Natural Language Processing (CSEP 517): Distributional Semantics

Roy Schwartz

University of Washington
roysch@cs.washington.edu

May 15, 2017

To-Do List

- Read: (Jurafsky and Martin, 2016a,b)

Distributional Semantics Models

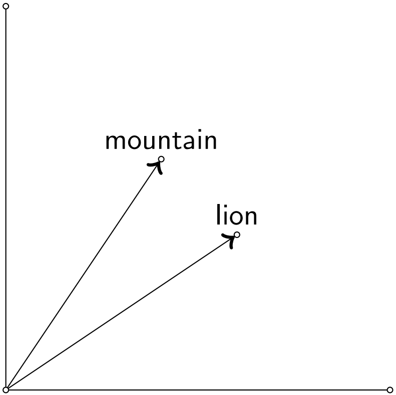
Aka, Vector Space Models, **Word Embeddings**

$$\mathbf{v}_{\text{mountain}} = \begin{pmatrix} 0.23 \\ \text{-}0.21 \\ 0.15 \\ 0.61 \\ \vdots \\ 0.02 \\ \text{-}0.12 \end{pmatrix}, \mathbf{v}_{\text{lion}} = \begin{pmatrix} 0.72 \\ 0.2 \\ 0.71 \\ 0.13 \\ \vdots \\ \text{-}0.1 \\ \text{-}0.11 \end{pmatrix}$$

Distributional Semantics Models

Aka, Vector Space Models, **Word Embeddings**

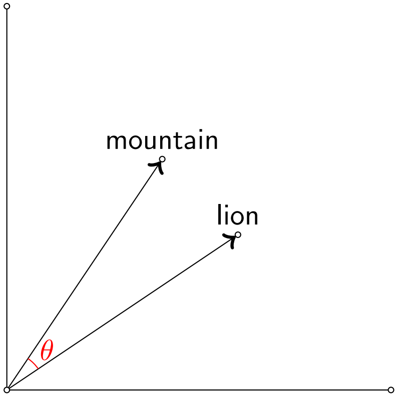
$$\mathbf{v}_{\text{mountain}} = \begin{pmatrix} 0.23 \\ \text{-0.21} \\ 0.15 \\ 0.61 \\ \vdots \\ 0.02 \\ \text{-0.12} \end{pmatrix}, \mathbf{v}_{\text{lion}} = \begin{pmatrix} 0.72 \\ 0.2 \\ 0.71 \\ 0.13 \\ \vdots \\ \text{-0.1} \\ \text{-0.11} \end{pmatrix}$$



Distributional Semantics Models

Aka, Vector Space Models, **Word Embeddings**

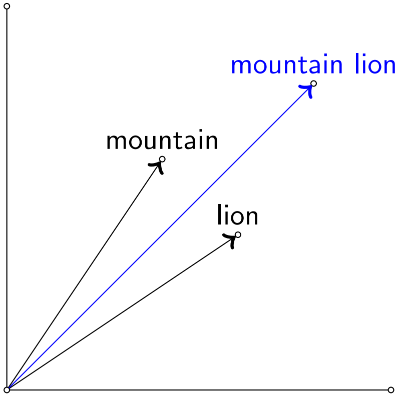
$$\mathbf{v}_{\text{mountain}} = \begin{pmatrix} 0.23 \\ \text{-}0.21 \\ 0.15 \\ 0.61 \\ \vdots \\ 0.02 \\ \text{-}0.12 \end{pmatrix}, \mathbf{v}_{\text{lion}} = \begin{pmatrix} 0.72 \\ 0.2 \\ 0.71 \\ 0.13 \\ \vdots \\ \text{-}0.1 \\ \text{-}0.11 \end{pmatrix}$$



Distributional Semantics Models

Aka, Vector Space Models, **Word Embeddings**

$$\mathbf{v}_{\text{mountain}} = \begin{pmatrix} 0.23 \\ -0.21 \\ 0.15 \\ 0.61 \\ \vdots \\ 0.02 \\ -0.12 \end{pmatrix}, \mathbf{v}_{\text{lion}} = \begin{pmatrix} 0.72 \\ 0.2 \\ 0.71 \\ 0.13 \\ \vdots \\ -0.1 \\ -0.11 \end{pmatrix}$$



Distributional Semantics Models

Aka, Vector Space Models, **Word Embeddings**

Applications

Deep learning models:
Machine Translation
Question Answering
Syntactic Parsing
...

