Distributional Semantics

Seminar Hot & Odd Topics in Semantics

Polina Stadnikova

Saarland University

23th April 2018

Outline

How do we represent word meanings formally?

Formal vs. Distributional

Set theory

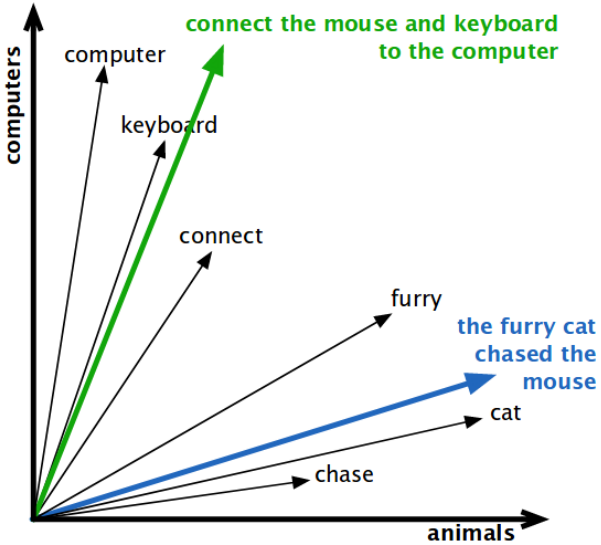
All cats chase mice:

$\forall x \forall y((cat'(x) \wedge mouse'(y)) \rightarrow chase'(x, y))$

Each mouse is connected to a computer:

$\forall x(mouse'(x) \rightarrow \exists y(comp'(y) \wedge connect'(x, y))$

Vector models



Formal vs. Distributional

Set theory

Represent object, their properties and relations between them, meanings = constraints

Vector models

Capture distance and similarity, meanings = vectors in a semantic space, **more fine-grained representation**

The basic idea of distributional semantics

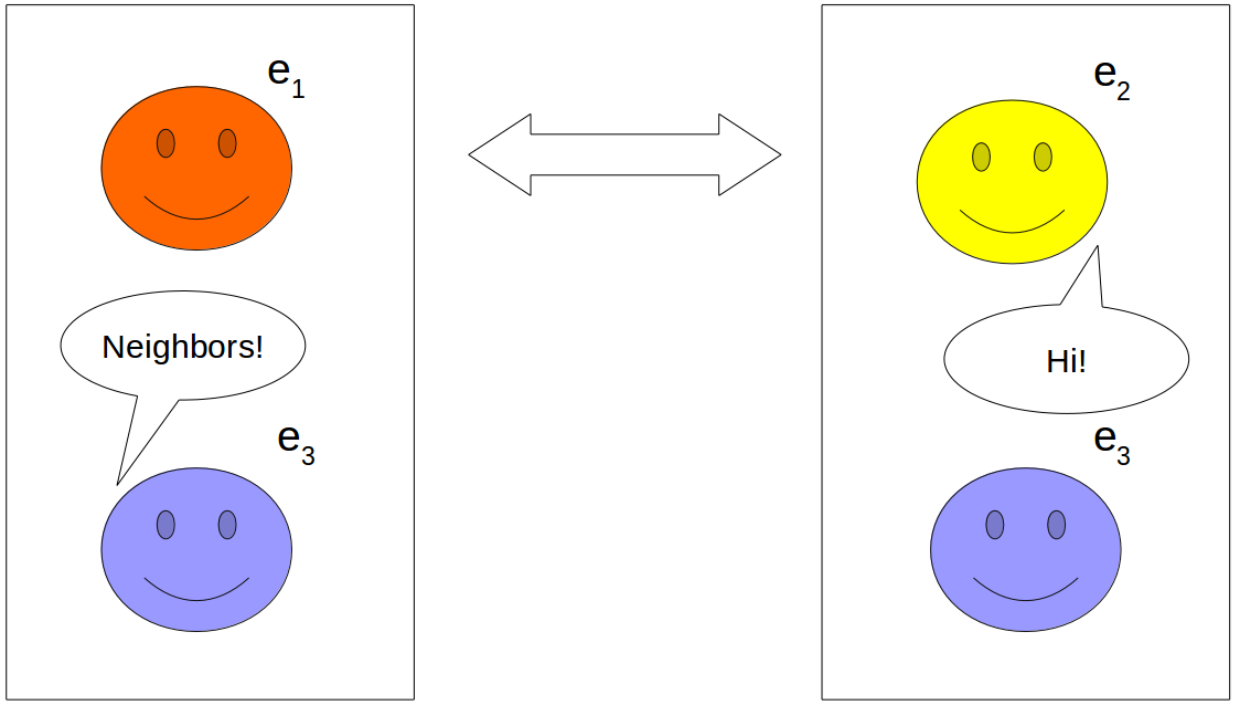
"You shall know a word by the company it keeps!"

