Word Representations via Gaussian Embedding

Luke Vilnis
Andrew McCallum
University of Massachusetts Amherst

Vector word embeddings

- **Low-Level NLP** [Turian et al. 2010, Collobert et al. 2011]
- **Named Entity Extraction** [Passos et al. 2014]
- **Machine Translation** [Kalchbrenner & Blunsom 2013, Cho et al. 2014]
- **Question Answering** [Weston et al. 2015]
- …

Vector word embeddings

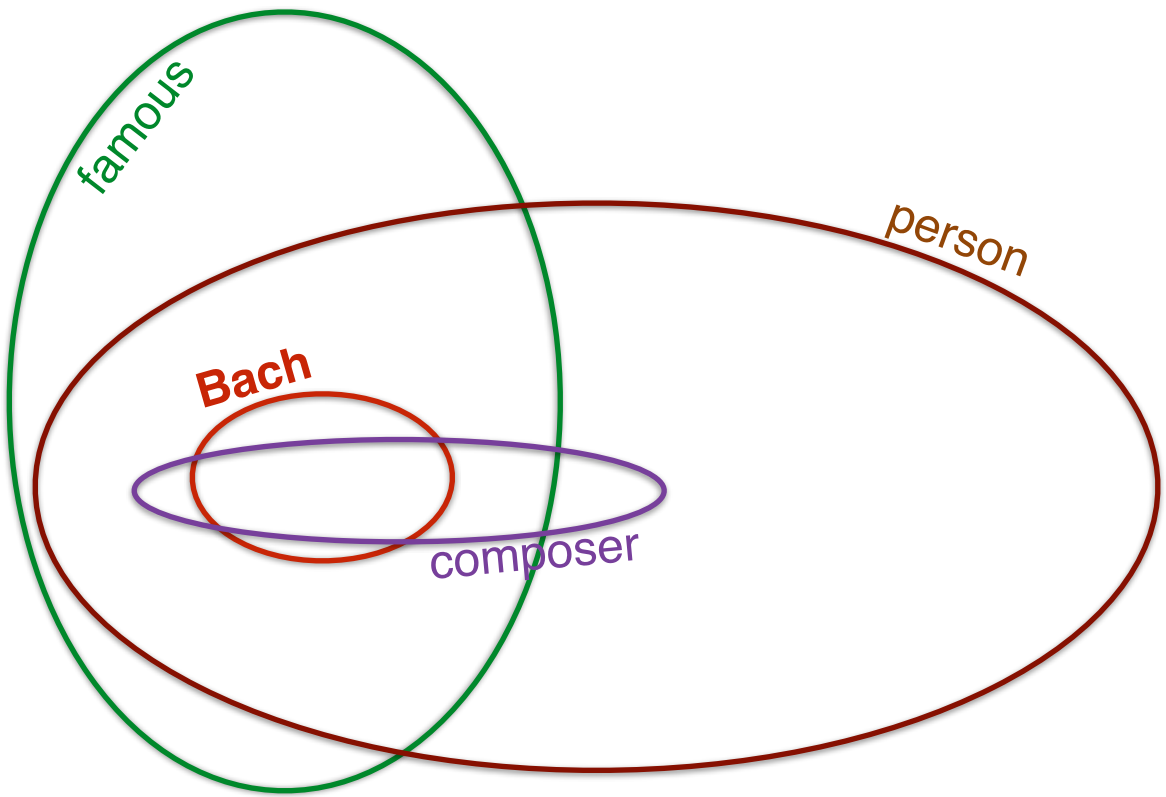
What's missing?

