Vector-based Models of Semantic Composition

Jeff Mitchell and Mirella Lapata

School of Informatics
University of Edinburgh

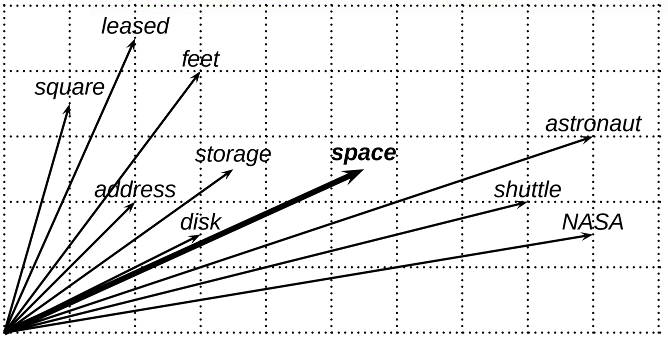
RANLP 2009, Borovets

Outline

Distributional Hypothesis

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

- A word's context provides information about its meaning.
- Words are similar if they share similar linguistic contexts.
- Distributional vs. semantic similarity.



A Simple Semantic Space

Stuart B. Opotowsky was named vice president, tax, for this company with interests in insurance, tobacco, hotels and broadcasting.

A Simple Semantic Space

Stuart B. Opotowsky was named vice president, tax, for this company with interests in insurance, tobacco, hotels and broadcasting.

- Select 2,000 most common content words as contexts.

A Simple Semantic Space

Stuart B. Opotowsky was named vice president, tax, for this company with interests in insurance, tobacco, hotels and broadcasting.

- Select 2,000 most common content words as contexts.
- Five word context window each side of the target word.

A Simple Semantic Space

	vice	president	tax	interests	...
company	1	1	1	1	...

- Select 2,000 most common content words as contexts.
- Five word context window each side of the target word.

A Simple Semantic Space

	vice	president	tax	interests	...
company	25	103	19	55	...

- Select 2,000 most common content words as contexts.
- Five word context window each side of the target word.

A Simple Semantic Space

	vice	president	tax	interests	...
company	0.06	0.26	0.05	0.14	...

- Select 2,000 most common content words as contexts.
- Five word context window each side of the target word.
- Convert counts to probabilities: $p(c|w)$.

A Simple Semantic Space

	vice	president	tax	interests	...
company	1.52	2.32	1.14	1.06	...

- Select 2,000 most common content words as contexts.
- Five word context window each side of the target word.
- Convert counts to probabilities: $p(c|w)$.
- Divide through by probabilities of each context word: $\frac{p(c|w)}{p(c)}$.

A Simple Semantic Space

	vice	president	tax	interests	...
company	1.52	2.32	1.14	1.06	...

- Select 2,000 most common content words as contexts.
- Five word context window each side of the target word.
- Convert counts to probabilities: $p(c|w)$.
- Divide through by probabilities of each context word: $\frac{p(c|w)}{p(c)}$.
- Cosine similarity: $sim(\mathbf{w}_1, \mathbf{w}_2) = \frac{\mathbf{w}_1 \cdot \mathbf{w}_2}{|\mathbf{w}_1||\mathbf{w}_2|}$.

An Alternative: Topic Models

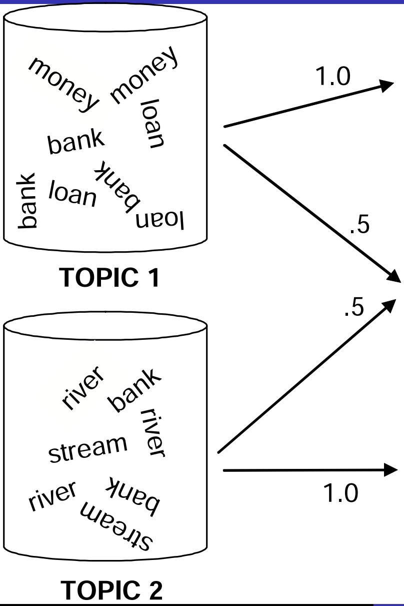
Key Idea: documents are mixtures of topics, topics are probability distributions over words (Blei et al., 2003; Griffiths and Steyvers, 2002; 2003; 2004).

Topic models are generative and structured. For a new document:

1. Choose a distribution over topics
2. Choose a topic at random according to distribution
3. draw a word from that topic

Statistical techniques used to invert the process: infer set of topics that were responsible for generating a collection of documents.