— J. R. Firth (1957)

Distributional hypothesis

- Words that occur in similar contexts tend to have similar meanings
- Set of contexts for a word = distribution
- Distribution over contexts represents meaning

Distributional methodology



e_1 and e_2 have similar distributional properties

Distributional properties

"In the language itself, there are only differences."

— F. de Saussure (1916/1983)

Two types of differences:

- **Syntagmatic**
  rely on positioning of entities (words co-occur at the same time)
- **Paradigmatic**
  rely on substitutability (words do not co-occur, they can substitute each other)

Distributional properties

	Paradigmatic relations **Selections: "x or y or..."**			
Syntagmatic relations	she	adores	green	paint
Combinations:	he	likes	blue	dye
"x and y and..."	they	love	red	colour

Refined distributional hypothesis

- Distributions using co-occurrence information contain syntagmatic relations
- Distributions using information about shared neighbors contain paradigmatic relations

Syntagmatic distributions

Giggs scored the first **goal** *of the* **football** *tournament at Wembley , North London.*
Bellamy was largely a passenger in the **football** *match , playing no part in either* **goal**.

Collocations, words co-occurring within the same *context*...
How do we define context?

Syntagmatic distributions

What is a context?

- set of documents
- a document
- some words

Large contexts enforce randomness in statistics, small contexts yield poor statistics and ignore some co-occurrences.

More fine-grained context?

- direction (on which side from the target word?)
- position in the context window
- preprocessing (PoS-tagging, dependency parsing, lemmatization)

Syntagmatic distributions

Different contexts for the target word *goal*:

Contextual elements for target word *goal* using a 7-word window method:
{*scored, the, first, of, football*}

Contextual elements with parts-of-speech:
{*scored*|VBD, *the*|DET, *first*|JJ, *of*|IN, *football*|NN}

Contextual elements with direction (L for left, R for right):
{*scored*|L, *the*|L, *first*|L, *of*|R, *the*|R, *football*|R}

Contextual elements with position (e.g. 1L is 1 word to the left):
{*scored*|3L, *the*|2L, *first*|1L, *of*|1R, *the*|2R, *football*|3R}

Contextual elements as grammatical relations:
{*first*|`ncmod`, *the*|`det`, *scored*|`dobj`}

Paradigmatic distributions

The London 2012 **soccer** *tournament began yesterday , with plenty of goals in the opening matches.*
Giggs scored the first goal of the **football** *tournament at Wembley , North London.*

Synonyms, antonyms, words occurring in similar contexts.

Syntagmatic vs. Paradigmatic

One context,
co-occurring words

vs.

Similar contexts,
not co-occurring words

cut blade spoon **knife** noni cutterhead nimuk

hammer hat shovel pencil **knife** spoon blanket

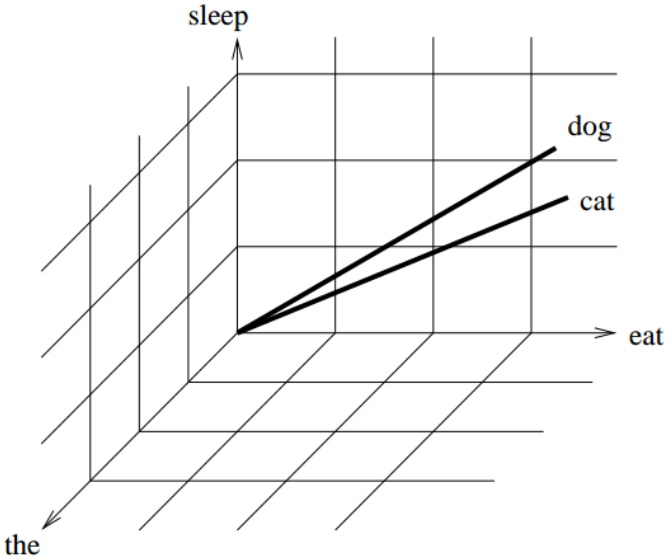
Here comes math...