Linguistic Study

Lexical Semantics
Multilingual Studies
Evolution of Language
...

Outline

Outline

Distributional Semantics Hypothesis

Harris (1954)

Words that have similar contexts are likely to have similar meaning

Distributional Semantics Hypothesis

Harris (1954)

Words that have similar contexts are likely to have similar meaning

Vector Space Models

- Representation of words by vectors of real numbers
- $\forall w \in \mathcal{V}, \mathbf{v}_w$ is function of the contexts in which w occurs
- Vectors are computed using a large text corpus
  - No requirement for any sort of annotation in the general case

$V_{1.0}$: Count Models

Salton (1971)

- Each element $\mathbf{v}_{w_i} \in \mathbf{v}_w$ represents the co-occurrence of w with another word i
  - $\mathbf{v}_{\text{dog}} =$ (cat: 10, leash: 15, loyal: 27, bone: 8, piano: 0, cloud: 0, ...)
- Vector dimension is typically very large (vocabulary size)
- Main motivation: lexical semantics

Count Models

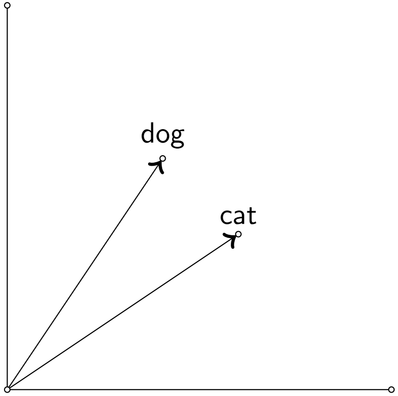
Example

$$\mathbf{v}_{\mathsf{dog}} = \begin{pmatrix} 0 \\ 0 \\ 15 \\ 17 \\ \vdots \\ 0 \\ 102 \end{pmatrix}, \mathbf{v}_{\mathsf{cat}} = \begin{pmatrix} 0 \\ 2 \\ 11 \\ 13 \\ \vdots \\ 20 \\ 11 \end{pmatrix}$$

Count Models

Example

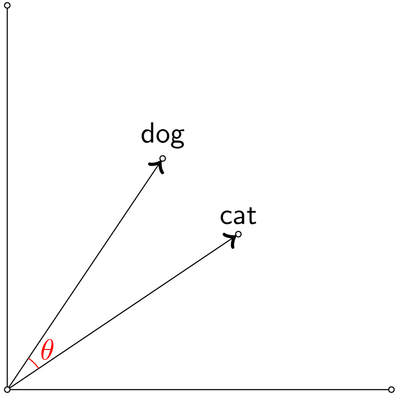
$$\mathbf{v}_{\text{dog}} = \begin{pmatrix} 0 \\ 0 \\ 15 \\ 17 \\ \vdots \\ 0 \\ 102 \end{pmatrix}, \mathbf{v}_{\text{cat}} = \begin{pmatrix} 0 \\ 2 \\ 11 \\ 13 \\ \vdots \\ 20 \\ 11 \end{pmatrix}$$



Count Models

Example

$$\mathbf{v}_{\text{dog}} = \begin{pmatrix} 0 \\ 0 \\ 15 \\ 17 \\ \vdots \\ 0 \\ 102 \end{pmatrix}, \mathbf{v}_{\text{cat}} = \begin{pmatrix} 0 \\ 2 \\ 11 \\ 13 \\ \vdots \\ 20 \\ 11 \end{pmatrix}$$



Variants of Count Models

- Reduce the effect of high frequency words by applying a weighting scheme
  - Pointwise mutual information (PMI), TF-IDF

Variants of Count Models

- Reduce the effect of high frequency words by applying a weighting scheme
  - Pointwise mutual information (PMI), TF-IDF
- Smoothing by dimensionality reduction
  - Singular value decomposition (SVD), principal component analysis (PCA), matrix factorization methods

Variants of Count Models

- Reduce the effect of high frequency words by applying a weighting scheme
  - Pointwise mutual information (PMI), TF-IDF
- Smoothing by dimensionality reduction
  - Singular value decomposition (SVD), principal component analysis (PCA), matrix factorization methods
- What is a context?
  - Bag-of-words context, document context (Latent Semantic Analysis (LSA)), dependency contexts, pattern contexts

