From words to phrases in Distributional Semantic Models

Raffaella Bernardi
Università di Trento
1. Logic view on Natural Language Semantics

The main questions are:

1. What does a given sentence mean?
2. How is its meaning built?
3. How do we infer some piece of information out of another?

Logic view answers: The meaning of a sentence 1. is its truth value, 2. is built from the meaning of its words; 3. is represented by a FOL formula, hence inferences can be handled by logic entailment.

Moreover,

- The meaning of most words refers to objects in the domain – it’s the set of entities, or set of pairs/triples of entities.
- Composition is obtained by function-application.
- Syntax guides the building of the meaning representation.
2. **Distributional Models**

You can tell a word by the company it keeps (Firth, 1957)

he curtains open and the moon shining in on the barely
ars and the cold, close moon ". And neither of the w
ough the night with the moon shining so brightly, it
made in the light of the moon. It all boils down, wr
surely under a crescent moon, thrilled by ice-white
sun, the seasons of the moon? Home, alone, Jay pla
m is dazzling snow, the moon has risen full and cold
un and the temple of the moon, driving out of the hug
in the dark and now the moon rises, full and amber a
bird on the shape of the moon over the trees in front
But I could n’t see the moon or the stars, only the
rning, with a sliver of moon hanging among the stars
they love the sun, the moon and the stars. None of
the light of an enormous moon. The plash of flowing w
man ’s first step on the moon; various exhibits, aer
the inevitable piece of moon rock. Housing The Airsh
oud obscured part of the moon. The Allied guns behind
2.1. Semantic Space Model

It’s a quadruple \( \langle B, A, S, V \rangle \), where:

- **B** is the set of “basis elements” – the dimensions of the space.

- **A** is a lexical association function that assigns co-occurrence frequency of words to the dimensions.

- **S** is a similarity measure.

- **V** is an optional transformation that reduces the dimensionality of the semantic space.
2.2. Toy example: vectors in a 2 dimensional space

\[ B = \{\text{shadow, shine,}\}; A = \text{frequency}; S: \text{angle measure (or Euclidean distance.)} \]

Smaller is the angle, more similar are the terms.
2.3. Space, dimensions, co-occurrence frequency

Word Meaning Let’s take a 6 dimensional space: $B = \{planet, night, full, shadow, shine, crescent\}$:

<table>
<thead>
<tr>
<th></th>
<th>planet</th>
<th>night</th>
<th>full</th>
<th>shadow</th>
<th>shine</th>
<th>crescent</th>
</tr>
</thead>
<tbody>
<tr>
<td>moon</td>
<td>10</td>
<td>22</td>
<td>43</td>
<td>16</td>
<td>29</td>
<td>12</td>
</tr>
<tr>
<td>sun</td>
<td>14</td>
<td>10</td>
<td>4</td>
<td>15</td>
<td>45</td>
<td>0</td>
</tr>
<tr>
<td>dog</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>10</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The “meaning” of “moon” is the $\vec{moon}$ in the 6-dimensional space:


(Many) space dimensions Usually, the space dimensions are the most $k$ frequent words (minus stop words.). They can be plain words, words with their PoS, words with their syntactic relation (viz. the corpus used can be analysed at different levels.)

Co-occurrence frequency Instead of plain counts, the values can be more significant weights that take into account frequency and relevance of the words within the corpus. (e.g. tf-idf, mutual information, log-likelihood ratio etc.).
2.4. Background: Angle and Cosine

When the **angle measure** increases, the **cosine measure** decreases. (Hence, higher is the cosine, more similar are the terms.)

The cosine of an angle $\alpha$ in a right triangle is the ratio between the side adjacent to the angle and the hypothenuse. It is independent from the size of the triangle.

\[
\cos \alpha = \frac{OH}{OP}
\]

With $OP = 1$, $\cos \alpha = OH$

\[\cos(0^\circ) = 1\]
\[\cos(90^\circ) = 0\]
\[\cos(180^\circ) = -1\]
2.5. **Cosine similarity**

\[
\cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}| |\vec{y}|} = \frac{\sum_{i=1}^{n} x_i \times y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \times \sqrt{\sum_{i=1}^{n} y_i^2}}
\]

in words: the inner product of the vectors, normalized by the vectors' length.