Probabilistic Generative Process

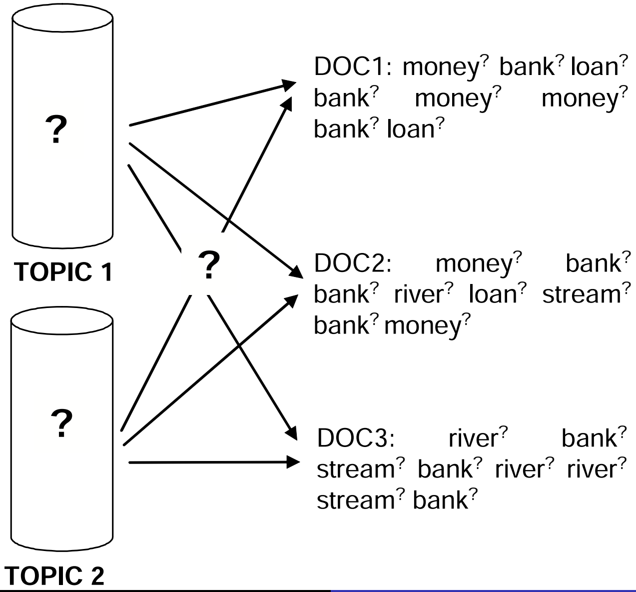


DOC1: money^{1} bank^{1} loan^{1} bank^{1} money^{1} money^{1} bank^{1} loan^{1}

DOC2: money^{1} bank^{1} bank^{2} river^{2} loan^{1} stream^{2} bank^{1} money^{1}

DOC3: river^{2} bank^{2} stream^{2} bank^{2} river^{2} river^{2} stream^{2} bank^{2}

Statistical Inference



Meaning Representation

	Topic 1	Topic 2	Topic 3	Topic n
company	0.69	0.02	0.32	...
practical	0.03	0.44	0.11	...
difficulty	0.06	0.17	0.09	...

- Topics are the dimensions of the space (500, 1000)
- Vector components: probability of word given topic
- Topics correspond to coarse-grained sense distinctions
- Cosine similarity can be used (probabilistic alternatives)

Semantic Space Models

Semantic space models are extremely popular across disciplines!

- Semantic Priming (Lund and Burgess, 1996)
- Text comprehension (Landauer and Dumais, 1997)
- Word association (McDonald, 2000)
- Information Retrieval (Salton et al., 1975)
- Thesaurus extraction (Grefenstette, 1994)
- Word Sense disambiguation (Schütze, 1998)
- Text Segmentation (Hirst, 1997)

Catch: representation of the meaning of **single words**. What about **phrases** or **sentences**?

Compositionality

Meaning of whole is function of meaning of its parts (Frege, 1957).

Compositionality

Meaning of whole is function of meaning of its parts (Frege, 1957).

$$\begin{array}{ccc} \textbf{a} & \textbf{horse} & \textbf{ran} \\ \downarrow & \downarrow & \downarrow \\ \lambda u.\lambda v.\exists x(u@x \wedge v@x) & \lambda y.HORSE(y) & \lambda z.RUN(z) \end{array}$$

Compositionality

Meaning of whole is function of meaning of its parts (Frege, 1957).

$$\begin{tikzcd}
\textbf{a} \arrow[d] & \textbf{horse} \arrow[d] & \textbf{ran} \arrow[d] \\
\lambda u.\lambda v.\exists x(u@x \wedge v@x) \arrow[dr] & \lambda y.HORSE(y) \arrow[d] & \lambda z.RUN(z) \arrow[dl] \\
& \exists x(HORSE(x) \wedge RUN(x)) &
\end{tikzcd}$$

Compositionality

Meaning of whole is function of meaning of its parts (Frege, 1957).

$$
\begin{tikzcd}
\textbf{a} \arrow[d] & \textbf{horse} \arrow[d] & \textbf{ran} \arrow[d] \\
\lambda u.\lambda v.\exists x(u@x \wedge v@x) \arrow[dr] & \lambda y.HORSE(y) \arrow[d] & \lambda z.RUN(z) \arrow[dl] \\
 & \exists x(HORSE(x) \wedge RUN(x)) &
\end{tikzcd}
$$

- Logic can account for sentential meaning (Montague, 1974).
- Differences in meaning are **qualitative** rather than **quantitative**.
- Cannot express **degrees of similarity**.

Compositionality

Partee (1995): the meaning of the whole is a function of the meaning of the parts and of the way they are **syntactically** combined.

Compositionality

Partee (1995): the meaning of the whole is a function of the meaning of the parts and of the way they are **syntactically** combined.

Lakoff (1977): the meaning of the whole is a **greater** than the meaning of the parts.

Compositionality

Partee (1995): the meaning of the whole is a function of the meaning of the parts and of the way they are **syntactically** combined.

Lakoff (1977): the meaning of the whole is a **greater** than the meaning of the parts.

Frege (1884): never ask the meaning of a word in **isolation** but only **in the context** of a statement.

Compositionality

Partee (1995): the meaning of the whole is a function of the meaning of the parts and of the way they are **syntactically** combined.

Lakoff (1977): the meaning of the whole is a **greater** than the meaning of the parts.

Frege (1884): never ask the meaning of a word in **isolation** but only **in the context** of a statement.

Pinker (1994): composition of simple elements must allow the construction of **novel meanings** which go beyond those of the individual elements.

Related Work

It was not the sales manager who hit the bottle that day, but the office worker with the serious drinking problem.

That day the office manager, who was drinking, hit the problem sales worker with a bottle, but it was not serious.

Related Work

It was not the sales manager who hit the bottle that day, but the office worker with the serious drinking problem.

That day the office manager, who was drinking, hit the problem sales worker with a bottle, but it was not serious.