- Breadth
- Asymmetry

Gaussian word embeddings

Advantages

- Breadth
- Asymmetry



Gaussian word embeddings

Advantages

- Breadth
- Asymmetry

for each word i

v_i

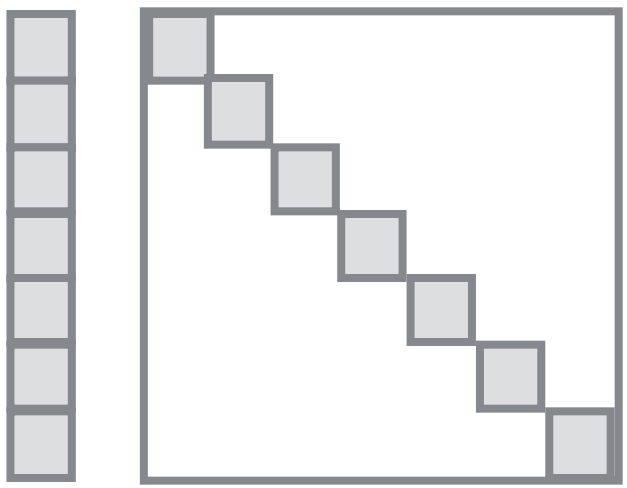
Gaussian word embeddings

Advantages

- Breadth
- Asymmetry

for each word i

$$\mathcal{N}(x; \mu_i, \Sigma_i)$$



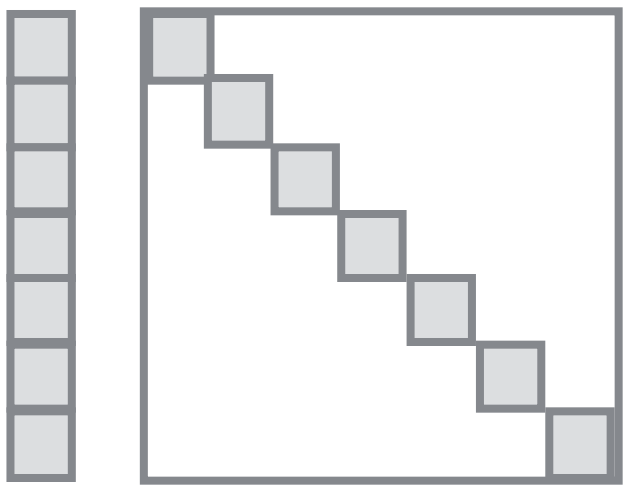
Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- Asymmetry

for each word i

$$\mathcal{N}(x; \mu_i, \Sigma_i)$$



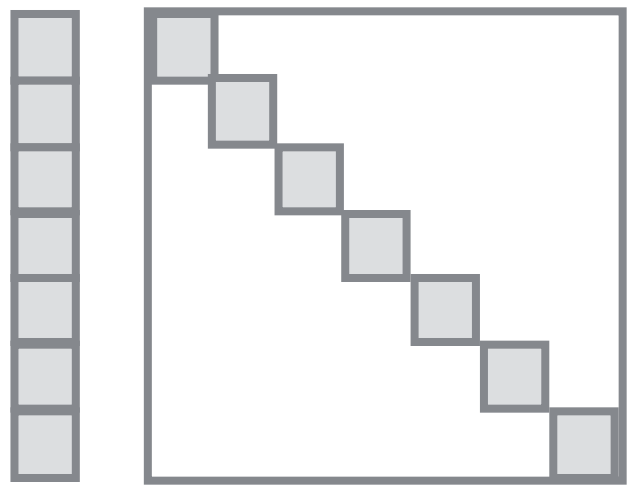
Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- Asymmetry

for each word i

$$\mathcal{N}(x; \mu_i, \Sigma_i) \quad \propto \quad -\log\det(\Sigma_i) - (\mu_i - x)^\top \Sigma_i^{-1} (\mu_i - x)$$



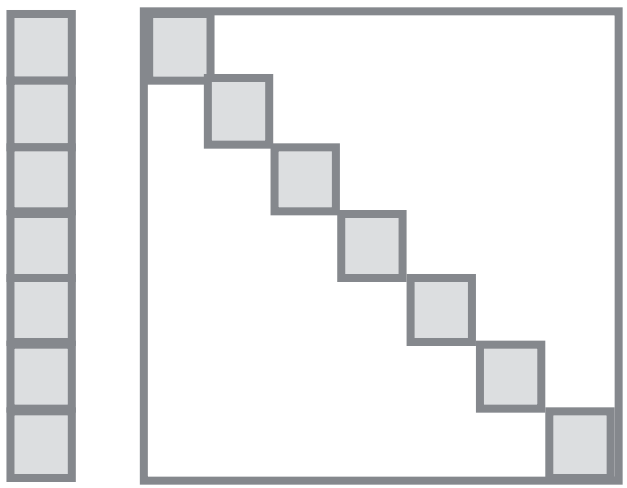
Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- Asymmetry

for each word i

$$\mathcal{N}(x; \mu_i, \Sigma_i) \quad \propto \quad \underbrace{-\log\det(\Sigma_i)}_{\text{logarithmic penalty on volume due to normalization}} \overbrace{-(\mu_i - x)^\top \Sigma_i^{-1}(\mu_i - x)}^{\text{Mahalanobis distance measured by } \boldsymbol{\Sigma}_i}$$



