Modeling and Learning Semantic Co-Compositionality through Prototype Projections and Neural Networks

Masashi Tsubaki, Kevin Duh, Masashi Shimbo, Yuji Matsumoto

Nara Institute of Science and Technology (NAIST), Japan

Two contributions in our work

New model of
compositionality
in word vector space

Unsupervised word vector
re-training algorithm
considering compositionality

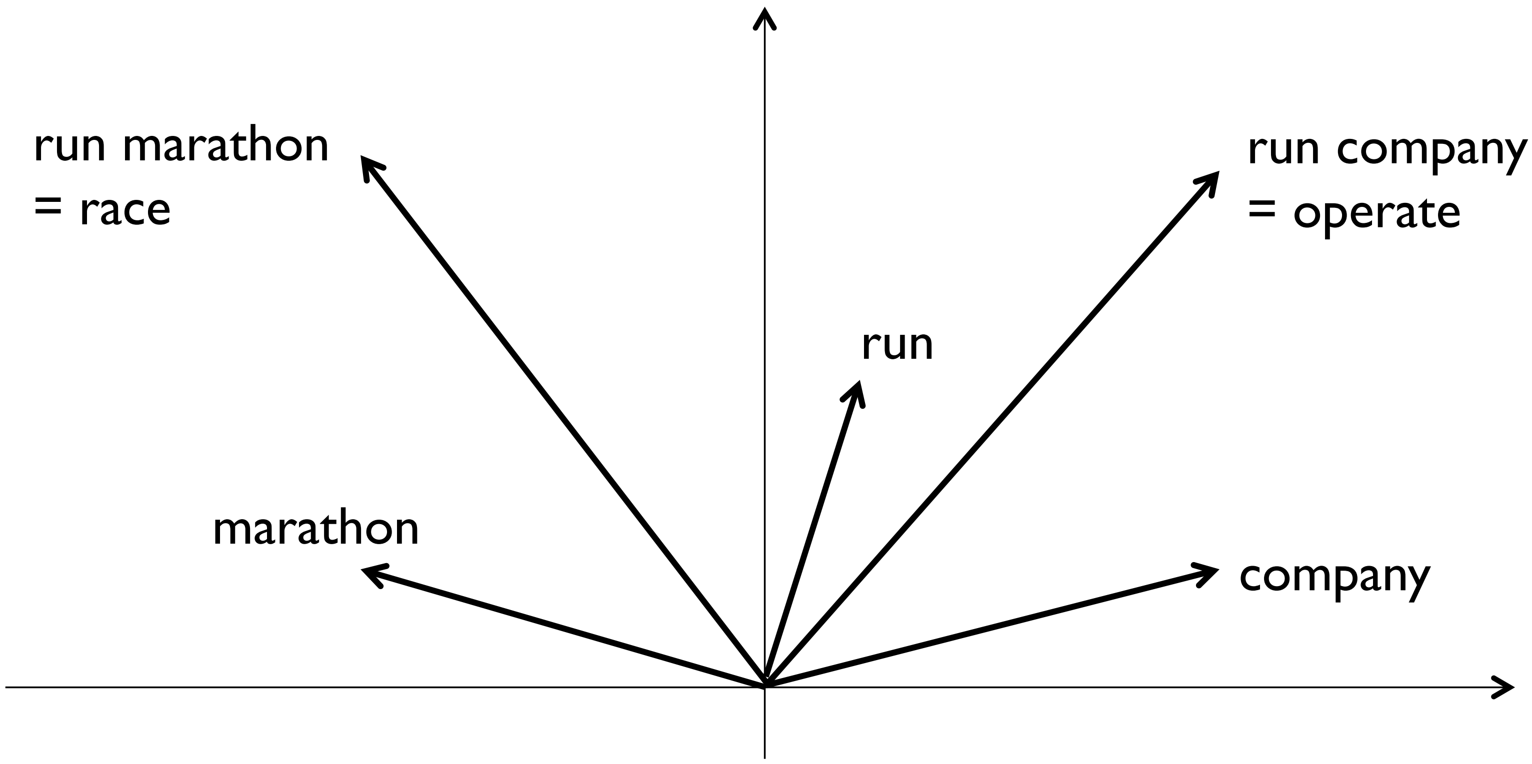
Two contributions in our work

New model of compositionality in word vector space

Unsupervised word vector re-training algorithm considering compositionality

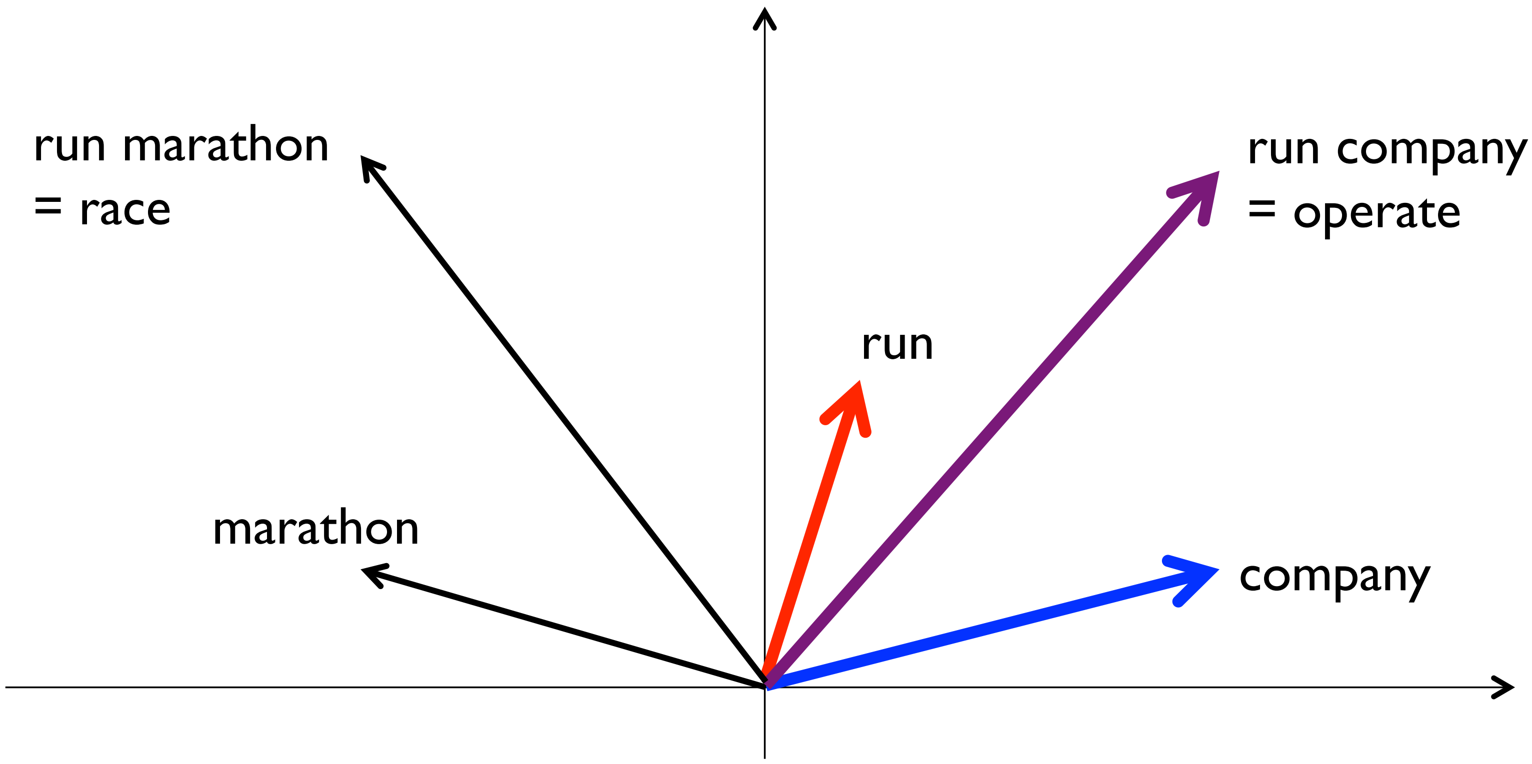
Modeling of compositionality in word vector space

From word to phrase representation with matrix-vector operation
[Mitchell and Lapata 08], [Baroni and Zamparell 10], [Socher+ 12], [Van de Cruys+ 13]



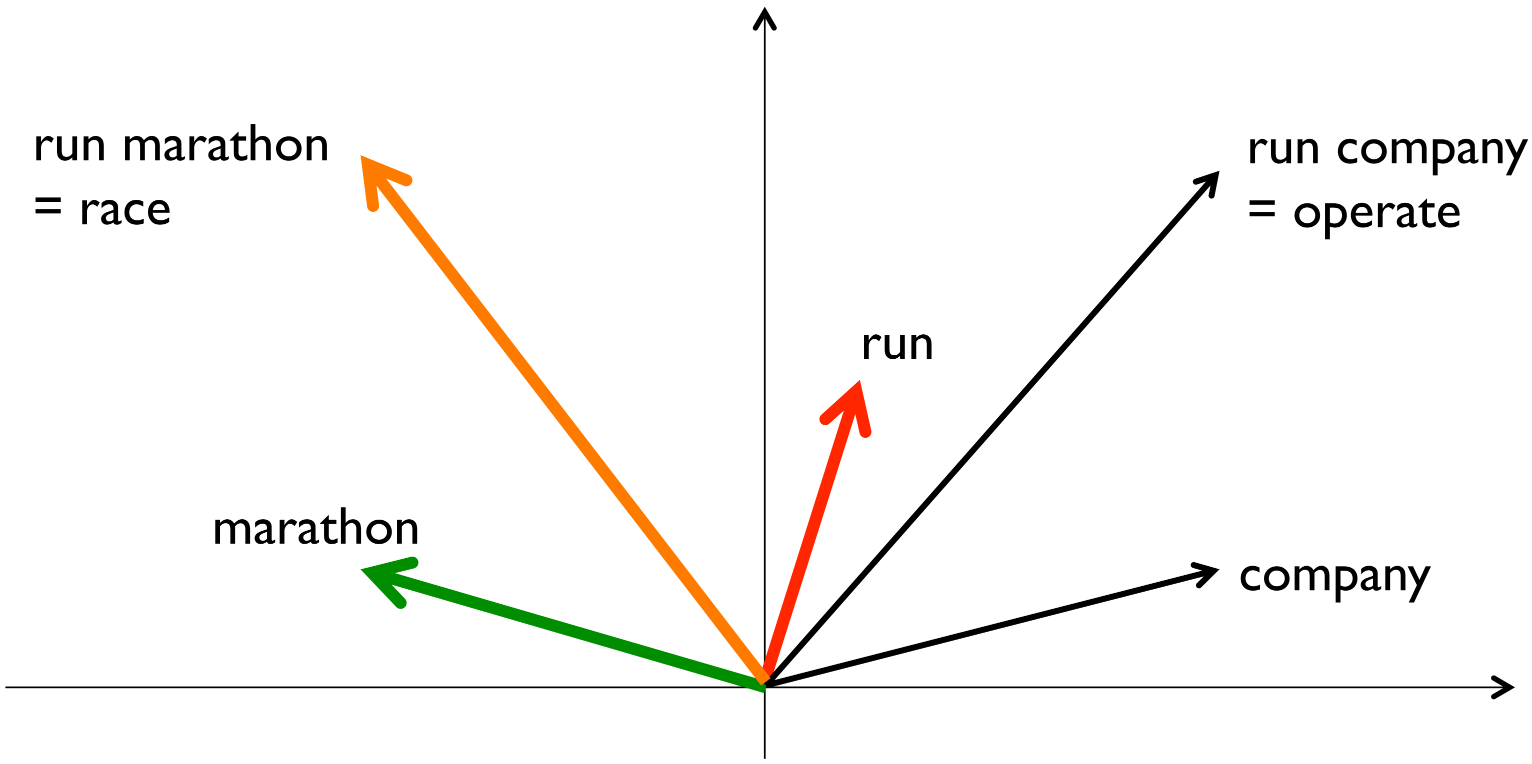
Modeling of compositionality in word vector space

From word to phrase representation with matrix-vector operation
[Mitchell and Lapata 08], [Baroni and Zamparell 10], [Socher+ 12], [Van de Cruys+ 13]



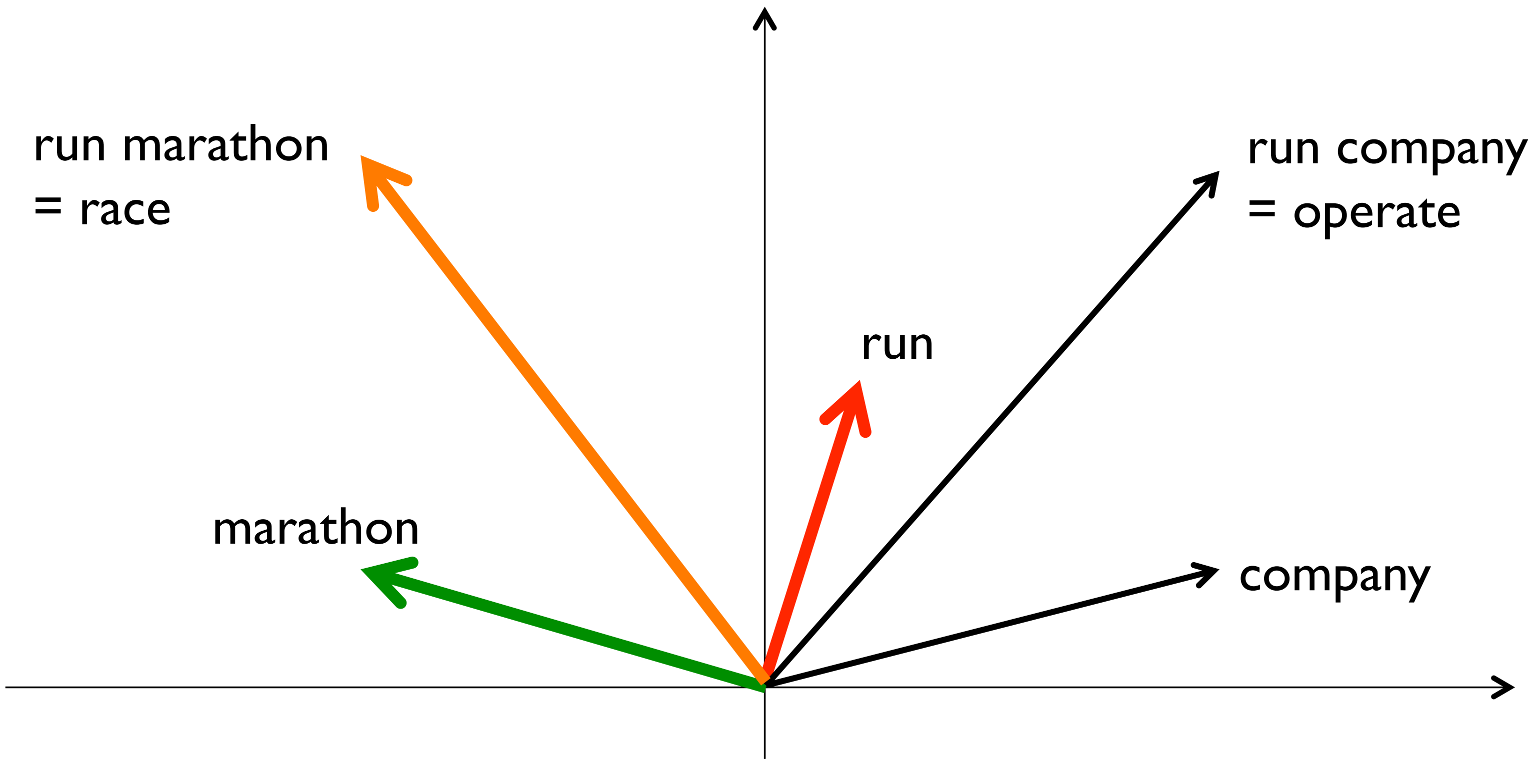
Modeling of compositionality in word vector space

From word to phrase representation with matrix-vector operation
[Mitchell and Lapata 08], [Baroni and Zamparell 10], [Socher+ 12], [Van de Cruys+ 13]



