -1(x_1) + 1(x_2) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 - x_1 \end{bmatrix} = \vec{b}$$

$A$ is a “difference matrix”: the output vector $\vec{b}$ contains differences of the input vector $\vec{x}$ on which “the matrix has acted.”
Background: Vector and Matrix

A matrix acts on a vector: Exercise

Given the matrix $A$ and the vector $\mathbf{v}$ below, compute the multiplication $A\mathbf{v}$

$$A = \begin{bmatrix} 3 & 5 & 6 \\ 4 & 7 & 10 \end{bmatrix}$$

$$\mathbf{v} = (2, 4)$$
### Compositionality in DS

#### Different Models

<table>
<thead>
<tr>
<th></th>
<th>horse</th>
<th>run</th>
<th>horse + run</th>
<th>horse ⊗ run</th>
<th>run(horse)</th>
</tr>
</thead>
<tbody>
<tr>
<td>gallop</td>
<td>15.3</td>
<td>24.3</td>
<td>39.6</td>
<td>371.8</td>
<td>24.6</td>
</tr>
<tr>
<td>jump</td>
<td>3.7</td>
<td>15.2</td>
<td>18.9</td>
<td>56.2</td>
<td>19.3</td>
</tr>
<tr>
<td>dissolve</td>
<td>2.2</td>
<td>20.1</td>
<td>22.3</td>
<td>44.2</td>
<td>12.4</td>
</tr>
</tbody>
</table>

- Additive and/or Multiplicative Models: Mitchell & Lapata (2008), Guevara (2010)
- For others, see Mitchell and Lapata (2010) overview, and Frege in Space related work section.
Compositionality as vectors composition

General class of models:

\[ \hat{\mathbf{p}} = f(\mathbf{u}, \mathbf{v}, R, K) \]

- \( \hat{\mathbf{p}} \) can be in a different space than \( \mathbf{u} \) and \( \mathbf{v} \).
- \( K \) is background knowledge
- \( R \) syntactic relation.

Putting constraints will provide us with different models.
Mitchell and Lapata (2010)

- Word-based (5 words on either side of the lexical item under consideration).
- The composition of two vectors \( \vec{u} \) and \( \vec{v} \) is some function \( f(\vec{u}, \vec{v}) \).
  - M & L try:
    - addition \( p_i = \vec{u}_i + \vec{v}_i \)
    - multiplication \( p_i = \vec{u}_i \cdot \vec{v}_i \)
    - tensor product \( p_{ij} = \vec{u}_i \cdot \vec{v}_j \)
    - circular convolution \( p_{ij} = \sigma_j \vec{u}_j \cdot \vec{v}_{i-j} \)
    - ... etc

- Task-based evaluation: similarity ratings (noun noun, adj noun, verb object phrases.). Sperman correlation human and models.
Compositionality as vectors composition

1. Not only the $i$th components of $\vec{u}$ and $\vec{v}$ contribute to the $i$th component of $\vec{p}$. Circular convolution:

   $$p_i = \sum_j u_j \cdot v_{i-j}$$

2. Role of $K$, e.g. consider the argument’s distributional neighbours Kitsch 2001:

   $$\vec{p} = \vec{u} + \vec{v} + \sum \vec{n}$$

3. Asymmetry weights pred and arg differently:

   $$p_i = \alpha u_j + \beta v_i$$

4. the $i$th component of $\vec{u}$ should be scaled according to its relevance to $\vec{v}$ and vice versa.

   multiplicative model
Discussion: the meaning of $f$

- How do we interpret $f(\vec{u}, \vec{v})$ linguistically?
- Intersection in formal semantics has a clear interpretation:  
  $\exists x [\text{cat}^\prime(x) \land \text{black}^\prime(x)]$
  There is a cat in the set of all cats which is also in the set of black things.
- But what with addition, multiplication?
**Multiplication**

- Multiplication is intersective.