<table>
<thead>
<tr>
<th></th>
<th>planet</th>
<th>night</th>
<th>full</th>
<th>shadow</th>
<th>shine</th>
<th>crescent</th>
</tr>
</thead>
<tbody>
<tr>
<td>moon</td>
<td>10</td>
<td>22</td>
<td>43</td>
<td>16</td>
<td>29</td>
<td>12</td>
</tr>
<tr>
<td>sun</td>
<td>14</td>
<td>10</td>
<td>4</td>
<td>15</td>
<td>45</td>
<td>0</td>
</tr>
<tr>
<td>dog</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>10</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
\cos(\text{moon}, \text{sūn}) = \frac{(10 \times 14) + (22 \times 10) + (43 \times 4) + (16 \times 15) + (29 \times 45) + (12 \times 0)}{\sqrt{10^2 + 22^2 + 43^2 + 16^2 + 29^2 + 12^2} \times \sqrt{14^2 + 10^2 + 4^2 + 15^2 + 45^2 + 0^2}} = 0.54
\]

\[
\cos(\text{moon}, \text{dog}) = \ldots = 0.50
\]

to account for the effects of sparseness (viz. the 0 values) weighted values are used and dimensions are reduced (e.g. by Singular Value Decomposition.)
2.6. **DM success on Lexical meaning**

DM captures pretty well synonyms. DM used over TOEFL test:

- Foreigners average result: 64.5%
- Macquarie University Staff (Rapp 2004):
  - Ave. 5 not native speakers: 86.75%
  - Ave. 5 native speakers: 97.75%
- DM:
  - DM (dimension: words): 64.4%
  - Best system: 92.5%
2.7. DM: Limitations

Focus on words, only recently on composition of words into phrases. Most used approach:

\[ \text{waters} + \text{runs} \text{ (additive model)} \text{ or } \text{waters} \times \text{runs} \text{ (multiplicative model)} \text{.} \]

Our aim  Learn from the logic view to compose DM words meaning representations into DM representations of phrases.
3. Back to the Logic View: Meaning Composition

The meaning of a sentence 1. is its truth value, 2. is built from the meaning of its words; 3. is represented by a FOL formula, hence we use Logic entailment to handle inferences. Moreover,

- The meaning of most words refers to objects in the domain – it’s the set of entities, or set of pairs/triples of entities.

- Composition is obtained by function-application – due to “complete” vs. “incomplete” words distinction.

- Syntax guides the building of the meaning representation. Lambek: function application (elimination) and abstraction (introduction rule).

These (blue) ideas have been incorporated into the DM framework.
3.1. Pre-group view on Distributional Model

Grefenstette, Sadrzadeh, Clark, Coecke, Pulman [2008-2011]

**Assumption 1:** words of different syntactic categories live in different spaces.

- $N^S$: space of nouns. The meaning of elements in this space is captured by a **vector**.
- $(N \otimes N)^S$: TV space. The meaning of elements in this space is captured by a **matrix**.

**Assumption 2:** The matrices in the $(N \otimes N)^S$ are built out of the vectors in $N^S$ – the meaning of a transitive verb is obtained from the meaning of the nouns that occur as its subject and object.
3.1.1. **Nouns’ space**  By means of example, they take the space of nouns to be characterized by the words that in the corpus are in a dependency relation with the nouns (adjective, verbs, etc.).

\[ N^S = \{ f_i | f_i - link - w_n \text{ in the dependency parsed corpus, for all nouns} \} \]

For instance,

\[ N^S = \{ \text{arg-fluffy, arg-ferocious, obj-buys, arg-shrewed, arg-valuable} \} \]

the meaning of a word living in \( N^S \), i.e. nouns, is the vector obtained computing for each dimension (feature) the tf-idf value (how relevant is the co-occurrence of the word with the feature for the given corpus.). \([w_n]\) = \( \vec{w} = \{ f_i : \text{tf-idf} | f_i \in N^S \} \). E.g.

\[
\begin{align*}
[\text{cat}] &= \vec{c\text{at}} = \{ \text{arg-fluffy: 7, arg-ferocious:1, obj-buys: 4, arg-shrewed:3, arg-valuable:1} \} \\
[\text{dog}] &= \vec{d\text{og}} = \{ \text{arg-fluffy: 3, arg-ferocious:6, obj-buys: 2, arg-shrewed:1, arg-valuable:2} \}
\end{align*}
\]
3.1.2. **Transitive verbs’ space**  The novel contribution w.r.t. “traditional” DM view: The space of transitive verbs is characterized by the pairs of noun’s features.