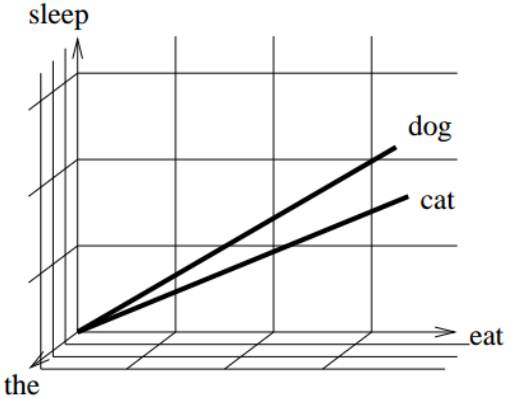
$$
\begin{array}{l}
 \\ \textit{automobile} \\ \textit{car} \\ \textit{soccer} \\ \textit{football}
\end{array}
\begin{array}{c}
\begin{array}{ccccccc}
\textit{wheel} & \textit{transport} & \textit{passenger} & \textit{tournament} & \textit{London} & \textit{goal} & \textit{match}
\end{array} \\
\begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 2 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 2 & 1
\end{pmatrix}
\end{array}
$$

Word representations = vectors = rows from co-occurrence matrices
Vector coefficients = co-occurrence frequencies (raw or weighted counts)
In practice, large matrices are reduced (SVD)

Weighted frequencies

Some words have more discriminating power.





Weighted frequencies

The basic idea:

If a word occurs only in few documents/contexts, it provides a stronger evidence for a particular meaning → should have a higher weight.

How to weight frequencies?

Divide term frequency by the number of documents/context in which the term occurs (TF IDF approach).

Similarity

Do distributional representations really capture semantic similarity?

(Very) simple method

- $Sim(\vec{q}, \vec{d}) = \sum_i q_i * d_i$

Normalization by length?

- $Sim(\vec{q}, \vec{d}) = \cos(\vec{q}, \vec{d})$
- avoids bias towards longer contexts
- still can fail in some applications (synonyms vs. antonyms)