- But it is commutative in a word-based model:
  \[
  \text{The cat chases the mouse} = \text{The mouse chases the cat}
  \]

- Note that in a syntax-based model, things could work out:
  \[
  \text{cat}_{\text{subj}} \text{chase}_{\text{head}} \text{mouse}_{\text{obj}} \neq \text{mouse}_{\text{subj}} \text{chase}_{\text{head}} \text{cat}_{\text{obj}}
  \]
Multiplying to zero

Multiplication has issues retaining information when composing several words. Most dimensions become 0 or close to 0:

\[
\begin{pmatrix}
0.45 \\
0.23 \\
0.00 \\
0.14 \\
0.76
\end{pmatrix}
\begin{pmatrix}
0.11 \\
0.43 \\
0.54 \\
0.00 \\
0.39
\end{pmatrix}
= 
\begin{pmatrix}
0.05 \\
0.10 \\
0.00 \\
0.00 \\
0.30
\end{pmatrix}
\begin{pmatrix}
0.05 \\
0.10 \\
0.00 \\
0.00 \\
0.30
\end{pmatrix}
\begin{pmatrix}
0.00 \\
0.89 \\
0.57 \\
0.23 \\
0.42
\end{pmatrix}
= 
\begin{pmatrix}
0.00 \\
0.09 \\
0.00 \\
0.00 \\
0.13
\end{pmatrix}
\]
Addition is not intersective: the whole meaning of both $\vec{u}$ and $\vec{v}$ are included in the resulting phrase.

Commutativity is a problem, as with multiplication.

No sense disambiguation and no indication as to how an adjective, for instance, modifies a particular noun (i.e. the distributions of red car and red cheek both include high weights on the blush dimension).

Too much information.

Still, in practice, simple addition has shown good performance on a variety of tasks...
Scottish castles in a DS space

- 20 nearest neighbours of “Scottish castle” (additive model):
  - 'castle', 'scottish', 'scotland', 'castles', 'dunkeld', 'huntly',
  - 'perthshire', 'linlithgow', 'gatehouse', 'crieff', 'inverness',
  - 'covenanter', 'haddington', 'moray', 'jacobites', 'atholl', 'holyrood',
  - 'jedburgh', 'braemar', 'lanark'
Compositionality in DS: Expectation

Pointwise models

Compositionality: DP IV

- 120 experimental items consisting of 15 reference verbs each coupled with 4 nouns and 2 (high- and low-similarity) landmarks
- Similarity of sentence with reference vs. landmark rated by 49 subjects on 1-7 scale

<table>
<thead>
<tr>
<th>Noun</th>
<th>Reference</th>
<th>High</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>The fire</td>
<td>glowed</td>
<td>burned</td>
<td>beamed</td>
</tr>
<tr>
<td>The face</td>
<td>glowed</td>
<td>beamed</td>
<td>burned</td>
</tr>
<tr>
<td>The child</td>
<td>strayed</td>
<td>roamed</td>
<td>digressed</td>
</tr>
<tr>
<td>The discussion</td>
<td>strayed</td>
<td>digressed</td>
<td>roamed</td>
</tr>
<tr>
<td>The sales</td>
<td>slumped</td>
<td>declined</td>
<td>slouched</td>
</tr>
<tr>
<td>The shoulders</td>
<td>slumped</td>
<td>slouched</td>
<td>declined</td>
</tr>
</tbody>
</table>

Table 1: Example Stimuli with High and Low similarity landmarks
Compositionality: DP IV

Models vs. Human judgment: different ranging scale. Additive model, Non compositional baseline, weighted additive and Kintsch (2001) don’t distinguish between High (close) and Low (far) landmarks. Multiplicative and combined models are closed to human ratings. The former does not require parameter optimization.

<table>
<thead>
<tr>
<th>Model</th>
<th>High</th>
<th>Low</th>
<th>ρ</th>
</tr>
</thead>
<tbody>
<tr>
<td>NonComp</td>
<td>0.27</td>
<td>0.26</td>
<td>0.08</td>
</tr>
<tr>
<td>Add</td>
<td>0.59</td>
<td>0.59</td>
<td>0.04</td>
</tr>
<tr>
<td>Weight Add</td>
<td>0.35</td>
<td>0.34</td>
<td>0.09</td>
</tr>
<tr>
<td>Kintsch</td>
<td>0.47</td>
<td>0.45</td>
<td>0.09</td>
</tr>
<tr>
<td>Multiply</td>
<td>0.42</td>
<td>0.28</td>
<td>0.17</td>
</tr>
<tr>
<td>Combined</td>
<td>0.38</td>
<td>0.28</td>
<td>0.19</td>
</tr>
<tr>
<td>Human Judg</td>
<td>4.94</td>
<td>3.25</td>
<td>0.40</td>
</tr>
</tbody>
</table>
### Compositionality as vector combination: problems