Modeling of compositionality in word vector space

From word to phrase representation with matrix-vector operation
[Mitchell and Lapata 08], [Baroni and Zamparell 10], [Socher+ 12], [Van de Cruys+ 13]



New model inspired by Co-Compositionality

Main Idea : Co-Compositionality [Pustejovsky 1995]

Co-compositionality

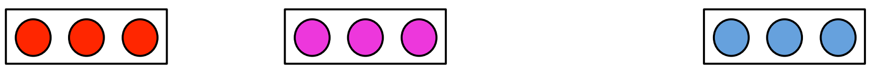
Verb and object are allowed to modify each other's meanings and generate the overall semantics

Main Idea : Co-Compositionality [Pustejovsky 1995]

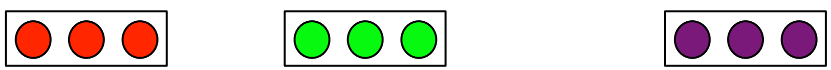
Co-compositionality

Verb and object are allowed to modify each other's meanings and generate the overall semantics

f(run , company) = operate



f(run , marathon) = race

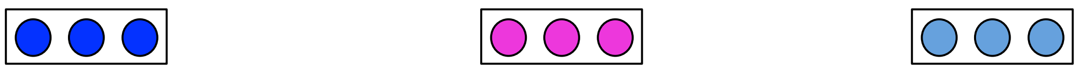


Main Idea : Co-Compositionality [Pustejovsky 1995]

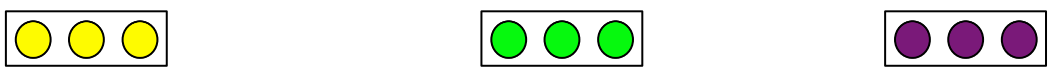
Co-compositionality

Verb and object are allowed to modify each other's meanings and generate the overall semantics

$$f(\ \mathrm{run}_{\mathrm{company}}\ ,\ \mathrm{company}\ ) = \mathrm{operate}$$



$$f(\ \mathrm{run}_{\mathrm{marathon}}\ ,\ \mathrm{marathon}\ ) = \mathrm{race}$$

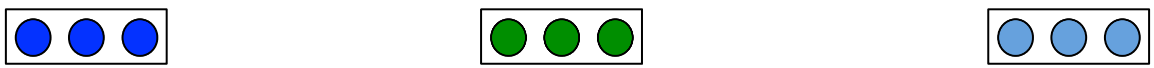


Main Idea : Co-Compositionality [Pustejovsky 1995]

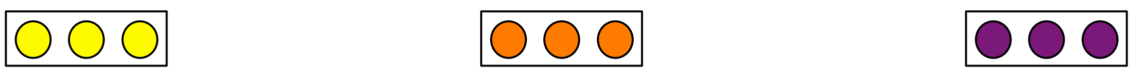
Co-compositionality

Verb and object are allowed to modify each other's meanings and generate the overall semantics

$$f(\, run_{company} \,,\, company_{run} \,) = operate$$



$$f(\, run_{marathon} \,,\, marathon_{run} \,) = race$$

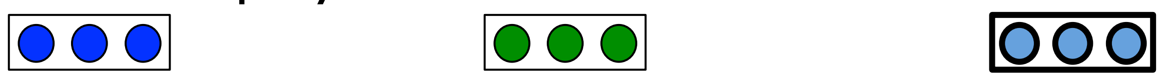


Main Idea : Co-Compositionality [Pustejovsky 1995]

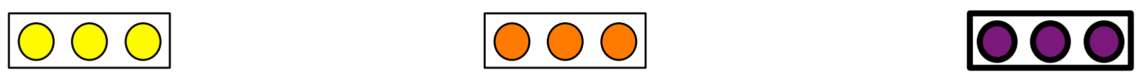
Co-compositionality

Verb and object are allowed to modify each other's meanings and generate the overall semantics

$$f(\,\mathrm{run}_{\mathrm{company}}\,,\,\mathrm{company}_{\mathrm{run}}\,) = \textbf{operate}$$



$$f(\,\mathrm{run}_{\mathrm{marathon}}\,,\,\mathrm{marathon}_{\mathrm{run}}\,) = \textbf{race}$$

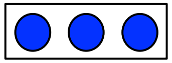
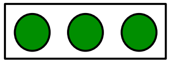
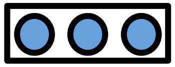


Main Idea : Co-Compositionality [Pustejovsky 1995]

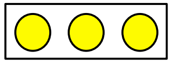
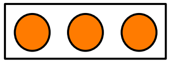
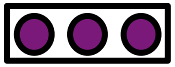
Co-compositionality

Verb and object are allowed to modify each other's meanings and generate the overall semantics

f($\text{run}_{\text{company}}$, $\text{company}_{\text{run}}$) = **operate**

f($\text{run}_{\text{marathon}}$, $\text{marathon}_{\text{run}}$) = **race**

Question

How do we implement co-compositionality in vector space ?

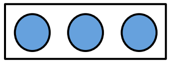
Prototype Projection for Co-Compositionality

Prototype Projection

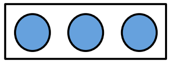
Matrix-vector operation as an implementation for Co-Compositionality

Co-Compositionality with Prototype Projections

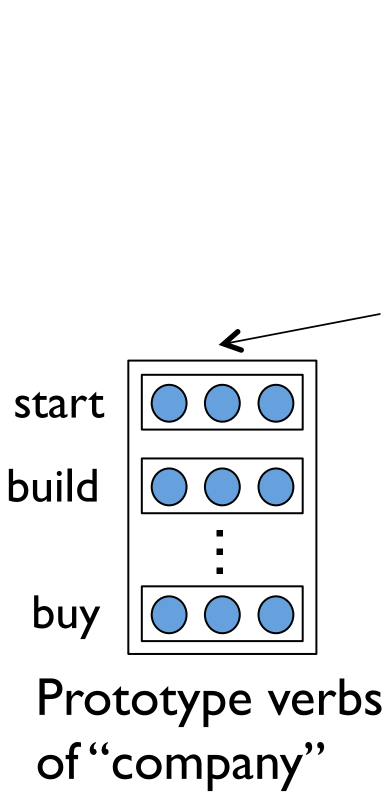
run



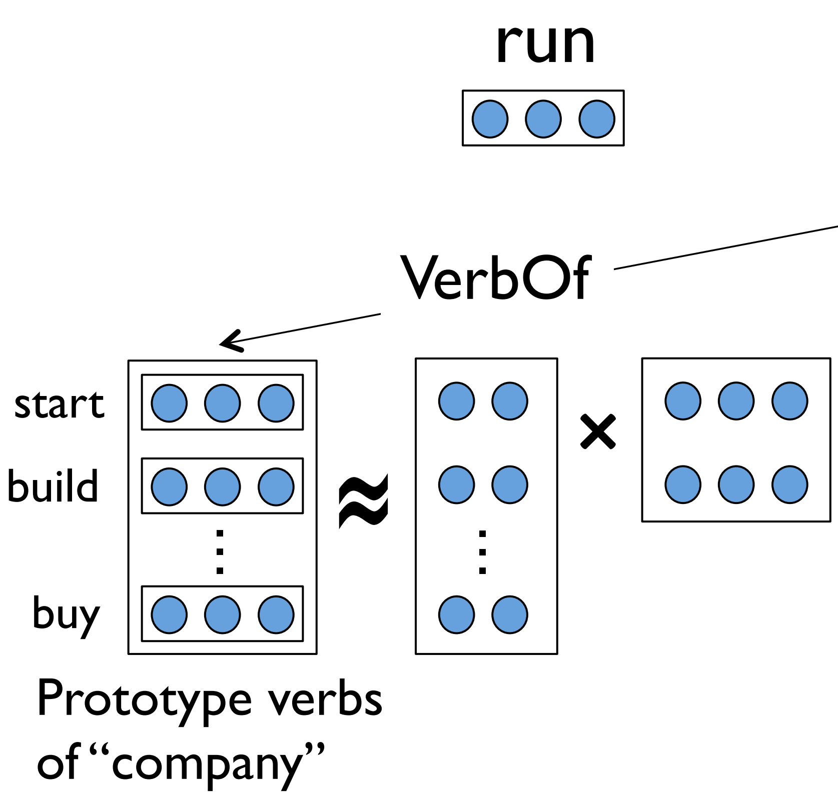
company



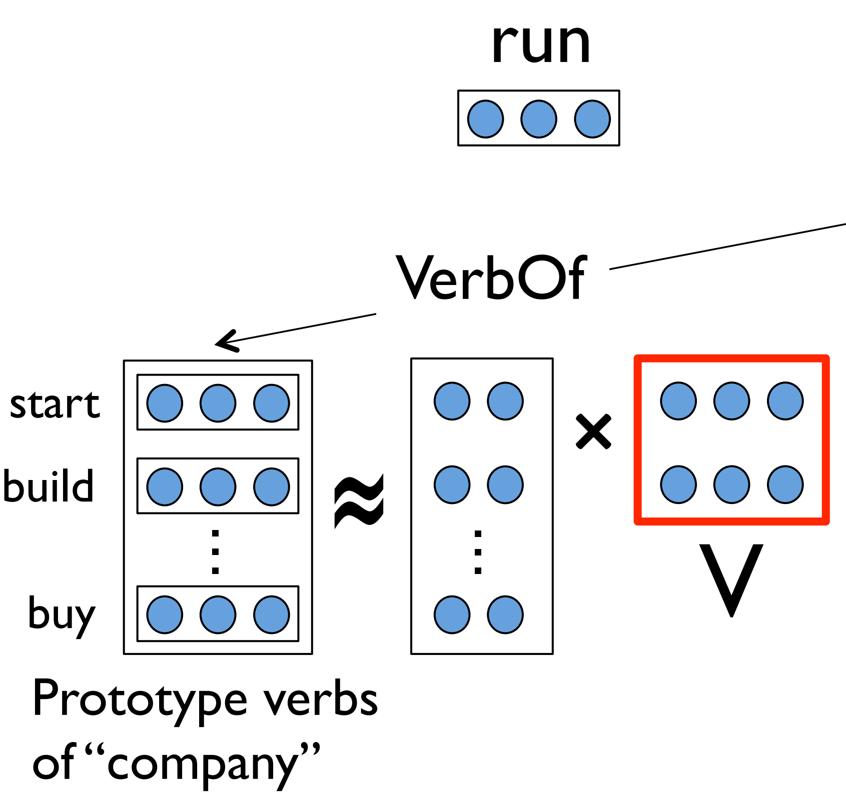
Co-Compositionality with Prototype Projections