- Vector averaging: $\mathbf{p} = \frac{1}{2}(\mathbf{u} + \mathbf{v})$ (Foltz et al., 1998; Landauer et al., 1997); **syntax insensitive**

Related Work

It was not the sales manager who hit the bottle that day, but the office worker with the serious drinking problem.

That day the office manager, who was drinking, hit the problem sales worker with a bottle, but it was not serious.

- Vector averaging: $\mathbf{p} = \frac{1}{2}(\mathbf{u} + \mathbf{v})$ (Foltz et al., 1998; Landauer et al., 1997); **syntax insensitive**
- Add a neighbor to the sum: $\mathbf{p} = \mathbf{u} + \mathbf{v} + \mathbf{n}$ (Kintsch, 2001); **meaning of predicate depends on its argument**

Related Work

It was not the sales manager who hit the bottle that day, but the office worker with the serious drinking problem.

That day the office manager, who was drinking, hit the problem sales worker with a bottle, but it was not serious.

- Vector averaging: $\mathbf{p} = \frac{1}{2}(\mathbf{u} + \mathbf{v})$ (Foltz et al., 1998; Landauer et al., 1997); **syntax insensitive**
- Add a neighbor to the sum: $\mathbf{p} = \mathbf{u} + \mathbf{v} + \mathbf{n}$ (Kintsch, 2001); **meaning of predicate depends on its argument**
- Tensor products: $\mathbf{p} = \mathbf{u} \otimes \mathbf{v}$ (Smolensky, 1990); **curse of dimensionality**

Related Work

It was not the sales manager who hit the bottle that day, but the office worker with the serious drinking problem.

That day the office manager, who was drinking, hit the problem sales worker with a bottle, but it was not serious.

- Vector averaging: $\mathbf{p} = \frac{1}{2}(\mathbf{u} + \mathbf{v})$ (Foltz et al., 1998; Landauer et al., 1997); **syntax insensitive**
- Add a neighbor to the sum: $\mathbf{p} = \mathbf{u} + \mathbf{v} + \mathbf{n}$ (Kintsch, 2001); **meaning of predicate depends on its argument**
- Tensor products: $\mathbf{p} = \mathbf{u} \otimes \mathbf{v}$ (Smolensky, 1990); **curse of dimensionality**
- Circular convolution: $\mathbf{p} = \mathbf{u} \circledast \mathbf{v}$ (Plate, 1991); **components are randomly distributed**

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, R, K)}$$

A Framework for Semantic Composition

$$\boxed{\color{red}{\mathbf{p}} = f(\mathbf{u}, \mathbf{v}, R, K)}$$

composition of $\mathbf{u}, \mathbf{v}$

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, R, K)}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, R, K)}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

background knowledge

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, R)}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

Assumptions:

1. eliminate background knowledge K

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, \mathit{OBJ})}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

Assumptions:

1. eliminate background knowledge K
2. vary syntactic relationship R

A Framework for Semantic Composition

$$\boxed{\color{red}{\mathbf{p}} = f(\mathbf{u}, \mathbf{v}, \mathit{OBJ})}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

Assumptions:

1. eliminate background knowledge K
2. vary syntactic relationship R
3. $\mathbf{p}$ is in same space as $\mathbf{u}$ and $\mathbf{v}$

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, OBJ)}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

Assumptions:

1. eliminate background knowledge K
2. vary syntactic relationship R
3. $\mathbf{p}$ is in same space as $\mathbf{u}$ and $\mathbf{v}$
4. $f()$ is a linear function of Cartesian product (**additive model**)

A Framework for Semantic Composition

$$\boxed{\mathbf{p} = f(\mathbf{u}, \mathbf{v}, OBJ)}$$

composition of $\mathbf{u}, \mathbf{v}$

syntactic relationship

Assumptions:

1. eliminate background knowledge K
2. vary syntactic relationship R
3. $\mathbf{p}$ is in same space as $\mathbf{u}$ and $\mathbf{v}$
4. $f()$ is a linear function of Cartesian product (**additive model**)
5. $f()$ is a linear function of tensor product (**multiplicative model**)

Models

Additive Models

$$\mathbf{p} = \mathbf{A}\mathbf{u} + \mathbf{B}\mathbf{v}$$

Instances

$$\mathbf{p} = \mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = \mathbf{u} + \mathbf{v} + \sum_i \mathbf{n}_i$$

$$\mathbf{p} = \alpha\mathbf{u} + \beta\mathbf{v}$$

$$\mathbf{p} = \mathbf{v}$$

Models

Additive Models

$\mathbf{p} = \mathbf{Au} + \mathbf{Bv}$

Instances

$\mathbf{p} = \mathbf{u} + \mathbf{v}$

$\mathbf{p} = \mathbf{u} + \mathbf{v} + \sum_i \mathbf{n}_i$

$\mathbf{p} = \alpha\mathbf{u} + \beta\mathbf{v}$

$\mathbf{p} = \mathbf{v}$

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0
problem	2	15	7	9	1

$\textbf{practical} + \textbf{difficulty} = [1\ 14\ 6\ 14\ 4]$

Models

Additive Models

$$\mathbf{p} = \mathbf{Au} + \mathbf{Bv}$$

Instances

$$\mathbf{p} = \mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = \mathbf{u} + \mathbf{v} + \sum_i \mathbf{n}_i$$

$$\mathbf{p} = \alpha\mathbf{u} + \beta\mathbf{v}$$

$$\mathbf{p} = \mathbf{v}$$

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0
problem	2	15	7	9	1

$$\textbf{practical} + \textbf{difficulty} = [1\ 14\ 6\ 14\ 4]$$

$$\textbf{practical} + \textbf{difficulty} + \textbf{problem} = [3\ 29\ 13\ 23\ 5]$$

