Embeddings Words and Senses Together via Joint Knowledge-Enhanced Training

Massimiliano Mancini, **Jose Camacho-Collados**, Ignacio Iacobacci and Roberto Navigli

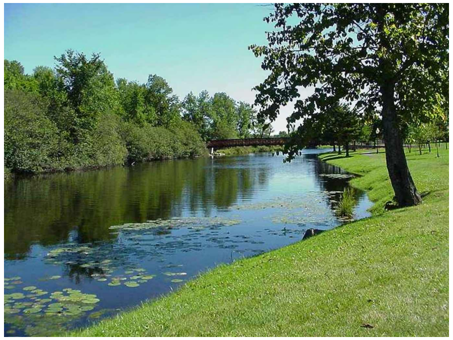
Dipartimento di Informatica

Sapienza Università di Roma

lcl.uniroma1.it/sw2v

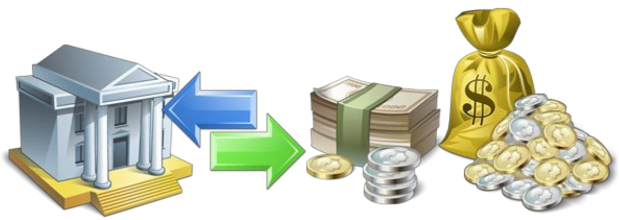
Motivation: Model senses instead of only words

*He withdrew money from the **bank**.*



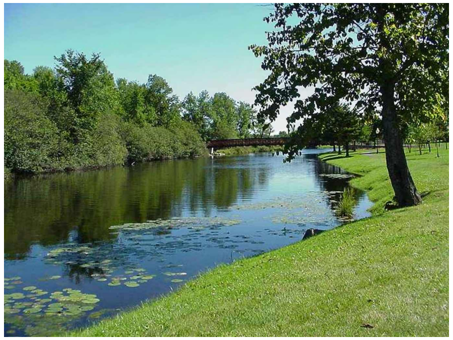
Motivation: Model senses instead of only words

*He withdrew money from the **bank**.*



bank#1

...

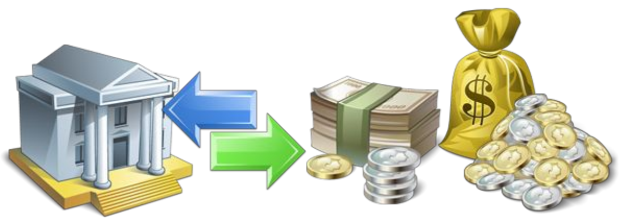


bank#2

...

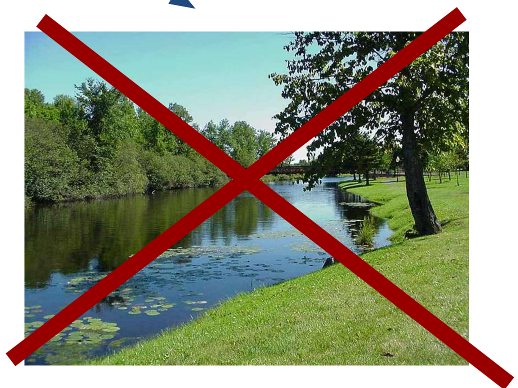
Motivation: Model senses instead of only words

*He withdrew money from the **bank**.*



bank#1

...



bank#2

...

Related Work

- **Unsupervised sense embeddings**

- **Knowledge-based sense embeddings**

Related Work

➢ **Unsupervised sense embeddings**

Learn sense embeddings exploiting **text corpora only** *(Huang et al. ACL 2012; Neelakantan et al. EMNLP 2014; Tian et al. COLING 2014; Li and Jurafsky, EMNLP 2015...)*. **Easily adaptable to new domains.**

Drawbacks:

- Senses not interpretable (+change from model to model)
- Knowledge from resources cannot be easily exploited
- Senses (esp. not frequent ones) not easy to discriminate

➢ **Knowledge-based sense embeddings**

Related Work

- ➢ **Unsupervised sense embeddings**
- ➢ **Knowledge-based sense embeddings**

Model **senses as defined on a sense inventory.**

Usually obtained as a **postprocessing of word embeddings** *(Chen et al. EMNLP 2014; Rothe and Schütze, ACL 2015...)*:

- Several training phases
- Infrequent senses not accurately captured

Related Work

- **Unsupervised sense embeddings**

- **Knowledge-based sense embeddings** (Our approach)

Related Work

- ➢ **Unsupervised sense embeddings**

- ➢ **Knowledge-based sense embeddings** (Our approach)

Idea

A word is the surface form of a sense: we can exploit this intrinsic relationship for **jointly training word and sense embeddings**.

Idea

A word is the surface form of a sense: we can exploit this intrinsic relationship for **jointly training word and sense embeddings**.

How?

Updating the representation of the word and its associated senses interchangeably.

Methodology

Given as input a **corpus** and a **semantic network**:

1. Use a semantic network to link to each word its *associated senses in context*.

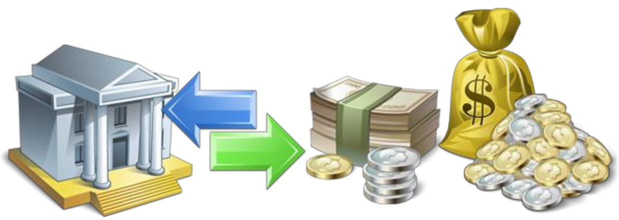
*He withdrew money from the **bank**.*

Methodology

Given as input a **corpus** and a **semantic network**:

1. Use a semantic network to link to each word its *associated senses in context*.

He withdrew money from the ***bank****.*



Methodology: Linking words and senses in context

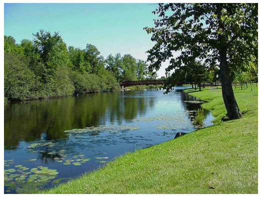
He **withdrew** **money** from the **bank**

retire

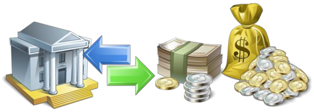
take out

cash

geography



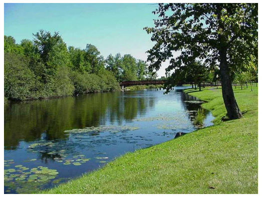
financial institution



building

Methodology: Linking words and senses in context

He **withdrew** **money** from the **bank**



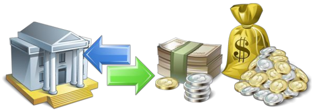
retire

take out

cash

geography

financial institution



building

Graph-based representation of the sentence using semantic networks (e.g. WordNet, BabelNet)

Methodology: Linking words and senses in context

He **withdrew** **money** from the **bank**

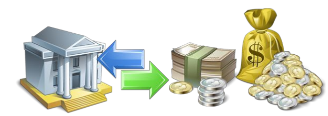
~~retire~~

take out

cash

~~geography~~

financial institution



building

Graph-based representation of the sentence using semantic networks (e.g. WordNet, BabelNet)

Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

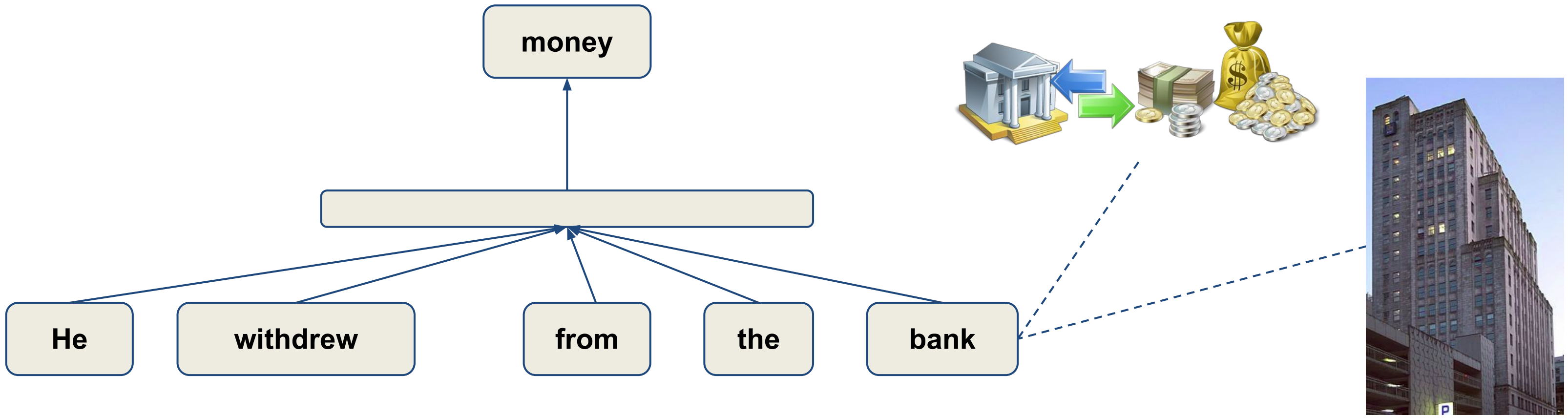
2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.

Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.

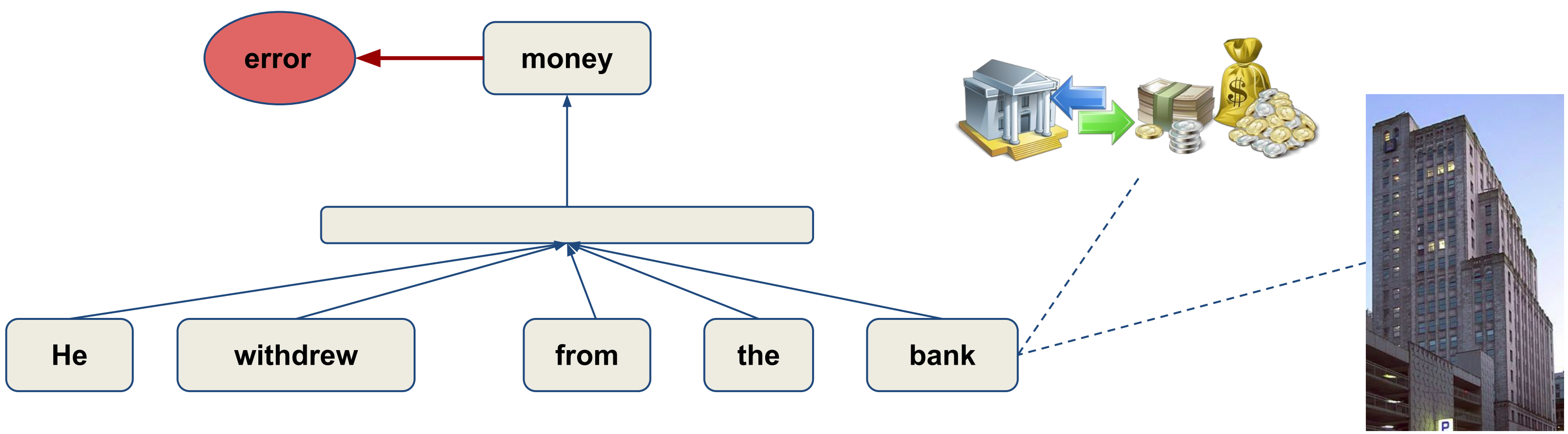


Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.

