The Role of Context Types and Dimensionality in Learning Word Embeddings

Oren Melamud, David McClosky, Siddharth Patwardhan, Mohit Bansal

NAACL, 2016

What’s a good word embedding for my task?

so many choices...

What's a good word embedding for my task?

Useful in supervised tasks:

- As pre-training initialization
- With limited supervised data

so many choices...

What's a good word embedding for my task?

Useful in supervised tasks:

- As pre-training initialization
- With limited supervised data

Applied to various tasks:

- Dependency Parsing
- Named Entity Recognition
- Co-reference Resolution
- Sentiment Analysis
- More...

so many choices...

Plethora of Word Embeddings

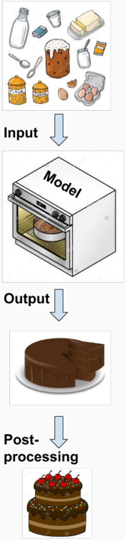
Easy to obtain

- Off-the-shelf
- Do-it-yourself toolkits

so many choices...

Plethora of Word Embeddings

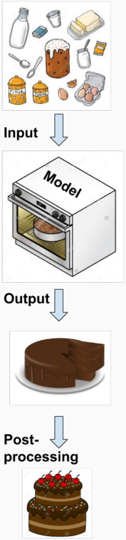
Lots of choices to make



Plethora of Word Embeddings

Lots of choices to make

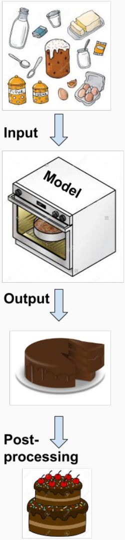
1. Input
   - Context type
     (BOW-N, syntactic, ...)
   - Learning corpus



Plethora of Word Embeddings

Lots of choices to make

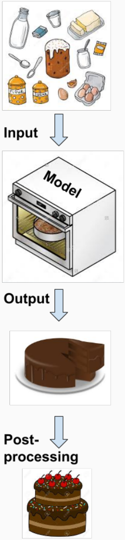
1. Input
   - Context type (BOW-N, syntactic, ...)
   - Learning corpus
2. Computational model
   - Model type (word2vec, GloVe, ...)
   - Hyperparameters



Plethora of Word Embeddings

Lots of choices to make

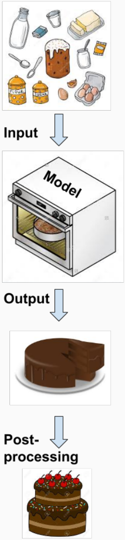
1. Input
   - Context type (BOW-N, syntactic, ...)
   - Learning corpus
2. Computational model
   - Model type (word2vec, GloVe, ...)
   - Hyperparameters
3. Output
   - Dimensionality (is higher always better?)



Plethora of Word Embeddings

Lots of choices to make

1. Input
   - Context type (BOW-N, syntactic, ...)
   - Learning corpus
2. Computational model
   - Model type (word2vec, GloVe, ...)
   - Hyperparameters
3. Output
   - Dimensionality (is higher always better?)
4. Post-processing
   - Ensembles, retrofitting, ...

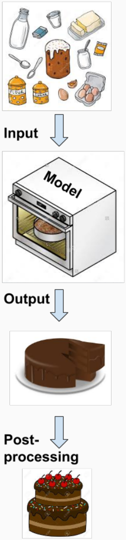


Our Focus

Choices we explore:

1. Input
   - Context type
     (BOW-N, syntactic, substitute)
   - Wikipedia + Gigaword + UMBC (web)
2. Computational model
   - word2vec
3. Output
   - Dimensionality
     (is higher always better?)
4. Post-processing
   - Embeddings combinations
     (concat, SVD, CCA)

Evaluated extensively on
intrinsic and extrinsic tasks



Our Focus

Research questions:

- Do intrinsic benchmarks predict extrinsic performance?

Our Focus

Research questions:

- Do intrinsic benchmarks predict extrinsic performance?
- Tune context type and dimensionality per extrinsic task?

Our Focus

Research questions:

- Do intrinsic benchmarks predict extrinsic performance?
- Tune context type and dimensionality per extrinsic task?
- Can we benefit from combining different context types?