word2vec

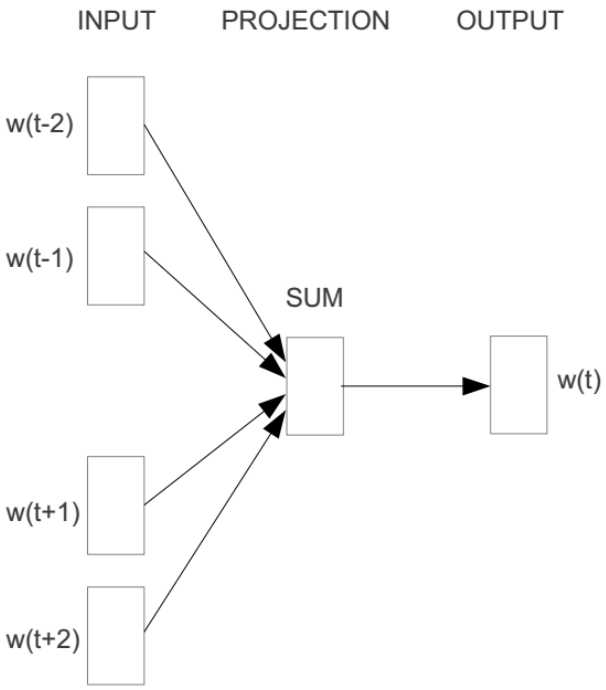
Motivation & goals

- Handle billions of words
- Capture different degrees of similarity

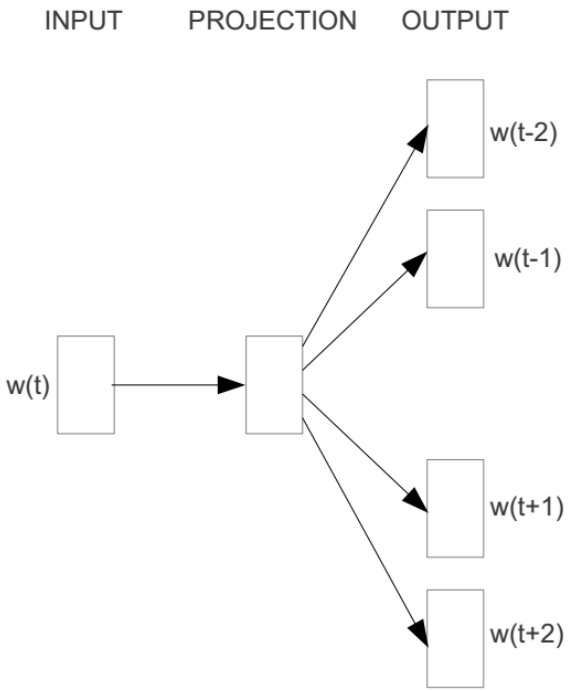
Methodology

- Neural networks!
- Use simple models to learn word vectors (CBOW and Skip-gram)
- Train a complex neural network on obtained vectors

word2vec



CBOW



Skip-gram

word2vec

CBOW

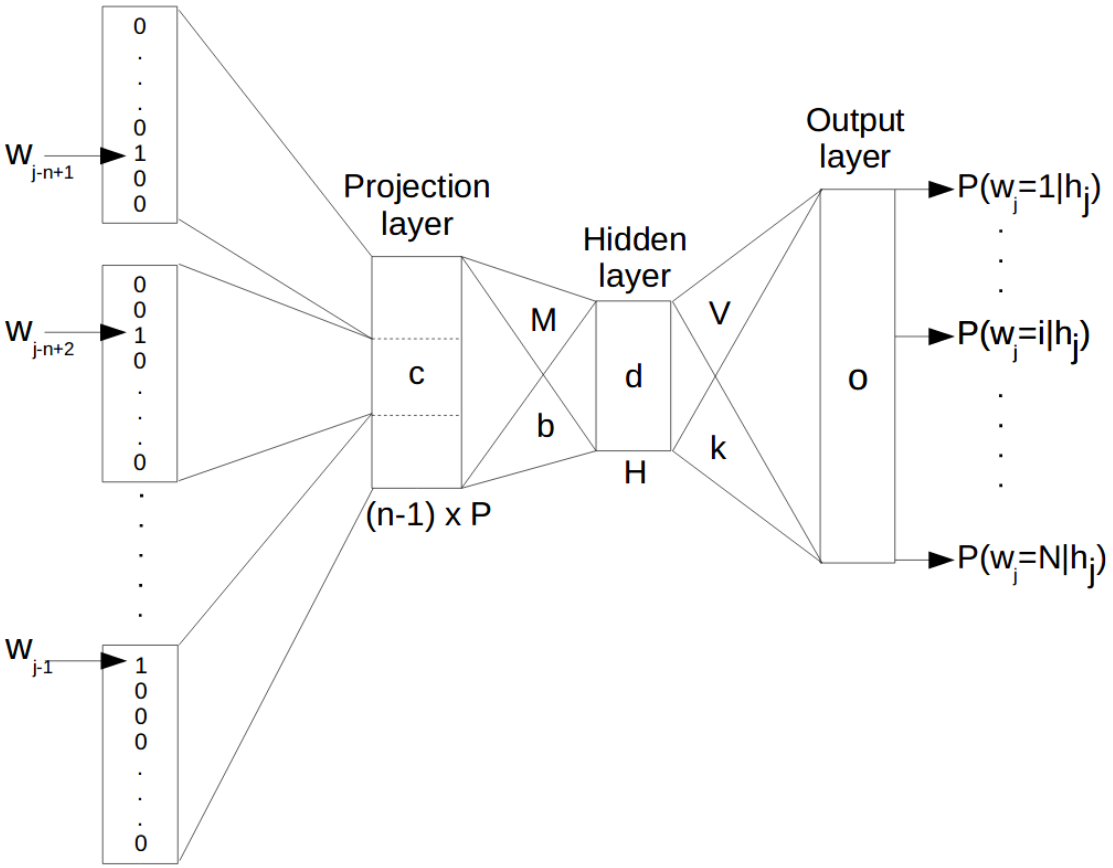
- Continuous bag-of-words model
- Given context, predict word
- *He scored a ___ in the first match*
- Faster, better for large datasets