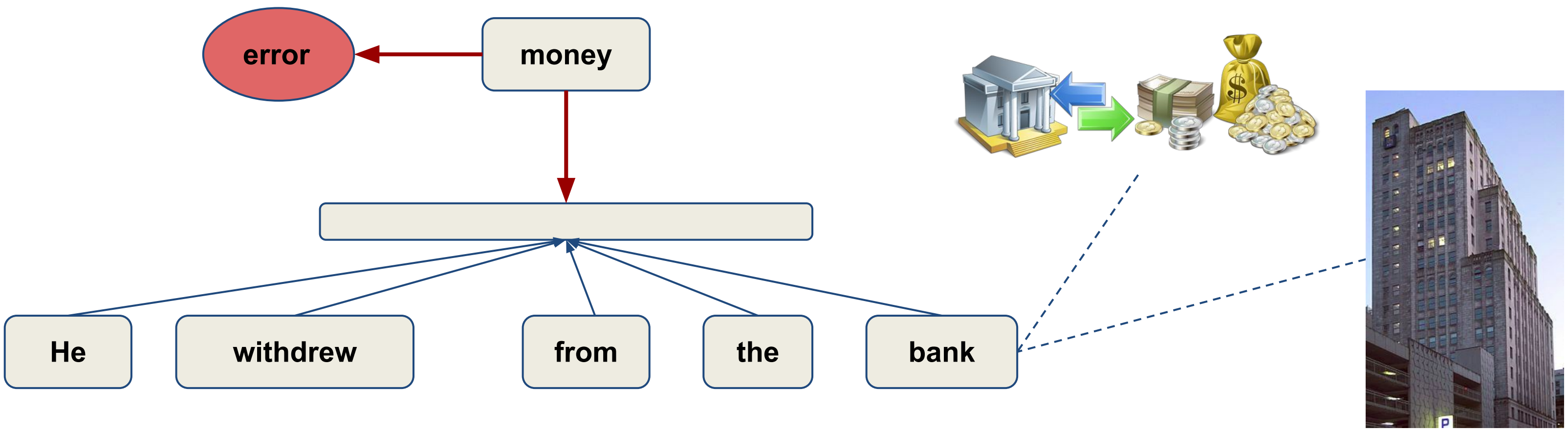


Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.

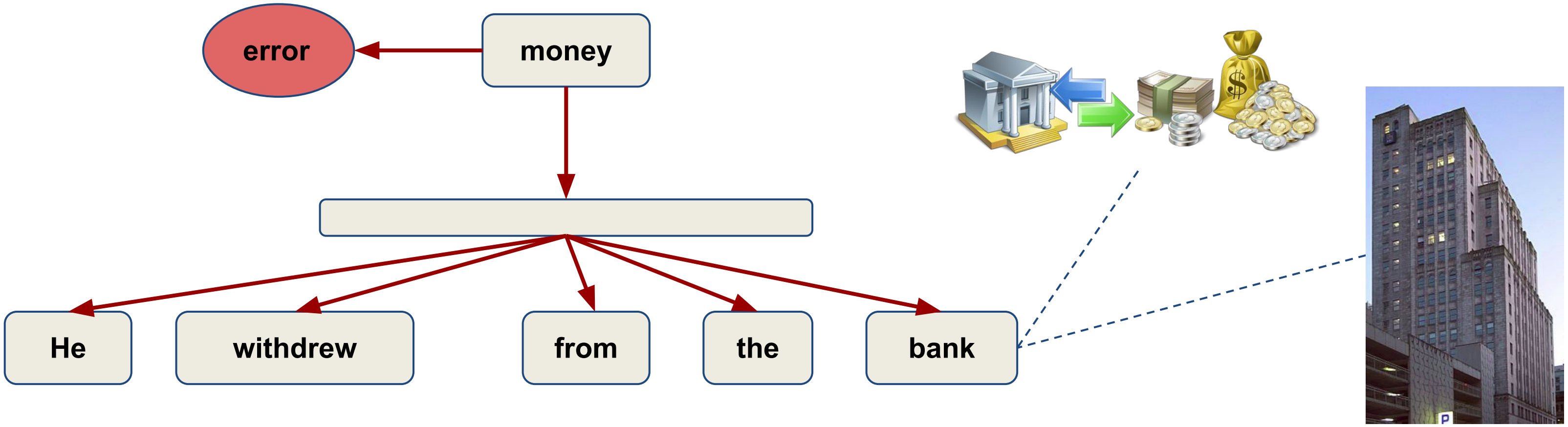


Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.

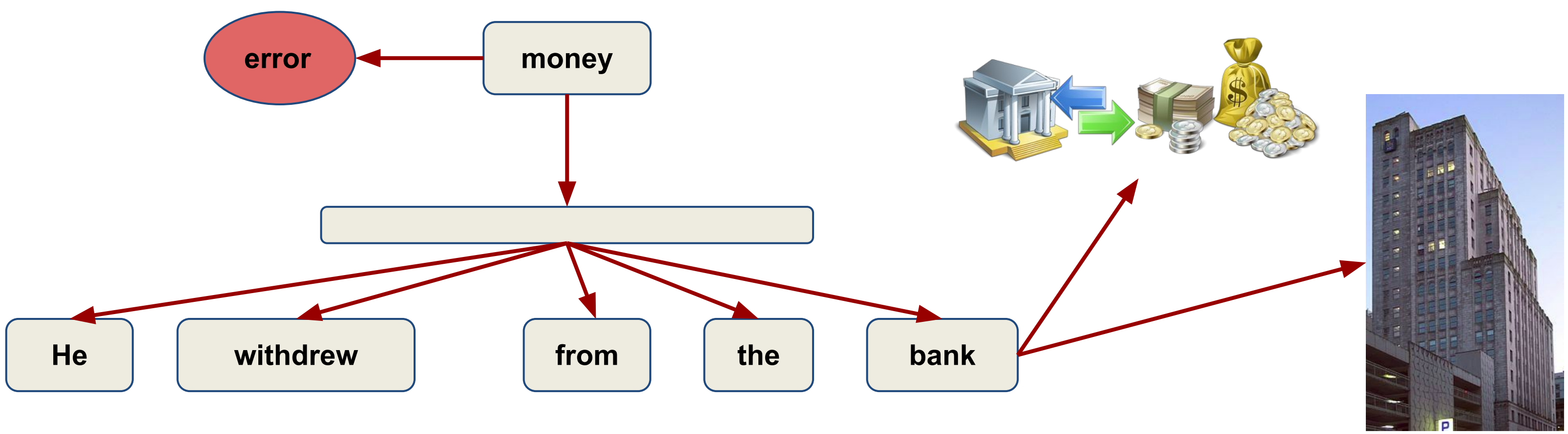


Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.



Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

2. Use a neural network where the update of word and sense embeddings is linked, exploiting *virtual* connections.

In this way it is possible to learn word and sense/synset embeddings jointly on a ***single training****.*

Methodology: Joint training of words and sense embeddings

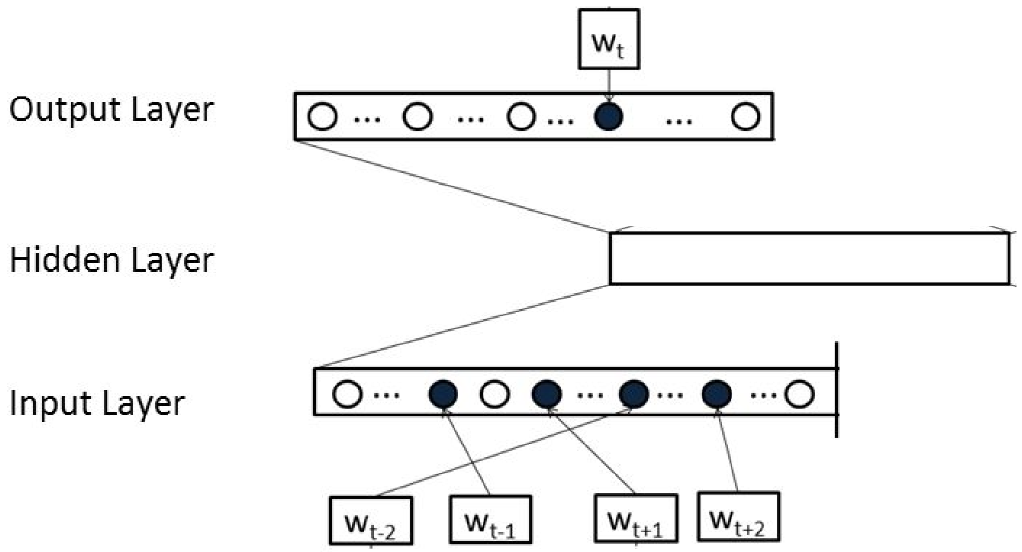
Once each word is connected to its set of senses *in context*, it is possible to **modify standard word embedding architectures** to take into account this information.

In this work we explore the CBOW architecture of Word2Vec (Mikolov et al. 2013) -> **SW2V** *(Senses and Words to Vectors)*.

Other neural network architectures could be explored as well (Skip-gram also included in the code).

Full architecture of W2V (Mikolov et al. 2013)

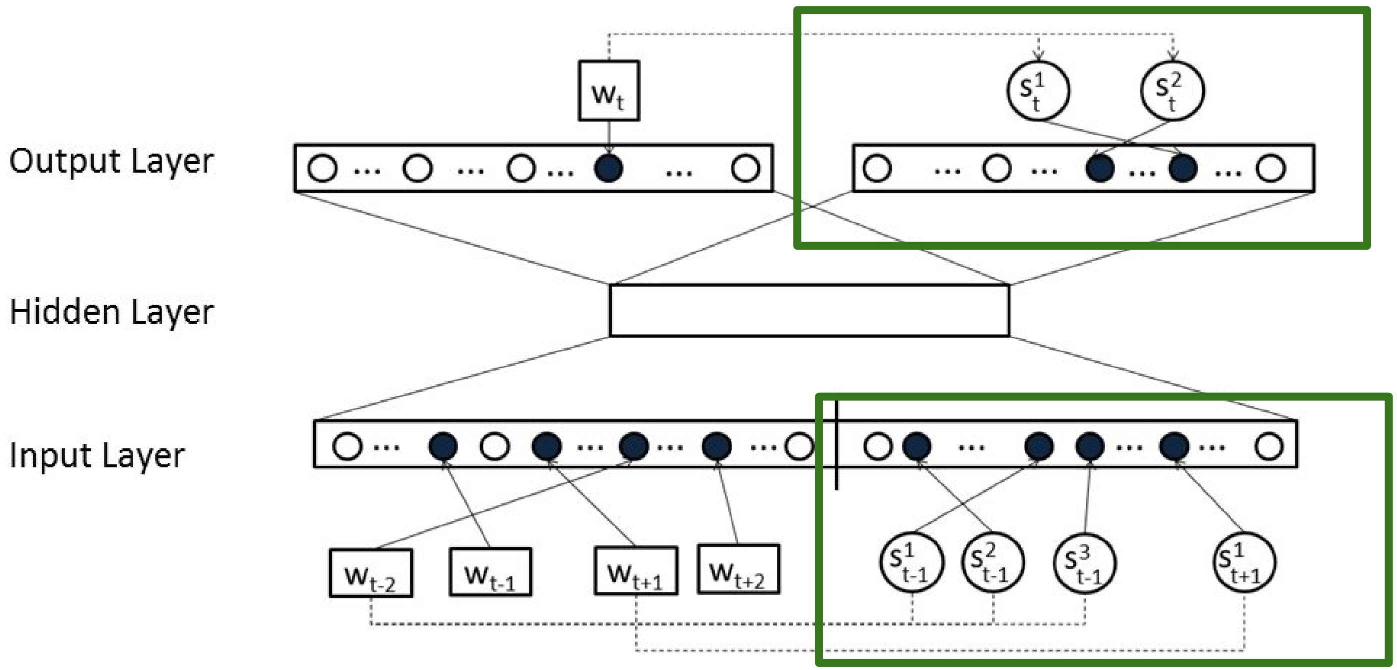
$$E = -\log(p(w_t | W^t))$$



Words and associated senses used both as input and output.