Additional Contribution

A new word2vec context type (substitute-based)

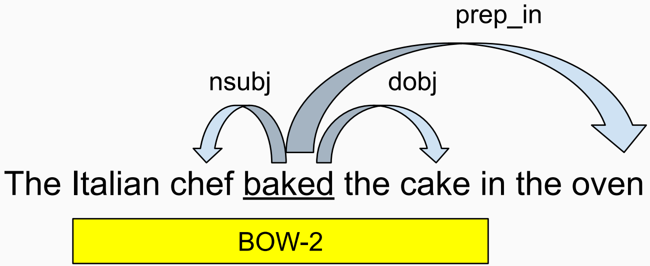
- Based on *n*-gram language modeling

Outline

- Context types and dimensionality
- Combining context types
- Conclusions

Context Types and Dimensionality

Common Context Types



BOW-2 Contexts	
t	c
baked	Italian
baked	chef
baked	the
baked	cake

Dependency Contexts	
t	c
baked	nsubj:chef
baked	dobj:cake
baked	prep_in:oven

Learning word2vec Skip-gram Embeddings

BOW-2 Contexts	
t	c
baked	Italian
baked	chef
baked	the
baked	cake

Dependency Contexts	
t	c
baked	nsubj:chef
baked	dobj:cake
baked	prep_in:oven

$$\sum_{(t,c) \in PAIRS} \left(\log \sigma(v'_c \cdot v_t) + \sum_{neg \in NEGS_{(t,c)}} \log \sigma(-v'_{neg} \cdot v_t) \right)$$

Substitute-based Contexts

Potential substitutes encode the context (Yuret, 2012)

The Italian chef <u>baked</u> the cake in the oven

Potential substitutes encode the context (Yuret, 2012)

The Italian chef _____ the cake in the oven

0.50 put
0.25 baked
0.15 cooked
0.10 forgot

Substitute-based Contexts

Potential substitutes encode the context (Yuret, 2012)

The Italian chef <u>baked</u> the cake in the oven

0.50 put
0.25 baked
0.15 cooked
0.10 forgot

Substitute Contexts		
t	s	$w_{t,s}$
baked	put	0.50
baked	baked	0.25
baked	cooked	0.15
baked	forgot	0.10

word2vec with Substitute-based Contexts

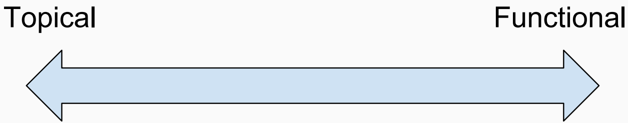
Substitute Contexts		
t	s	$w_{t,s}$
baked	put	0.50
baked	baked	0.25
baked	cooked	0.15
baked	forgot	0.10

$$\sum_{(t,s)\in PAIRS} \mathbf{w_{t,s}} \cdot \left(\log \sigma(v'_s \cdot v_t) + \sum_{neg\in NEGS_{(t,s)}} \log \sigma(-v'_{neg} \cdot v_t) \right)$$

‘Flavors’ of Similarity

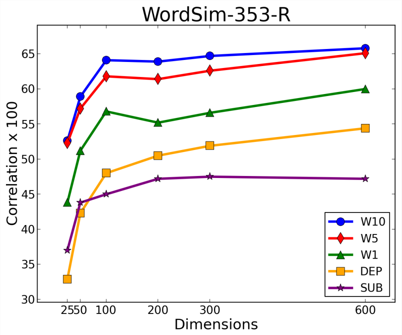
Top-5 closest words to ‘**playing**’

W-10	DEP	SUB
played	play	singing
play	played	rehearsing
plays	understudying	performing
professionally	caddying	composing
player	plays	running

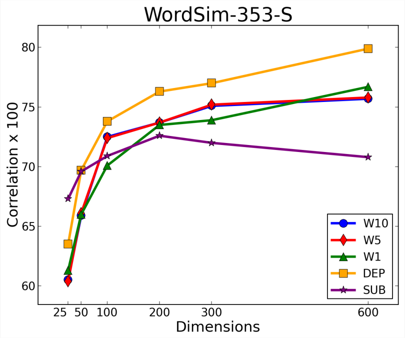


Small context windows also yield ‘functional’ similarity

Intrinsic Evaluations - Word Similarity



Topical
(lion:zoo)