Grammatical words: highly frequent

<table>
<thead>
<tr>
<th></th>
<th>planet</th>
<th>night</th>
<th>space</th>
<th>color</th>
<th>blood</th>
<th>brown</th>
</tr>
</thead>
<tbody>
<tr>
<td>the</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
</tr>
<tr>
<td>moon</td>
<td>24.3</td>
<td>15.2</td>
<td>20.1</td>
<td>3.0</td>
<td>1.2</td>
<td>0.5</td>
</tr>
<tr>
<td>the moon</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
</tr>
</tbody>
</table>
### Composition as vector combination: problems

**Grammatical words variation**

<table>
<thead>
<tr>
<th></th>
<th>car</th>
<th>train</th>
<th>theater</th>
<th>person</th>
<th>movie</th>
<th>ticket</th>
</tr>
</thead>
<tbody>
<tr>
<td>few</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
</tr>
<tr>
<td>a few</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
<td>&gt;1K</td>
</tr>
<tr>
<td>seats</td>
<td>24.3</td>
<td>15.2</td>
<td>20.1</td>
<td>3.0</td>
<td>1.2</td>
<td>0.5</td>
</tr>
<tr>
<td>few seats</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
</tr>
<tr>
<td>a few seats</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
<td>??</td>
</tr>
</tbody>
</table>

- There are few seats available. negative: hurry up!
- There are a few seats available. positive: take your time!
Recall:

- an intransitive verb is a set of entities, hence it's a one argument function. \( e \rightarrow t \)
- transitive verb: set of pairs of entities, hence it's a two argument function: \( e \rightarrow (e \rightarrow t) \)

\[
\begin{align*}
\text{S} & \quad \text{S} \\
\text{DP} & \quad \text{DP} \\
\text{IV} & \quad \text{DP} \backslash \text{S}
\end{align*}
\]

The function “walk” selects a subset of \( D_e \).
Compositionality in Formal Semantics

Adjectives

Syntax:

\[
\begin{array}{c}
\text{N} \\
\overbrace{\text{ADJ}} \quad \overbrace{\text{N}} \\
\text{N/N} \\
\text{N}
\end{array}
\]

ADJ is a function that modifies a noun:

\[[\text{Red}] \cap [\text{Moon}]\]
Baroni and Zamparelli (2010)

- Functional model for adjective-noun composition.
- Composition is the multiplication of vectors/matrices learned from access to phrasal distributions.
- ‘Internal’ evaluation: composition is evaluated against phrasal distributions.
Assumptions

- Given enough data, distributions for phrases should be obtained in the same way as for single words.
- I.e. it is fair to assume that if we have seen enough instances of *black cat*, the context of the phrase should give us an indication of its meaning (perhaps it is more related to witches than *cat* and *ginger cat*).
- Let’s say we have a vector $\vec{a}$ (*black*) and a $\vec{n}$ (*cat*), and also a $\vec{an}$ (*black cat*), we can hypothesise a composition method which combines $\vec{a}$ and $\vec{n}$ to get $\vec{an}$ (standard machine learning).
Assumptions

- There is no single composition operation for adjectives. Each adjective acts on nouns in a different way:
  - *red car*: the outside of the car is evenly painted with the colour red (visual);
  - *fast car*: the engine of the car is powerful (functional);
  - *expensive car*: the price of the car is high (abstract/relational).

- Even single adjectives will combine with various nouns in different ways:
  - *red car*: outside of the car, even paint;
  - *red watermelon*: inside of the watermelon, probably not as red as the car;
  - *red nose*: a little redder than usual, probably due to a cold.
Baroni and Zamparelli’s 2010 proposal

Implementing the idea of function application in a vector space

- Functions as **linear maps** between vector spaces
- Functions are matrices, function application is function-by-vector multiplication
### Compositionality in DS: Function application

Baroni and Zamparelli (2010)

**Distributional Semantics (e.g. 2 dimensional space):**

<table>
<thead>
<tr>
<th>N/N: matrix</th>
<th>N: vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{red} )</td>
<td>( \text{moon} )</td>
</tr>
</tbody>
</table>
| \( \begin{array}{c|cc}
\text{d1} & n1 & n2 \\
\text{d2} & m1 & m2 \\
\end{array} \) | \( \begin{array}{c|cc}
\text{d1} & k1 \\
\text{d2} & k2 \\
\end{array} \) |