Full architecture of SW2V (this work)

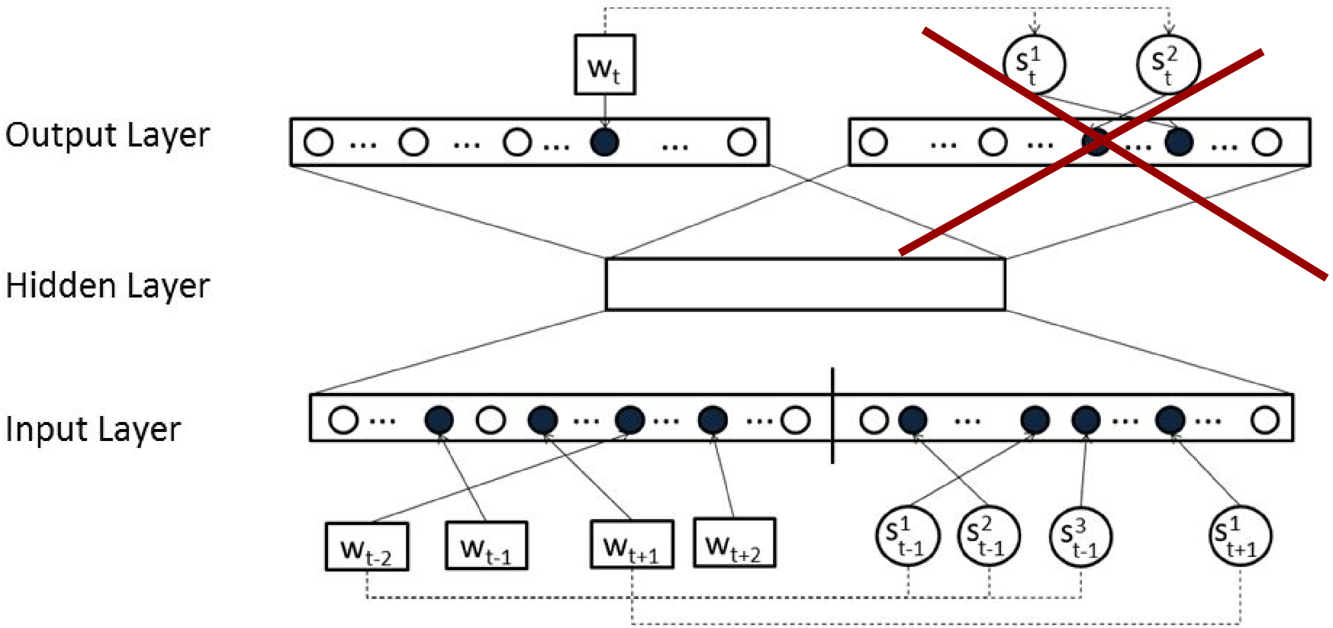
$$E = -\log(p(w_t|W^t, S^t)) - \sum_{s \in S^t} \log(p(s|W^t, S^t))$$



Words and associated senses used both as input and output.

Output layer alternatives: only words

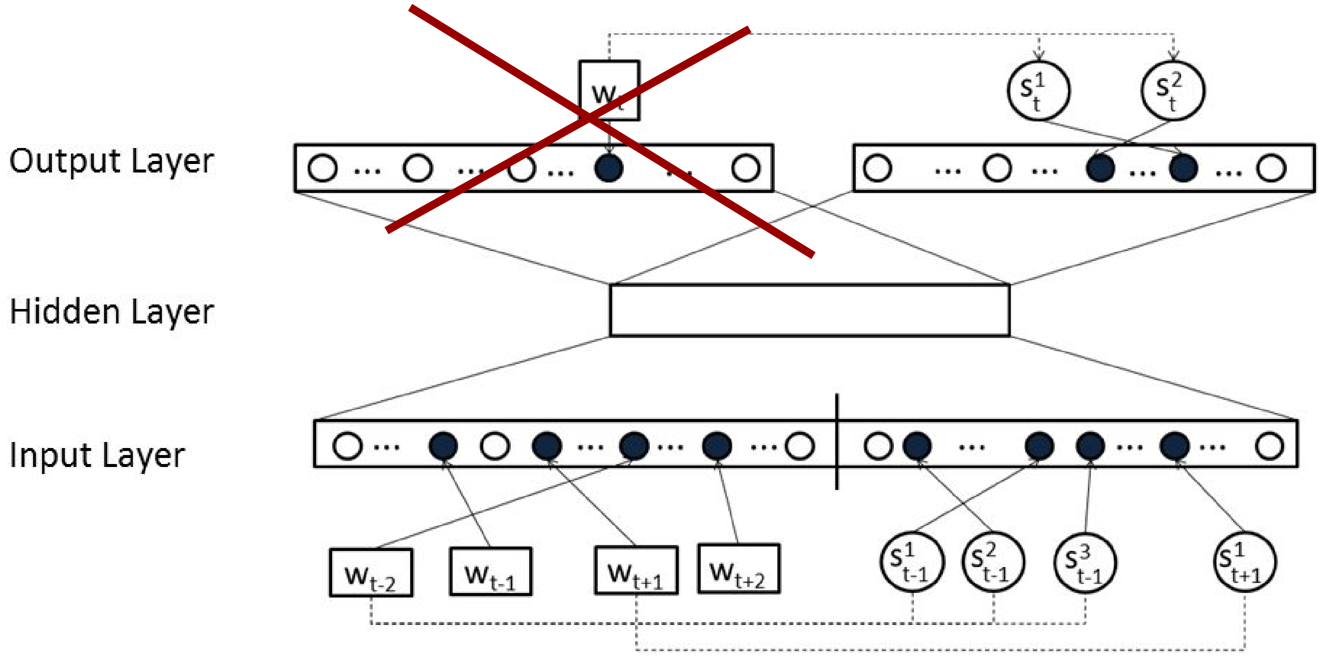
$$E=-\log(p(w_t|W^t,S^t)) \; - \cancel{\sum_{s\in S^t} \log(p(s|W^t,S^t))}$$



The architecture does not try to predict senses. No loss contribution from them.

Output layer alternatives: only senses

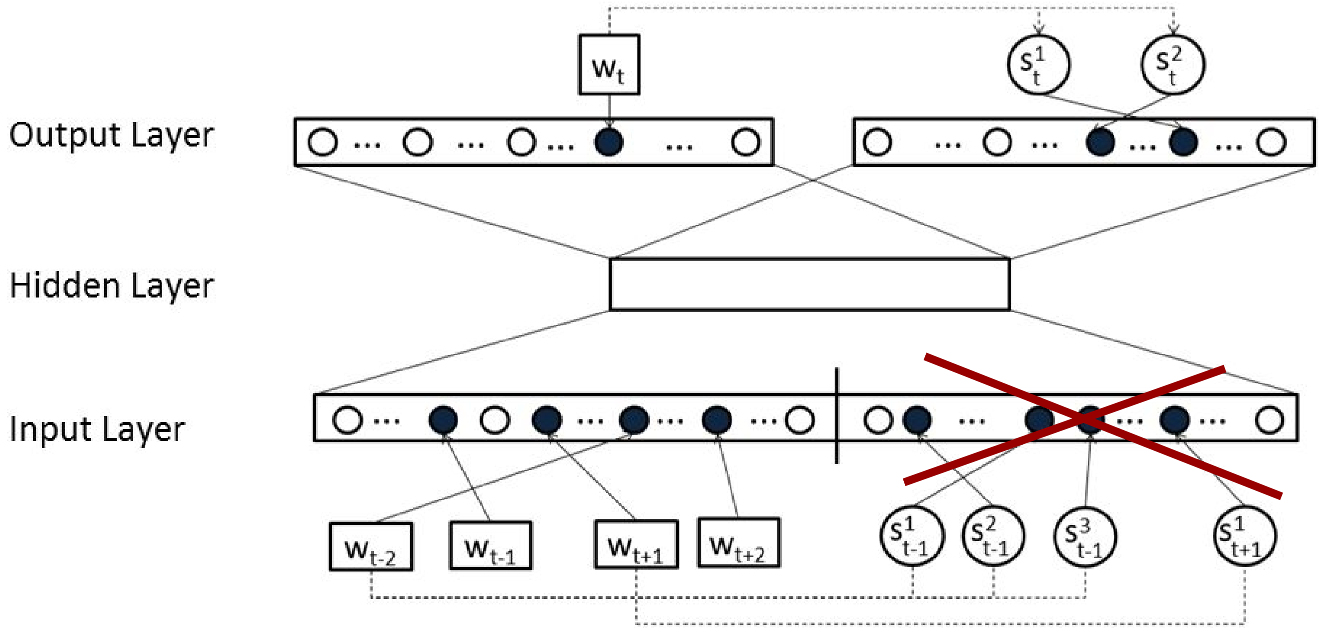
$$E=-\cancel{\log(p(w_t|W^t,S^t))} - \sum_{s \in S_t} \log(p(s|W^t,S^t))$$



The architecture does not try to predict words. No loss contribution from them.

Input layer alternatives: only words

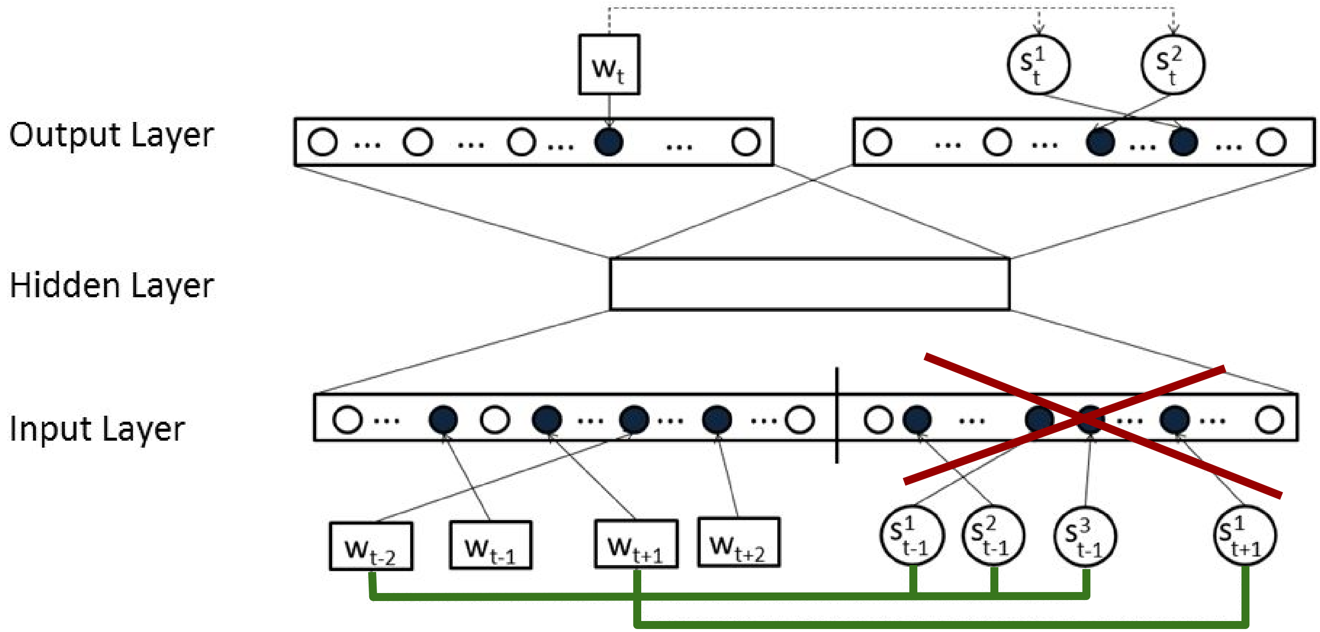
$$E=-\log(p(w_t|W^t,\cancel{S^t})) - \sum_{s\in S_t} \log(p(s|W^t,\cancel{S^t}))$$



Senses are not included in the input layer. Only words contribute to the hidden state. This way, during backpropagation sense embeddings do **not** receive any gradient.