Functional
(lion:cat)

* Similar results for SimLex-999

- Context type matters
- Higher dimensionality is generally better

Extrinsic Evaluations

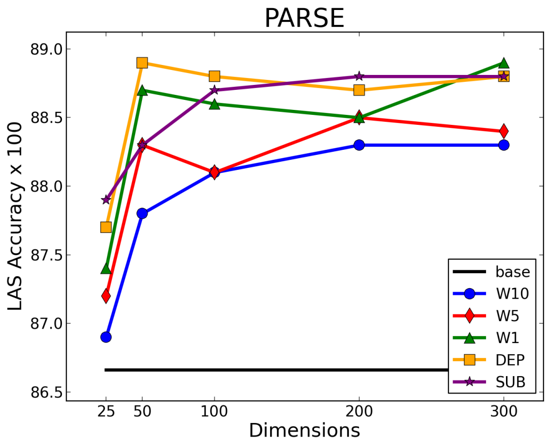
Can we find similar patterns in extrinsic tasks?

Extrinsic Evaluations

System	Benchmark
Stanford NN Dependency Parser Chen & Manning (2014)	PTB
Named Entity Recognition Turian et al. (2010)	CoNLL-2003 shared task
Co-reference Resolution Durrett & Klein (2013) Full features + embeddings	CoNLL-2012 shared task
Sentiment Analysis Average of embeddings with logistic regression	Sentence-level Sentiment Treebank Socher et al. (2013)

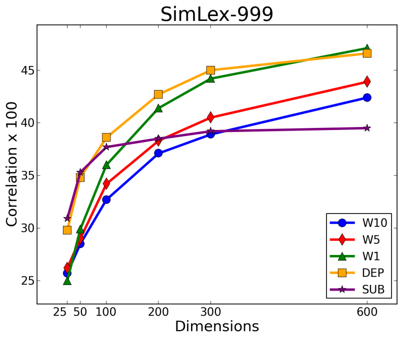
*Only dev-set experiments

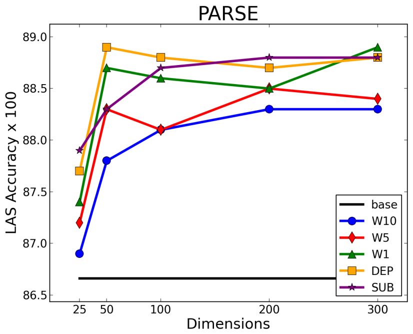
Extrinsic Evaluations - Parsing



- Preference for ‘functional’ embeddings
- Best performance at $d = 50$ (due to limited supervision?)

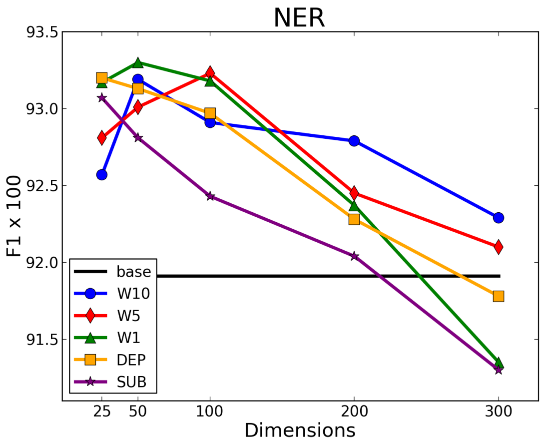
Extrinsic Evaluations - Parsing





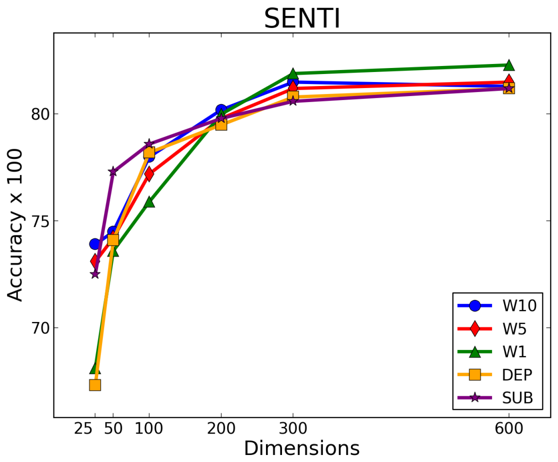
- Similar context type preferences
- But different dimensionality preferences