Function app. by the matrix product and returns a vector:  
\[
\text{red}(\overrightarrow{\text{moon}}) = \sum_{i=1}^{n} \text{red}_i \text{moon}_i 
\]

<table>
<thead>
<tr>
<th>N: vector</th>
<th>N: vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{red moon} )</td>
<td>( \text{red moon} )</td>
</tr>
</tbody>
</table>
| \( \begin{array}{c|c}
\text{d1} & (n1, n1) \cdot (k1, k2) \\
\text{d2} & (m1, m2) \cdot (k1, k2) \\
\end{array} \) | \( \begin{array}{c|c}
\text{d1} & (n1k1) + (n2k2) \\
\text{d2} & (m1k1) + (m2k2) \\
\end{array} \) |
Vectors are induced from the corpus by a lexical association co-frequency function. [Well established]

Matrices are learned by regression (Baroni & Zamparelli (2010)). E.g.: “red” is learned, using linear regression, from the pairs (N, red-N).

n and the moon shining i
with the moon shining s
rainbowed moon . And the
crescent moon , thrille
in a blue moon only , wi
now , the moon has risen
d now the moon rises , f
y at full moon , get up
crescent moon . Mr Angu

f a large red moon , Campana
, a blood red moon hung over
glorious red moon turning t
The round red moon , she ’s
l a blood red moon emerged f
n rains , red moon blows , w
monstrous red moon had climb
. A very red moon rising is
under the red moon a vampire

...
Compositionality in DS: Function application

Learning matrices

red (R) is a matrix whose values are unknown (I use capitol letters for unknown):

\[
\begin{bmatrix}
R_{11} & R_{12} \\
R_{21} & R_{22}
\end{bmatrix}
\]

We have harvested the vectors \(\vec{moo}n\) and \(\vec{ar}my\) representing “moon” and “army”, resp. and the vectors \(\vec{n}_1 = (n_{11}, n_{12})\) and \(\vec{n}_2 = (n_{21}, n_{22})\) representing “red moon”, “red army”. Since we know that e.g.

\[
R \vec{moo}n = \begin{bmatrix}
R_{11}moon_1 + R_{12}moon_2 \\
R_{21}moon_1 + R_{22}moon_2
\end{bmatrix} = \begin{bmatrix}
n_{11} \\
n_{12}
\end{bmatrix} = \vec{n}_1
\]

taking all the data together, we end up having to solve the following multiple regression problems to determine the \(R\) values (\(r_{11}, r_{12}\) etc.)

\[
\begin{align*}
R_{11}moon_1 + R_{12}moon_2 &= n'_{11} \\
R_{11}army_1 + R_{12}army_2 &= n'_{21} \\
R_{21}moon_1 + R_{22}moon_2 &= n'_{12} \\
R_{21}army_1 + R_{22}army_2 &= n'_{22}
\end{align*}
\]

which are solved by assigning weights to the unknown
System

- Test by measuring distance between a given adjective-noun combination and the corresponding phrasal distribution on unseen data.
Compositionality in DS: ADJ N

Comparison Compositional DS models

Summing up, Baroni & Zamparelli 2010 have

- trained separate models for each adjective;
- (a) composed the learned matrix (function) with a noun vector (argument) by matrix product \( \cdot \) – the adjective weight matrix with the noun vector value;
- composed adjectives with nouns using: (b) additive and (c) multiplicative model –starting from adjective and noun vectors;
- harvested vectors for “adjective-noun” from the corpus;
- compared (a) “learned_matrix \cdot vector\_noun” (“function application”) vs. (b) “vector\_adj + vector\_noun” vs. (c) “vector\_adj \odot vector\_noun”;
- shown that – among (a), (b), (c) – (a) gives results more similar to the “harvested vector\_adj-noun” than the other two methods.
Compositionality in DS: ADJ N

Observed ADJ N vs. Composed ADJ(N): (a) when observed and composed are close

Comparison observed vector (induced from corpus) with the result of the matrix product by comparing their neighbour:

<table>
<thead>
<tr>
<th>adj N</th>
<th>observed neighbor</th>
<th>predicted neighbor</th>
</tr>
</thead>
<tbody>
<tr>
<td>common understanding</td>
<td>common approach</td>
<td>common vision</td>
</tr>
<tr>
<td>different authority</td>
<td>different objective</td>
<td>different description</td>
</tr>
<tr>
<td>different partner</td>
<td>different organisation</td>
<td>different department</td>
</tr>
<tr>
<td>general question</td>
<td>general issue</td>
<td>general issue</td>
</tr>
<tr>
<td>historical introduction</td>
<td>historical background</td>
<td>historical background</td>
</tr>
<tr>
<td>necessary qualification</td>
<td>necessary experience</td>
<td>necessary experience</td>
</tr>
<tr>
<td>new actor</td>
<td>new cast</td>
<td>new case</td>
</tr>
<tr>
<td>recent request</td>
<td>recent enquiry</td>
<td>recent enquiry</td>
</tr>
<tr>
<td>small drop</td>
<td>droplet</td>
<td>drop</td>
</tr>
<tr>
<td>young engineer</td>
<td>young designer</td>
<td>young engineering</td>
</tr>
</tbody>
</table>
### Compositionality in DS: ADJ N

Observed ADJ N vs. Composed ADJ(N): (b) when observed and composed are far

<table>
<thead>
<tr>
<th>adj N</th>
<th>observed neighbor</th>
<th>predicted neighbor</th>
</tr>
</thead>
<tbody>
<tr>
<td>American affair</td>
<td>American development</td>
<td>American policy</td>
</tr>
<tr>
<td>current dimension</td>
<td>left (a)</td>
<td>current element</td>
</tr>
<tr>
<td>good complaint</td>
<td>current complaint</td>
<td>good beginning</td>
</tr>
<tr>
<td>great field</td>
<td>excellent field</td>
<td>great distribution</td>
</tr>
<tr>
<td>historical thing</td>
<td>different today</td>
<td>historical reality</td>
</tr>
<tr>
<td>important summer</td>
<td>summer</td>
<td>big holiday</td>
</tr>
<tr>
<td>large pass</td>
<td>historical region</td>
<td>large dimension</td>
</tr>
<tr>
<td>special something</td>
<td>little animal</td>
<td>special thing</td>
</tr>
<tr>
<td>white profile</td>
<td>chrome (n)</td>
<td>white show</td>
</tr>
<tr>
<td>young photo</td>
<td>important song</td>
<td>young image</td>
</tr>
</tbody>
</table>
From Formal to Distributional Semantics

FS domains and DS spaces

- FS:
  - Atomic vs. functional types
  - Typed denotational domains
  - Correspondence between syntactic categories and semantic types

Could we import these ideas in DS?

- Vectors vs. matrices
- Typed semantic spaces
- Correspondence between syntactic categories and semantic types
A fundamental difference between formal and distributional semantics:

- Formal semantics encodes truth in a model (and just doesn’t know where the model comes from...)
- Distributional semantics encodes usage (including lies).
At best, we can hope to measure consistency/contradictions.

If *Obama* is found in many contexts related to being born in Africa *and* to being born in America, both Obama born in Kenya and Obama born in Hawaii will end up with mediocre weights.
Entailment in DS

Entailment in FS

FS starting point is logical entailment between propositions, hence it’s based on the referential meaning of sentences \(D_t = \{0, 1\}\).

All domains are partially ordered, e.g.:

- \(D_t = \{0, 1\}\) and \(0 \leq t 1\),
- \(D_{e \rightarrow t} : \{\text{student}, \text{person}\}\),
  s.t. \([\text{student}] = \{a, b\}\) and \([\text{person}] = \{a, b, c\}\),
  by def: \([\text{student}] \leq_{e \rightarrow t} [\text{person}]\) since
  \(\forall \alpha \in D_e \ [\text{student}](\llbracket \alpha \rrbracket) \leq_t [\text{person}](\llbracket \alpha \rrbracket),\)
Entailment

- Lexical entailment: already some successful results.
- Phrase entailment: a few studies done.
- Sentential entailment: vd. SICK
Entailment
DS success on Lexical entailment

Cosine similarity has shown to be a valid measure for the synonymy relation, but it does not capture the “is-a” relation properly: it’s symmetric!

Kotlerman, Dagan, Szpektor and Zhitomirsky-Geffet 2010 see is-a relation as “feature inclusion” (where “features” are the space dimensions) and propose an asymmetric measure based on empirical harvested vectors. Intuition behind their measure:

1. Is-a score higher if included features are ranked high for the narrow term.
2. Is-a score higher if included features are ranked high in the broader term vector as well.
3. Is-a score is lower for short feature vectors.