Skip-gram

- Continuous skip-gram model
- Given word, predict context
- ___ ___ ___ *goal* ___ ___ ___ ___
- Better for small datasets and infrequent words

word2vec

Feedforward Neural Net Language Model



word2vec

Recurrent Neural Net Language Model

- Does not have a projection layer
- Hidden layer is connected to itself → short-term memory (can remember previous inputs)
- No need to specify the input length → can handle more complex patterns
- Outperforms Feedforward NNLMs

word2vec

Experiments

- Word similarity task
- 1.6 billion words dataset
- Different types of similarity (*big-bigger/small-smaller* vs. *big-biggest/small-smallest*)
- Define different types of semantic and syntactic questions
- Perform operations like addition and subtraction on vectors

Example

- What is the word that is similar to small in the same sense as biggest is similar to big?
- $\vec{word} = \vec{biggest} - \vec{big} + \vec{small}$
- find the word closest to $\vec{word}$

word2vec

Wrap-up

- Outperforms state of the art models
- Learns high quality representations from large datasets with lower computational cost
- Captures both semantic and syntactic similarity
- Applications: machine translation, question answering, information retrieval

We can increase vector dimensionality and training data size at the same time!

GloVe

Motivation

- Criticism on word2vec skip-gram models: scan each context window, do not use co-occurrence statistics directly

Goal

- Effectively use statistics
- Capture semantic and syntactic relations

Methodology

- Non-zero elements of a global word-word co-occurrence matrix
- Assumption: ratio of probabilities encode some form of meaning

GloVe

Probability ratio:

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

Train the log-bilinear model by minimizing the difference between the dot product of vectors and ratio of co-occurrence probabilities.

GloVe

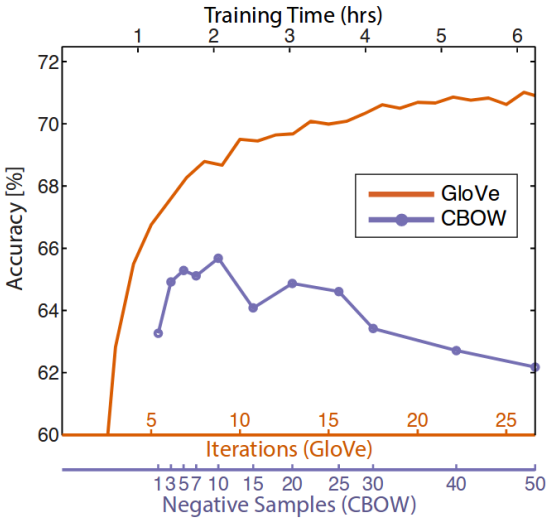
Experiments

- Training data - billions of tokens
- Word analogy task from word2vec: *A is to B as C is to ___ ?*
- Different similarity tasks
- Named Entity Recognition

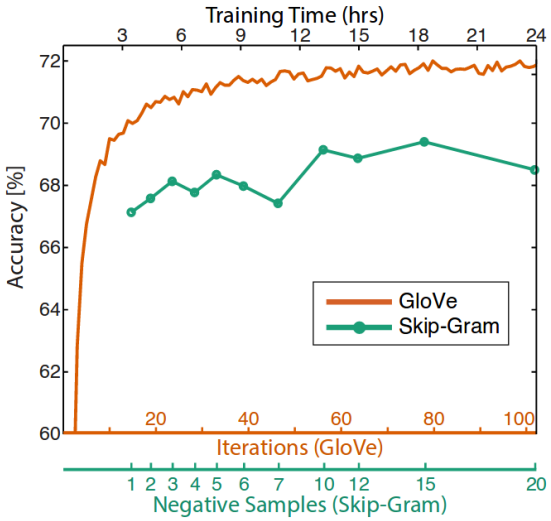
GloVe

Wrap-up

- Outperforms other models on all three tasks
- Fast training on large corpora
- Good performance even with small corpora



(a) GloVe vs CBOW



(b) GloVe vs Skip-Gram

Cross-lingual learning

- *There are so many languages...*
- *Do you think distributional semantics can help?*
- *Let's see...It can measure similarity between words!*
- *Hmm...Machine translation people would be happy...*

Cross-lingual learning

Why do we care?