Extrinsic Evaluations - NER



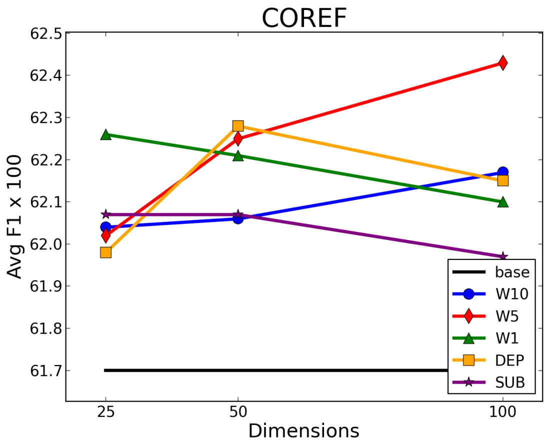
- Best performance at $d = 50$
- No clear context type preference

Extrinsic Evaluations - Sentiment Analysis



- No context type preference
- Higher dimensionality is better

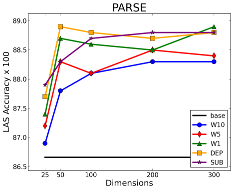
Extrinsic Evaluations - Coreference Resolution

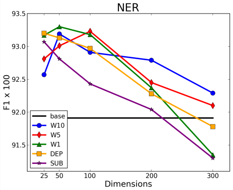


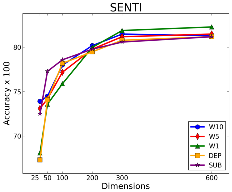
- Small performance diffs (competitive non-embedding features)

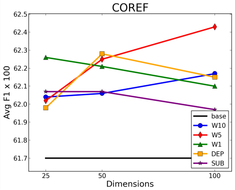
Extrinsic Evaluations - Summary

- Correlation with intrinsic results



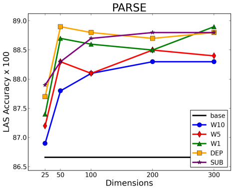


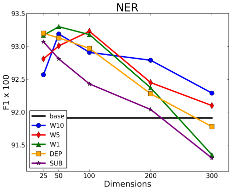


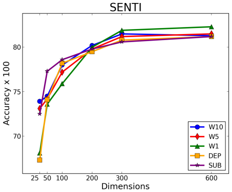


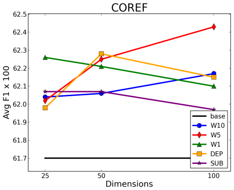
Extrinsic Evaluations - Summary

- Correlation with intrinsic results
- Dimensionality preferences



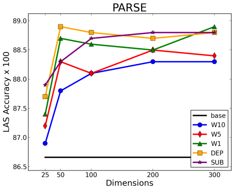


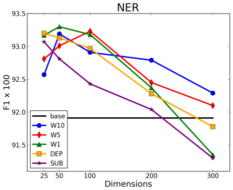


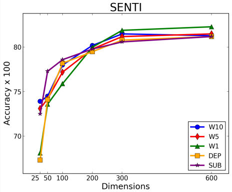


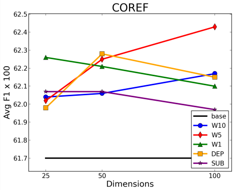
Extrinsic Evaluations - Summary

- Correlation with intrinsic results
- Dimensionality preferences
- Context type preferences