Input layer alternatives: only words

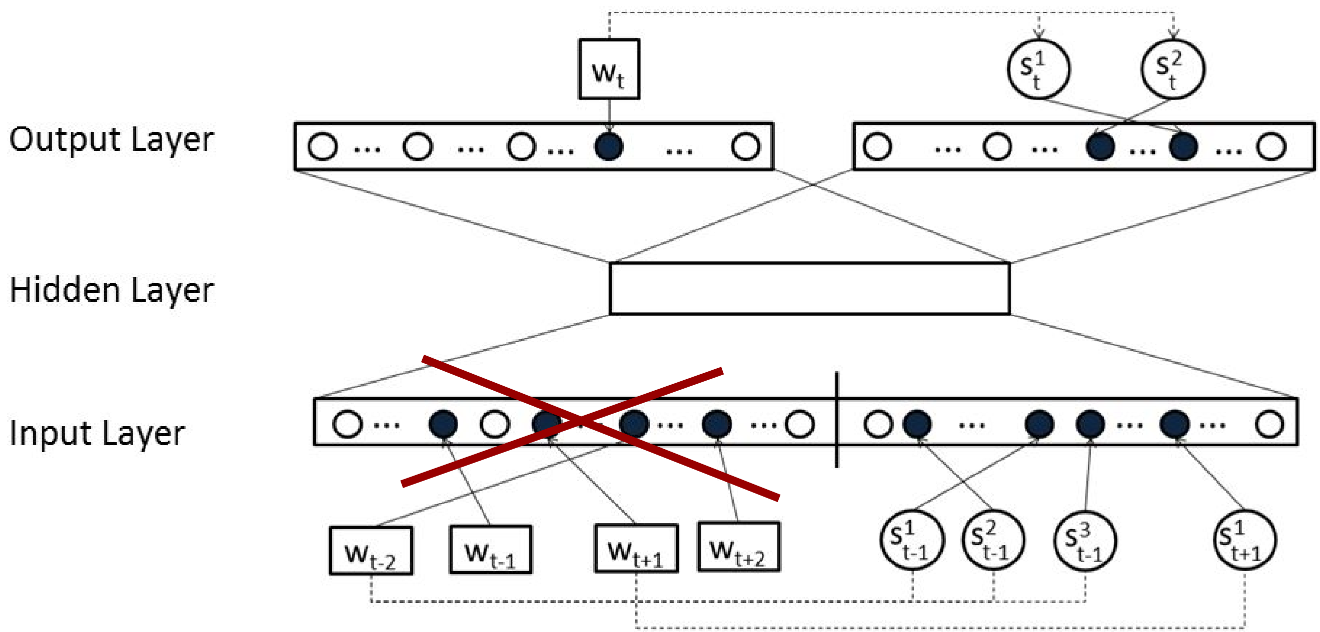
$$E=-\log(p(w_t|W^t,\cancel{S^t})) - \sum_{s\in S_t} \log(p(s|W^t,\cancel{S^t}))$$



During backpropagation, sense embeddings will receive the **same** gradient **of the word they are associated with**.

Input layer alternatives: only senses

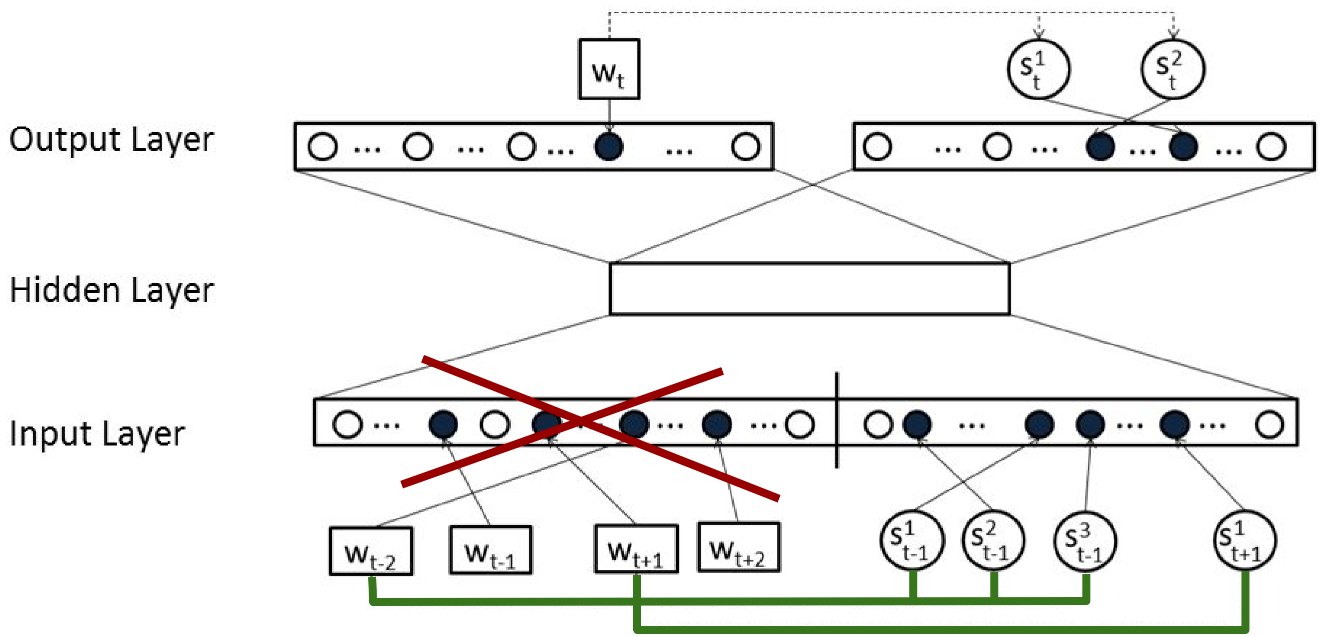
$$E = -\log(p(w_t|\cancel{W^t},S^t)) - \sum_{s\in S^t} \log(p(s|\cancel{W^t},S^t))$$



Words are not included in the input layer. Only senses contribute to the hidden state. This way, during backpropagation word embeddings do **not** receive any gradient.

Input layer alternatives: only senses

$$E=-\log(p(w_t|\cancel{W^t},S^t)) - \sum_{s\in S^t} \log(p(s|\cancel{W^t},S^t))$$



During backpropagation, their embeddings will receive the **same** gradient **of their associated senses**.

Analysis: Model configurations

We used word similarity for analyzing the **performance of sense embeddings** on each of the nine configurations.

- Best configuration -

- **Input layer:** Only senses
- **Output layer:** Both words and senses

Why? *(Intuition)* Co-occurrence information gets duplicated if both words and senses are included in the input layer.

Evaluation: Experimental setting

- **Best configuration** used in all experiments
- **Standard hyperparameters**
- Semantic networks used: **WordNet** and **BabelNet**
- Corpora used: **UMBC** and **Wikipedia**
- Experiments on:
    - **Word and sense interconnectivity** (qualitative)
    - **Word similarity**
    - **Sense clustering**

Evaluation: Comparison systems

Sense embeddings:

- ➢ *Chen et al. (2014)*
- ☆ ➢ AutoExtend *(Rothe and Schütze, 2015)*
- ➢ SensEmbed *(Iacobacci et al. 2015)*
- ➢ NASARI *(Camacho-Collados et al. 2016)*

Evaluation: Comparison systems

Sense embeddings:

- *Chen et al. (2014)*
- ☆ AutoExtend *(Rothe and Schütze, 2015)*
- SensEmbed *(Iacobacci et al. 2015)*
- NASARI *(Camacho-Collados et al. 2016)*

WordNet

BabelNet

Evaluation: Comparison systems

Sense embeddings:

- ➢ *Chen et al. (2014)*
- ☆ ➢ AutoExtend *(Rothe and Schütze, 2015)*
- ➢ SensEmbed *(Iacobacci et al. 2015)*
- ➢ NASARI *(Camacho-Collados et al. 2016)*

WordNet

BabelNet

Word embeddings:

- ➢ Word2Vec (Mikolov et al. 2013)
- ☆ ➢ Retrofitting (Faruqui et al. 2015)

Evaluation: Comparison systems

Sense embeddings:

- *Chen et al. (2014)*
- ☆ AutoExtend *(Rothe and Schütze, 2015)*
- SensEmbed *(Iacobacci et al. 2015)*
- NASARI *(Camacho-Collados et al. 2016)*

Word embeddings:

- Word2Vec (Mikolov et al. 2013)
- ☆ Retrofitting (Faruqui et al. 2015)

WordNet

BabelNet

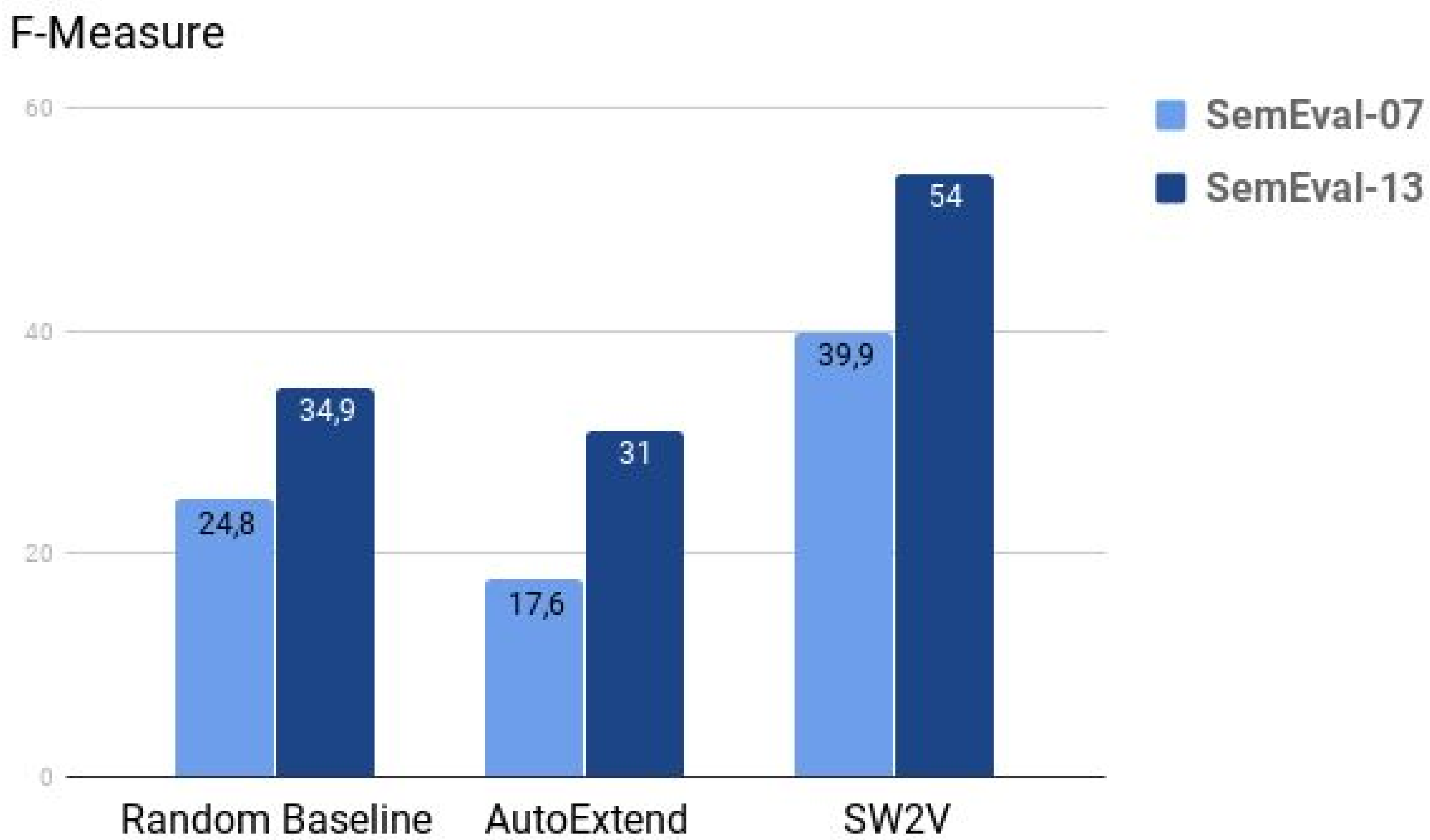
Evaluation: Word and sense interconnectivity

How coherent is the shared vector space of word and sense embeddings?