Outline

Vector Space Models

Lexical Semantic Applications

Word Embeddings

Compositionality

Current Research Problems

Vector Space Models

Evaluation

- Vector space models as features
  - Synonym detection
    - TOEFL (Landauer and Dumais, 1997)
  - Word clustering
    - CLUTO (Karypis, 2002)

Vector Space Models

Evaluation

- Vector space models as features
  - Synonym detection
    - TOEFL (Landauer and Dumais, 1997)
  - Word clustering
    - CLUTO (Karypis, 2002)
- Vector operations
  - Semantic Similarity
    - RG-65 (Rubenstein and Goodenough, 1965), wordsim353 (Finkelstein et al., 2001), MEN (Bruni et al., 2014), SimLex999 (Hill et al., 2015)
  - Word Analogies
    - Mikolov et al. (2013)

Semantic Similarity

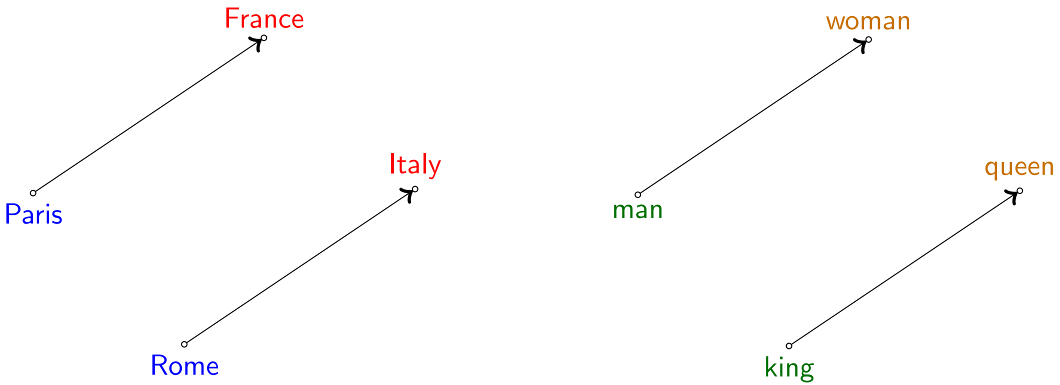
w_1	w_2	human score	model score
tiger	cat	7.35	0.8
computer	keyboard	7.62	0.54
...	...	...	...
architecture	century	3.78	0.03
book	paper	7.46	0.66
king	cabbage	0.23	-0.42

Table: Human scores taken from wordsim353 (Finkelstein et al., 2001)

- Model scores are cosine similarity scores between vectors
- Model's performance is the Spearman/Pearson correlation between human ranking and model ranking

Word Analogy

Mikolov et al. (2013)



Outline

$V_{2.0}$: Predict Models

(Aka Word Embeddings)

- A new generation of vector space models
- Instead of representing vectors as cooccurrence counts, train a supervised machine learning algorithm to predict $p(\text{word}|\text{context})$
- Models learn a latent vector representation of each word
  - These representations turn out to be quite effective vector space representations
  - *Word embeddings*

Word Embeddings

- Vector size is typically a few dozens to a few hundreds
- Vector elements are generally uninterpretable
- Developed to initialize feature vectors in deep learning models
  - Initially language models, nowadays virtually every sequence level NLP task
  - Bengio et al. (2003); Collobert and Weston (2008); Collobert et al. (2011); word2vec (Mikolov et al., 2013); GloVe (Pennington et al., 2014)

Word Embeddings

- Vector size is typically a few dozens to a few hundreds
- Vector elements are generally uninterpretable
- Developed to initialize feature vectors in deep learning models
  - Initially language models, nowadays virtually every sequence level NLP task
  - Bengio et al. (2003); Collobert and Weston (2008); Collobert et al. (2011); word2vec (Mikolov et al., 2013); GloVe (Pennington et al., 2014)

word2vec

Mikolov et al. (2013)

- A software toolkit for running various word embedding algorithms

Based on (Goldberg and Levy, 2014)

word2vec

Mikolov et al. (2013)

- A software toolkit for running various word embedding algorithms
- Continuous bag-of-words: $\underset{\theta}{\operatorname{argmax}} \prod_{w \in \text{corpus}} p(w|C(w);\theta)$

Based on (Goldberg and Levy, 2014)

word2vec

Mikolov et al. (2013)