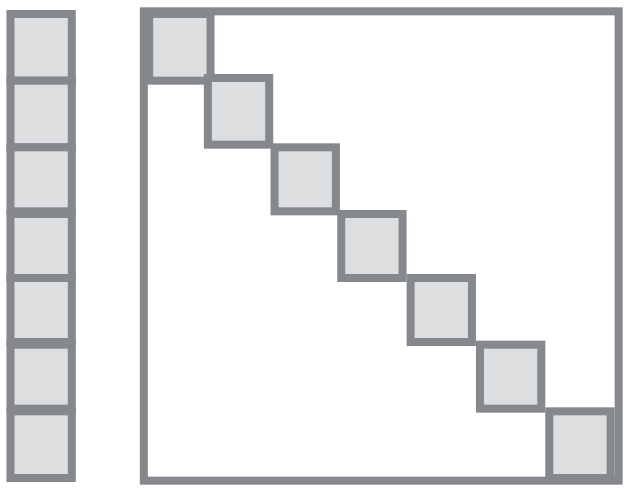
Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- **Asymmetry:** KL-divergence

for each word i

$$\mathcal{N}(x; \mu_i, \Sigma_i) \quad \propto \quad \underbrace{-\log\det(\Sigma_i)}_{\text{logarithmic penalty on volume due to normalization}} - \overbrace{(\mu_i - x)^\top \Sigma_i^{-1} (\mu_i - x)}^{\text{Mahalanobis distance measured by } \boldsymbol{\Sigma}_i}$$



Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- **Asymmetry:** KL-divergence

$$KL(\mathcal{N}_i \| \mathcal{N}_j) =$$

$$\int_x \mathcal{N}(x; \mu_i, \Sigma_i) \log \frac{\mathcal{N}(x; \mu_i, \Sigma_i)}{\mathcal{N}(x; \mu_j, \Sigma_j)} dx$$

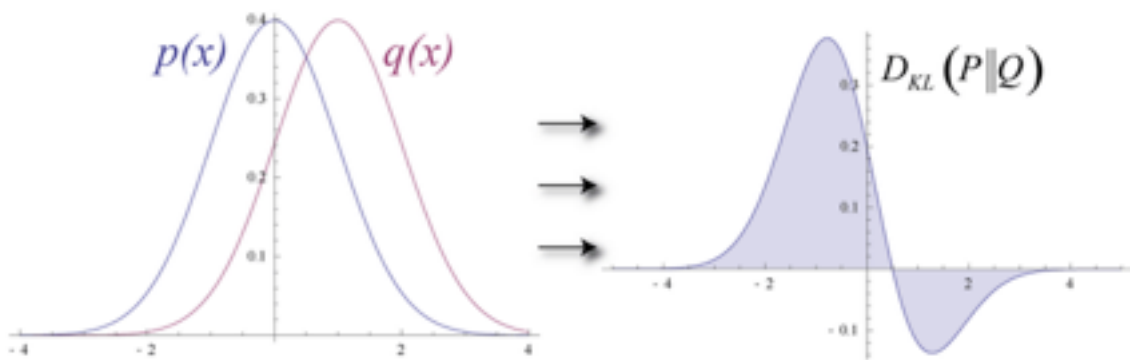
Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- **Asymmetry:** KL-divergence

$$KL(\mathcal{N}_i \| \mathcal{N}_j) =$$

$$\int_x \mathcal{N}(x; \mu_i, \Sigma_i) \log \frac{\mathcal{N}(x; \mu_i, \Sigma_i)}{\mathcal{N}(x; \mu_j, \Sigma_j)} dx$$



[Wikipedia]

Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- **Asymmetry:** KL-divergence

$$KL(\mathcal{N}_i \| \mathcal{N}_j) \propto$$

$$-\mathrm{tr}(\Sigma_i^{-1}\Sigma_j) - (\mu_i - \mu_j)^\top \Sigma_i^{-1}(\mu_i - \mu_j) - \log \frac{\det(\Sigma_i)}{\det(\Sigma_j)}$$

Gaussian word embeddings

Advantages

- **Breadth:** covariance matrix
- **Asymmetry:** KL-divergence

$$KL(\mathcal{N}_i \| \mathcal{N}_j) \propto$$

$$\underbrace{-\mathrm{tr}(\Sigma_i^{-1}\Sigma_j)}_{\text{directions of variance should be aligned, i should be “large” and j “small”}} \underbrace{-(\mu_i - \mu_j)^\top \Sigma_i^{-1}(\mu_i - \mu_j)}_{\text{distance between means is “small” as measured by i}} \underbrace{-\log \frac{\det(\Sigma_i)}{\det(\Sigma_j)}}_{\text{logarithmic penalty on volume due to normalization}}$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …

$$E(\text{word}_i, \text{word}_j) = \langle v_i, v_j \rangle$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …

(**composer**, musician)

$$E(\text{word}_i, \text{word}_j) = \langle v_i, v_j \rangle$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …

(**composer**, musician)