Intuition: the Most Frequent Sense (MFS) should be close to the word embedding -> Reasonably strong MFS baseline for WSD

Evaluation on two WSD datasets using the **embeddings as a MFS baseline** (closest sense embedding to its associated word embedding is selected).

Evaluation: Word and sense interconnectivity



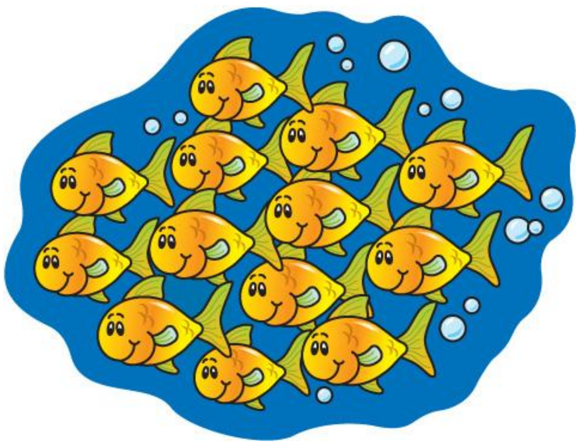
Word and sense interconnectivity: Example I

$company^2_n$ (military unit)

AutoExtend	SW2V
$company^9_n$	$battalion^1_n$
company	battalion
$company^8_n$	$regiment^1_n$
$company^6_n$	$detachment^4_n$
$company^7_n$	$platoon^1_n$
$company^1_v$	$brigade^1_n$
firm	regiment
$business^1_n$	$corps^1_n$
$firm^2_n$	brigade
$company^1_n$	platoon

Ten closest word and sense embeddings to the sense *company* (military unit)

Word and sense interconnectivity: Example II



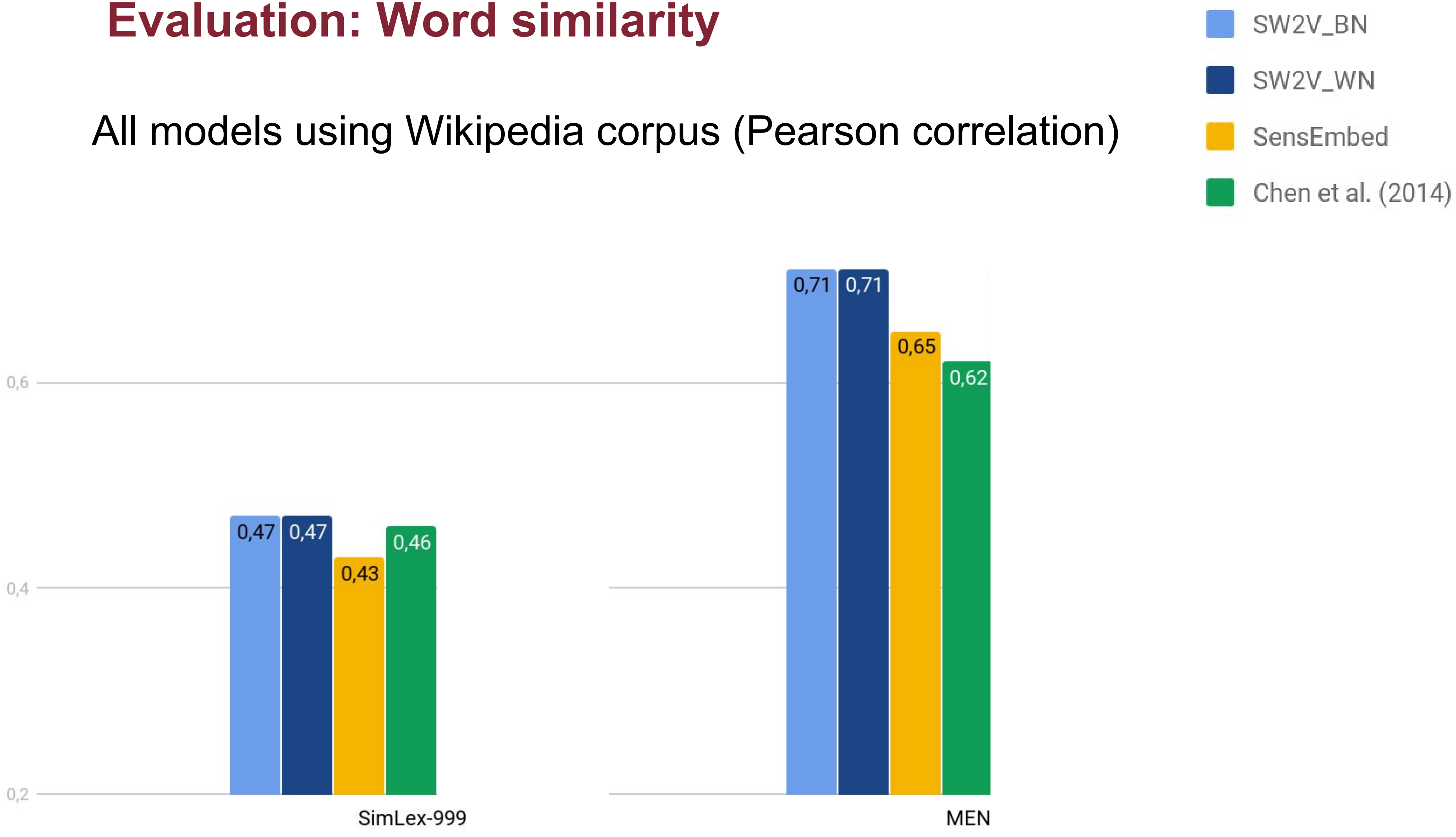
school^7_n (group of fish)

AutoExtend	SW2V
school	schools^7_n
school^4_n	sharks^1_n
school^6_n	sharks
school^1_v	shoals^3_n
school^3_n	fish^1_n
elementary	dolphins^1_n
schools	pods^3_n
elementary^3_a	eels
school^5_n	dolphins
elementary^1_a	whales^2_n

Ten closest word and sense embeddings to the sense school (group of fish)

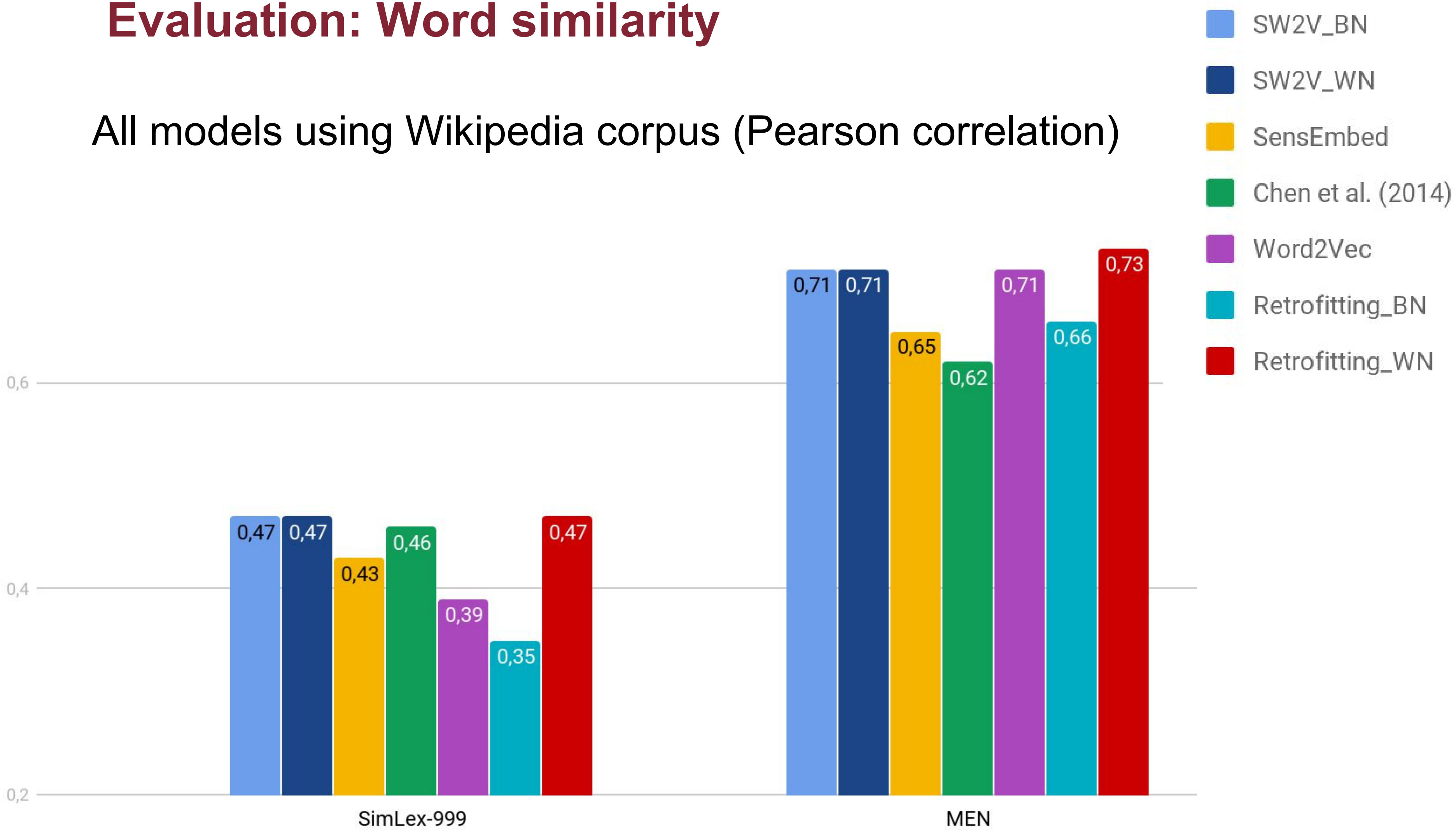
Evaluation: Word similarity

All models using Wikipedia corpus (Pearson correlation)