Models

Additive Models

$$\mathbf{p} = \mathbf{A}\mathbf{u} + \mathbf{B}\mathbf{v}$$

Instances

$$\mathbf{p} = \mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = \mathbf{u} + \mathbf{v} + \sum_i \mathbf{n}_i$$

$$\mathbf{p} = \alpha\mathbf{u} + \beta\mathbf{v}$$

$$\mathbf{p} = \mathbf{v}$$

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0
problem	2	15	7	9	1

$$\textbf{practical} + \textbf{difficulty} = [1\ 14\ 6\ 14\ 4]$$

$$\textbf{practical} + \textbf{difficulty} + \textbf{problem} = [3\ 29\ 13\ 23\ 5]$$

$$0.4 \cdot \textbf{practical} + 0.6 \cdot \textbf{difficulty} = [0.6\ 5.6\ 3.2\ 6.4\ 1.6]$$

Models

Additive Models

$$\mathbf{p} = \mathbf{A}\mathbf{u} + \mathbf{B}\mathbf{v}$$

Instances

$$\mathbf{p} = \mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = \mathbf{u} + \mathbf{v} + \sum_i \mathbf{n}_i$$

$$\mathbf{p} = \alpha\mathbf{u} + \beta\mathbf{v}$$

$$\mathbf{p} = \mathbf{v}$$

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0
problem	2	15	7	9	1

$\mathbf{practical} + \mathbf{difficulty} = [1\ 14\ 6\ 14\ 4]$

$\mathbf{practical} + \mathbf{difficulty} + \mathbf{problem} = [3\ 29\ 13\ 23\ 5]$

$0.4 \cdot \mathbf{practical} + 0.6 \cdot \mathbf{difficulty} = [0.6\ 5.6\ 3.2\ 6.4\ 1.6]$

$\mathbf{difficulty} = [1\ 8\ 4\ 4\ 0]$

Models

Multiplicative Models

$$\mathbf{p} = \mathbf{Cuv}$$

Instances

$$\mathbf{p} = \mathbf{u} \odot \mathbf{v}$$
$$p_i = u_i v_i$$

$$\mathbf{p} = \mathbf{u} \otimes \mathbf{v}$$
$$p_{i,j} = u_i \cdot v_j$$

$$\mathbf{p} = \mathbf{u} \circledast \mathbf{v}$$
$$p_i = \sum_j u_j \cdot v_{i-j}$$

Models

Multiplicative Models

$$\mathbf{p} = \mathbf{Cuv}$$

Instances

$$\mathbf{p} = \mathbf{u} \odot \mathbf{v}$$
$$p_i = u_i v_i$$

$$\mathbf{p} = \mathbf{u} \otimes \mathbf{v}$$
$$p_{i,j} = u_i \cdot v_j$$

$$\mathbf{p} = \mathbf{u} \circledast \mathbf{v}$$
$$p_i = \sum_j u_j \cdot v_{i-j}$$

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0

$$\textbf{practical} \odot \textbf{difficulty} = [0\ 48\ 8\ \ 40\ 0]$$

Models

Multiplicative Models

$\mathbf{p} = \mathbf{C}\mathbf{u}\mathbf{v}$

Instances

$\mathbf{p} = \mathbf{u} \odot \mathbf{v}$

$p_i = u_i v_i$

$\mathbf{p} = \mathbf{u} \otimes \mathbf{v}$

$p_{i,j} = u_i \cdot v_j$

$\mathbf{p} = \mathbf{u} \circledast \mathbf{v}$

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

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0

$\textbf{practical} \odot \textbf{difficulty} = [0\ 48\ 8\ \ 40\ 0]$

$$\textbf{practical} \otimes \textbf{difficulty} = \begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 6 & 48 & 24 & 24 & 0 \\ 2 & 16 & 8 & 8 & 0 \\ 10 & 80 & 40 & 40 & 0 \\ 4 & 32 & 16 & 16 & 0 \end{matrix}$$