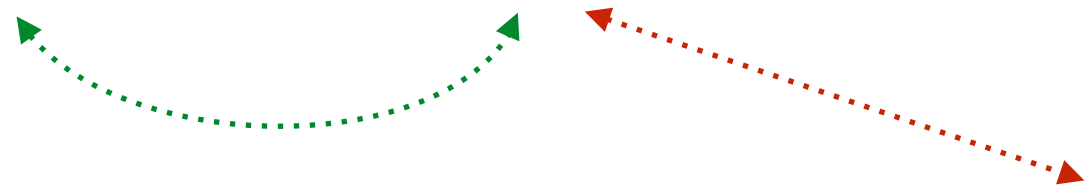
(**composer**, random dictionary word)

$$E(\text{word}_i, \text{word}_j) = \langle v_i, v_j \rangle$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …



(**composer**, musician)

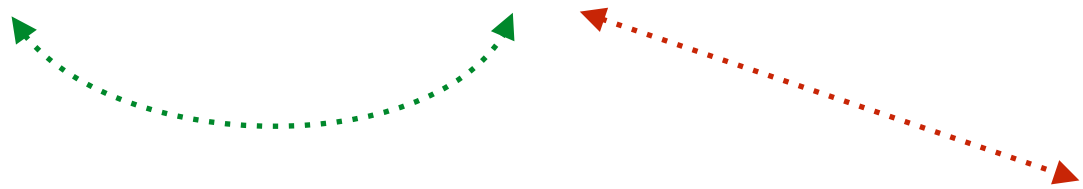
(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \langle v_i, v_j \rangle$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …



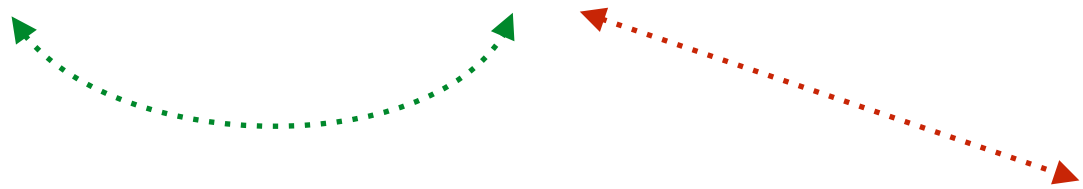
E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \langle v_i, v_j \rangle$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …



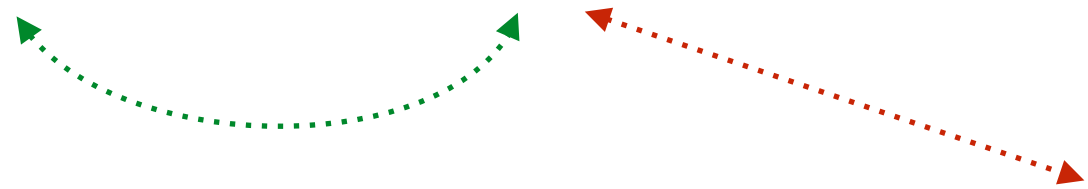
E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \sum_k v_i^{(k)} v_j^{(k)}$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …



E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_k v_i(k) v_j(k) dk$$

Learning vector embeddings

e.g. [Mikolov et al. 2013]

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_k v_i(k) v_j(k) dk$$

Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_k v_i(k) v_j(k) dk$$

Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_x \mathcal{N}(x; \mu_i, \Sigma_i)\mathcal{N}(x; \mu_j, \Sigma_j)dx$$

[**PPK**, Jebara et al. 2003]

Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_x \mathcal{N}(x; \mu_i, \Sigma_i)\mathcal{N}(x; \mu_j, \Sigma_j)dx$$

$$= \mathcal{N}(0; \mu_i - \mu_j, \Sigma_i + \Sigma_j)$$

[**PPK**, Jebara et al. 2003]

Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_x \mathcal{N}(x; \mu_i, \Sigma_i)\mathcal{N}(x; \mu_j, \Sigma_j)dx$$

$$= \mathcal{N}(0; \mu_i - \mu_j, \Sigma_i + \Sigma_j)$$

$$\propto -\log\det(\Sigma_i + \Sigma_j) - (\mu_i - \mu_j)^\top(\Sigma_i + \Sigma_j)^{-1}(\mu_i - \mu_j)$$

[**PPK**, Jebara et al. 2003]

Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_x \mathcal{N}(x; \mu_i, \Sigma_i)\mathcal{N}(x; \mu_j, \Sigma_j)dx$$

$$= \mathcal{N}(0; \mu_i - \mu_j, \Sigma_i + \Sigma_j)$$

[**PPK**, Jebara et al. 2003]

$$\propto \underbrace{-\log\det(\Sigma_i + \Sigma_j)}_{\text{log-volume of ellipse}} \underbrace{-(\mu_i - \mu_j)^\top(\Sigma_i + \Sigma_j)^{-1}(\mu_i - \mu_j)}_{\text{Mahalanobis distance between means}}$$

Learning Gaussian embeddings

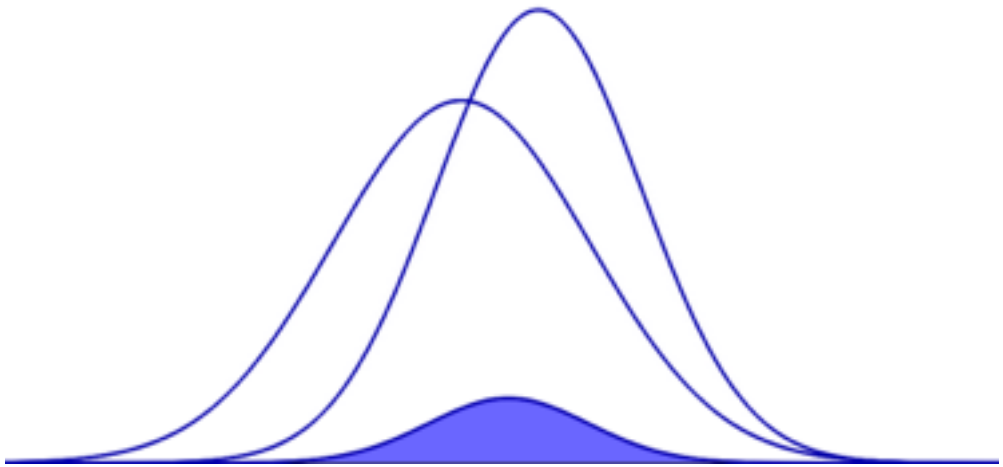
… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$E(\text{word}_i, \text{word}_j) = \int_x \mathcal{N}(x; \mu_i, \Sigma_i)\mathcal{N}(x; \mu_j, \Sigma_j)dx$$

[**PPK**, Jebara et al. 2003]

$$= \mathcal{N}(0; \mu_i - \mu_j, \Sigma_i + \Sigma_j)$$



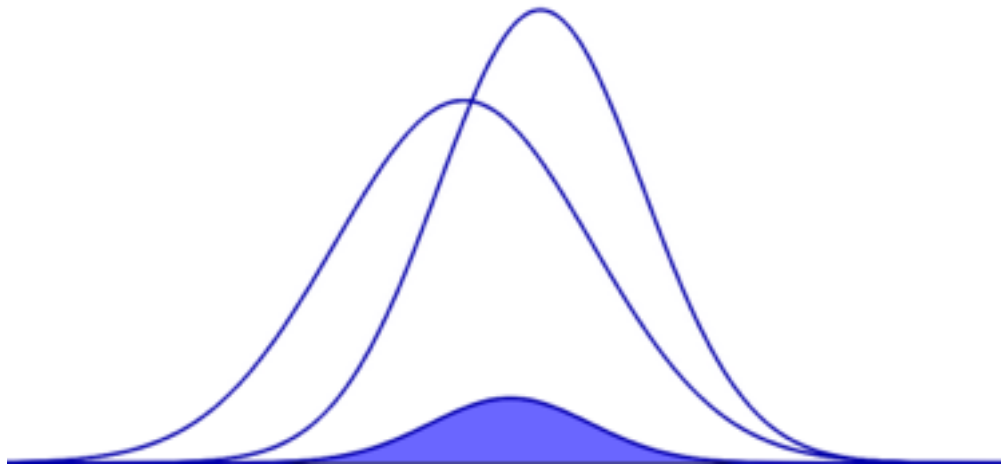
Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$\text{Loss}_{\text{PPK}}(w, c_{pos}, c_{neg}) =$$

$$\max(0, m - E_{\text{PPK}}(w, c_{pos}) + E_{\text{PPK}}(w, c_{neg}))$$



Learning Gaussian embeddings

… German musician and **composer** of the Baroque …

E(**composer**, musician) > E(**composer**, banana)

$$\text{Loss}_{\text{KL}}(w, c_{pos}, c_{neg}) =$$

$$\max(0, m + \text{KL}(c_{pos} \| w) - \text{KL}(c_{neg} \| w))$$

(asymmetric supervision)