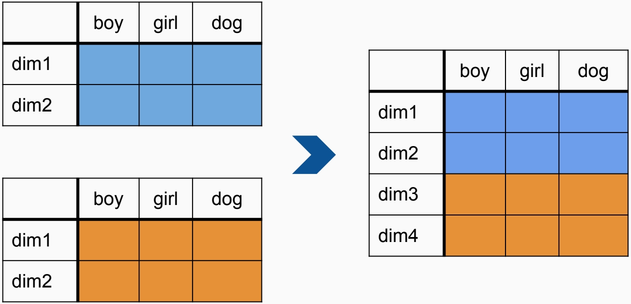




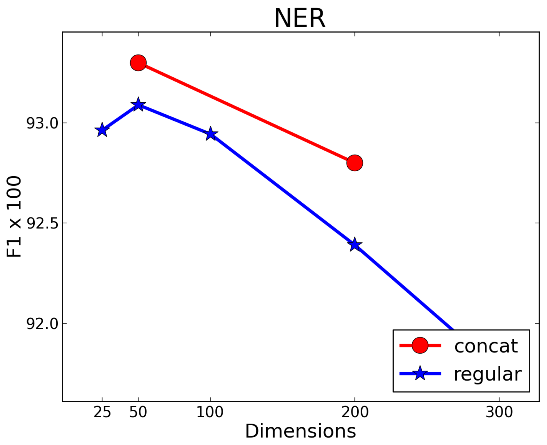
Context Combinations

Embeddings Concatenation

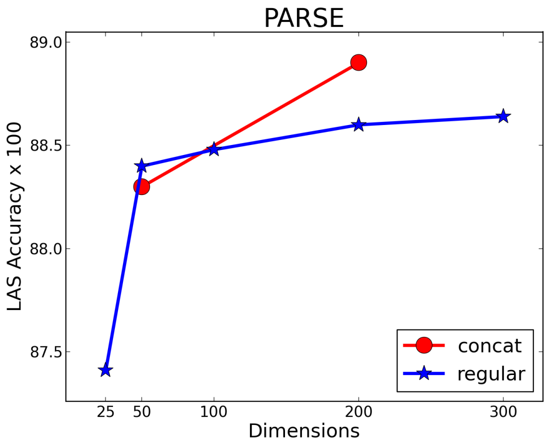
Let the classifier choose the valuable information:



Concatenation

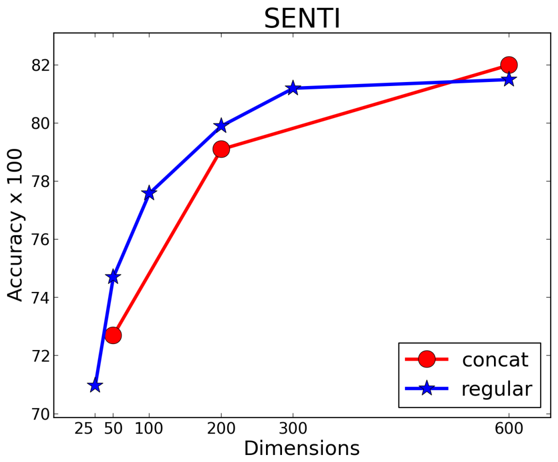


Concatenation



Concat helps when 'regular' increase in dimensionality is 'exhausted'

Concatenation



Concat helps when 'regular' increase in dimensionality is 'exhausted'

Concatenation

‘Topical’+‘Functional’ concats worked best

- W10 + SUB
- W10 + W1
- W10 + DEP

Compressed Combinations

- Compression via SVD or CCA degrades performance
- Better let the task-specific classifier ‘choose’ the relevant information

Conclusions

Summary

- Do intrinsic benchmarks predict extrinsic performance?

Summary

- Do intrinsic benchmarks predict extrinsic performance? NO

Summary

- Do intrinsic benchmarks predict extrinsic performance? NO
- Tune context type and dimensionality per extrinsic task?

Summary

- Do intrinsic benchmarks predict extrinsic performance? NO
- Tune context type and dimensionality per extrinsic task? YES

Summary

- Do intrinsic benchmarks predict extrinsic performance? NO
- Tune context type and dimensionality per extrinsic task? YES
- Can we benefit from combining different context types?

Summary

- Do intrinsic benchmarks predict extrinsic performance? NO
- Tune context type and dimensionality per extrinsic task? YES
- Can we benefit from combining different context types? MAYBE

Summary

- Do intrinsic benchmarks predict extrinsic performance? NO
- Tune context type and dimensionality per extrinsic task? YES
- Can we benefit from combining different context types? MAYBE

Thank you

and

happy cooking!