Models

Multiplicative Models

$$\mathbf{p} = \mathbf{Cuv}$$

Instances

$$\mathbf{p} = \mathbf{u} \odot \mathbf{v}$$
$$p_i = u_i v_i$$

$$\mathbf{p} = \mathbf{u} \otimes \mathbf{v}$$
$$p_{i,j} = u_i \cdot v_j$$

$$\mathbf{p} = \mathbf{u} \circledast \mathbf{v}$$
$$p_i = \sum_j u_j \cdot v_{i-j}$$

	music	solution	economy	craft	create
practical	0	6	2	10	4
difficulty	1	8	4	4	0

$$\textbf{practical} \odot \textbf{difficulty} = [0\ 48\ 8\ \ 40\ 0]$$

$$\textbf{practical} \otimes \textbf{difficulty} = \begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ 6 & 48 & 24 & 24 & 0 \\ 2 & 16 & 8 & 8 & 0 \\ 10 & 80 & 40 & 40 & 0 \\ 4 & 32 & 16 & 16 & 0 \end{array}$$

$$\textbf{practical} \circledast \textbf{difficulty} = [116\ 50\ 66\ 62\ 80]$$

Models

Dilation Models

$$\mathbf{p} = \mathbf{Cuv} = \mathbf{Uv}$$
$$U_{ij} = 0, U_{ii} = u_i$$

$$\mathbf{x} = \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} \qquad \mathbf{y} = \mathbf{v} - \mathbf{x} = \mathbf{v} - \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u}$$

$$\mathbf{v}' = \lambda\mathbf{x} + \mathbf{y} = (\lambda - \mathbf{1})\frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = (\lambda - \mathbf{1})(\mathbf{u}\cdot\mathbf{v})\mathbf{u} + (\mathbf{u}\cdot\mathbf{u})\mathbf{v}$$

Models

Dilation Models

$$\mathbf{p} = \mathbf{C}\mathbf{u}\mathbf{v} = \mathbf{U}\mathbf{v}$$
$$U_{ij} = 0, U_{ii} = u_i$$

$$\mathbf{x} = \frac{\mathbf{u} \cdot \mathbf{v}}{\mathbf{u} \cdot \mathbf{u}}\mathbf{u} \qquad \mathbf{y} = \mathbf{v} - \mathbf{x} = \mathbf{v} - \frac{\mathbf{u} \cdot \mathbf{v}}{\mathbf{u} \cdot \mathbf{u}}\mathbf{u}$$

$$\mathbf{v}' = \lambda\mathbf{x} + \mathbf{y} = (\lambda - \mathbf{1})\frac{\mathbf{u} \cdot \mathbf{v}}{\mathbf{u} \cdot \mathbf{u}}\mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = (\lambda - \mathbf{1})(\mathbf{u} \cdot \mathbf{v})\mathbf{u} + (\mathbf{u} \cdot \mathbf{u})\mathbf{v}$$

Models

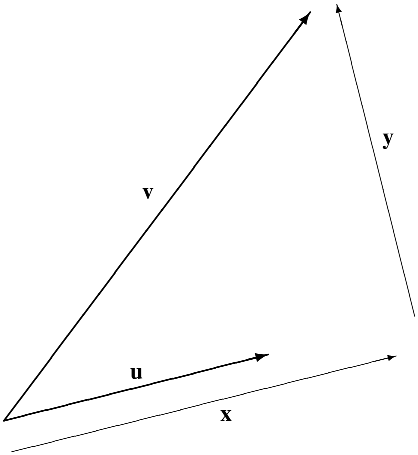
Dilation Models

$$\mathbf{p} = \mathbf{Cuv} = \mathbf{Uv}$$
$$U_{ij} = 0, U_{ii} = u_i$$

$$\mathbf{x} = \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} \qquad \mathbf{y} = \mathbf{v} - \mathbf{x} = \mathbf{v} - \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u}$$

$$\mathbf{v}' = \lambda\mathbf{x} + \mathbf{y} = (\lambda - 1)\frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = (\lambda - 1)(\mathbf{u}\cdot\mathbf{v})\mathbf{u} + (\mathbf{u}\cdot\mathbf{u})\mathbf{v}$$



Models

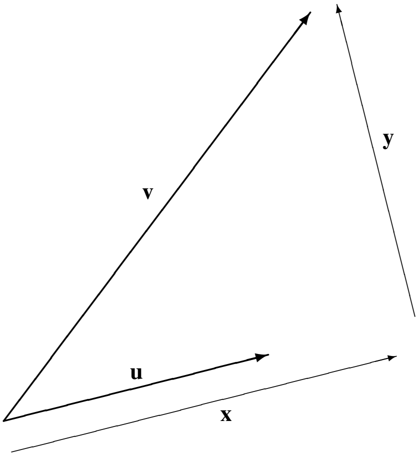
Dilation Models

$$\mathbf{p} = \mathbf{C}\mathbf{u}\mathbf{v} = \mathbf{U}\mathbf{v}$$
$$U_{ij} = 0, U_{ii} = u_i$$

$$\mathbf{x} = \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} \qquad \mathbf{y} = \mathbf{v} - \mathbf{x} = \mathbf{v} - \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u}$$

$$\mathbf{v}' = \lambda\mathbf{x} + \mathbf{y} = (\lambda - 1)\frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = (\lambda - 1)(\mathbf{u}\cdot\mathbf{v})\mathbf{u} + (\mathbf{u}\cdot\mathbf{u})\mathbf{v}$$