Related work

- Asymmetric, sparse, distributional [Baroni et al. 2012]
- Dense can be better [Baroni et al. 2014]
- Symmetric, dense [Bengio et al. 2003, Mikolov et al. 2013, many others]
- Bayesian matrix factorization [Salakhutdinov & Mnih 2008]
- (Mixture) density networks [Bishop 1994]
- Gaussian process neural nets [Damianou & Lawrence 2013]

Experimental results

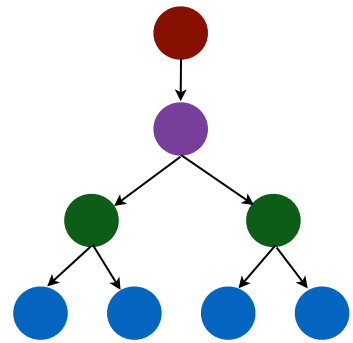
- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

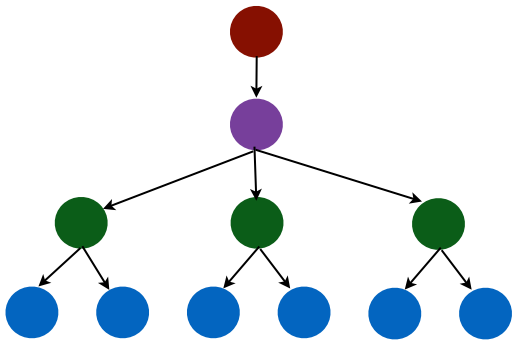
Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Synthetic hierarchy

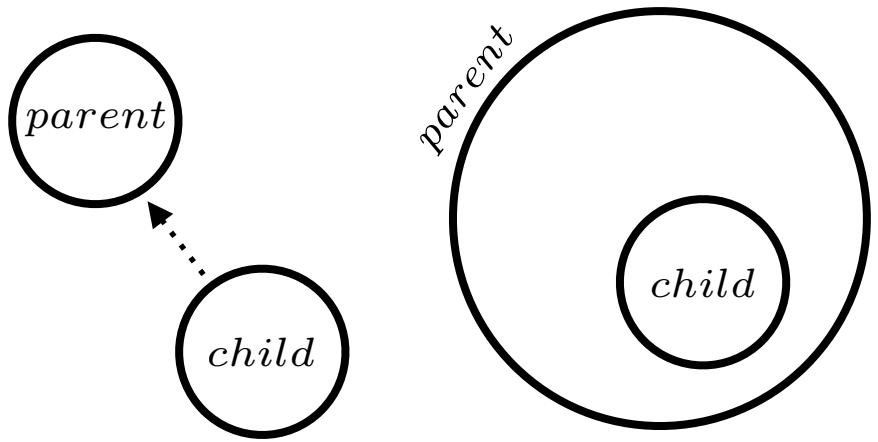
Train data





Objective

$$child \vdash parent$$

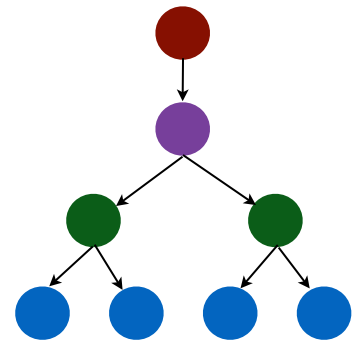


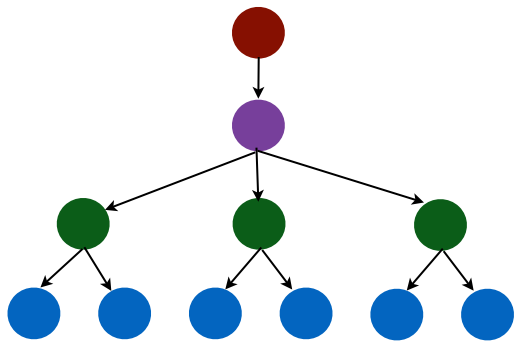
$$\mathrm{KL}(v_{child} \| v_{parent})$$