Co-Compositionality with Prototype Projections

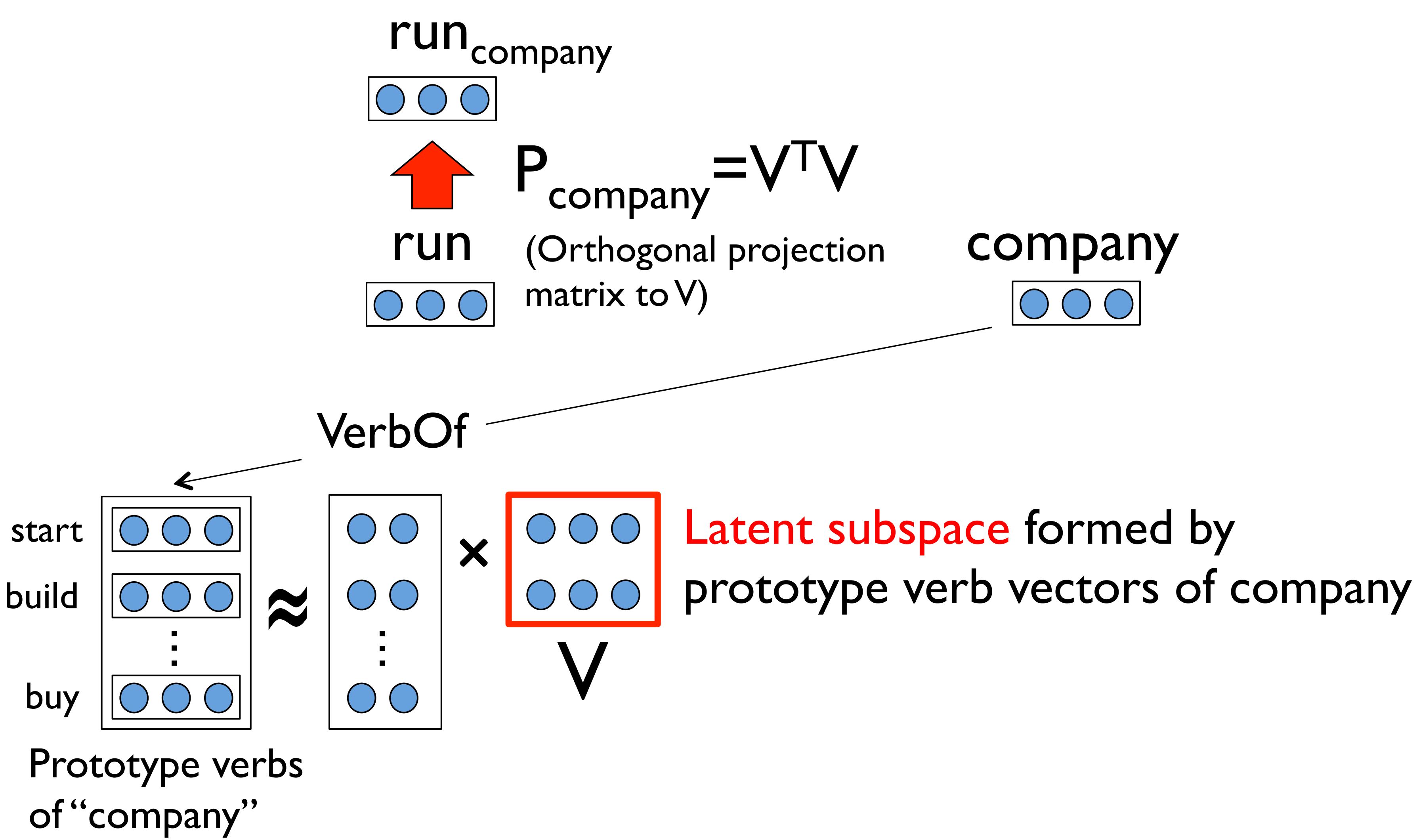


Co-Compositionality with Prototype Projections

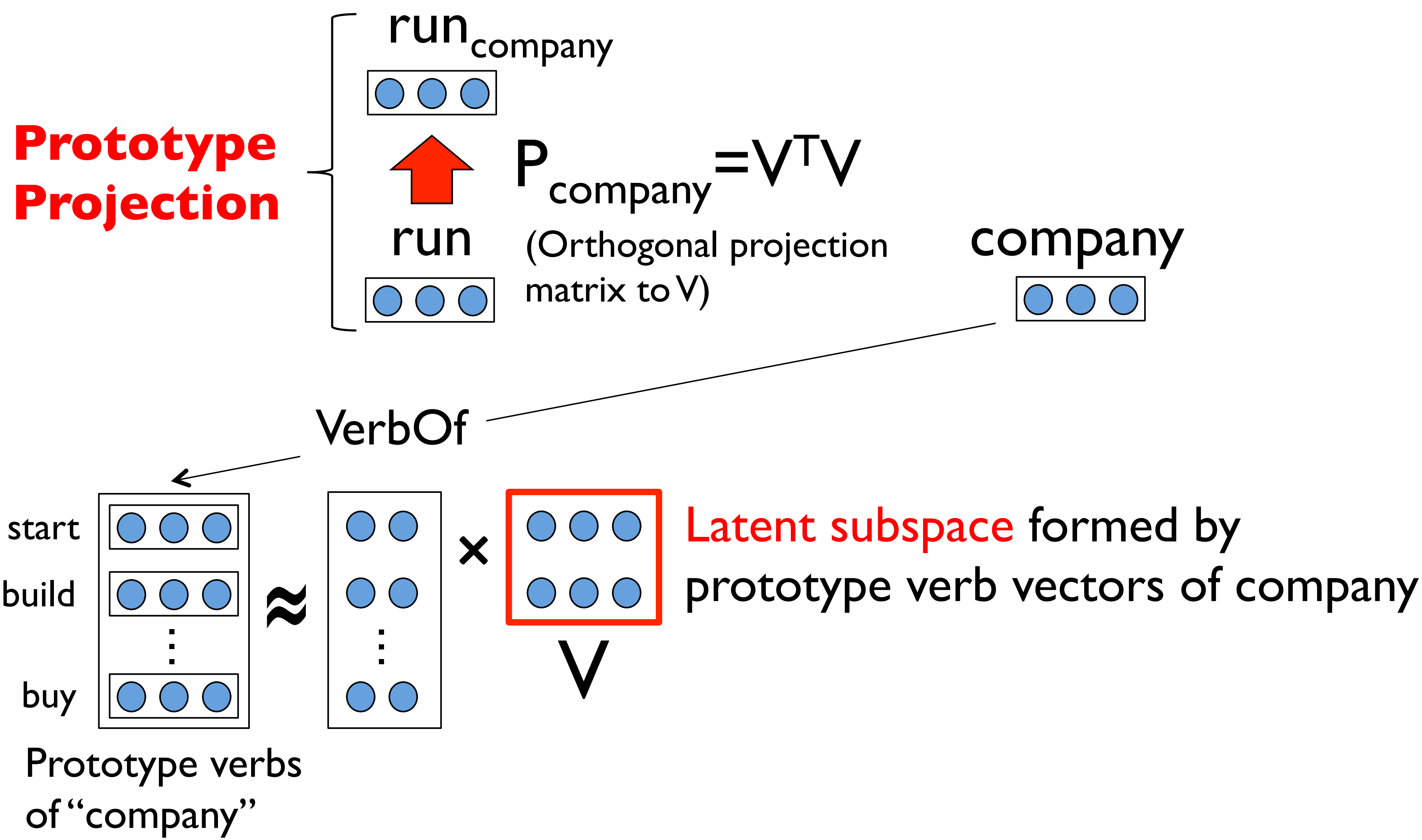


Latent subspace formed by prototype verb vectors of company

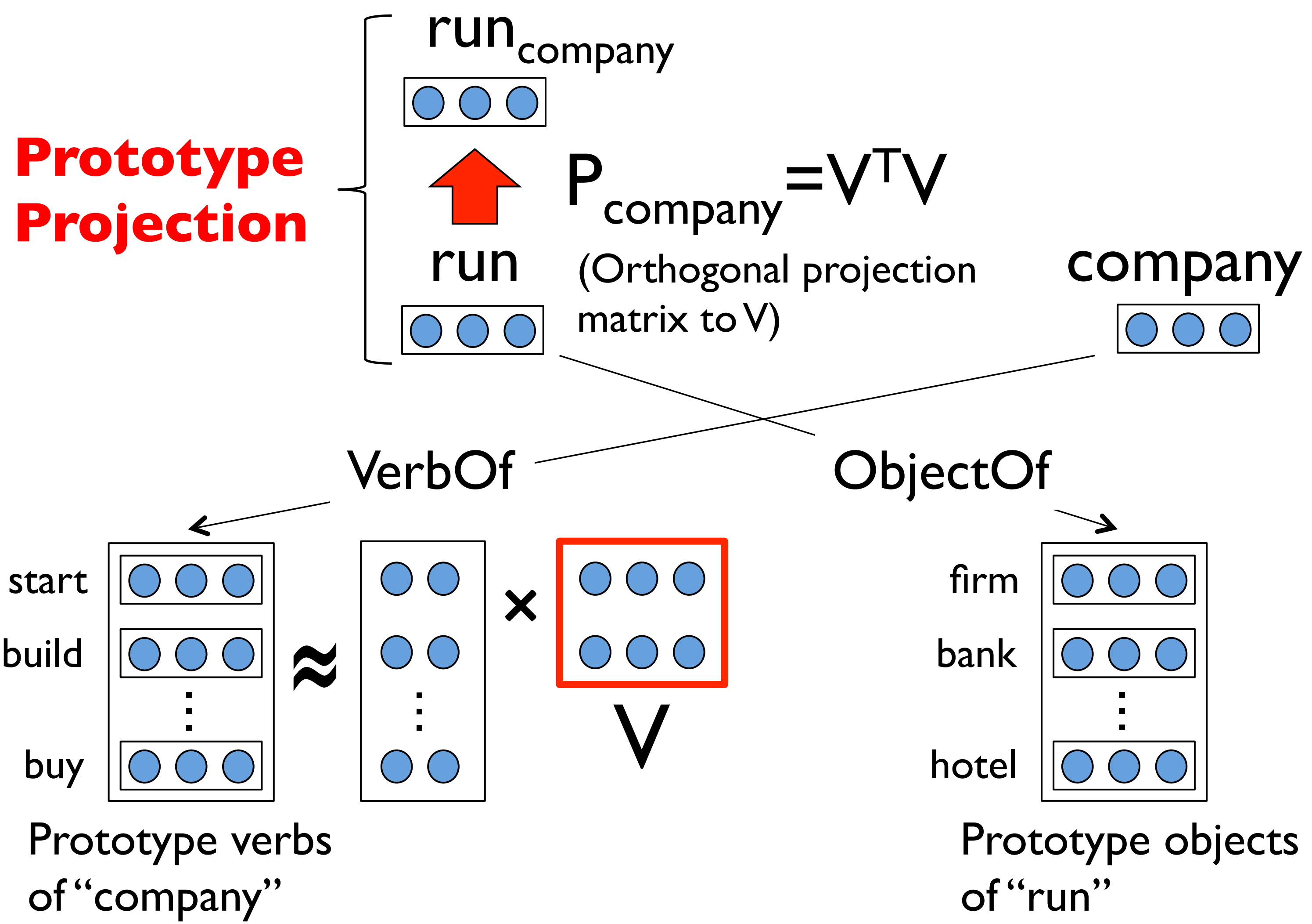
Co-Compositionality with Prototype Projections



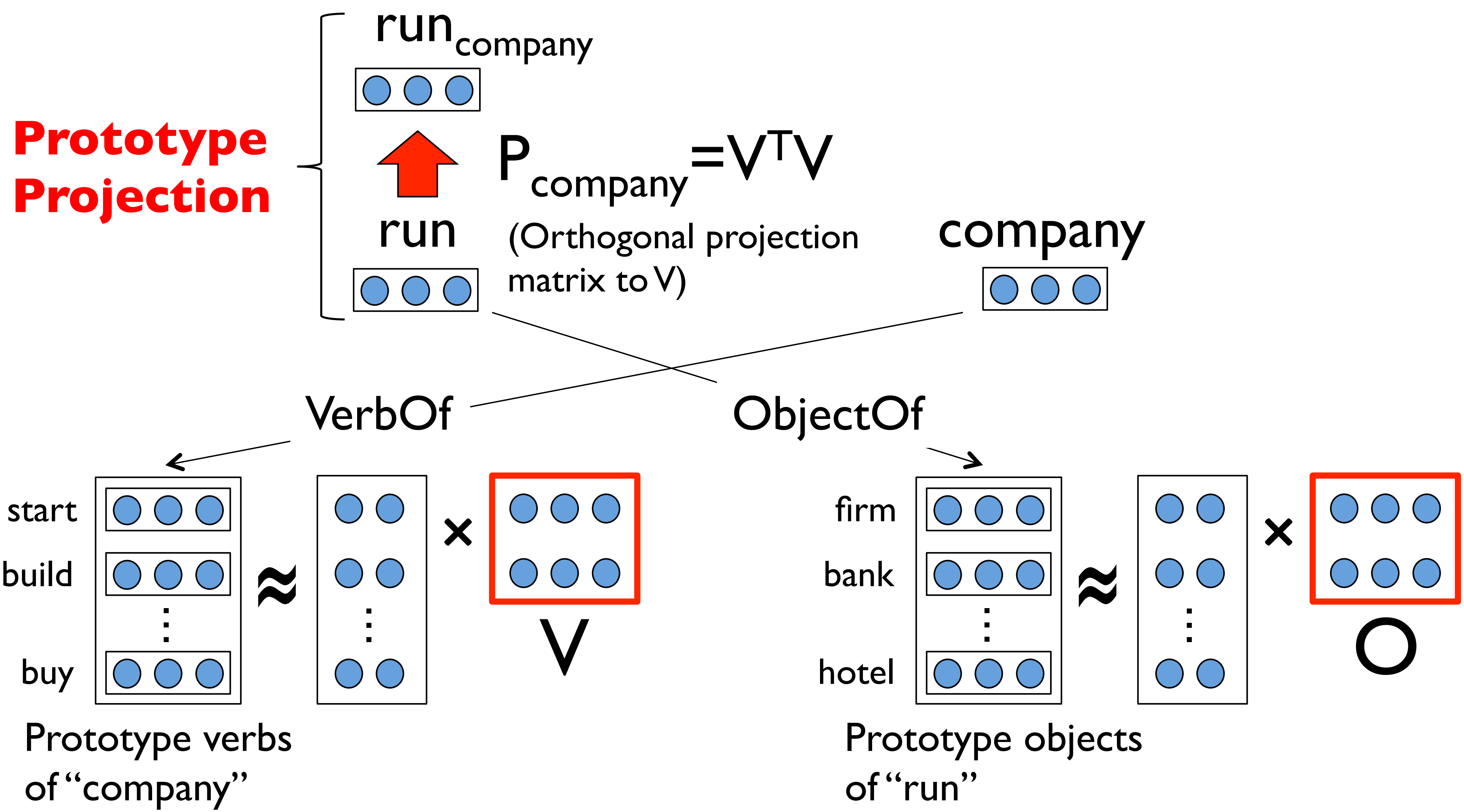
Co-Compositionality with Prototype Projections



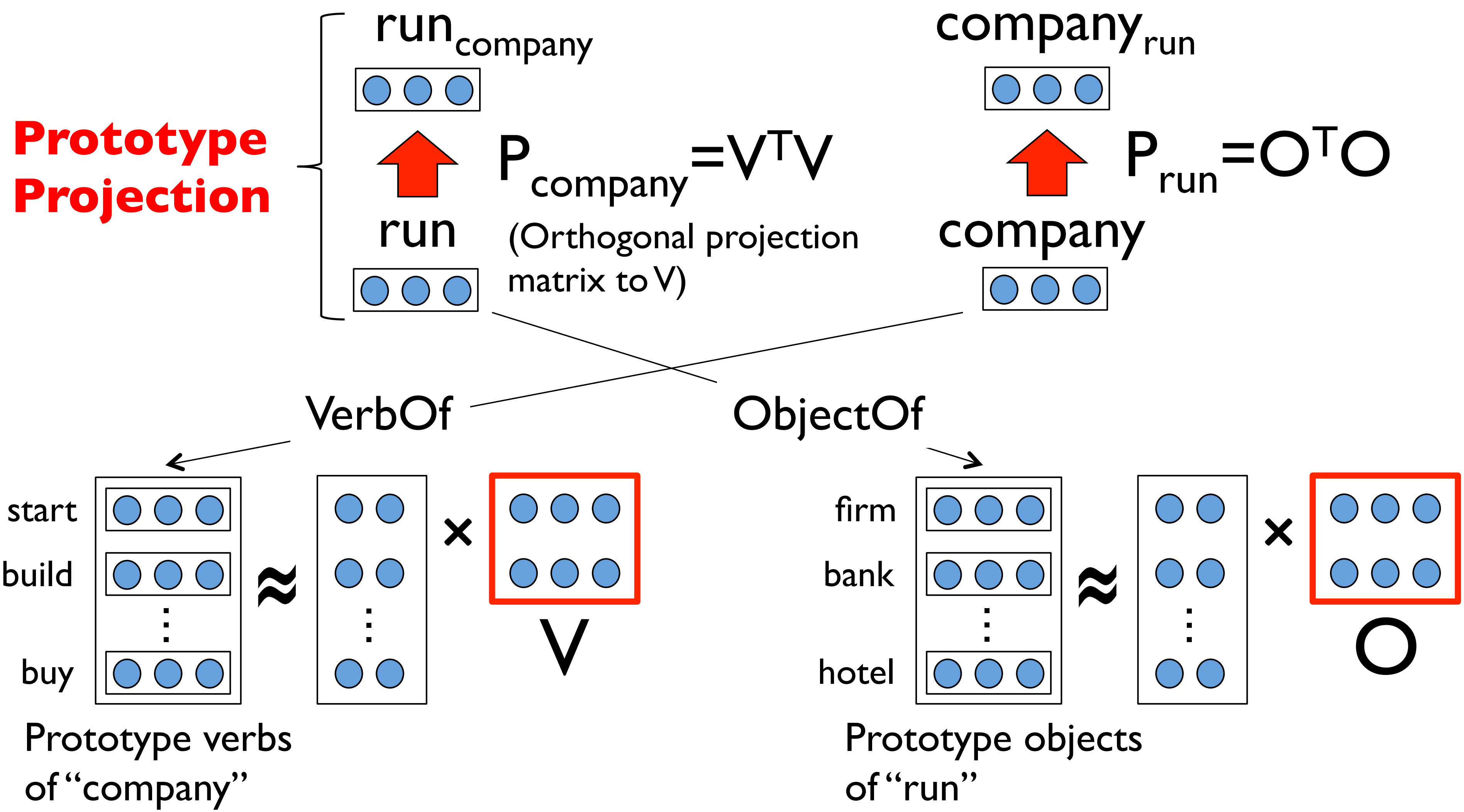
Co-Compositionality with Prototype Projections



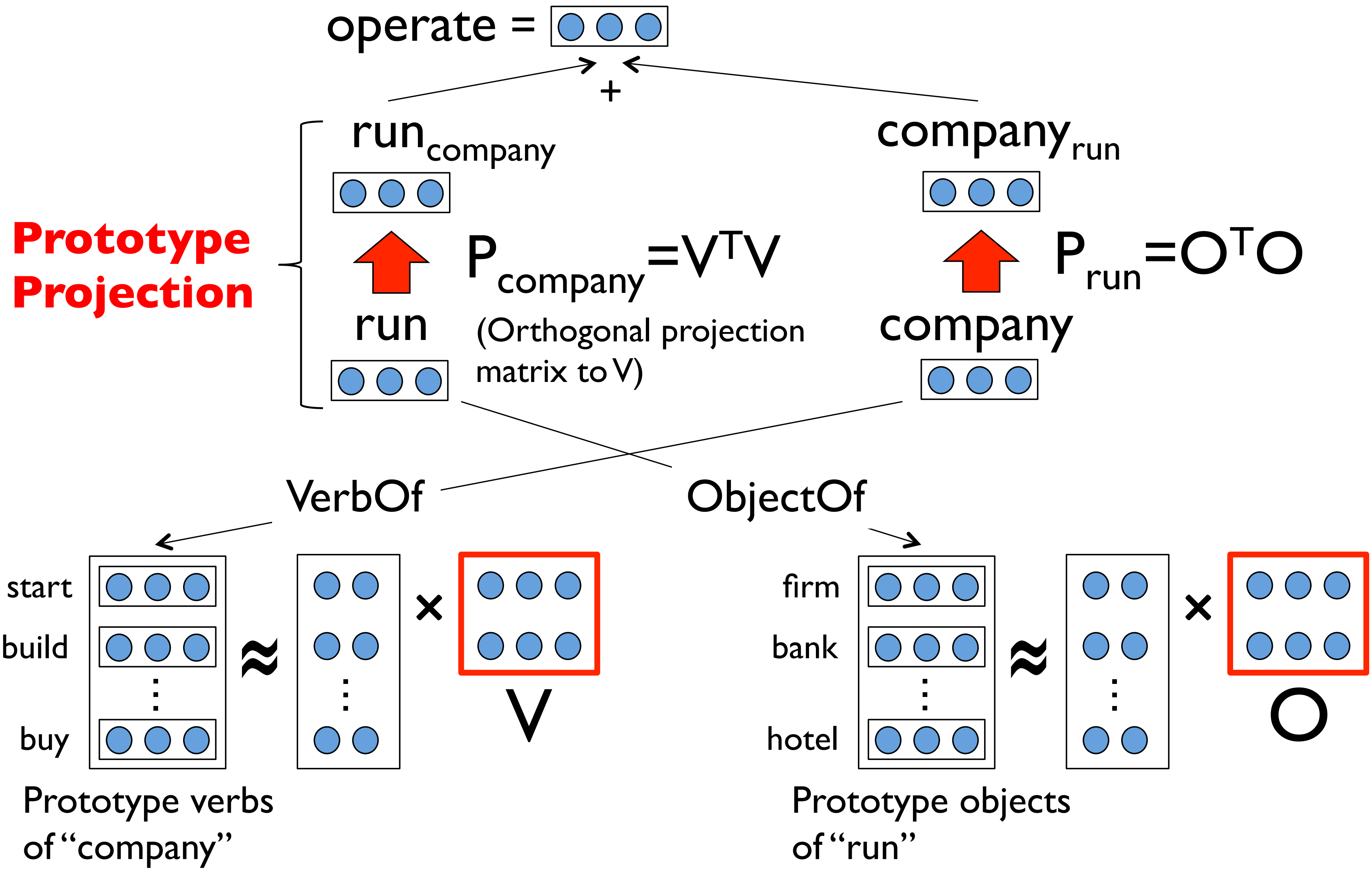
Co-Compositionality with Prototype Projections