- A software toolkit for running various word embedding algorithms
- Continuous bag-of-words: $\underset{\theta}{\operatorname{argmax}} \prod_{w \in \text{corpus}} p(w|C(w);\theta)$
- Skip-gram: $\underset{\theta}{\operatorname{argmax}} \prod_{(w,c) \in \text{corpus}} p(c|w;\theta)$

Based on (Goldberg and Levy, 2014)

word2vec

Mikolov et al. (2013)

- A software toolkit for running various word embedding algorithms
- Continuous bag-of-words: $\underset{\theta}{\operatorname{argmax}} \prod_{w \in \text{corpus}} p(w|C(w);\theta)$
- Skip-gram: $\underset{\theta}{\operatorname{argmax}} \prod_{(w,c) \in \text{corpus}} p(c|w;\theta)$
- Negative sampling: randomly sample **negative** (*word,context*) pairs, then:

$$\underset{\theta}{\operatorname{argmax}} \prod_{(w,c) \in \text{corpus}} p(c|w;\theta) \cdot \prod_{(w,c')} (1 - p(c'|w;\theta))$$

Based on (Goldberg and Levy, 2014)

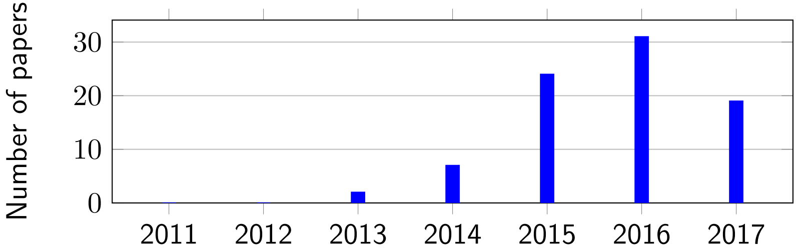
Skip-gram with Negative Sampling (SGNS)

- Obtained significant improvements on a range of lexical semantic tasks
- Is very fast to train, even on large corpora
- Nowadays, by far the most popular word embedding approach[1]

[1] Along with GloVe (Pennington et al., 2014)

Embeddings in ACL

Number of Papers in ACL Containing the Word “Embedding”



Count vs. Predict

- Don't count, Predict! (Baroni et al., 2014)

Count vs. Predict

- Don't count, Predict! (Baroni et al., 2014)
- But...
  Neural embeddings are implicitly matrix factorization tools (Levy and Goldberg, 2014)

Count vs. Predict

- Don't count, Predict! (Baroni et al., 2014)
- But...
  Neural embeddings are implicitly matrix factorization tools (Levy and Goldberg, 2014)
- So?...
  It's all about *hyper-parameter* (Levy et al., 2015)

Count vs. Predict

- Don't count, Predict! (Baroni et al., 2014)
- But...
  Neural embeddings are implicitly matrix factorization tools (Levy and Goldberg, 2014)
- So?...
  It's all about *hyper-parameter* (Levy et al., 2015)
- The bottom line:
  word2vec and GloVe are very good *implementations*

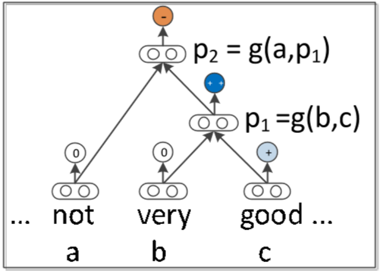
Outline

Compositionality

- Basic approach: average / weighted average
  - $\mathbf{v}_{\text{good}} + \mathbf{v}_{\text{day}} = \mathbf{v}_{\text{good day}}$

Compositionality

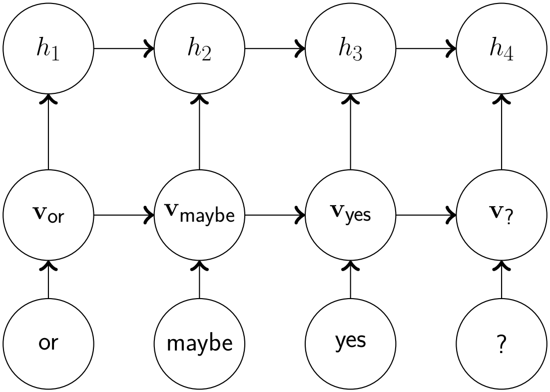
Recursive Neural Networks (Goller and Kuchler, 1996)