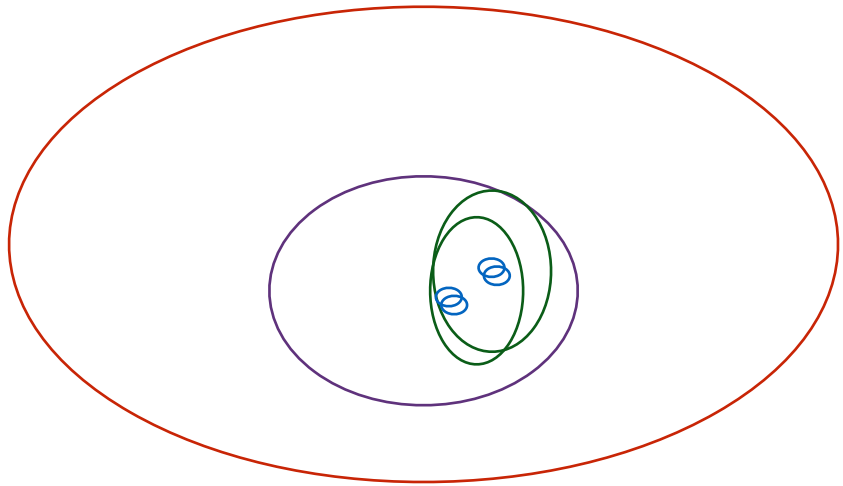
Synthetic hierarchy

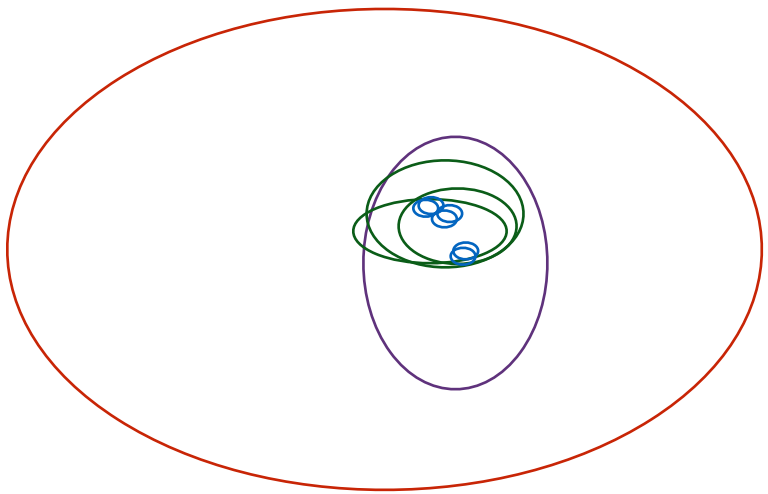
Train data

Learned model









KL objective accurately learns all containments

Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Entailment

Binary labeled dataset of entailment pairs [Baroni et al. 2012]

adrenaline is-a neurotransmitter
archbishop is-a clergyman
horse is-a mammal (+)
pizza is-a food

Entailment

Binary labeled dataset of entailment pairs [Baroni et al. 2012]

adrenaline is-a neurotransmitter
archbishop is-a clergyman
horse is-a mammal (+)
pizza is-a food

aircrew is-not-a playlist
bamboo is-not-a bear (-) no relation

Entailment

Binary labeled dataset of entailment pairs [Baroni et al. 2012]

adrenaline is-a neurotransmitter
archbishop is-a clergyman
horse is-a mammal (+)
pizza is-a food

aircrew is-not-a playlist
bamboo is-not-a bear (-) no relation

food is-not-a pizza
molecule is-not-a carbohydrate (-) reversed
gathering is-not-a seminar

Entailment

- **Model:** diagonal (D) and spherical (S) variances
- **Train:** ~1b tokens Wikipedia + 3b tokens of newswire
- **Evaluate:** optimal F1 operating point, average precision

Entailment

- **Model:** diagonal (D) and spherical (S) variances
- **Train:** ~1b tokens Wikipedia + 3b tokens of newswire
- **Evaluate:** optimal F1 operating point, average precision

Model	Test	Similarity	Best F1	AP
Baroni et al. (2012)	E	balAPinc	**75.1**	–
Learned (D)	E	KL	79.01	**.80**
Learned (S)	E	KL	**79.34**	.78

Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Symmetric word similarity

- Word similarity tasks (e.g. WordSim-353)

(money, bank, 8.5)
(psychology, Freud, 8.21)
(media, radio, 7.42)
(drug, abuse, 6.85)
(Mars, scientist, 5.63)
(cup, object, 3.69)
(professor, cucumber, 0.31)

- **Evaluate:** Spearman's ρ

Symmetric word similarity

Dataset	Vector	Spherical Gaussian		Diagonal Gaussian	
	SG (100d)	LG/50/m/S	LG/50/d/S	LG/50/m/D	LG/50/d/D
SimLex	31.13	**32.23**	29.84	**31.25**	30.50
WordSim	59.33	**65.49**	62.03	**62.12**	61.00
WordSim-S	70.19	**76.15**	73.92	**74.64**	72.79
WordSim-R	54.64	**58.96**	54.37	**54.44**	53.36
MEN	70.70	**71.31**	69.65	**71.30**	70.18
MC	66.76	**70.41**	69.17	67.01	**68.50**
RG	69.38	71.00	**74.76**	70.41	**77.00**
YP	35.76	41.50	**42.55**	36.05	**39.30**
Rel-122	51.26	**53.74**	51.09	52.28	**53.54**
Average	56.57	**60.09**	58.60	57.72	**58.46**