\[ TV^S = \{(f_i, f_j)|f_i, f_j \in N^S\} \]

the meaning of a word living in \(TV^S\), i.e. transitive verbs, is a **superposition**, viz. it is the **matrix** obtained by taking for each \((f_i, f_j)\) in \(TV^S\) the sum of the result of the multiplication of the value of the properties of the subjects and objects of the verb.

\[
[w_{rv}] = \{(f_i, f_j): \sum(f_i^{x_n} \times f_j^{y_n})|(f_i, f_j) \in TV^S\}
\]

where \(x_n\) and \(y_n\) are the subject and object of “w” within the same sentence as found in the dependency parsed corpus, and \(f_i^{x_n}\) (resp. \(f_j^{y_n}\)) are the tf-idf weight associated to \(f_i\) (resp. \(f_j\)) in the \(\vec{x}_n\) (resp. \(\vec{y}_n\)).
3.1.3. Example: transitive verb  Let’s take a corpus with only one sentence with the verb “chase”, viz. “dogs chase cats”.

Recall, the meaning of “dog” and “cats” are the vectors:

<table>
<thead>
<tr>
<th></th>
<th>arg-fluffy</th>
<th>arg-ferocious</th>
<th>obj-buys</th>
<th>arg-shrewd</th>
<th>arg-valuable</th>
</tr>
</thead>
<tbody>
<tr>
<td>dogs</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>cats</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

The meaning of “chase” is represented by the matrix below.

<table>
<thead>
<tr>
<th></th>
<th>arg-fluffy</th>
<th>arg-ferocious</th>
<th>obj-buys</th>
<th>arg-shrewd</th>
<th>arg-valuable</th>
</tr>
</thead>
<tbody>
<tr>
<td>arg-fluffy</td>
<td>(3 x 7) + 0</td>
<td>(3 x 1) + 0</td>
<td>(3 x 4) + 0</td>
<td>(3 x 3) + 0</td>
<td>(3 x 1) + 0</td>
</tr>
<tr>
<td>arg-ferocious</td>
<td>(6 x 7) + 0</td>
<td>(6 x 1) + 0</td>
<td>(6 x 4) + 0</td>
<td>(6 x 3) + 0</td>
<td>(6 x 1) + 0</td>
</tr>
<tr>
<td>obj-buys</td>
<td>(2 x 7) + 0</td>
<td>(2 x 1) + 0</td>
<td>(2 x 4) + 0</td>
<td>(2 x 3) + 0</td>
<td>(2 x 1) + 0</td>
</tr>
<tr>
<td>arg-shrewd</td>
<td>(1 x 7) + 0</td>
<td>(1 x 1) + 0</td>
<td>(1 x 4) + 0</td>
<td>(1 x 3) + 0</td>
<td>(1 x 1) + 0</td>
</tr>
<tr>
<td>arg-valuable</td>
<td>(2 x 7) + 0</td>
<td>(2 x 1) + 0</td>
<td>(2 x 4) + 0</td>
<td>(2 x 3) + 0</td>
<td>(2 x 1) + 0</td>
</tr>
</tbody>
</table>

If in the corpus there were other sentences with “chase” the values above need to be added to those resulting from the other subject and object pairs (i.e. the addition was not with 0.) -superposition.
3.1.4. **Matrix vector composition**  The composition of TV with the subject and the object is obtained by

1. $\mathbf{sub} \otimes \mathbf{obj}$ which results into a matrix. Note $\mathbf{sub} \otimes \mathbf{obj} \neq \mathbf{obj} \otimes \mathbf{sub}$

2. $\mathbf{TV} \odot (\mathbf{sub} \otimes \mathbf{obj})$ which again results into a matrix – Sentences live in the $(N \otimes N)$ space.

Given $\mathbf{dogs}$ and $\mathbf{cats}$ and the matrix of “chase”:

\[
\begin{array}{c|cc}
\text{d1} & \text{d2} \\
\hline
\text{dogs} & 3 & 6 \\
\text{cats} & 7 & 1
\end{array}
\quad
\begin{array}{c|cc}
\text{chase} & \text{d1} & \text{d2} \\
\hline
\text{d1} & n1 & n2 \\
\text{d2} & m1 & m2
\end{array}
\]

the matrices of $\mathbf{dogs} \otimes \mathbf{cats}$ and of the sentence $(\mathbf{chase} \odot (\mathbf{dogs} \otimes \mathbf{cats}))$ are

\[
\begin{array}{c|cc}
\text{d1} & \text{d2} \\
\hline
\mathbf{dogs} \otimes \mathbf{cats} & 3 \times 7 & 3 \times 1 \\
\mathbf{d1} & 6 \times 7 & 6 \times 1
\end{array}
\quad
\begin{array}{c|ccc}
\text{d1} & \text{d2} \\
\hline
\text{dogs \ chase \ cats} & n1 \times 3 \times 7 & n2 \times 3 \times 1 \\
\mathbf{d1} & m1 \times 6 \times 7 & m2 \times 6 \times 1
\end{array}
\]
3.2. Different learning strategies for complete vs. incomplete words