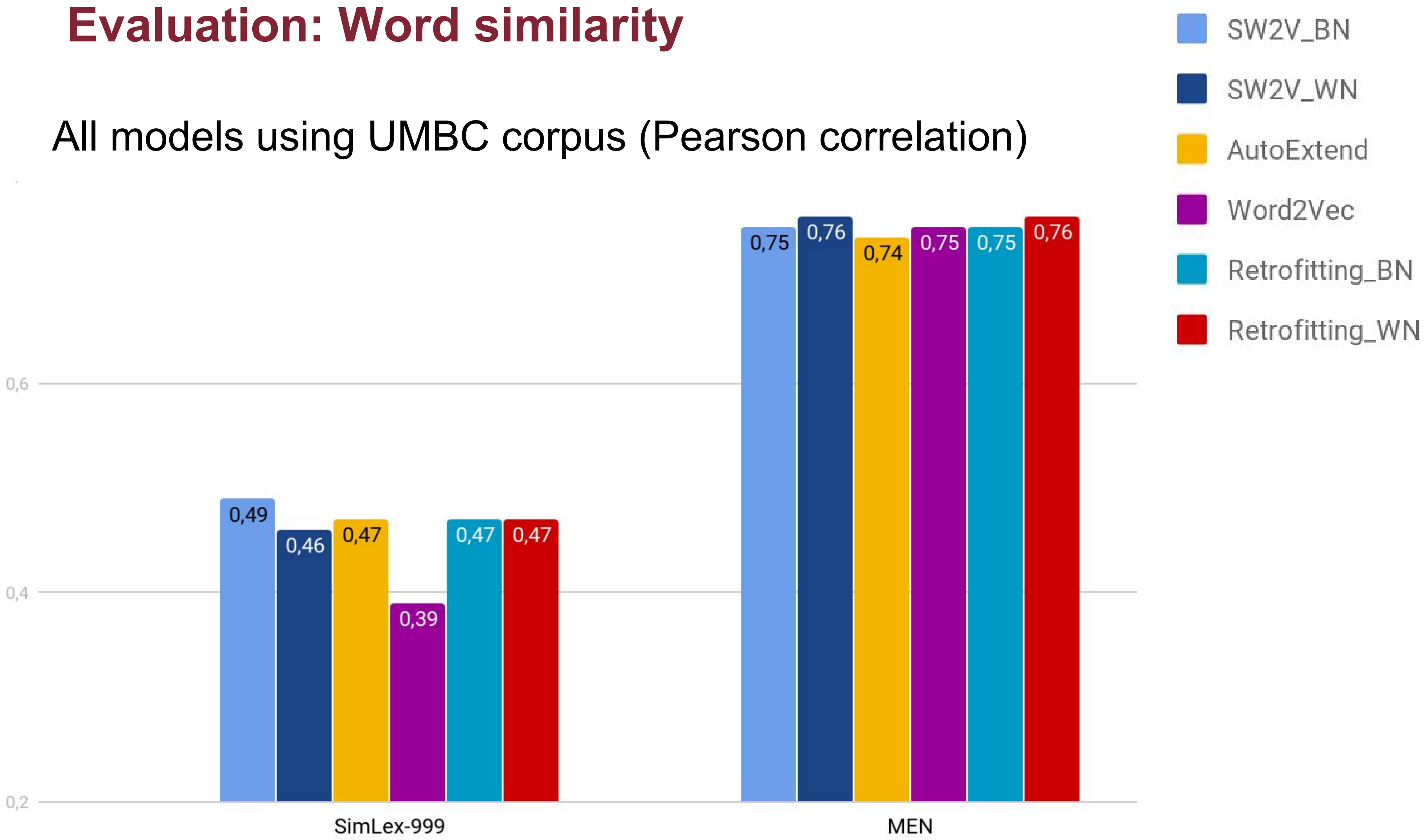
Evaluation: Word similarity

All models using Wikipedia corpus (Pearson correlation)



Evaluation: Word similarity

All models using UMBC corpus (Pearson correlation)

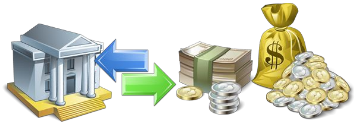


Evaluation: Sense clustering

Some sense inventories make a fine-grained distinction between senses, which can be harmful on downstream applications (Hovy et al. 2013, Pilehvar et al. 2017).

Example: *Bank*

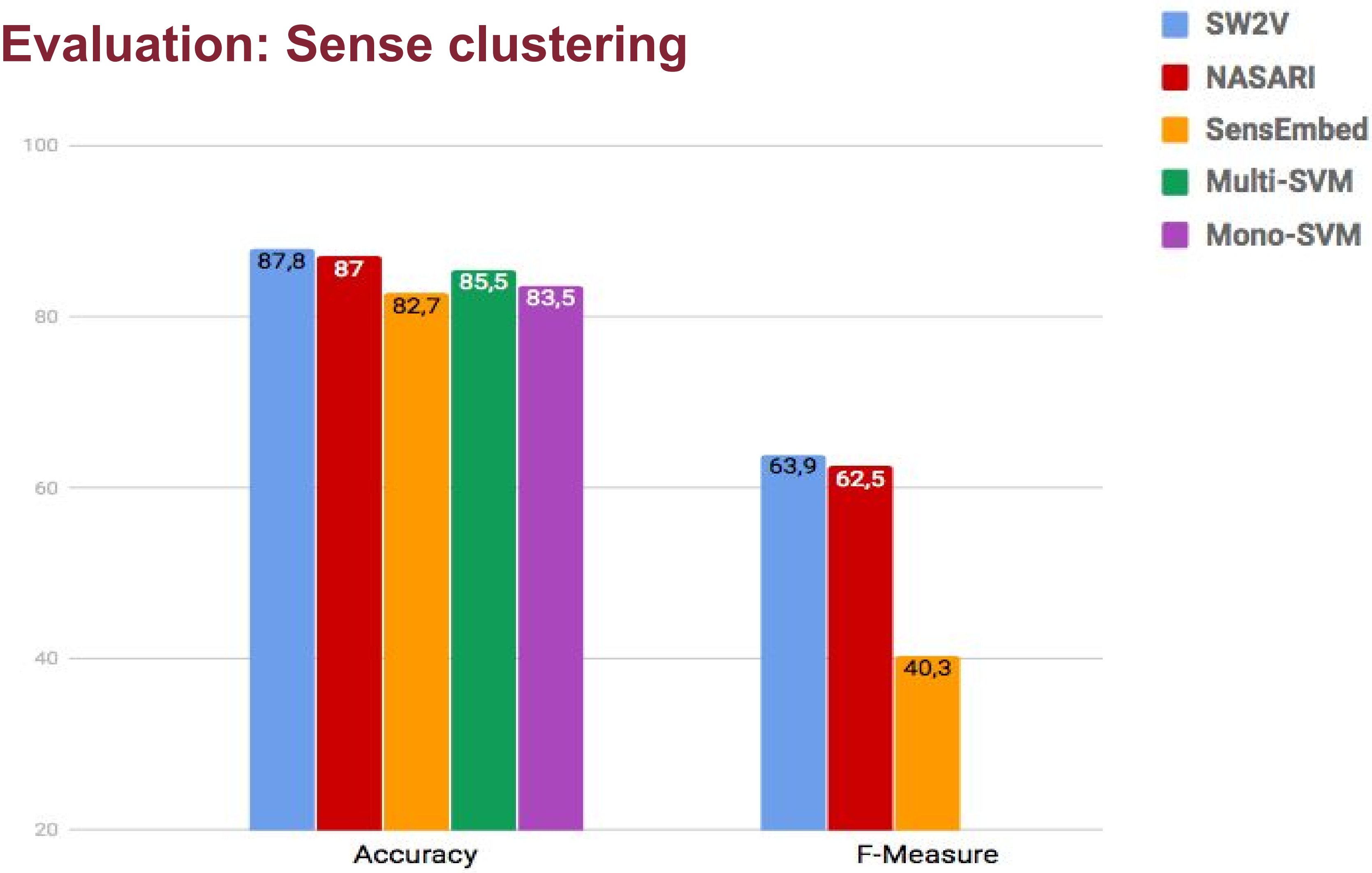
Institution



Physical building

Evaluation datasets (Dandala et al. 2013): Highly ambiguous words from past SemEval competitions.

Evaluation: Sense clustering



Conclusion

We presented SW2V: a neural architecture for **jointly learning word and sense embeddings** in the same vector space using text corpora and knowledge obtained from semantic networks.

Conclusion

We presented SW2V: a neural architecture for **jointly learning word and sense embeddings** in the same vector space using text corpora and knowledge obtained from semantic networks.

Future work:

- Exploiting our model for other linked representations such as **multilingual** or **Image-to-Text embeddings**.
- **Word Sense Disambiguation** and **Entity Linking**.
- Integrating our embeddings into **downstream NLP applications**, following the lines of *Pilehvar et al. (ACL 2017)*.

Conclusion

We presented SW2V: a neural architecture for **jointly learning word and sense embeddings** in the same vector space using text corpora and knowledge obtained from semantic networks.

Future work:

- Exploiting our model for other linked representations such as **multilingual** or **Image-to-Text embeddings**.
- **Word Sense Disambiguation** and **Entity Linking**.
- Integrating our embeddings into **downstream NLP applications**, following the lines of *Pilehvar et al. (ACL 2017)*.

http://lcl.uniroma1.it/sw2v

Thank you!

Code and pre-trained models available at

http://lcl.uniroma1.it/sw2v

SECRET SLIDES

Outline

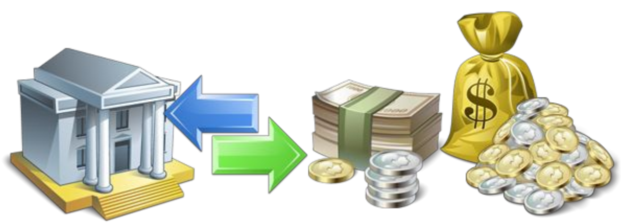
- Related work
- Our approach: SW2V (*Senses and Words to Vectors*)
  - Linking words and senses in context
  - Joint training of words and sense embeddings
- Evaluation

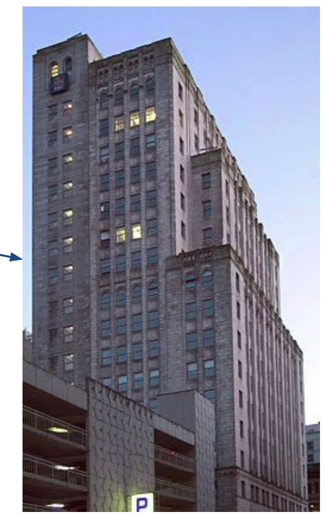
Methodology

Given as input a corpus and a semantic network:

1. Use a semantic network to link to each word its *associated senses in context*.

*He withdrew money from the **bank**.*





Joint training of word and sense embeddings

Once each word is connected to its set of senses *in context*, it is possible to modify standard word embedding models to take into account this information.

Formally, given a target word at position *t* we have a set of words:

$$W=\{w_{t-n}, \dots , w_t, \dots, w_{t+n}\} \quad \text{with} \qquad W^t=W \setminus w_t$$

and a set of associated senses:

$$S = \{S_{t-n}, \dots , S_t, \dots, S_{t+n}\} \quad \text{and} \qquad S^t=S \setminus S_t$$

with $S_i=\{s_i^1, \dots , s_i^{k,i}\}$ the senses associated with the i_{th} word.