Very positive results compared to WordNet-based measures. They have focused on nouns.
Entailment

Entailment at phrasal level in DS

Baroni, Bernardi, Do and Shan (EACL 2012):

- Dagan et. al. measure
  - does generalize to expressions of the noun category, tested on $N_1 \leq N_2$ and $\text{ADJ } N_1 \leq N_1$.
  - does not generalize to expressions of other categories, tested on QPs.

- FS different partial order for different domains; DS different partial orders for different semantic spaces.
## Entailment

**SVM learned QP entailment**

<table>
<thead>
<tr>
<th>Quantifier pair</th>
<th>Correct</th>
<th>Quantifier pair</th>
<th>Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>many $\models$ several</td>
<td>19%</td>
<td>many $\not\models$ most</td>
<td>65%</td>
</tr>
<tr>
<td>many $\models$ some</td>
<td>86%</td>
<td>many $\not\models$ no</td>
<td>52%</td>
</tr>
<tr>
<td>each $\models$ some</td>
<td>99%</td>
<td>both $\not\models$ many</td>
<td>73%</td>
</tr>
<tr>
<td>most $\models$ many</td>
<td>18%</td>
<td>both $\not\models$ most</td>
<td>94%</td>
</tr>
<tr>
<td>much $\models$ some</td>
<td>88%</td>
<td>both $\not\models$ several</td>
<td>15%</td>
</tr>
<tr>
<td>every $\models$ many</td>
<td>87%</td>
<td>either $\not\models$ both</td>
<td>62%</td>
</tr>
<tr>
<td>all $\models$ many</td>
<td>88%</td>
<td>many $\not\models$ all</td>
<td>97%</td>
</tr>
<tr>
<td>all $\models$ most</td>
<td>85%</td>
<td>many $\not\models$ every</td>
<td>98%</td>
</tr>
<tr>
<td>all $\models$ several</td>
<td>99%</td>
<td>few $\not\models$ many</td>
<td>20%</td>
</tr>
<tr>
<td>all $\models$ some</td>
<td>99%</td>
<td>few $\not\models$ all</td>
<td>97%</td>
</tr>
<tr>
<td>both $\models$ either</td>
<td>2%</td>
<td>several $\not\models$ all</td>
<td>99%</td>
</tr>
<tr>
<td>both $\models$ some</td>
<td>56%</td>
<td>some $\not\models$ many</td>
<td>49%</td>
</tr>
<tr>
<td>several $\models$ some</td>
<td>76%</td>
<td>some $\not\models$ all</td>
<td>99%</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td><strong>77%</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Subtotal 77%**

**P:** 77%, **R:** 77%, **F:** 77%, **A:** 78%**
Entailment
Partially ordered spaces

The results show that:

- DS models do contain information needed to detect the entailment relation among other categories too, e.g. QP.
- Not the same dimensions/not the same relations among dimensions are at work for different partial orders ($\leq_{QP}$ vs. $\leq_{N}$)

Questions: which are the dimensions involved in the entailment relation for the various categories? Can we hope to find an abstract definition based on atomic and function types as in FS?
Conclusions

Ideas imported from FS into DS

(a) Meaning flows from the words;
(b) “Complete” (vectors) vs. Incomplete words (matrices);
(c) Meaning representations are guided by the syntactic structure.
(d) Different partial order for different semantic spaces
A few references

- M. Baroni and R. Zamparelli (2010). Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. Proceedings of EMNLP

COMPOSES [http://clic.cimec.unitn.it/composes/]
Neural Network and CDSM

(Socher et al., 2012, Kalchbrenner et al., 2014, Cheng and Kartsaklis, 2015)

Pollack (1990); Socher et al. (2011;2012):

\[
\vec{y} = f(\mathbf{W} \vec{w}_1; \vec{w}_2) + \vec{b}
\]

\[
\vec{s} = f(\mathbf{W} \vec{w}_3; \vec{y}) + \vec{b}
\]

NN models, in particular RNN, in which the compositional operator is part of a neural network and is usually optimized against a specific objective. You will learn them in ML for NLP.
Conclusion

Back to our Goals

1. provide students with an overview of the field with focus on the syntax-semantics interface;
2. bring students to be aware on the one hand of several lexicalized formal grammars, on the other hand of computational semantics models and be able to combine some of them to capture the natural language syntax-semantics interface;
3. evaluate several applications [Started] with a special focus to DSM and Language and Vision Models;
4. make students acquainted with writing scientific reports. [Next 4 classes]