Co-Compositionality with Prototype Projections

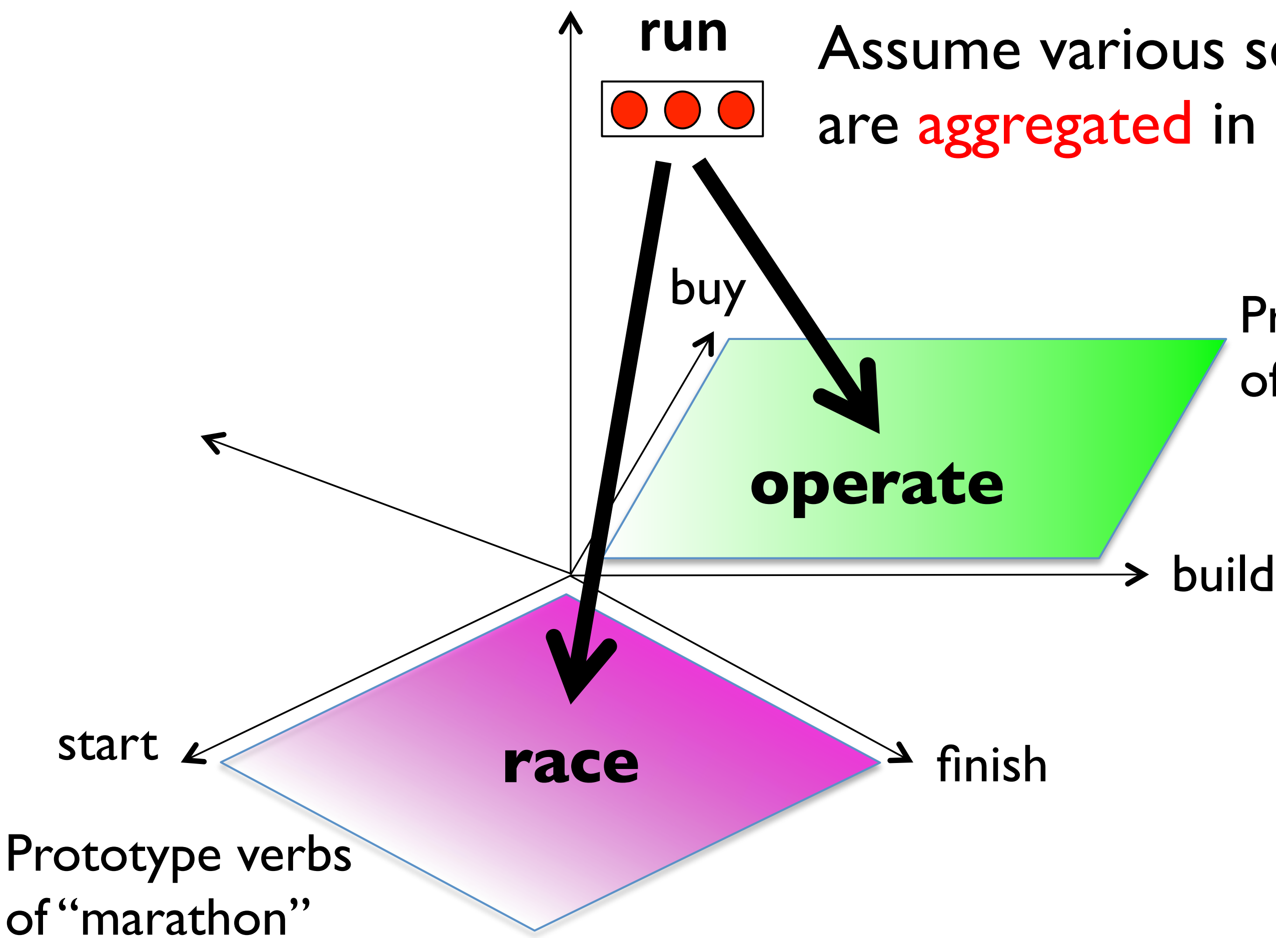


Co-Compositionality with Prototype Projections



Intuitive image of prototype projection

Assume various senses of "run" are aggregated in one vector



Tease out the proper semantics from aggregate representation by projection to latent space

Evaluation : Verb disambiguation in subject-verb-object triples

Evaluation dataset [Grefenstette and Sadrzadeh 11]

Subj-Verb-Obj	Landmark verb	Similarity of human judgment
People-run-company	operate	7
People-run-company	move	2

200 subject-verb-object triples judged by 25 participants

Evaluation : Verb disambiguation in subject-verb-object triples

Evaluation dataset [Grefenstette and Sadrzadeh 11]

Subj-Verb-Obj	Landmark verb	Similarity of human judgment
People-run-company	operate	7
People-run-company	move	2

200 subject-verb-object triples judged by 25 participants

Final co-compositional vector for subject-verb-object

$$subj + cocompositioned(verb, obj)$$

Evaluation : Verb disambiguation in subject-verb-object triples

Evaluation dataset [Grefenstette and Sadrzadeh 11]

Subj-Verb-Obj	Landmark verb	Similarity of human judgment
People-run-company	operate	7
People-run-company	move	2

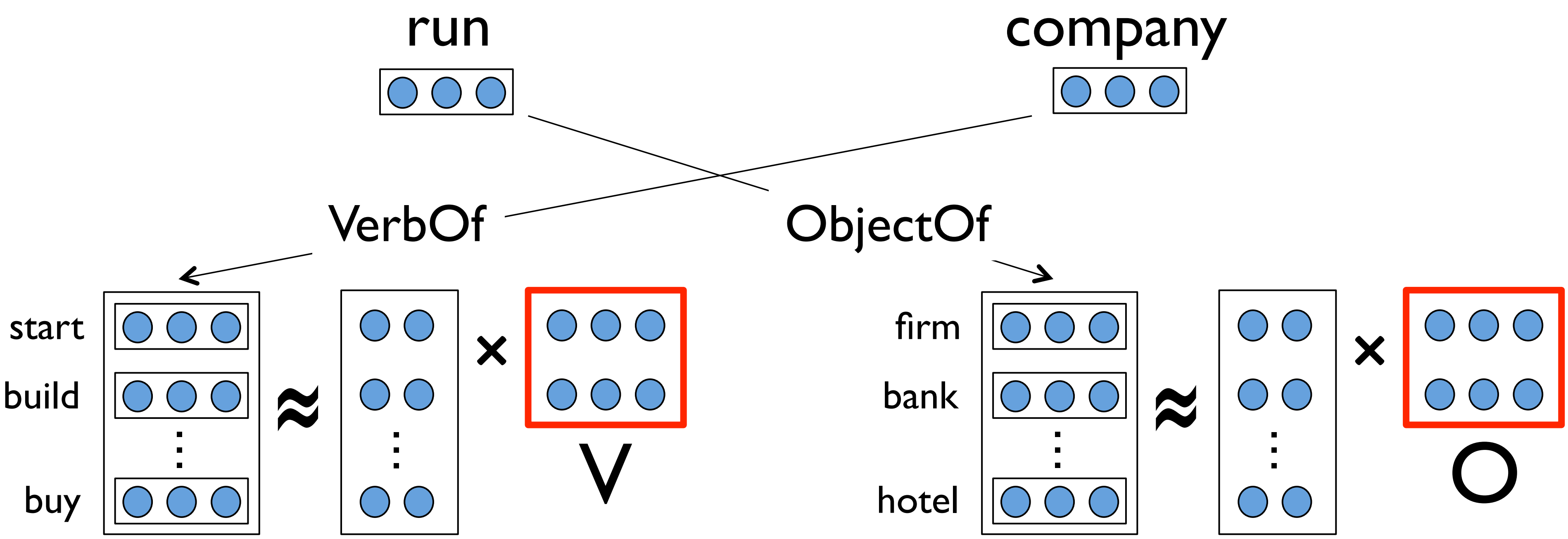
200 subject-verb-object triples judged by 25 participants

Final co-compositional vector for subject-verb-object

$$subj + cocompositioned(verb, obj)$$

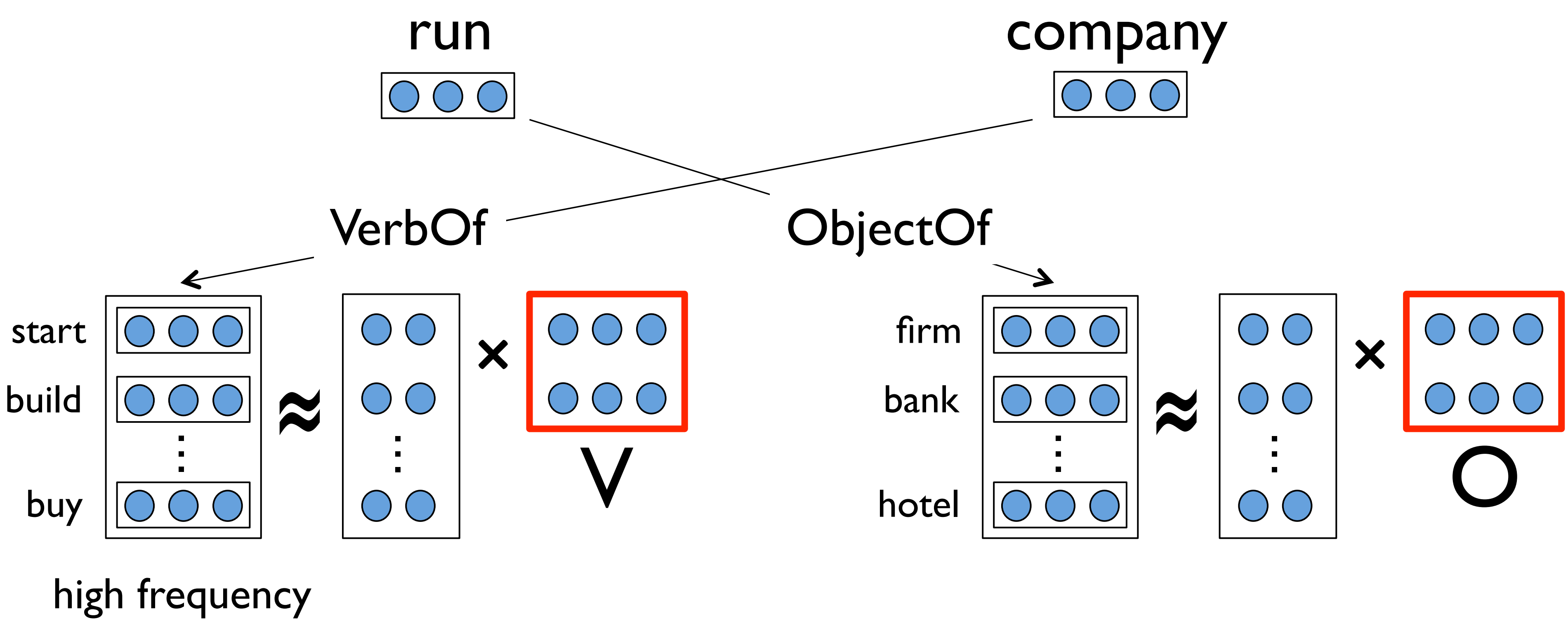
Models are evaluated by Spearman's rank correlation between vectors' computed similarity and human judgment

Implementation details



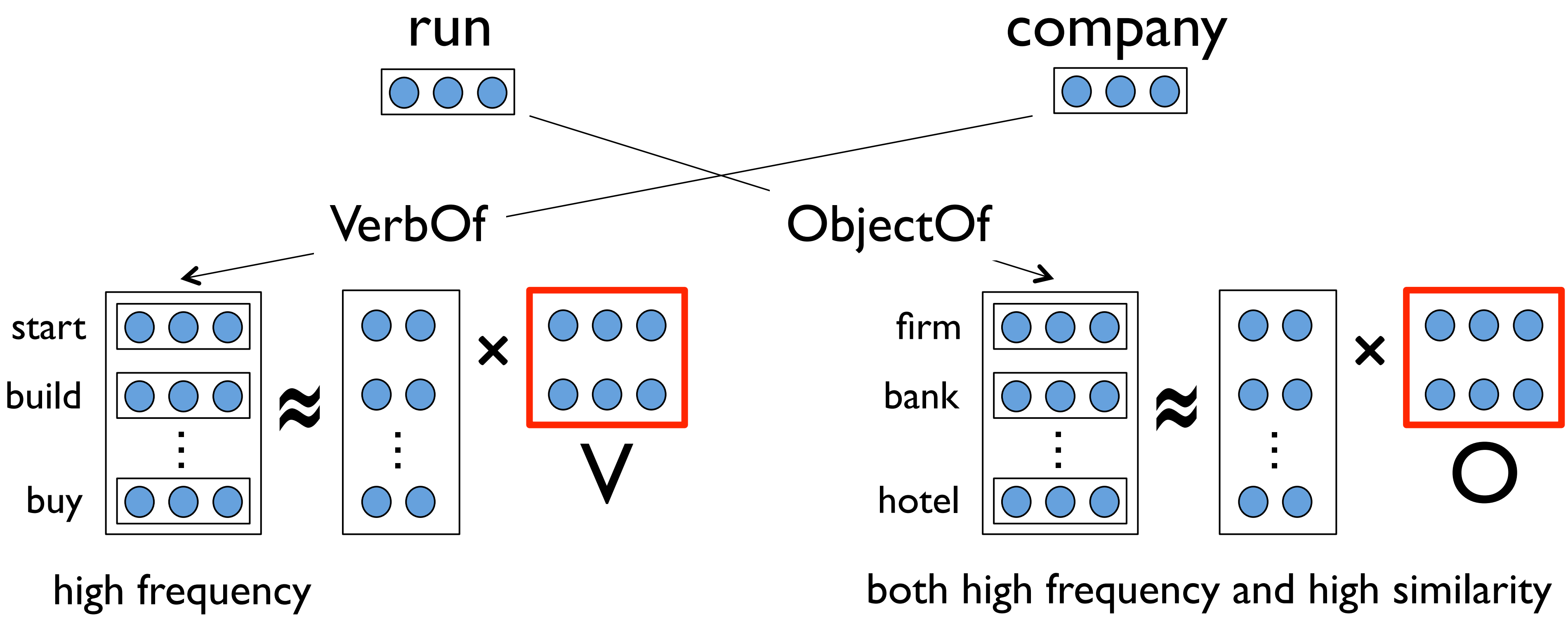
Extracted 20 prototype words from ukWaC corpus