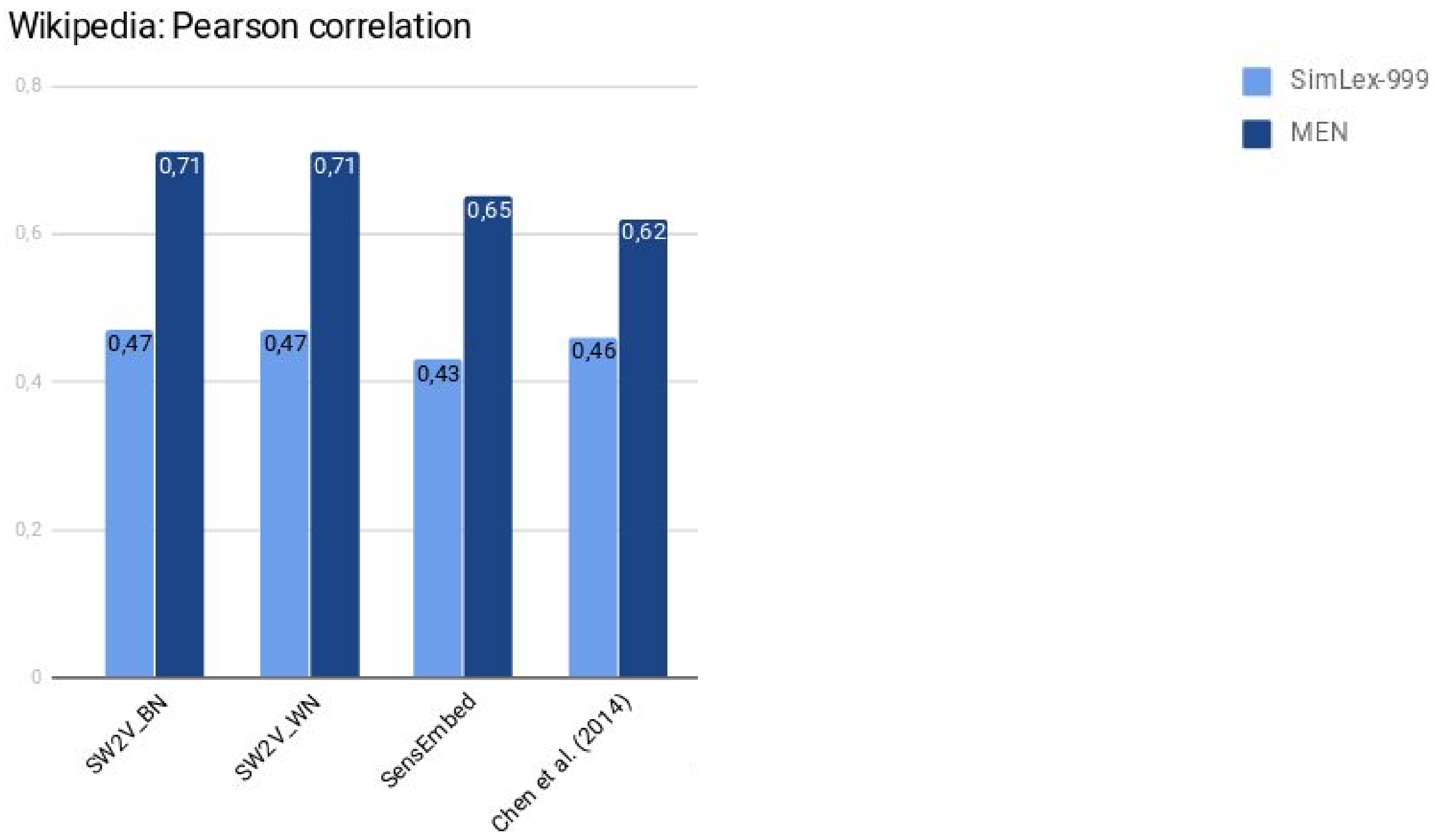
We aim at minimizing: $E=-\log(p(w_t|W^t,S^t)) - \sum_{s\in St} \log(p(s|W^t,S^t))$

Evaluation: Word similarity

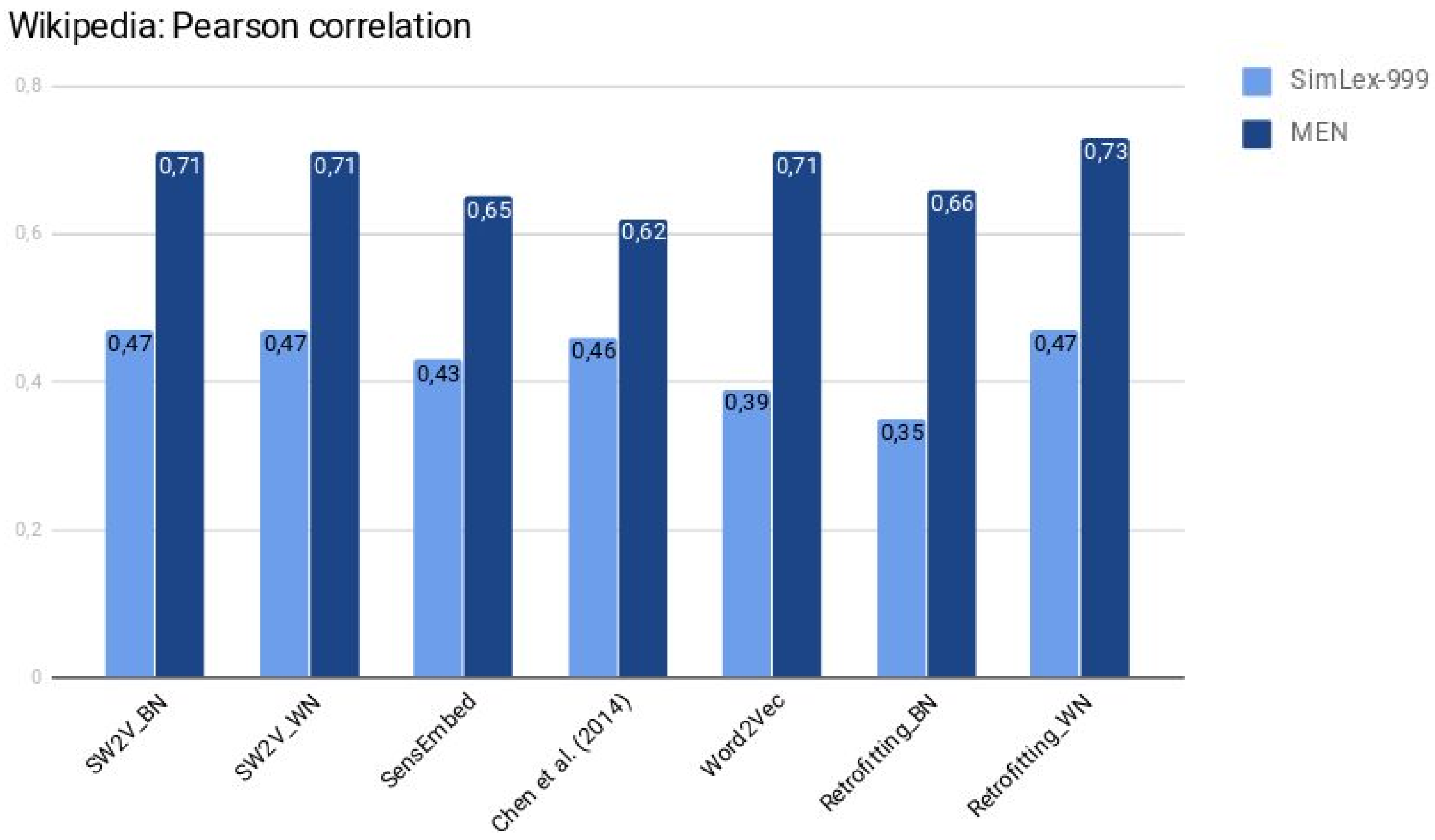
Sense Embeddings		SimLex-999		MEN	
System	**Corpus**	*r*	*p*	r	*p*
SW2V$_{BN}$	UMBC	**0.49**	**0.47**	0.75	0.75
SW2V$_{WN}$	UMBC	0.46	0.45	**0.76**	**0.76**
AutoExtend	UMBC	0.47	0.45	0.74	0.75
AutoExtend	Google-News	0.46	0.46	0.68	0.70
SW2V$_{BN}$	Wikipedia	0.47	0.43	0.71	0.73
SW2V$_{WN}$	Wikipedia	0.47	0.43	0.71	0.72
SensEmbed	Wikipedia	0.43	0.39	0.65	0.70
Chen et al. (2014)					

Word Embeddings		SimLex-999		MEN	
System	**Corpus**	*r*	*p*	r	*p*
Word2Vec	UMBC	0.39	0.39	0.75	0.75
Retrofitting$_{BN}$	UMBC	0.47	0.46	0.75	**0.76**
Retrofitting$_{WN}$	UMBC	0.47	0.46	**0.76**	**0.76**
Word2Vec	Wikipedia	0.39	0.38	0.71	0.72
Retrofitting$_{BN}$	Wikipedia	0.35	0.32	0.66	0.66
Retrofitting$_{WN}$	Wikipedia	0.47	0.44	0.73	0.73