Baroni & Zamparelli 2010:

- a “complete” word is represented by a vector.
- an “incomplete” word is represented by a matrix.

They look into Adjective-Noun composition. Hence, only on functions from “atomic” to “atomic” categories (from noun to noun – from vectors to vectors!)

**Intuition** Learn the vectors and matrices in different ways.

- induce the vectors (complete words’ meaning) from the corpus
- learn the matrix (ATOMIC → ATOMIC function’s meaning) from the argument and the value of the function application pairs.
3.3. Learning the function/matrix

The linear map for “red” is learnt, using linear regression, from the pairs (N, red-N).
3.4. Function application as inner product

From the vectors input pairs, linear regression gives us the values of the “red” matrix

<table>
<thead>
<tr>
<th>input pairs</th>
<th>Learned matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>moon</td>
<td>red</td>
</tr>
<tr>
<td>d1 301 d2 92</td>
<td>d1 n1 n2</td>
</tr>
<tr>
<td>red moon</td>
<td>d2 m1 m2</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Function application is performed by the inner product and returns a vector:

\[
\vec{red} \cdot \vec{moon} = \sum_{i=1}^{n} red_i \times moon_i
\]

<table>
<thead>
<tr>
<th></th>
<th>d1</th>
<th>d2</th>
</tr>
</thead>
<tbody>
<tr>
<td>red moon</td>
<td>(n1 \times 301) + (n2 \times 92)</td>
<td>(m1 \times 301) + (m2 \times 92)</td>
</tr>
</tbody>
</table>

To double check the validity of the approach: the result \( \vec{red} \cdot \vec{moon} \) has been compared to the vector induced from the corpus: positive results.
3.4.1. DM Composition: “function application” Baroni & Zamparelli 2010, they have

- trained separate models for each adjective;
- (a) composed the learned matrix (function) with a noun vector (argument) by inner product (·) the adjective weight matrix with the noun vector value;
- composed adjectives with nouns using: (b) the additive and (c) the multiplicative model – starting from adjective and noun vectors;
- harvested vectors for “adjective-noun” from the corpus;
- compared (a) “learned_matrix · vector_noun” (“function application”) vs. (b) “vector_adj + vector_noun” vs. (c) “vector_adj × vector_noun”;
- shown that – among (a), (b), (c) – (a) gives results more similar to the “harvested vector_adj-noun” than the other two methods.
3.5. DM: Meaning Composition

Ideas imported into DM (a) Meaning flows from the words; (b) “Complete” (argument) vs. Incomplete (function) words; (c) meaning representations are guided by the syntactic structure.

Lesson learned

a “complete” word is represented by a vector

vs.

an “incomplete” word is represented by a matrix.

Function application as inner product between the matrix and the vector.
4. Back to the logic view: Entailment

3. How do we infer some piece of information out of another? Logic view:

Entailment  Partially ordered domains

\[
\llbracket \text{tall student} \rrbracket \leq_{(e,t)} \llbracket \text{student} \rrbracket \quad \text{iff} \quad \forall \alpha \in D_e \\
\llbracket \text{tall student}(\alpha) \rrbracket \leq_t \llbracket \text{student}(\alpha) \rrbracket \quad \text{iff} \\
\llbracket \text{tall student}(\llbracket \alpha \rrbracket) \rrbracket \leq_t \llbracket \text{student}(\llbracket \alpha \rrbracket) \rrbracket \quad \text{iff} \\
\llbracket \text{tall student}(\llbracket \alpha \rrbracket) \rrbracket = 0 \text{ or } \llbracket \text{student}(\llbracket \alpha \rrbracket) \rrbracket = 1.
\]

Monotonicity  Let \( f : A \to B \) be a function and let \( \leq_A, \leq_B \) be partial orders on \( A \) and \( B \), respectively. Then,

a. \( f \) is “monotone increasing” (\( \uparrow \text{Mon} \)) iff \( \forall x, y \in A, x \leq_A y \) implies \( f(x) \leq_B f(y) \).

b. \( f \) is “monotone decreasing” (\( \downarrow \text{Mon} \)) iff \( \forall x, y \in A, x \leq_A y \) implies \( f(y) \leq_B f(x) \).