Implementation details



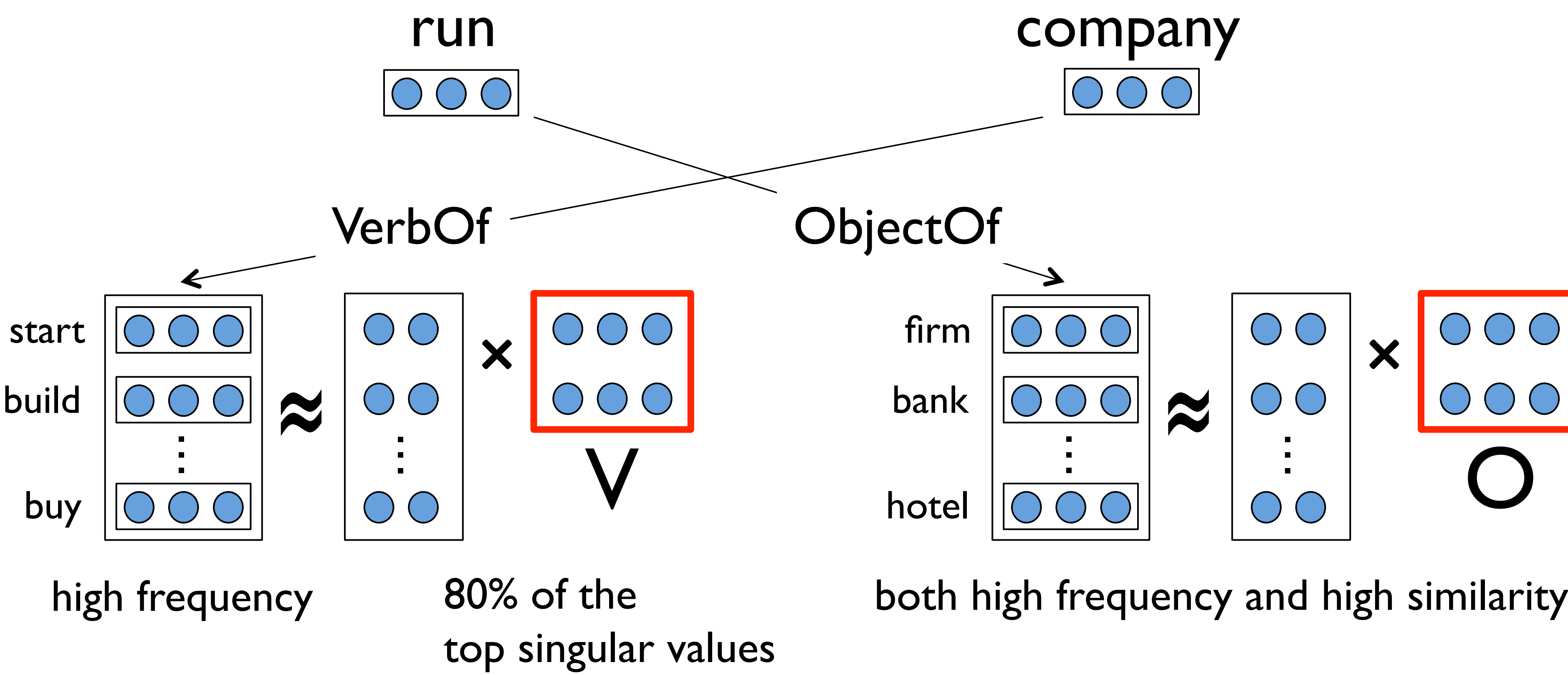
Extracted 20 prototype words from ukWaC corpus

Implementation details



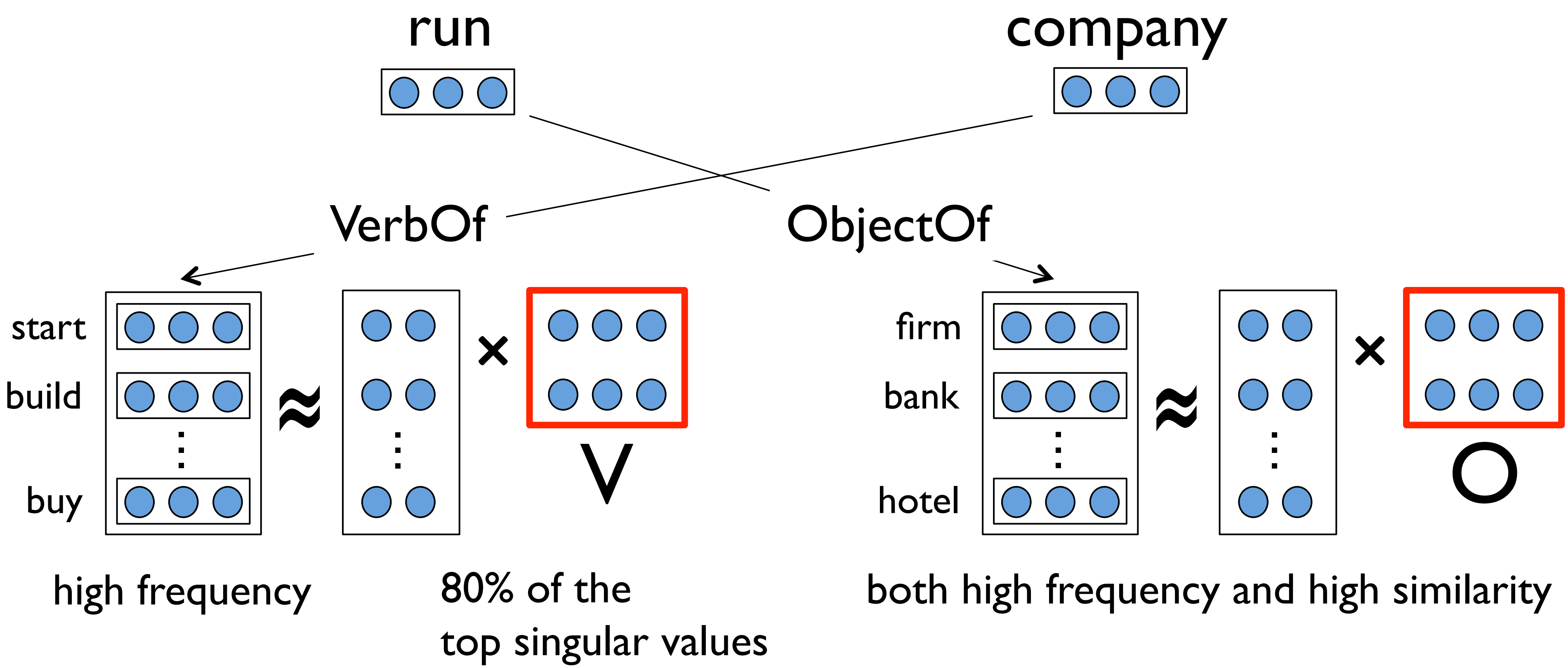
Extracted 20 prototype words from ukWaC corpus

Implementation details



Extracted 20 prototype words from ukWaC corpus

Implementation details



Extracted 20 prototype words from ukWaC corpus

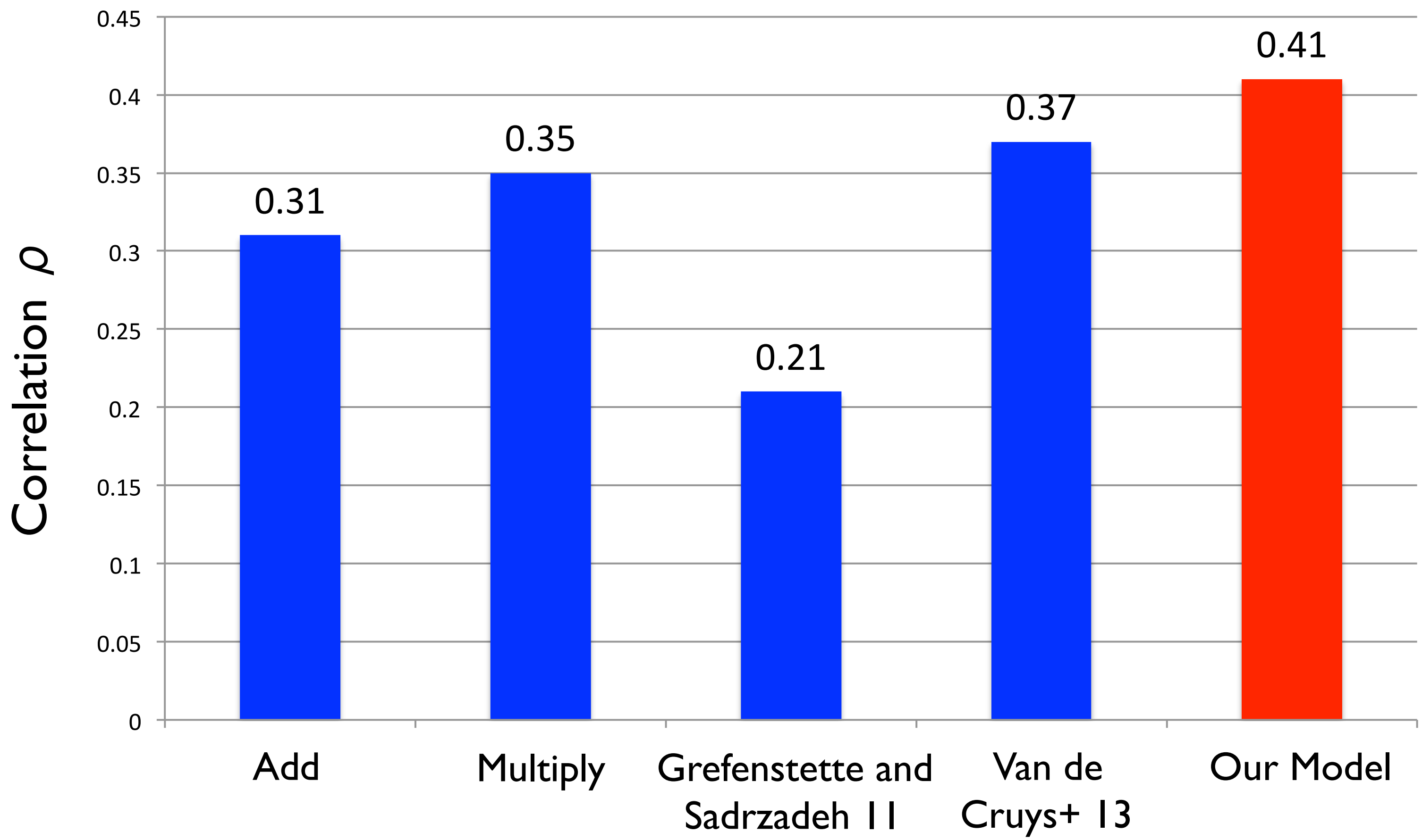
Word representation [Blacoe and Lapata 12]
①Distributional vector (2000 dim) ②Neural vector (50 dim)

Baselines : Models compared to ours

Add [Mitchell and Lapata 08]	$sbj + verb + obj$
Multiply [Mitchell and Lapata 08]	$sbj \times verb \times obj$
Grefenstette and Sadrzadeh 11	Mathematical model based on abstract categorical framework
Van de Cruys+13	Multi-way interaction model based on non-negative matrix factorization

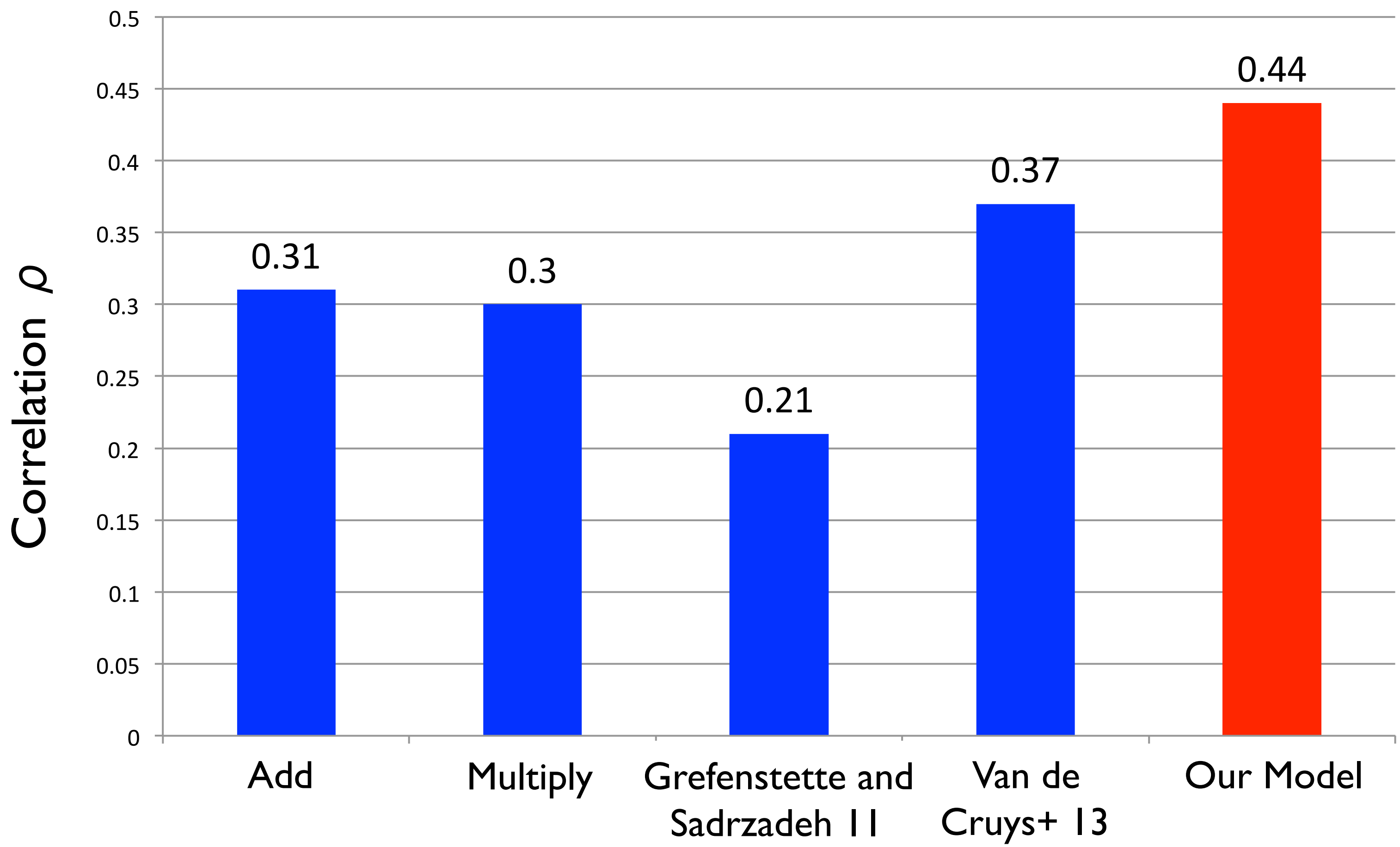
Correlation with human judgment (Distributional vector)

Achieves high performance (ρ = 0.41)