skip-gram

sphere, μ

sphere, μ, Σ

diagonal, μ

diagonal, μ, Σ

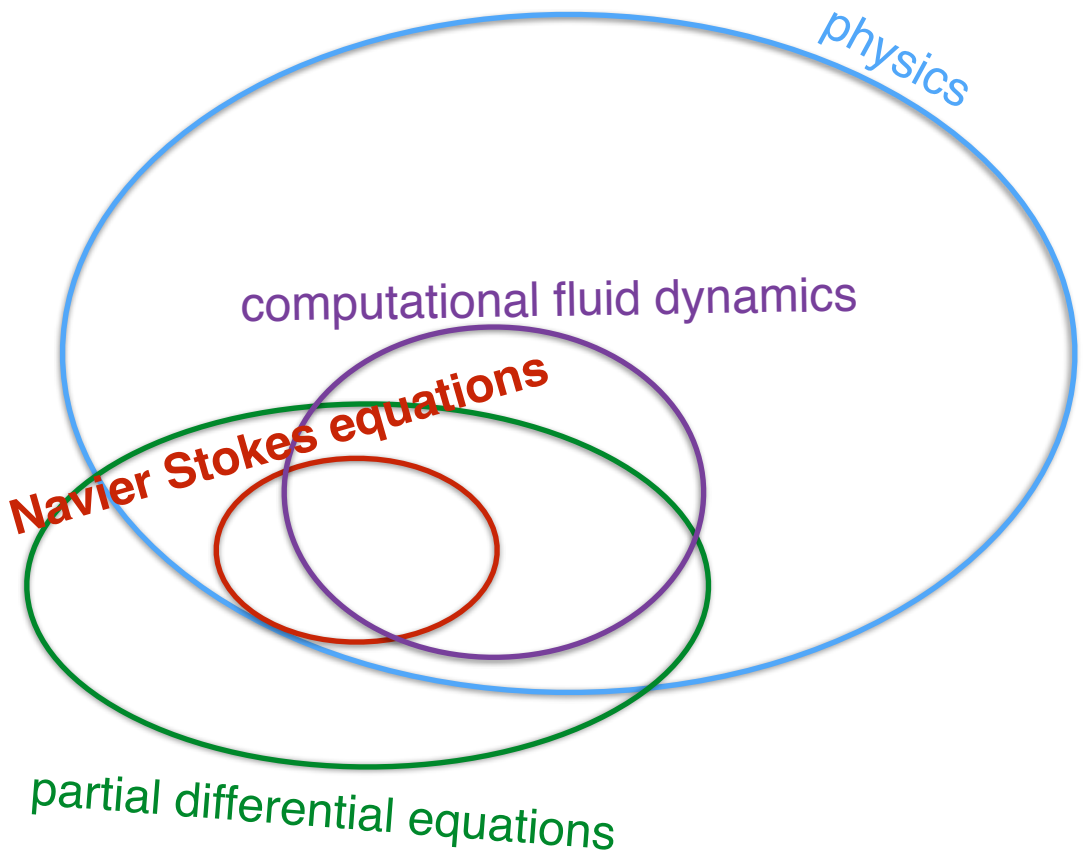
Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Experimental results

- Synthetic hierarchies
- Entailment
- Word similarity tasks
- Scientific key phrase finding

Scientific key-phrase finding



Scientific key-phrase finding

What makes a good key-phrase?

- High frequency
- Predictive

Scientific key-phrase finding

What makes a good key-phrase?

- High frequency
- Predictive

Phrases	Frequent?	Predictive?
conventional wisdom suggests pre-defined categories	No	No
paper describes experimental results	Yes	No
EXPTIME complete autocorrelation function	No	Yes
operational semantics regular languages	Yes	Yes

Scientific key-phrase finding

Sample key-phrases from scientific paper abstracts:

frequent, predictive

rare, uninformative

linear matrix inequality
satisfiability problem
encryption schemes
sparse matrix
vector spaces
exploratory study
theoretical basis
major contributions
hot topic

Thank you! Conclusion

- Introduced **Gaussian word embeddings**:
  - Capture **asymmetry**
  - Capture **broadness** of meaning and **uncertainty**
  - **Expressive**, dense, distributed representation
  - **Scalable** learning
    - 4 billion tokens, 1 core, 8 hours
- **Future work:**
  - Multi-peaked, unnormalized, non-Gaussian
  - Relations, documents, semantic frames
  - Non-NLP domains for density representations