- Resource-rich vs. resource-lean languages
- Reasoning about word meaning in multilingual context
- Knowledge transfer between languages

Basic idea

- Learn word representations from multilingual data
- Adapt existing monolingual approaches

Functions

- Capture similarity
- Serve as features for cross-lingual transfer

Cross-lingual learning

Data

- Parallel data: word or sentence alignments (always available? no)
- Comparable data: sources are similar to some extent (to what extent? is correct comparison ensured?)

Method

- Count-based (remember GloVe?)
- Prediction-based (remember word2vec?)

Cross-lingual learning

Søgaard et al.(2015):

No parallel data

- Articles on the same topic in different languages are linked to the same concept in the Wikipedia ontology
- Use a common subset of concepts
- Describe words by Wikipedia concepts
- Similar words occur in articles linked to the same concept

Count-based

- Concept-word co-occurrence matrix

Outperforms two state-of-art approaches on 14 from 17 datasets in 4 tasks.

Cross-lingual learning

Applications from Søgaard et al.(2015)

- Document classification
- PoS-tagging
- Dependency parsing
- Word alignment

Word sense induction

- Task: cluster sentences in English containing the same word according to different meanings of the word
- Better results with multilingual word embeddings than with bi-/monolingual

Final wrap-up

Now we know how to...

- **Represent word meanings as vectors**
  Context = basis vectors which define space
  Words to describe = vectors in space

Final wrap-up

Now we know how to...

- **Represent word meanings as vectors**
  Context = basis vectors which define space
  Words to describe = vectors in space

- **Measure similarity of vectors**
  ...and not to forget the importance of normalization

Final wrap-up

Now we know how to...

- **Represent word meanings as vectors**
  Context = basis vectors which define space
  Words to describe = vectors in space

- **Measure similarity of vectors**
  ...and not to forget the importance of normalization

- **Use word representations in NLP**
  machine translation, question answering systems, word sense disambiguation, ...

Final wrap-up

We also learned that...

Defining a context might be complicated

...and depends on the application

Final wrap-up

We also learned that...

Defining a context might be complicated

...and depends on the application

There is no big difference between count-based and prediction-based approaches

...even if it's fun to argue about this

Final wrap-up

We also learned that...

Defining a context might be complicated
...and depends on the application

There is no big difference between count-based and prediction-based approaches
...even if it's fun to argue about this

Vector representation of word meanings help to extact lexical realtions
synonymy, antonymy, hyponymy, etc.

Final wrap-up

Moreover...

There are also multilingual word representations
and they contribute to performance improvement

References

- Stephen Clark (2015), *Vector Space Models of Lexical Meaning.* In: Handbook of Contemporary Semantic Theory — second edition, edited by Shalom Lappin and Chris Fox. Chapter 16, pp.493-522.
- Magnus Sahlgren (2008),*The Distributional Hypothesis*. In: Italian Journal of Linguistics.
- Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean (2013),*Efficient Estimation of Word Representations in Vector Space*. In: Proceedings of ICLR, pages 1-12.
- Jeffrey Pennington, Richard Socher, and Christopher Manning (2014), *GloVe: Global Vectors for Word Representation*. In: Proceedings of EMNLP, pages 1532-1543.
- Anders Søgaard, Zeljko Agic, Hector Martinez Alonso, Barbara Plank, and Bernd Bohnet (2015), *Inverted indexing for cross-lingual NLP*. In: Proceedings of ACL, Vol. 1, pages 1713-1722.
- Sebastian Ruder, Ivan Vulic, and Anders Søgaard (2017), *A Survey of Cross-lingual Embedding Models*.
- Shyam Upadhyay, Kai-Wei Chang, Matt Taddy, Adam Kalai, and James Zou (2017), *Beyond bilingual: Multi-sense word embeddings using multilingual context*. In: Proceedings of the 2nd Workshop on RepL4NLP, pages 139–145.

Thanks for your attention!

Questions?