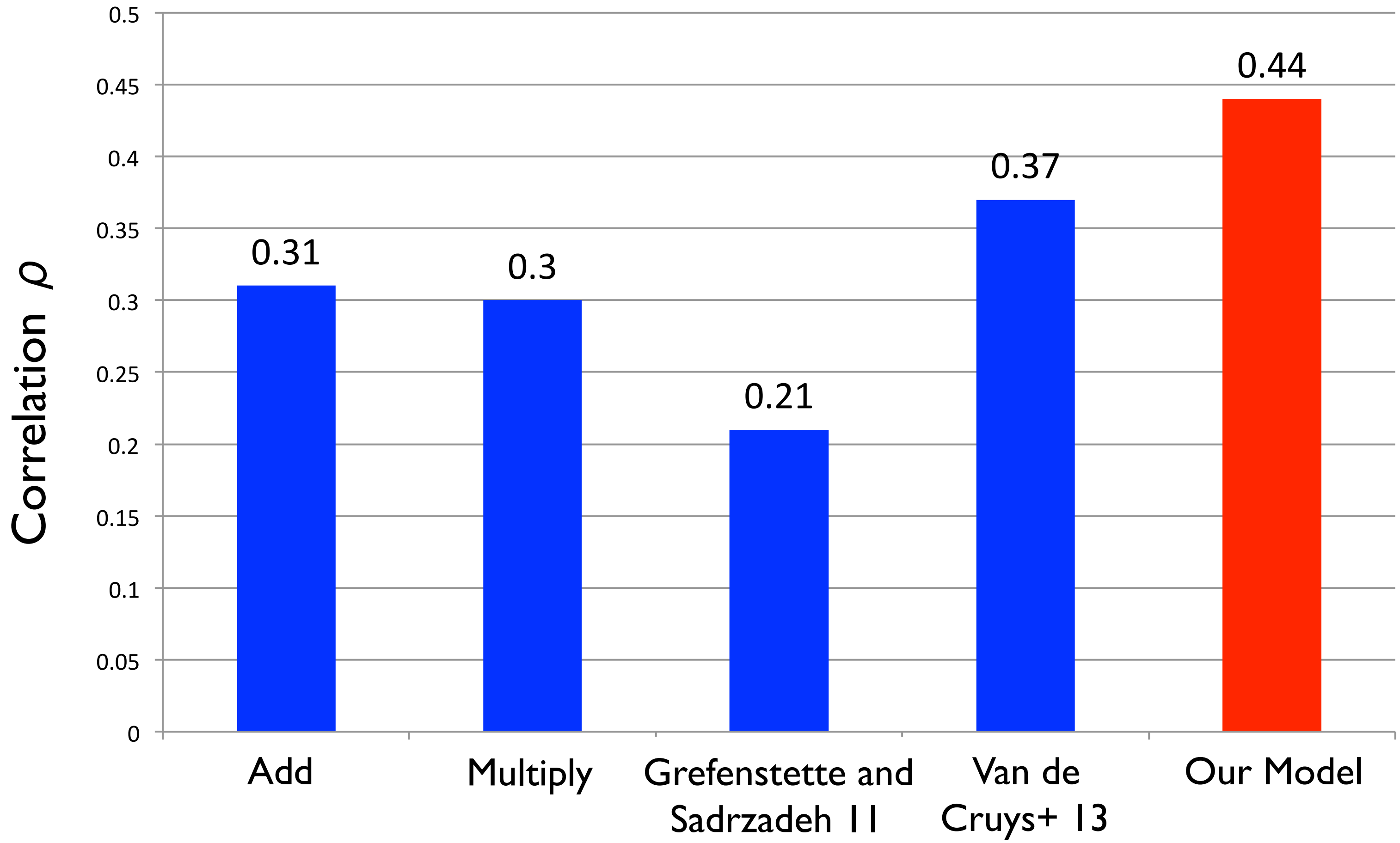
Correlation with human judgment (Neural vector)

State of the art performance (ρ = 0.44)



Correlation with human judgment (Neural vector)

State of the art performance (ρ = 0.44)



Co-Compositionality is useful for word sense disambiguation

Prototype projection is effective implementation for Co-Compositionality

Two contributions in our work

New model of compositionality in word vector space

Unsupervised word vector re-training algorithm considering compositionality

Two contributions in our work

New model of
compositionality
in word vector space

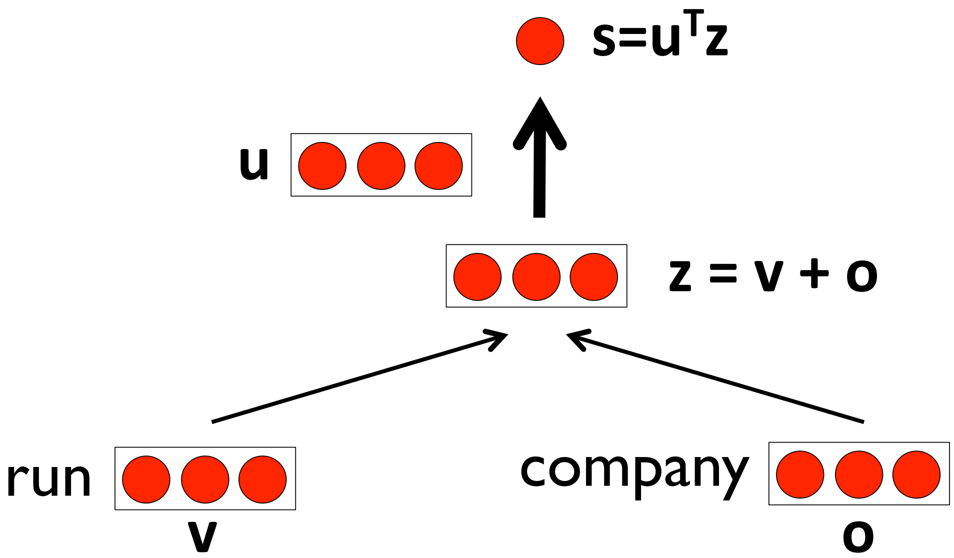
Unsupervised word vector
re-training algorithm
considering compositionality

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector

Compositional Neural Language Model

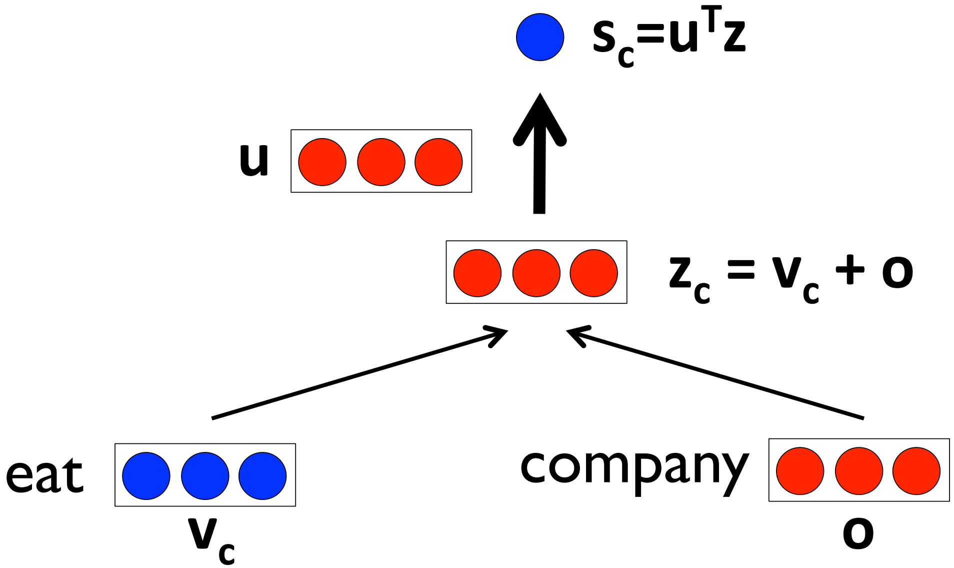
Re-training word representation with decomposition of phrase vector



①Compute the score s of correct phrase

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector

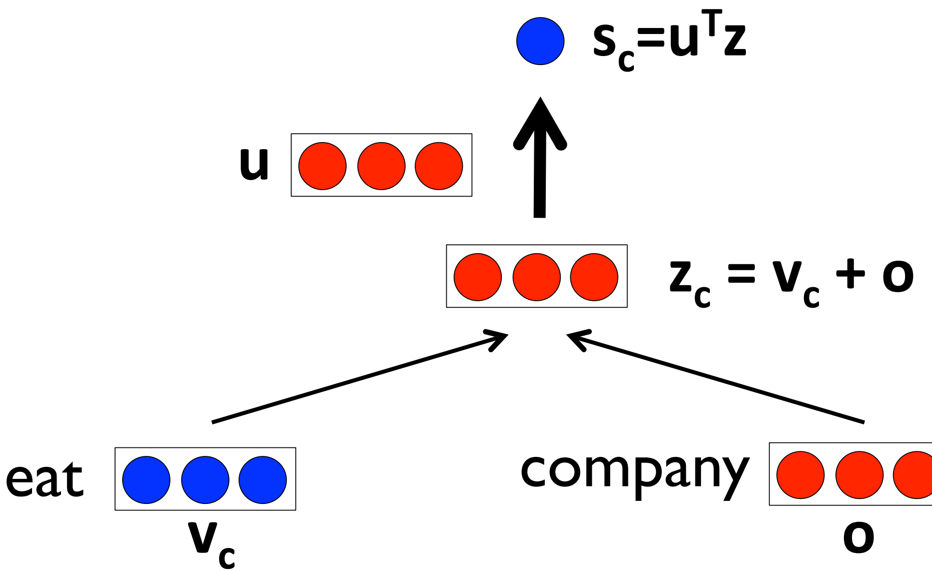


①Compute the score s of correct phrase

②Compute the score s_c of corrupted incorrect phrase

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector



①Compute the score s of correct phrase

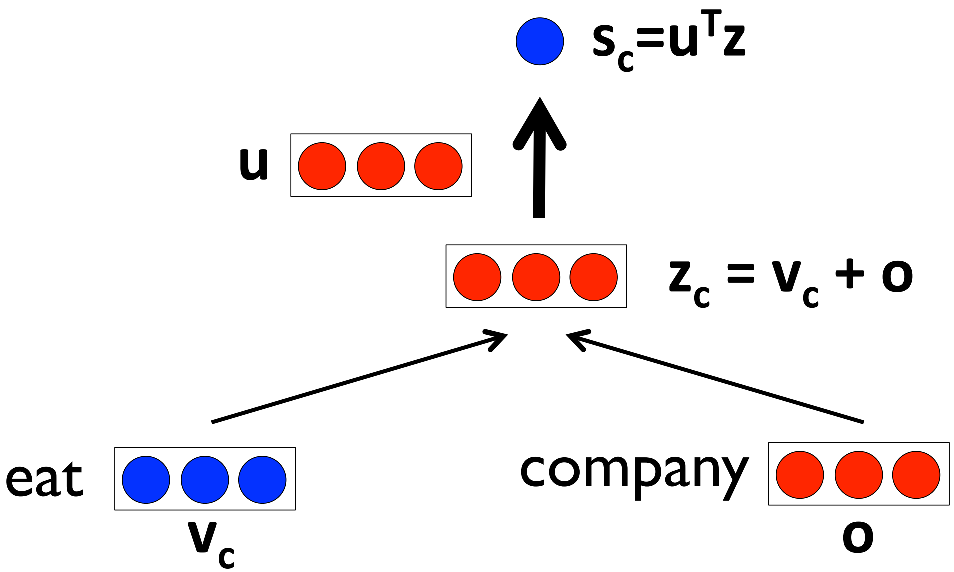
②Compute the score s_c of corrupted incorrect phrase

$$J = \max\left(0, 1 - s + s_c\right)$$

Correct score > Incorrect score

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector



$$J = \max\left(0, 1 - s + s_c\right)$$

Correct score > Incorrect score

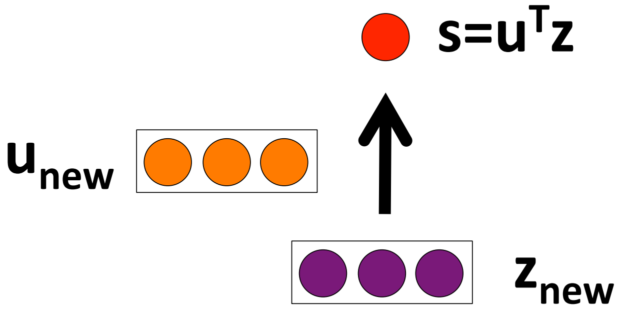
①Compute the score s of correct phrase

②Compute the score s_c of corrupted incorrect phrase

③Minimize cost function by SGD, $u \rightarrow u_{new}$, $z \rightarrow z_{new}$

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector



①Compute the score s of correct phrase

②Compute the score s_c of corrupted incorrect phrase

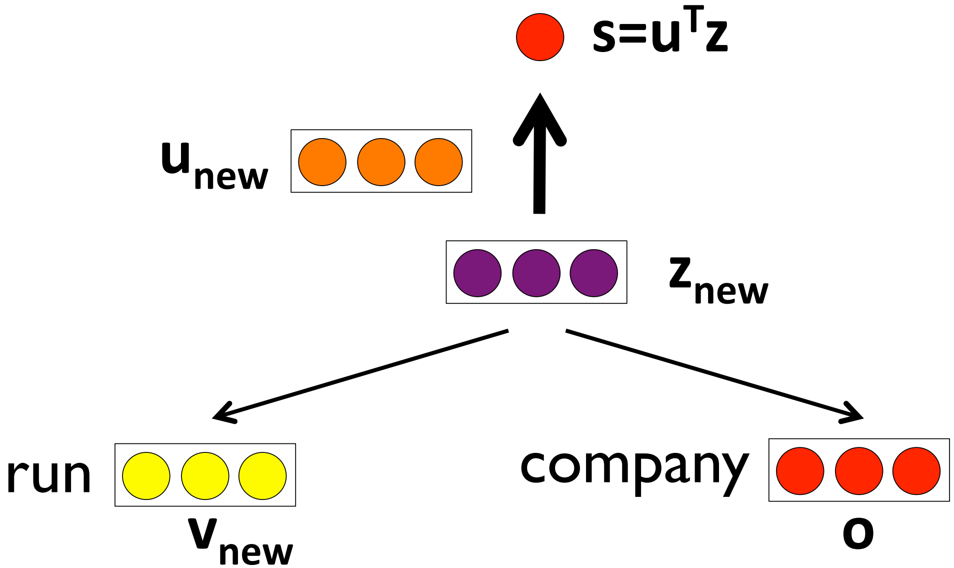
③Minimize cost function by SGD, $u \to u_{new}$, $z \to z_{new}$

$$J = \max\left(0, 1 - s + s_c\right)$$

Correct score $>$ Incorrect score

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector



$$J = \max\left(0, 1 - s + s_c\right)$$

Correct score > Incorrect score

①Compute the score s of correct phrase

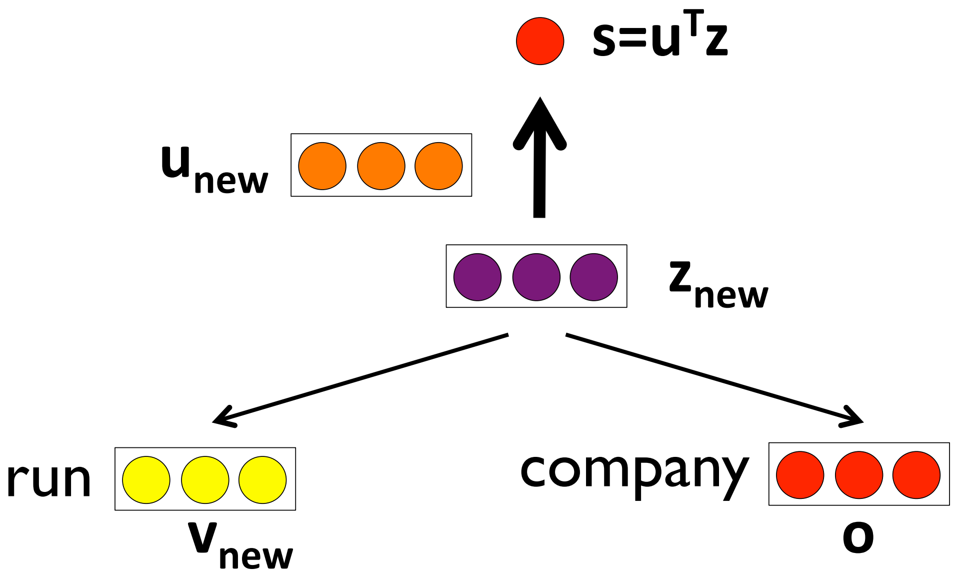
②Compute the score s_c of corrupted incorrect phrase