Models

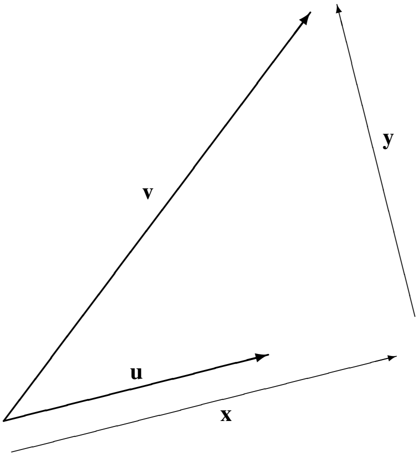
Dilation Models

$$\mathbf{p} = \mathbf{C}\mathbf{u}\mathbf{v} = \mathbf{U}\mathbf{v}$$
$$U_{ij} = 0, U_{ii} = u_i$$

$$\mathbf{x} = \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} \qquad \mathbf{y} = \mathbf{v} - \mathbf{x} = \mathbf{v} - \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u}$$

$$\mathbf{v}' = \lambda\mathbf{x} + \mathbf{y} = (\lambda - 1)\frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} + \mathbf{v}$$

$$\mathbf{p} = (\lambda - 1)(\mathbf{u}\cdot\mathbf{v})\mathbf{u} + (\mathbf{u}\cdot\mathbf{u})\mathbf{v}$$



Phrase Similarity Task

Originally proposed in Kintsch (2002):

- Elicit similarity judgments for adjective-noun, noun-noun, verb-object combinations.
- Phrase pairs from three bands: High, Medium, Low.
- Compute vectors for phrases, measure their similarity.
- Correlate model similarities with human ratings.

Phrase Similarity Task

Originally proposed in Kintsch (2002):

- Elicit similarity judgments for adjective-noun, noun-noun, verb-object combinations.
- Phrase pairs from three bands: High, Medium, Low.
- Compute vectors for phrases, measure their similarity.
- Correlate model similarities with human ratings.

	High	**Medium**	**Low**
old person			
kitchen door			
produce effect			

Phrase Similarity Task

Originally proposed in Kintsch (2002):

- Elicit similarity judgments for adjective-noun, noun-noun, verb-object combinations.
- Phrase pairs from three bands: High, Medium, Low.
- Compute vectors for phrases, measure their similarity.
- Correlate model similarities with human ratings.

	High	**Medium**	**Low**
old person	elderly lady	right hand	small house
kitchen door			
produce effect			

Phrase Similarity Task

Originally proposed in Kintsch (2002):

- Elicit similarity judgments for adjective-noun, noun-noun, verb-object combinations.
- Phrase pairs from three bands: High, Medium, Low.
- Compute vectors for phrases, measure their similarity.
- Correlate model similarities with human ratings.

	High	**Medium**	**Low**
old person	elderly lady	right hand	small house
kitchen door	bedroom window	office worker	housing department
produce effect			

Phrase Similarity Task

Originally proposed in Kintsch (2002):

- Elicit similarity judgments for adjective-noun, noun-noun, verb-object combinations.
- Phrase pairs from three bands: High, Medium, Low.
- Compute vectors for phrases, measure their similarity.
- Correlate model similarities with human ratings.

	High	**Medium**	**Low**
old person	elderly lady	right hand	small house
kitchen door	bedroom window	office worker	housing department
produce effect	achieve result	consider matter	start work

Experimental Setup

Similarity Ratings

- 36 pairs (adj-noun, noun-noun, verb-noun) $\times$ 3 bands (324 pairs in total, created automatically, substitutability test)
- Ratings collected using Webexp (90 participants)
- Participants use 7-point similarity scale

Semantic Space

- Compare simple semantic against LDA topic model (Blei et al. 2003)
- 2000 dimensions vs 100 topics, using cosine similarity measure
- Parameters for composition models tuned on dev set

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

- Multiplicative and dilation models best for simple space

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

- Multiplicative and dilation models best for simple space

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

- Multiplicative and dilation models best for simple space
- Dilation and Additive models best for LDA model

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

- Multiplicative and dilation models best for simple space
- Dilation and Additive models best for LDA model

Results (for verb-obj)

Model	Simple	LDA
Additive	0.30	0.40
Kintsch	0.23	0.31
Weighted Additive	0.34	0.40
Multiplicative	0.37	0.34
Tensor Product	0.33	0.33
Circular Convolution	0.10	0.12
Dilation	0.38	0.41
Head Only	0.24	0.17
Humans	0.55	

- Multiplicative and dilation models best for simple space
- Dilation and Additive models best for LDA model
- Circular convolution is worst performing model

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations
- **What are composition models good for?**

Modeling Brain Activity

Tom Mitchell and collaborators
Wang et al., 2003; Mitchell et al., 2004; Mitchell et al., 2008; Hutchinson et al., 2009; Chang et al., 2009; Rustandi, 2009

- Can we observe differences in neural activity as people think about different concepts?
- Can we use vector-based models to explain observed neural activity?