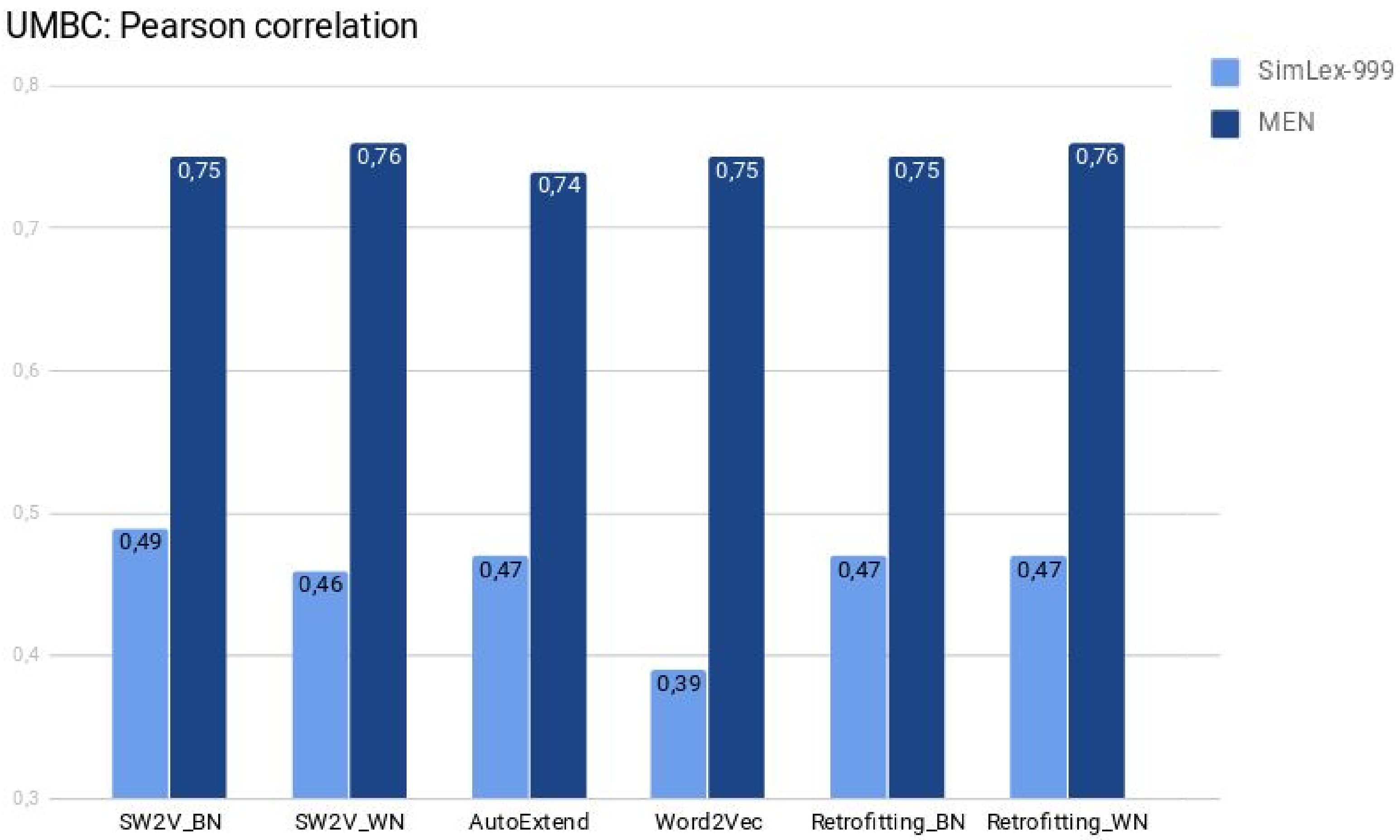
Evaluation: Word similarity



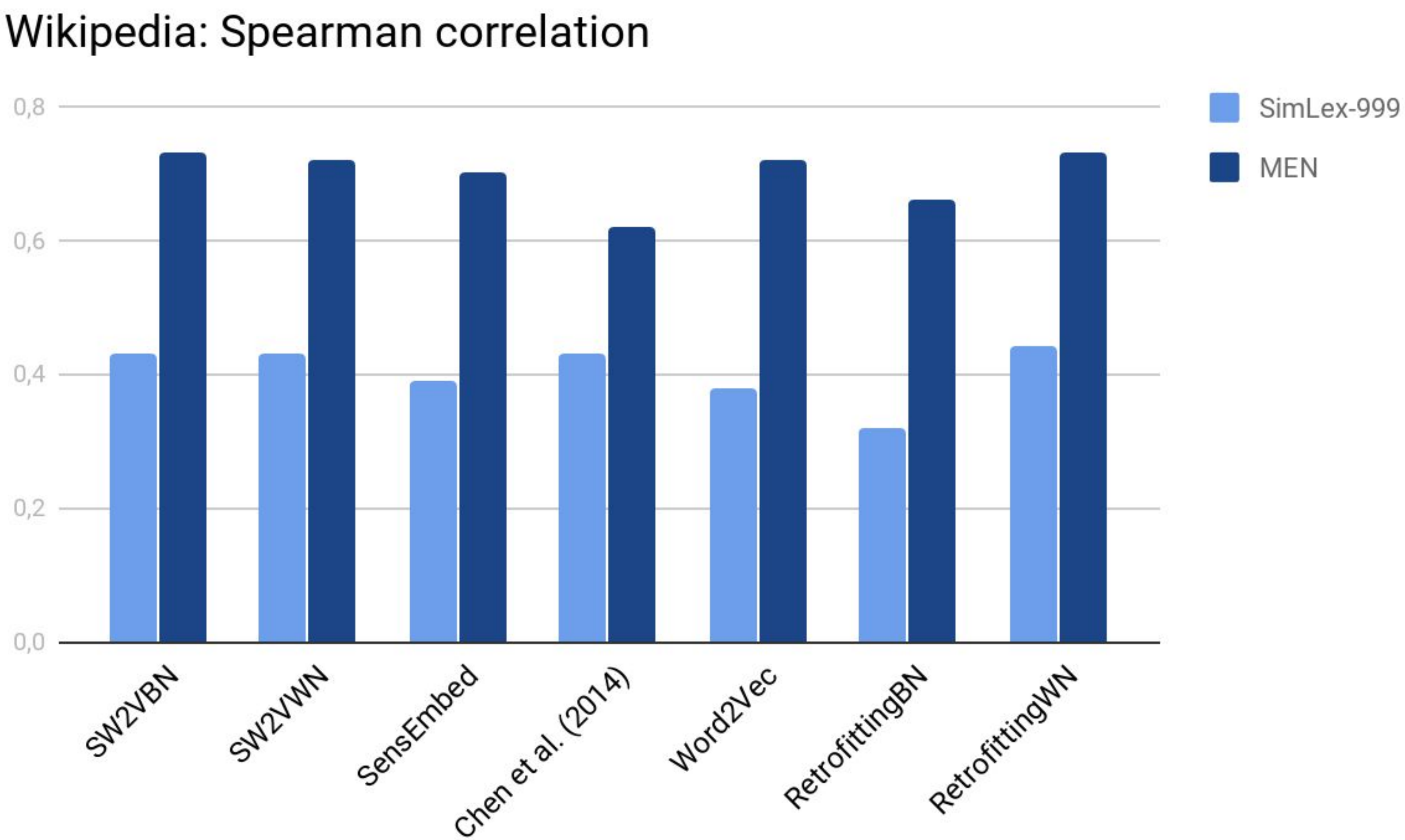
Evaluation: Word similarity



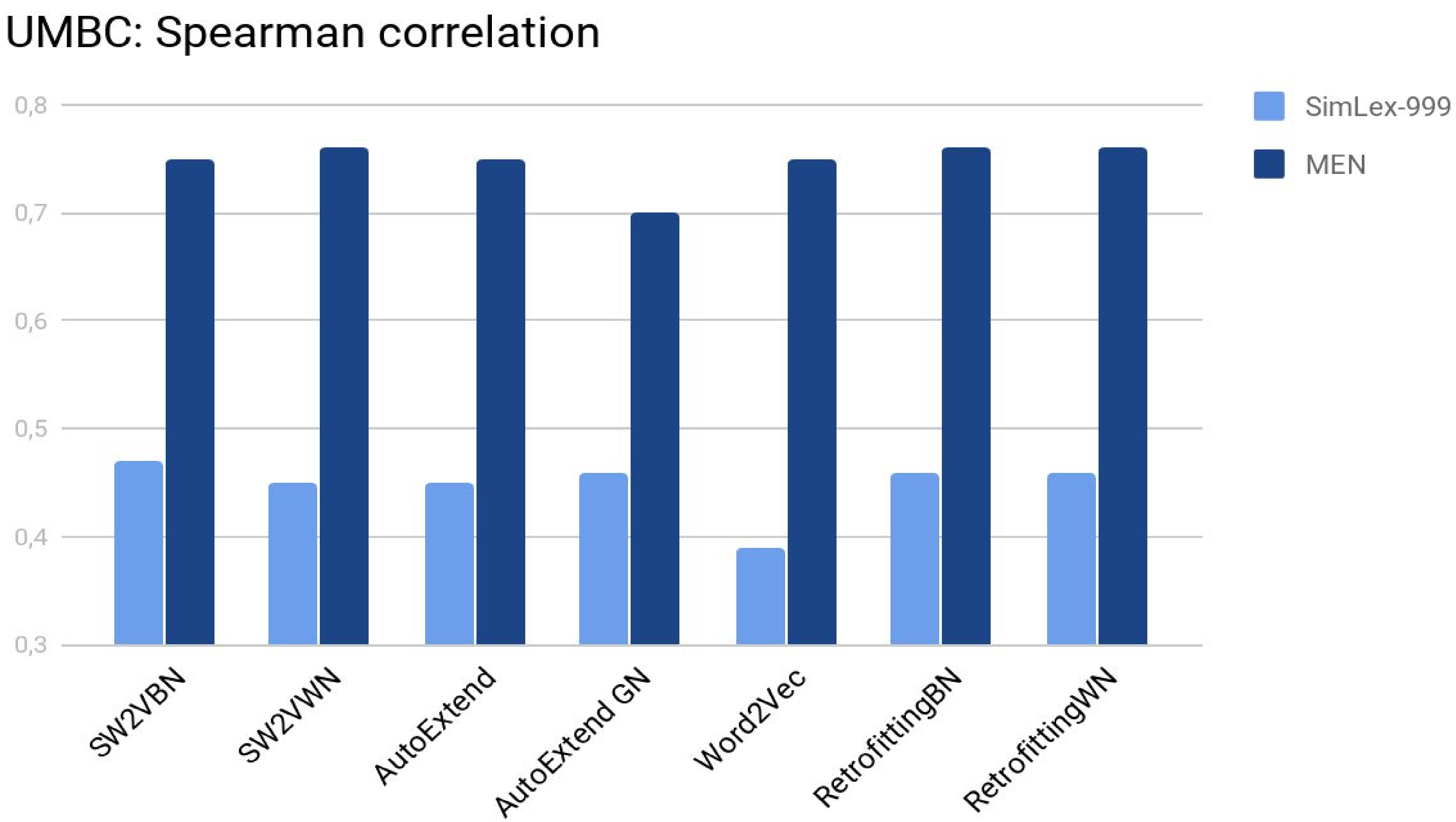
Evaluation: Word similarity



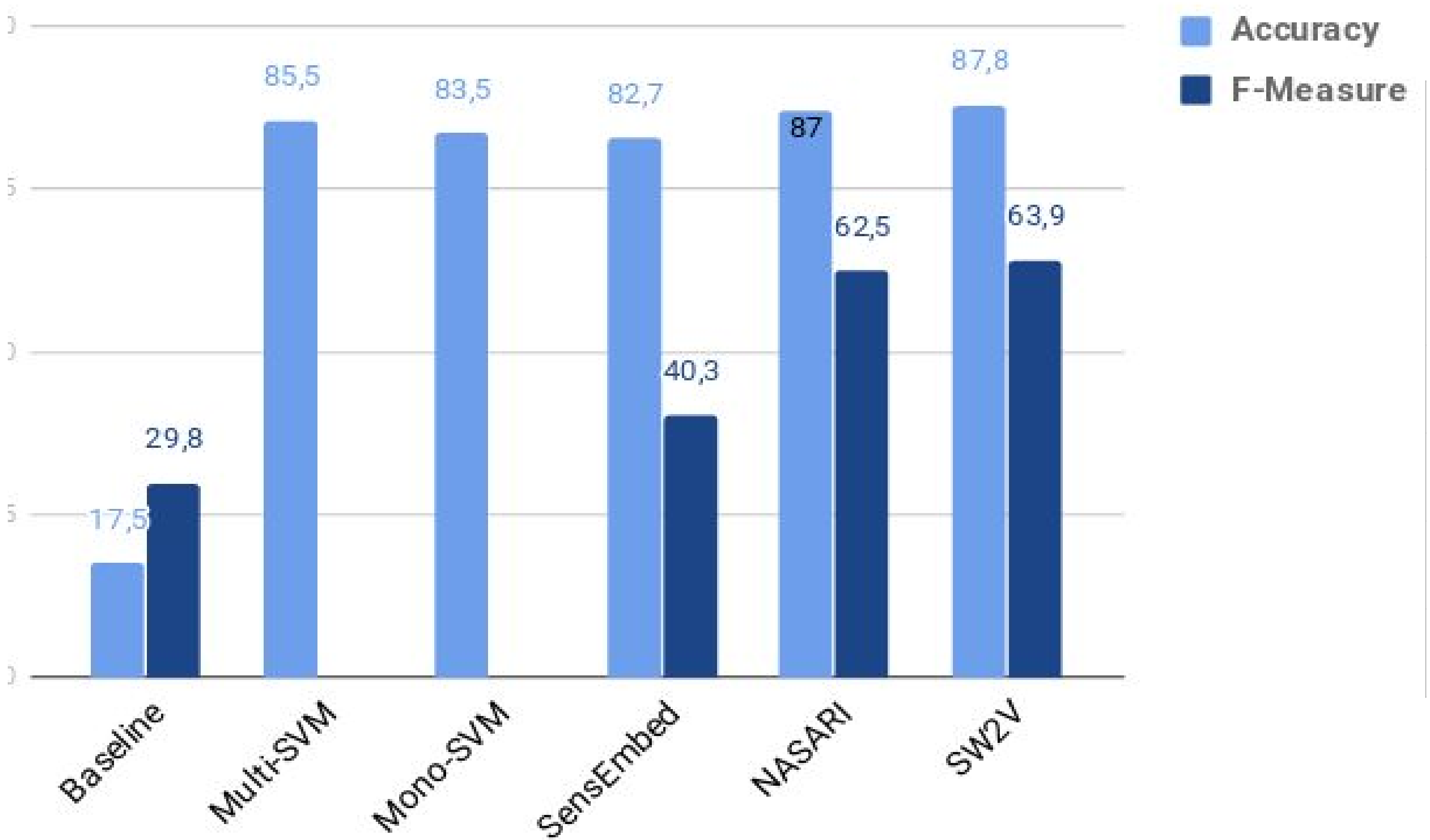
Evaluation: Word similarity



Evaluation: Word similarity



Evaluation: Sense clustering



Evaluation: Sense clustering

	Accuracy	F-Measure
SW2V	**87.8**	**63.9**
SensEmbed	82.7	40.3
NASARI	87.0	62.5
Multi-SVM	85.5	-
Mono-SVM	83.5	-
Baseline	17.5	29.8

Word and sense interconnectivity

	SemEval-07	SemEval-13
SW2V	**39.9**	**54.0**
AutoExtend	17.6	31.0
Baseline	24.8	34.9