③Minimize cost function by SGD, $u \rightarrow u_{new}$, $z \rightarrow z_{new}$

④New verb vector is $v_{new} = z_{new} - o$

Compositional Neural Language Model

Re-training word representation with decomposition of phrase vector



$$J = \max\left(0, 1 - s + s_c\right)$$

Correct score > Incorrect score

①Compute the score s of correct phrase

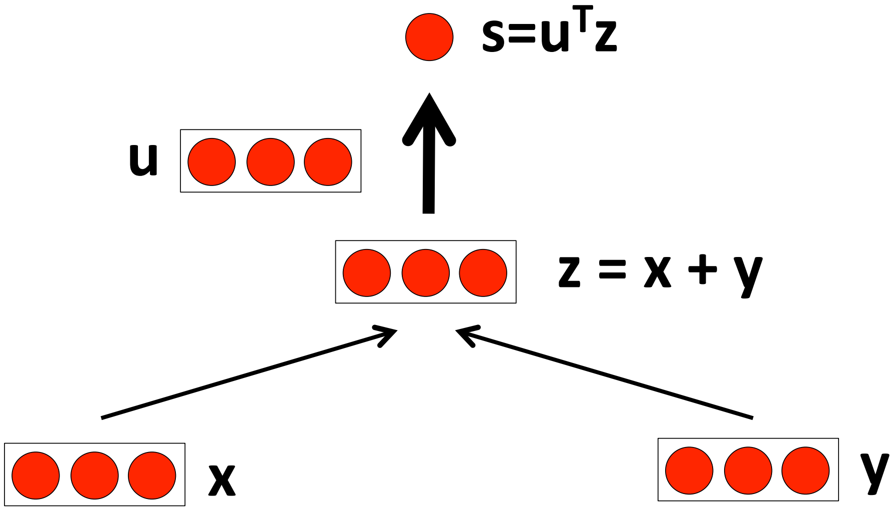
②Compute the score s_c of corrupted incorrect phrase

③Minimize cost function by SGD, $u \rightarrow u_{new}$, $z \rightarrow z_{new}$

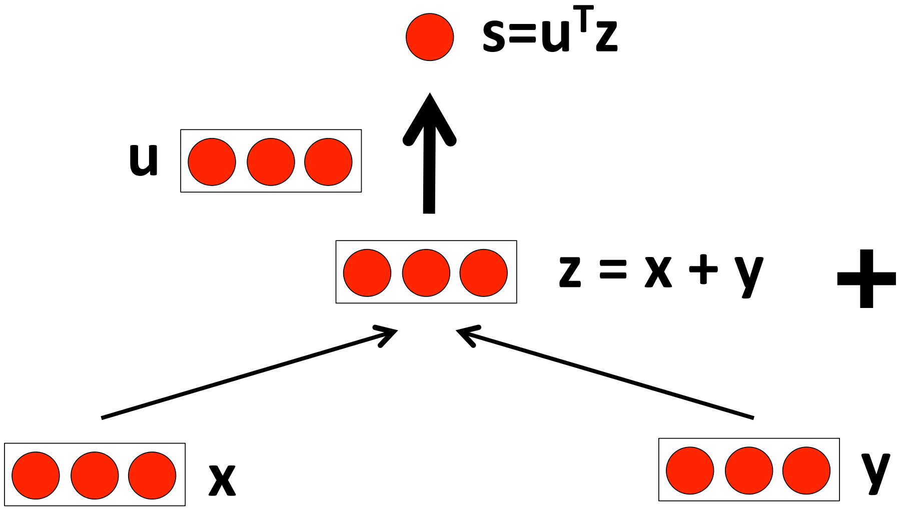
④New verb vector is $v_{new} = z_{new} - o$

New word representations considering compositionality

Compositional Neural Language Model



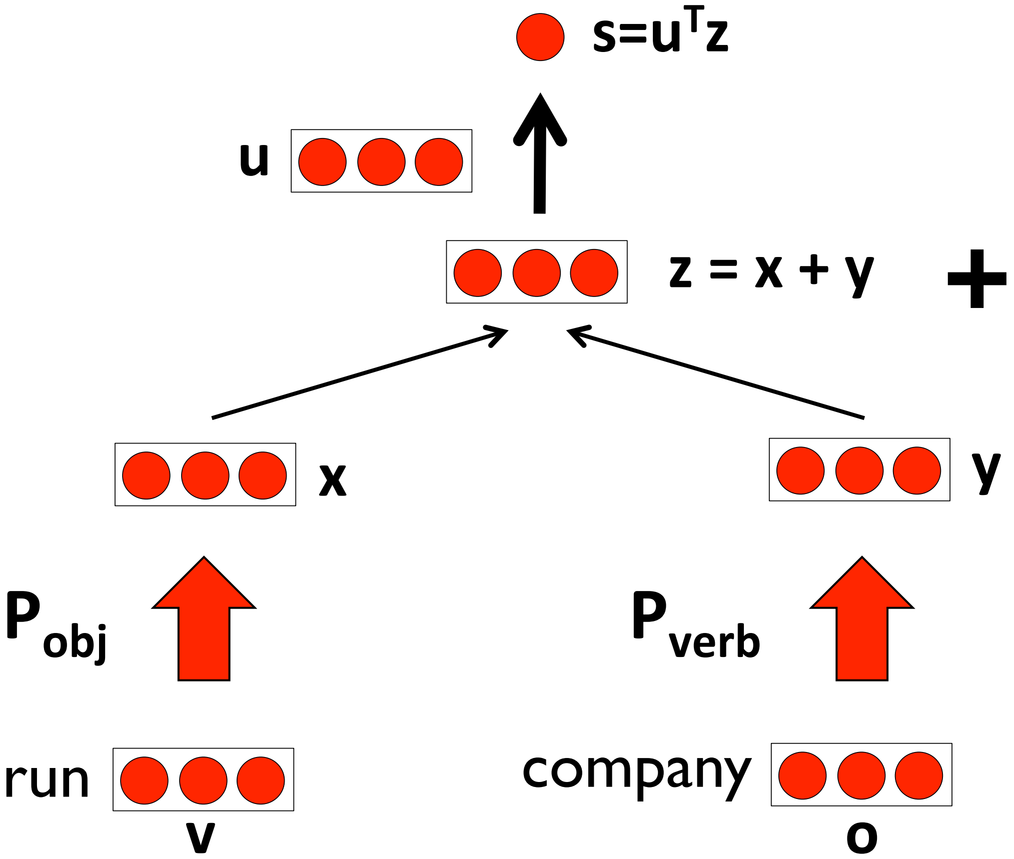
Compositional Neural Language Model



+

Co-Compositionality with Prototype Projection

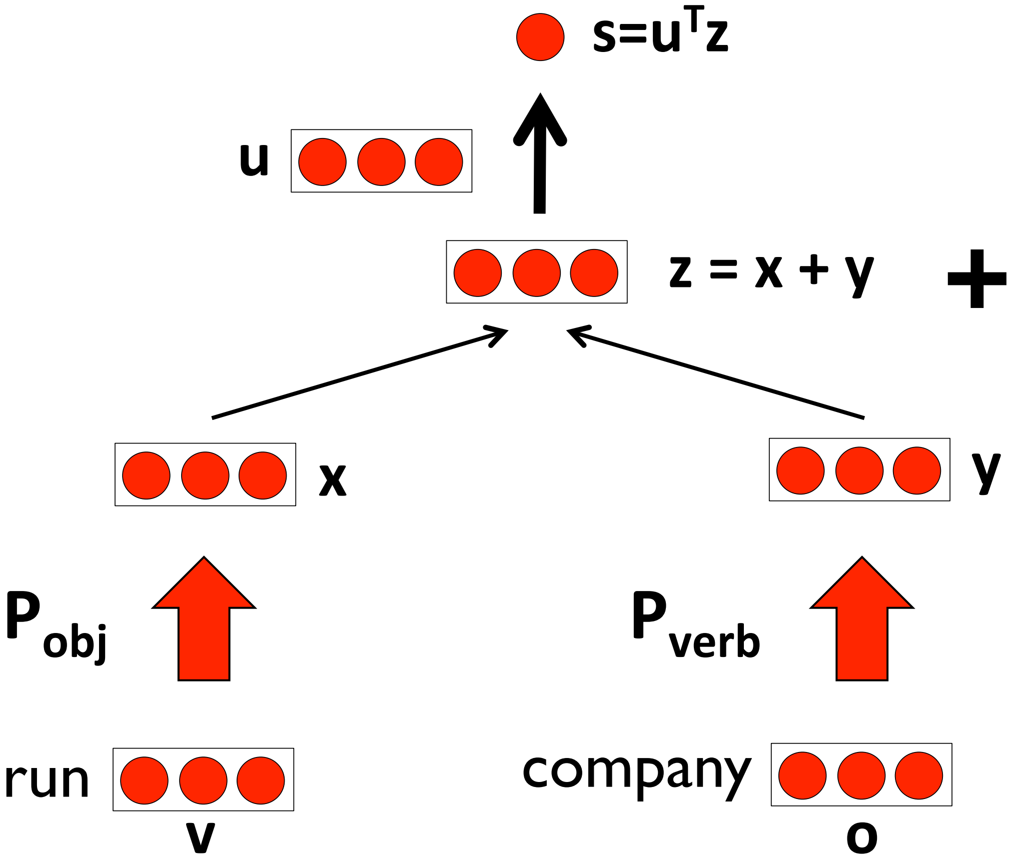
Compositional Neural Language Model



+ Co-Compositionality with Prototype Projection

Co-Compositional Neural Language Model

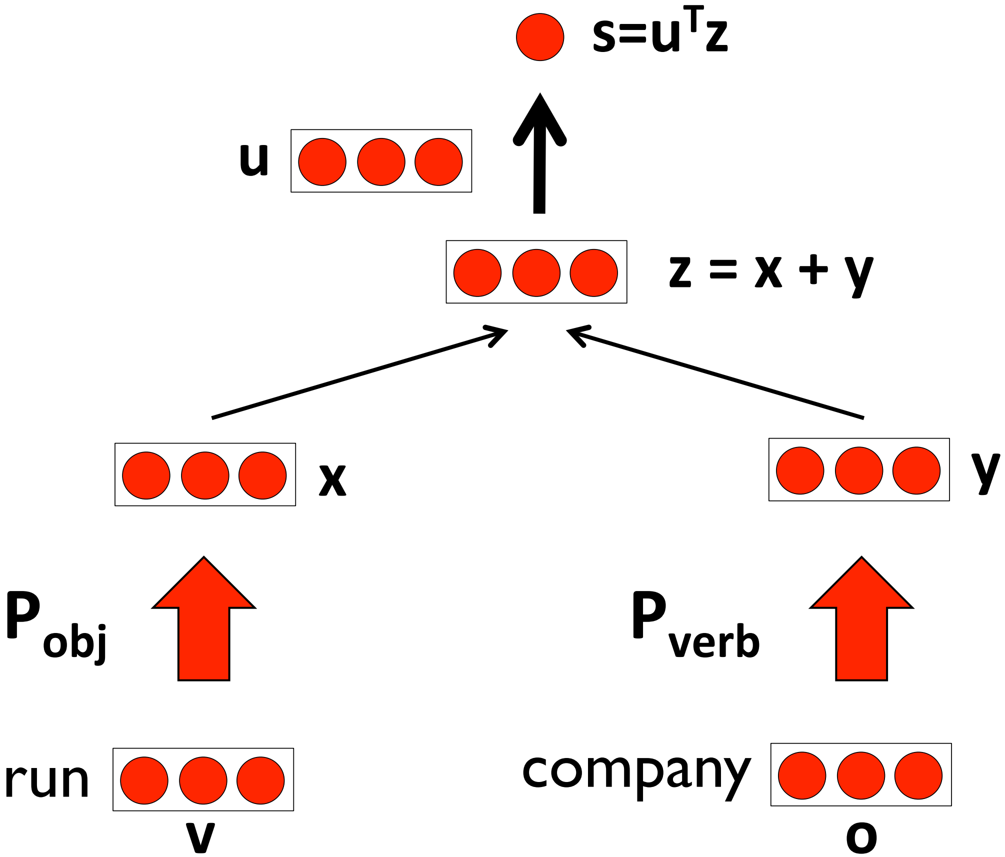
Compositional Neural Language Model with Prototype Projection



+ Co-Compositionality with Prototype Projection

Co-Compositional Neural Language Model

Compositional Neural Language Model with Prototype Projection



①Prototype projection for both verb and object

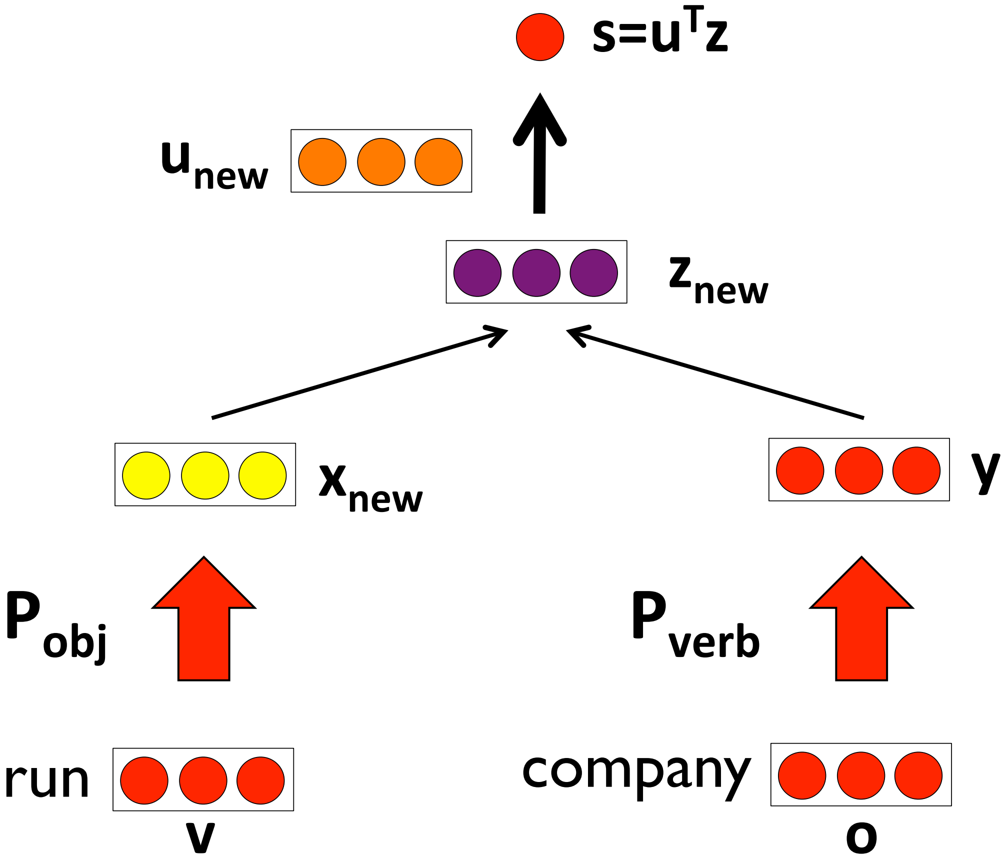
②Optimize parameters with same method as Compositional NLM

③Minimize

$$\min_{v}\left(\left\|x_{new}-P_{obj}v\right\|^{2}+\lambda\left\|v\right\|^{2}\right)$$

Co-Compositional Neural Language Model

Compositional Neural Language Model with Prototype Projection



①Prototype projection for both verb and object

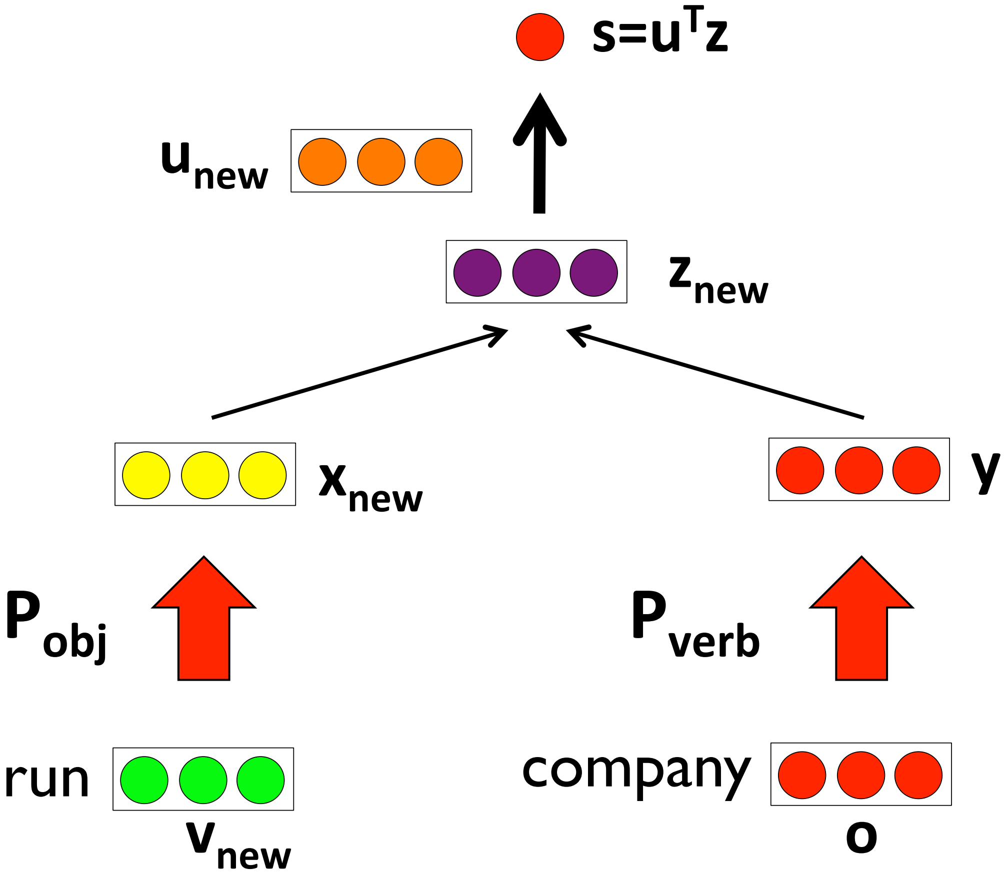
②Optimize parameters with same method as Compositional NLM