Picture taken from Socher et al. (2013)

Compositionality

Recurrent Neural Networks (Elman, 1990)



Recurrent Neural Networks

- In recent years, the most common method to represent sequence of texts is using RNNs
  - In particular, long short-term memory (LSTM, Hochreiter and Schmidhuber (1997)) and gated recurrent unit (GRU, Cho et al. (2014))

Recurrent Neural Networks

- In recent years, the most common method to represent sequence of texts is using RNNs
  - In particular, long short-term memory (LSTM, Hochreiter and Schmidhuber (1997)) and gated recurrent unit (GRU, Cho et al. (2014))
- Very recently, state-of-the-art models on tasks such as semantic role labeling and coreference resolution stated to rely solely on deep networks with word embeddings and LSTM layers (He et al., 2017)
  - These tasks traditionally relied on syntactic information
  - Many of these results come from the UW NLP group

Word Embeddings in RNNs

- Pre-trained embeddings (fixed or tuned)
- Random initialization
- A concatenation of both types

Alternatives to Word Embeddings

- Character embeddings
  - Machine translation (Ling et al., 2015)
  - Syntactic parsing (Ballesteros et al., 2015)
- Character n-grams (Neubig et al., 2013; Schütze, 2017)
- POS tag embeddings (Dyer et al., 2015)

Outline

50 Shades of Similarity

- What is *similarity*?
  - Synonymy: high — tall
  - Co-hyponymy: dog — cat
  - Association: coffee — cup
  - Dissimilarity: good — bad
  - Attributional similarity: banana — the sun (both are yellow)
  - Morphological similarity: going — crying (same verb tense)
  - Schwartz et al. (2015); Rubinstein et al. (2015); Cotterell et al. (2016)
- Definition is *application dependent*

What is a context?

- Most word embeddings rely on bag-of-word contexts
  - Which capture general word association
- Other options exists
  - Dependency links (Padó and Lapata, 2007)
  - Symmetric patterns (e.g., "X and Y", Schwartz et al. (2015, 2016))
  - Substitute vectors (Yatbaz et al., 2012)
  - Morphemes (Cotterell et al., 2016)
- Different context types translate to different relations between similar vectors

External Resources

- Guide vectors towards desired flavor of similarity
- Use dictionaries and/or thesauri
  - Part of the model (Yu and Dredze, 2014; Kiela et al., 2015)
  - Post-processing (Faruqui et al., 2015; Mrkšić et al., 2016)
- Multimodal embeddings

Multimodal Embeddings

- Combination of textual representation and perceptual representation
  - Most prominently visual
- Most approaches combine both types of vectors using methods such as canonical correlation analysis (CCA, e.g., Gella et al. (2016))
- The resulting embeddings often improve performance compared to text-only embeddings
  - They are also able to capture visual attributes such as size and color, which are often not captured by text only methods (Rubinstein et al., 2015)

Multilingual Embeddings

- Mapping embeddings in different languages into the same space
  - $\mathbf{v}_{\text{dog}} \sim \mathbf{v}_{\text{perro}}$
- Useful for multi-lingual tasks, as well as low-resource scenarios
- Most approaches use bilingual dictionaries or parallel corpora
- Recent approaches use more creative knowledge sources such as geospatial contexts (Cocos and Callison-Burch, 2017) and sentences ids in a parallel corpus (Levy et al., 2017)

Summary

- Distributional semantic models (aka *vector space models, word embeddings*) represent words using vectors of real numbers
- These methods are able to capture lexical semantics such as similarity and association
- They also serve as a fundamental building block in virtually all deep learning models in NLP
- Despite decades of research, many questions remain open

Summary

- Distributional semantic models (aka *vector space models, word embeddings*) represent words using vectors of real numbers
- These methods are able to capture lexical semantics such as similarity and association
- They also serve as a fundamental building block in virtually all deep learning models in NLP
- Despite decades of research, many questions remain open

Thank you!

Roy Schwartz `homes.cs.washington.edu/~roysch/` roysch@cs.washington.edu

References I

Miguel Ballesteros, Chris Dyer, and Noah A. Smith. Improved transition-based parsing by modeling characters instead of words with lstms. In *Proc. of EMNLP*, 2015.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proc. of ACL*, 2014.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *JMLR*, 3:1137–1155, 2003.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. Multimodal distributional semantics. *JAIR*, 49(2014):1–47, 2014.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In *Proc. of SSST*, 2014.