Some \textbf{tall student} wanders \quad (\uparrow) \quad \text{Every \textbf{student} wanders} \\
Some \textbf{student} wanders \quad (\uparrow) \quad \text{Every \textbf{tall student} wanders} \quad (\downarrow)
4.1. DM success on Lexical entailment

Lexical entailment Cosine similarity has shown to be a valid measure for the synonymy relation, but it does not capture the “is-a” relation – e.g. it’s symmetric! Kotlerman, Dagan, Szpektor and Zhitomirsky-Geffet 2010 see is-a relation as “feature inclusion” and propose an asymmetric measure. 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.
4.2. DM: Limitation

So far focus on lexical entailment

Our aim  DM entailment between meaning representations: from words to phrases.
4.3. Learning the entailment relation

Bernardi, Baroni, Ngoc, Shan – work in progress

<table>
<thead>
<tr>
<th>NOUN1 &lt; NOUN2</th>
<th>Training</th>
<th>Testing</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJ NOUN &lt; NOUN</td>
<td>2492 pairs</td>
<td>Noun1 &lt; Noun2</td>
<td>71%</td>
</tr>
<tr>
<td>Q1 NOUN &lt; Q2 NOUN</td>
<td>25067 pairs</td>
<td>2785 pairs</td>
<td>92%</td>
</tr>
<tr>
<td>Q↑ NOUN1 &lt; Q↑ NOUN2</td>
<td>tot. 2700 pairs</td>
<td>tot. 300 pairs</td>
<td>57%</td>
</tr>
<tr>
<td>Q↓ NOUN2 &lt; Q↓ NOUN1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Data Pairs were creating using:

Quantifiers: many, several, each, some, all, most, much, both, either, few, every, no.
Q↑: some, several, these, those vs. Q↓: few, all, no, every.
Nouns in is-a relation: taken from WordNet.
5. Connection with Moortgat’s talks

\[
\frac{N/N \vdash X_2 : N/N}{N/N \otimes N \vdash N : X_2 X_1} \quad (/E)
\]

\[
\frac{N/N \vdash N/N : X_3}{N/N \otimes (N/N \otimes N) \vdash X_3 (X_2 X_1) : N} \quad (/E)
\]

Instantiate the categories with one of the word belonging to them e.g. “black young dog”, the final meaning representation of the actual string is obtained by replacing the corresponding proof-term variables with the actual meaning representation.

**Logic view:** word meaning is represented by lambda terms (representing the set-theoretical interpretation), hence replace

\[
X_3 \text{ with } \lambda X.\lambda y.\text{black}(y) \land X(y), \quad X_2 \text{ with } \lambda Y.\lambda x.\text{young}(x) \land Y(x), \quad X_1 \text{ with } \lambda z.\text{dog}(z)
\]

\[
\sim \lambda x.\text{black}(x) \land \text{young}(x) \land \text{dog}(x)
\]

**DM view:** word meaning is represented by vectors, hence

\[
\vec{\text{black}} \cdot (\vec{\text{young}} \cdot \vec{\text{dog}}) \sim \text{a new vector.}
\]
6. Back to the Logic View: what else?

1. The meaning of a sentence is its truth value, 2. is built from the meaning of its words; 3. is represented by a FOL formula, hence we use logic entailment to handle inferences. Moreover,

- The meaning of most words refers to objects in the domain – it’s the set of entities, or set of pairs/triples of entities. Quantifiers are second order functions.

- Composition is obtained by function-application.

- Syntax guides the building of the meaning representation. Lambek: function application (elimination) and abstraction (introduction rule).

Open questions in DM view

What’s the meaning of a sentence? What’s the meaning of “entities”, e.g., “John”. Does a DM representation of e.g. quantifiers differ from a matrix? How can structure be de-composed in a DM representation?
7. Acknowledgments

Thanks go to Marco Baroni, Edward Grefenstette, Graham Katz, Alessandro Lenci, Michael Moortgat, Massimo Poesio, Ken Shan, Roberto Zamparelli.