③Minimize

$$\min_{v}\left(\left\|x_{new}-P_{obj}v\right\|^{2}+\lambda\left\|v\right\|^{2}\right)$$

Co-Compositional Neural Language Model

Compositional Neural Language Model with Prototype Projection



①Prototype projection for both verb and object

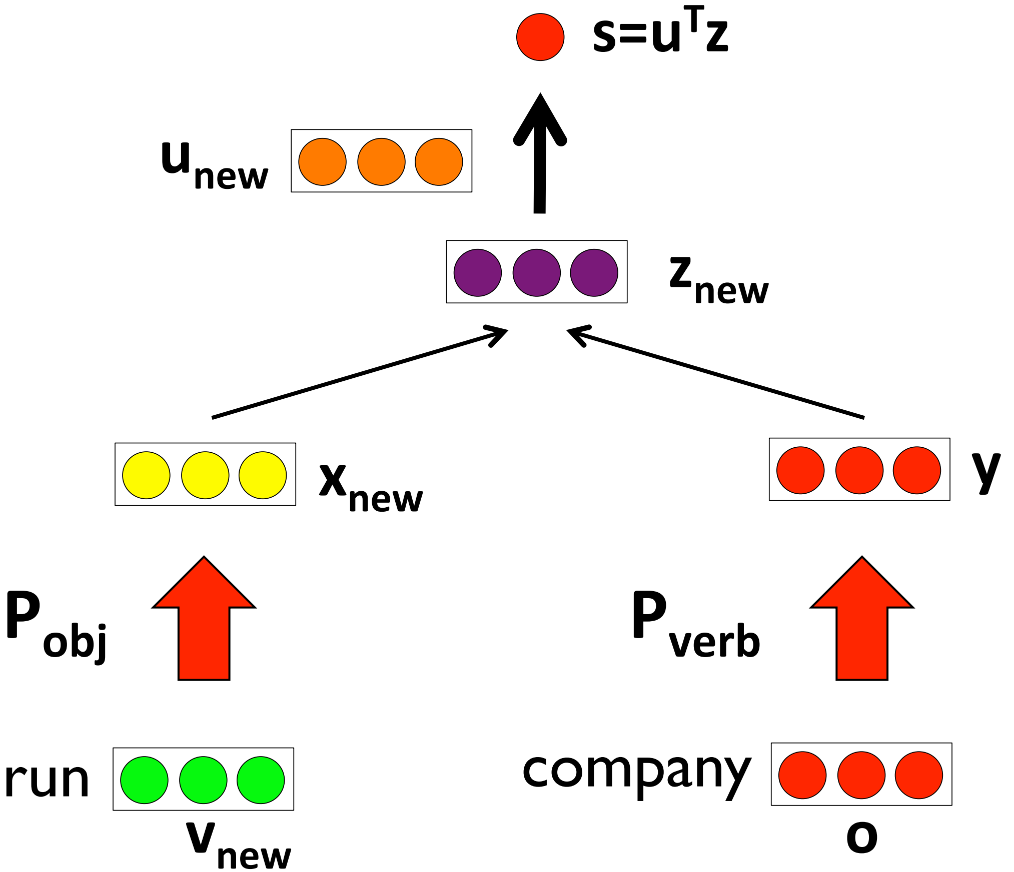
②Optimize parameters with same method as Compositional NLM

③Minimize

$$\min_{v}\left(\left\|x_{new} - P_{obj}v\right\|^2 + \lambda\left\|v\right\|^2\right)$$

Co-Compositional Neural Language Model

Compositional Neural Language Model with Prototype Projection



①Prototype projection for both verb and object

②Optimize parameters with same method as Compositional NLM

③Minimize

$$\min_{v}\left(\left\|x_{new} - P_{obj}v\right\|^2 + \lambda\left\|v\right\|^2\right)$$

New word representations considering co-compositionality

Evaluation : Verb disambiguation [Grefenstette and Sadrzadeh 11]

Original neural vector [Blacoe and Lapata 12]

vs.

Re-trained neural vector with our learning models

Evaluation : Verb disambiguation [Grefenstette and Sadrzadeh 11]

Original neural vector [Blacoe and Lapata 12]

vs.

Re-trained neural vector with our learning models

Training data

Extracted 5000 Verb-Obj pairs from ukWaC corpus

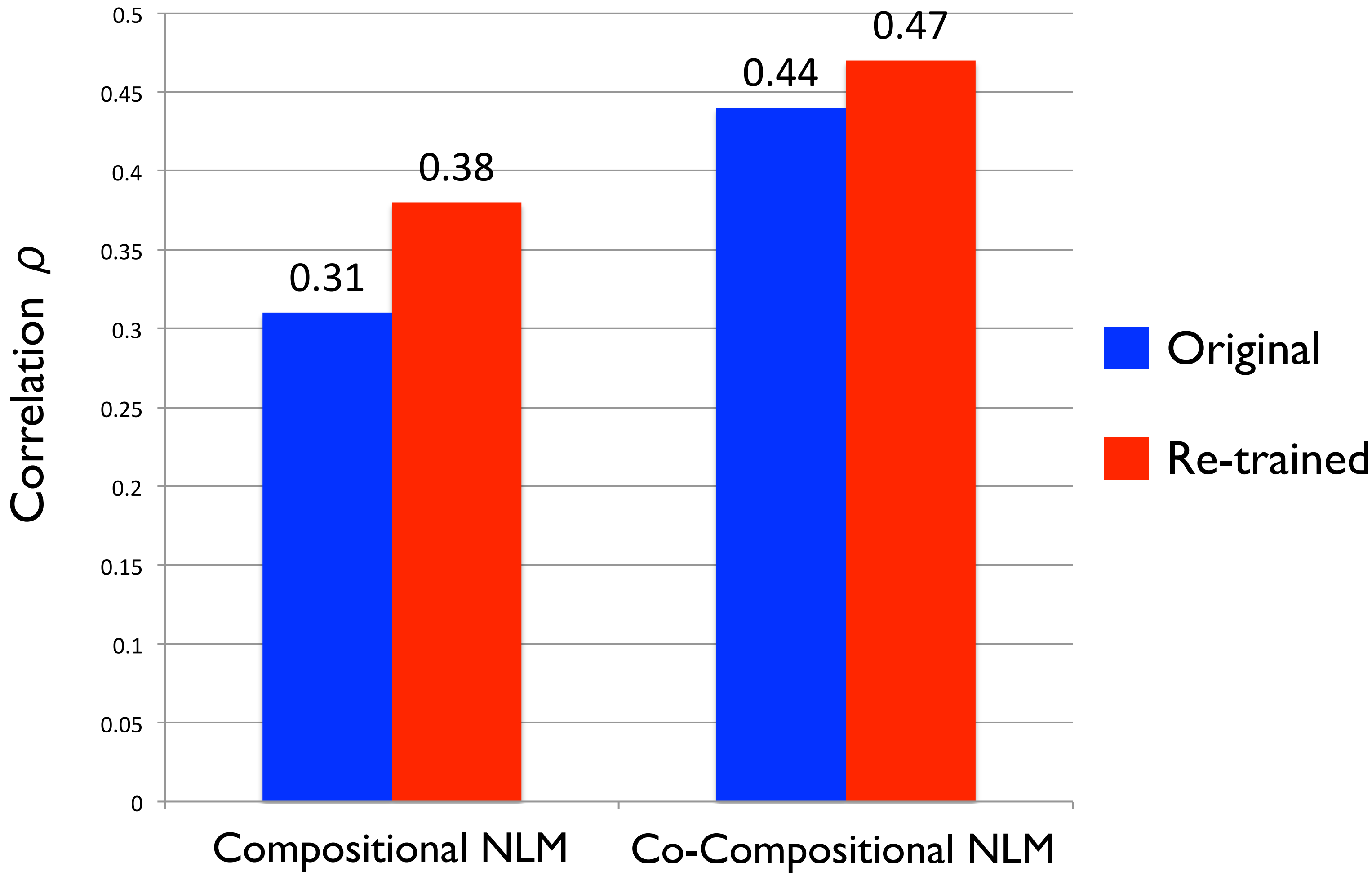
Hyper-parameters

Learning rate : 0.01, Regularization : 10^4

20 iterations (One iteration is one run through the training data)

Correlation with human judgment (Re-trained neural vector)

New state of the art performance (ρ = 0.47)



Higher performance with re-trained word representation

Conclusion

New model of compositionality in word vector space

Co-Compositionality with Prototype Projection

Unsupervised word vector re-training algorithm considering compositionality

Compositional & Co-Compositional Neural Language Models
Achieve state of the art on verb disambiguation task

Examples

verb	object	landmark	similarity(verb, landmark)	similarity(projected verb, landmark)
run	company	operate	0.40	0.70
meet	criterion	satisfy	0.49	0.71
spell	name	write	0.04	0.50

Table 1: Examples of verb-object pairs. Original verb and landmark verb similarity, prototype projected verb and landmark verb similarity, as measure by cosine using Collobert and Weston's word embeddings. *Meet* has a abstract meaning itself, but after prototype projection with matrix constructed by word vectors of $W(\mathit{VerbOf}, \mathit{criterion})$, *meet* is more close to meaning of *satisfy*.

Results of the different compositionality models

Model	ρ
Grefenstette and Sadrzadeh (2011)	0.21
Add (SDS)	0.31
Add (NLM)	0.31
Multiply (SDS)	0.35
Multiply (NLM)	0.30
Van de Cruys et al. (2013)	0.37
Erk and Padó (SDS)	0.39
Erk and Padó (NLM)	0.03
Co-Comp with f=Add (SDS)	0.41
Co-Comp with f=Add (NLM)	**0.44**
Co-Comp with f=Multiply (SDS)	0.37
Co-Comp with f=Multiply (NLM)	0.35
Upper bound	0.62

Table 3: Results of the different compositionality models on the similarity task. The number of prototype words $m = 20$ in all our models. Our model (f=Addition and NLM) achieves the new state-of-the-art performance for this task ($\rho = 0.44$).

The number of prototype words

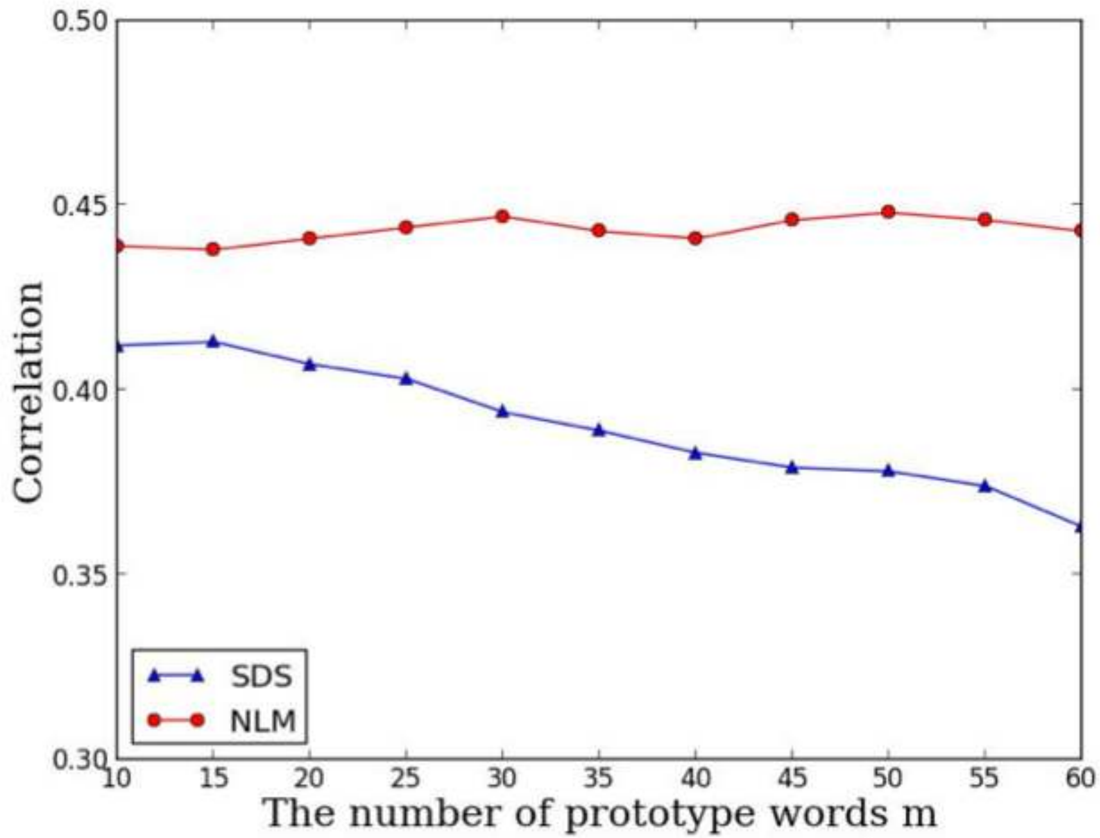


Figure 5: The relation between the number of prototype words and correlation of SDS or NLM. In general, NLM has higher correlation than SDS and is more robust across the m.

Variations in model configuration

Subj	Verb	Obj	NLM ρ	SDS ρ
prpj	prpj	prpj	0.39	0.37
+	prpj	prpj	0.44	0.41
	prpj	prpj	0.45	0.41
+	prpj	+	0.43	0.38
	prpj	+	0.43	0.38
+	+	+	0.31	0.31

Table 5: Variants of the full co-compositional model, based on how subject, verb, and object vector representations are included. prpj indicates that prototype projection is used. + indicates that the vector is added without projection first. Blank indicates that the vector is not used in the final compositional score.

Composition operator and parameter

Composition Operator	Parameter
Add: $w_1 u + w_2 v$	$w_1, w_2 \in \mathbb{R}$
Multiply: $u^{w_1} \odot v^{w_2}$	$w_1, w_2 \in \mathbb{R}$
FullAdd: $W_1 u + W_2 v$	$W_1, W_2 \in \mathbb{R}^{n \times n}$
LexFunc: $A_u v$	$A_u \in \mathbb{R}^{n \times n}$
FullLex: $\sigma([W_1 A_u v, W_2 A_v u])$	$A_u, A_v \in \mathbb{R}^{n \times n}$
	$W_1, W_2 \in \mathbb{R}^{n \times n}$
Ours (Add): $P_{(R,v)} u + P_{(R,u)} v$	SVD's (m, k)
Ours (Mult): $P_{(R,v)} u \odot P_{(R,u)} v$	SVD's (m, k)

Table 6: Comparison of composition operators that combine two word vector representations, $u, v \in \mathbb{R}^n$ and their learning parameters. Our model only needs two hyper-parameters: the number of prototype words m and dimensional reduction k in SVD