Anne Cocos and Chris Callison-Burch. The language of place: Semantic value from geospatial context. In *Proc. of EACL*, 2017.

Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proc. of ICML*, 2008.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *JMLR*, 12:2493–2537, 2011.

Ryan Cotterell, Hinrich Schütze, and Jason Eisner. Morphological smoothing and extrapolation of word embeddings. In *Proc. of ACL*, 2016.

References II

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. Transition-based dependency parsing with stack long short-term memory. In *Proc. of ACL*, 2015.

Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. Retrofitting word vectors to semantic lexicons. In *Proc. of NAACL*, 2015.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. In *Proc. of WWW*, 2001.

Spandana Gella, Mirella Lapata, and Frank Keller. Unsupervised visual sense disambiguation for verbs using multimodal embeddings. In *Proc. of NAACL*, 2016.

Yoav Goldberg and Omer Levy. word2vec explained: Deriving mikolov et al.?s negative-sampling word-embedding method, 2014. arXiv:1402.3722.

Christoph Goller and Andreas Kuchler. Learning task-dependent distributed representations by backpropagation through structure. In *Proc. of ICNN*, 1996.

Zelig Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. Deep semantic role labeling: What works and what's next. In *Proc. of ACL*, 2017.

Felix Hill, Roi Reichart, and Anna Korhonen. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 2015.

References III

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Dan Jurafsky and James H. Martin. Vector semantics (draft chapter). chapter 15. 2016a. URL `https://web.stanford.edu/~jurafsky/slp3/15.pdf`.

Dan Jurafsky and James H. Martin. Semantics with dense vectors (draft chapter). chapter 16. 2016b. URL `https://web.stanford.edu/~jurafsky/slp3/16.pdf`.

George Karypis. Cluto-a clustering toolkit. Technical report, DTIC Document, 2002.

Douwe Kiela, Felix Hill, and Stephen Clark. Specializing word embeddings for similarity or relatedness. In *Proc. of EMNLP*, 2015.

Thomas K. Landauer and Susan T. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211, 1997.

Omer Levy and Yoav Goldberg. Neural word embeddings as implicit matrix factorization. In *Proc. of NIPS*, 2014.

Omer Levy, Yoav Goldberg, and Ido Dagan. Improving distributional similarity with lessons learned from word embeddings. *TACL*, 3:211–225, 2015.

Omer Levy, Anders Søgaard, and Yoav Goldberg. A strong baseline for learning cross-lingual word embeddings from sentence alignments. In *Proc. of EACL*, 2017.

Wang Ling, Isabel Trancoso, Chris Dyer, and Alan W Black. Character-based neural machine translation, 2015. arXiv:1511.04586.

References IV

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. arXiv:1301.3781.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. Counter-fitting word vectors to linguistic constraints. In *Proc. of NAACL*, 2016.

Graham Neubig, Taro Watanabe, Shinsuke Mori, and Tatsuya Kawahara. Substring-based machine translation. *Machine Translation*, 27(2):139–166, 2013.

Sebastian Padó and Mirella Lapata. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199, 2007.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Proc. of EMNLP*, 2014.

Herbert Rubenstein and John B Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, 1965.

Dana Rubinstein, Effi Levi, Roy Schwartz, and Ari Rappoport. How well do distributional models capture different types of semantic knowledge? In *Proc. of ACL*, 2015.

Gerard Salton. *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1971.

Hinrich Schütze. Nonsymbolic text representation. In *Proc. of EACL*, 2017.

References V

Roy Schwartz, Roi Reichart, and Ari Rappoport. Symmetric pattern based word embeddings for improved word similarity prediction. In *Proc. of CoNLL*, 2015.

Roy Schwartz, Roi Reichart, and Ari Rappoport. Symmetric patterns and coordinations: Fast and enhanced representations of verbs and adjectives. In *Proc. of NAACL*, 2016.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. 2013.

Mehmet Ali Yatbaz, Enis Sert, and Deniz Yuret. Learning syntactic categories using paradigmatic representations of word context. In *Proc. of EMNLP*, 2012.

Mo Yu and Mark Dredze. Improving lexical embeddings with semantic knowledge. In *Proc. of ACL*, 2014.