Functional MRI

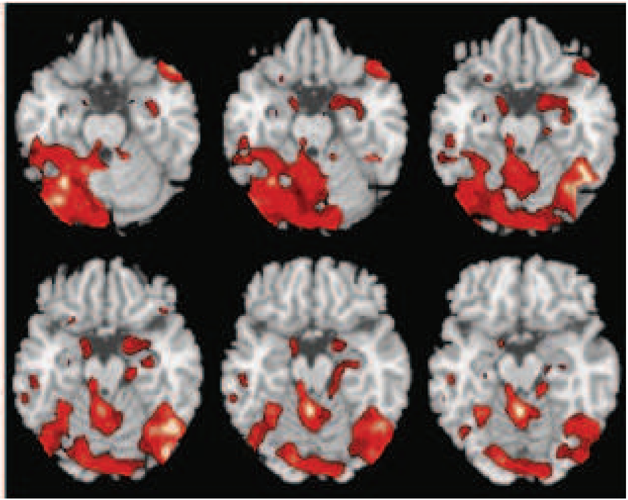
Functional MRI

Monitors brain activity when people comprehend words or phrases.

Functional MRI

Monitors brain activity when people comprehend words or phrases.
Measures changes related to blood flow and blood oxygenation.

Functional MRI



soft bear

strong dog

Chang et al. (ACL, 2009)

- Participants see adjective-noun phrases
- Adjectives emphasize semantic properties of nouns
- Use vector-based models to account for variance in neural activity.
- Train regression model to fit activation profile of stimuli
- Multiplicative model outperforms non-compositional and additive model.

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations
- **What are composition models good for?**

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations
- **What are composition models good for?**
  - modeling brain activity

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations
- **What are composition models good for?**
  - modeling brain activity
  - sentential priming, inductive inference

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations
- **What are composition models good for?**
  - modeling brain activity
  - sentential priming, inductive inference
  - textual entailment, information retrieval, language modeling

Interim Summary

- General framework of semantic composition
- Different composition functions appropriate for different representations (additive vs. multiplicative)
- Dilation models overall best, syntax sensitive, parametric
- Results generalize to noun-noun, adj-noun, verb-obj combinations
- **What are composition models good for?**
    - modeling brain activity
    - sentential priming, inductive inference
    - textual entailment, information retrieval, **language modeling**

Language Modeling

What is the next word?

Language Modeling

What is the next word?

He is now president and chief operating

Language Modeling

What is the next word?

He is now president and chief operating

'chief operating' is followed by 'officer' 99% of the time.

Language Modeling

What is the next word?

He is now president and chief operating officer

'chief operating' is followed by 'officer' 99% of the time.

Language Modeling

What is the next word?

He is now president and chief operating officer **of the**

'of the' is very frequent but not very predictive.

Language Modeling

What is the next word?

He is now president and chief operating officer **of the**

'of the' is very frequent but not very predictive.

Language Modeling

What is the next word?

He is now president and chief operating officer **of the**

Prior content indicative of domain the vocabulary is drawn from.

Language Modeling

What is the next word?

> He is now **president** and **chief operating officer** of the

Prior content indicative of domain the vocabulary is drawn from.

Language Modeling

What is the next word?

He is now president and chief operating officer of the company.

Given semantic representations for ‘president’, ‘chief’, ‘operating’ and ‘officer’ how do we combine them to make the most predictive representation of this history?

Language Modeling

- What about using this vector composition in a language model?
- Semantic space models already used for language modeling: Bellegarda (2000), Coccaro & Jurafsky (1998), Gildea & Hofmann (1999)
- All make assumptions about how to combine vectors, without empirical investigation.
- Focus on multiplicative and additive models.

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$p(company|president, chief, operating, officer)$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$$p(company|president, chief, operating, officer)$$

$$p(w|h) = sim(w, h)$$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$$p(company|president, chief, operating, officer)$$

$$p(w|h) = sim(w, h)$$

$$sim(w, h) \propto \mathbf{w} \cdot \mathbf{h}$$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$p(company|president, chief, operating, officer)$

$p(w|h) = sim(w, h)$

$sim(w, h) \propto \mathbf{w} \cdot \mathbf{h} = \sum w_i h_i$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$$p(company|president, chief, operating, officer)$$

$$p(w|h) = sim(w, h)$$

$$sim(w, h) \propto \mathbf{w} \cdot \mathbf{h} = \sum \frac{p(c_i|w)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)}$$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$$p(company|president, chief, operating, officer)$$

$$p(w|h) = p(w) \sum_i \frac{p(c_i|w)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$p(company|president, chief, operating, officer)$

$p(w|h) = \sum_i p(w|c_i)p(c_i|h)$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$p(company|president, chief, operating, officer)$

$p(w|h) = \sum_i p(w|c_i)p(c_i|h)$

$\mathbf{h}_n = f(\mathbf{w}_n, \mathbf{h}_{n-1})$

A Language Model Based on Vector Composition

He is now president and chief operating officer of the company

$p(company|president, chief, operating, officer)$

$p(w|h) = \sum_i p(w|c_i)p(c_i|h)$

$\mathbf{h}_n = f(\mathbf{w}_n, \mathbf{h}_{n-1})$
$\mathbf{h}_1 = \mathbf{w}_1$

Experimental Setup

- BLLIP Corpus
  - Training set - 38M words
  - Development set - 50K words
  - Test set - 50K words
- Numbers replaced with <NUM>
- Vocabulary of 20K word types
- Others replaced with <UNK>
- Perplexity of model predictions on test set
- Compare simple semantic space against LDA topic model

Perplexities

Model	Perplexity
Unigram	698
Additive	628
Multiplicative	540

Perplexities

Model	Perplexity
Unigram	698
Additive	628
Multiplicative	540

- Both models improve over ignoring history.

Perplexities

Model	Perplexity
Unigram	698
Additive	628
Multiplicative	540

- Both models improve over ignoring history.
- Multiplicative is better than additive.

Perplexities

Model	Perplexity
Unigram	698
Additive	628
Multiplicative	540

- Both models improve over ignoring history.
- Multiplicative is better than additive.
- But neither model is particularly effective.

Integrating with an Ngram model

Linear interpolation

- $\lambda p_1(w) + (1 - \lambda)p_2(w)$
- But this will be most effective when models comparable in predictiveness.

Integrating with an Ngram model

Linear interpolation

- $\lambda p_1(w) + (1 - \lambda)p_2(w)$
- But this will be most effective when models comparable in predictiveness.

Modify $p(w|h)$

Integrating with an Ngram model

Linear interpolation

- $\lambda p_1(w) + (1 - \lambda) p_2(w)$
- But this will be most effective when models comparable in predictiveness.

Modify $p(w|h)$

- $p(w_n) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$

Integrating with an Ngram model

Linear interpolation

- $\lambda p_1(w) + (1-\lambda)p_2(w)$
- But this will be most effective when models comparable in predictiveness.

Modify $p(w|h)$

- $p(w_n) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$
- $p(w_n|w_{n-1}, w_{n-2}) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$

Integrating with an Ngram model

Linear interpolation

- $\lambda p_1(w) + (1 - \lambda)p_2(w)$
- But this will be most effective when models comparable in predictiveness.

Modify $p(w|h)$

- $p(w_n) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$
- $p(w_n|w_{n-1}, w_{n-2}) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$

Integrating with an Ngram model

Linear interpolation

- $\lambda p_1(w) + (1-\lambda)p_2(w)$
- But this will be most effective when models comparable in predictiveness.

Modify $p(w|h)$

- $p(w_n) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$
- $p(w_n|w_{n-1}, w_{n-2}) \sum \frac{p(c_i|w_n)}{p(c_i)} \frac{p(c_i|h)}{p(c_i)} p(c_i)$

Perplexities

	Simple
Trigram	78.72
Additive	76.66
Multiplicative	75.01

Perplexities

	Simple
Trigram	78.72
Additive	76.66
Multiplicative	75.01

- Simple models improve over Trigram, multiplicative better.

Perplexities

	Simple	LDA
Trigram	78.72	78.72
Additive	76.66	76.60
Multiplicative	75.01	123.95

- Simple models improve over Trigram, multiplicative better.
- LDA improves over trigram for additive composition function.

Perplexities

	Simple	LDA
Trigram	78.72	78.72
Additive	76.66	76.60
Multiplicative	75.01	123.95

- Simple models improve over Trigram, multiplicative better.
- LDA improves over trigram for additive composition function.
- But deteriorates with multiplicative composition function.
  - LDA produces very sparse vectors.
  - Multiplication tends to increase sparsity.

Perplexities

	Simple	LDA
Trigram	78.72	78.72
Additive	76.66	76.60
Multiplicative	75.01	123.95

- Simple models improve over Trigram, multiplicative better.
- LDA improves over trigram for additive composition function.
- But deteriorates with multiplicative composition function.
  - LDA produces very sparse vectors.
  - Multiplication tends to increase sparsity.
- Simple semantic space produces best results overall.

Comparison to Parsing

- Our model incorporates long range semantic dependencies into a trigram model.
- Increases the probability of upcoming words which are semantically similar to the history.
- How about other methods of handling long range dependencies?
  - Language models based on syntactic structure.
  - Incorporate Roark's (2001) parser using linear interpolation.

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

- Parser yields reductions comparable to semantic model

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

- Parser yields reductions comparable to semantic model
- Incorporating both leads to further reductions

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

- Parser yields reductions comparable to semantic model
- Incorporating both leads to further reductions
- Syntactic and semantic models use different long range dependencies

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

- Parser yields reductions comparable to semantic model
- Incorporating both leads to further reductions
- Syntactic and semantic models use different long range dependencies
- LDA results follow similar pattern (additive is best)

Conclusions

Work so far

- Considered vector composition for phrase similarity and language modeling
- Compared a simple semantic space to LDA
- Different composition functions appropriate for each model
- Semantic dependencies complementary to syntactic ones

Future work

- Incorporate syntax into composition
- Optimize vectors and composition function on specific tasks

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

	LDA	Parser
Trigram	78.72	75.22
Additive	76.60	71.58
Multiplicative	123.93	87.93

Perplexities

	Simple	Parser
Trigram	78.72	75.22
Additive	76.66	73.45
Multiplicative	75.01	71.32

	LDA	Parser
Trigram	78.72	75.22
Additive	76.60	71.58
Multiplicative	123.93	87.93

LDA Topics