A Tensor-based Factorization Model of Semantic Compositionality

Tim Van de Cruys, Thierry Poibeau and Anna Korhonen

(ACL 2013)

Presented by Mamoru Komachi
<komachi@tmu.ac.jp>

The 5th summer camp of NLP
2013/08/31

The principle of compositionality

- Dates back to Gottlob Frege (1892)
- "… meaning of a complex expression is a function of the meaning of its parts and the way those parts are (syntactically) combined"

Compositionality is modeled as a multi-way interaction between latent factors

- Propose a method for computation of compositionality within a distributional framework
  - Compute a latent factor model for nouns
  - The latent factors are used to induce a latent model of three-way (*subject*, *verb*, *object*) interactions, represented by a core tensor
- Evaluate on a similarity task for transitive phrases (SVO)

Previous work

Distributional framework for semantic composition

Previous work: Mitchell and Lapata (ACL 2008)

- Explore a number of different models for vector composition:
  - Vector addition: $p_i = u_i + v_i$
  - Vector multiplication: $p_i = u_i \cdot v_i$
- Evaluate their models on a noun-verb phrase similarity task
  - Multiplicative model yields the best results
- One of the first approaches to tackle compositional phenomena (baseline in this work)

Previous work: Grefenstette and Sadrzadeh (EMNLP 2011)

- An instantiation of Coecke et al. (Linguistic Analysis 2010)
  - A sentence vector is a function of the Kronecker product of its word vectors

$$\overrightarrow{sub\,verb\,obj} = (\overrightarrow{sub} \circ \overrightarrow{obj}) * \overrightarrow{verb}$$

- Assume that relational words (e.g. adjectives or verbs) have a rich (multi-dimensional) structure
- Proposed model uses an intuition similar to theirs (the other baseline in this work)

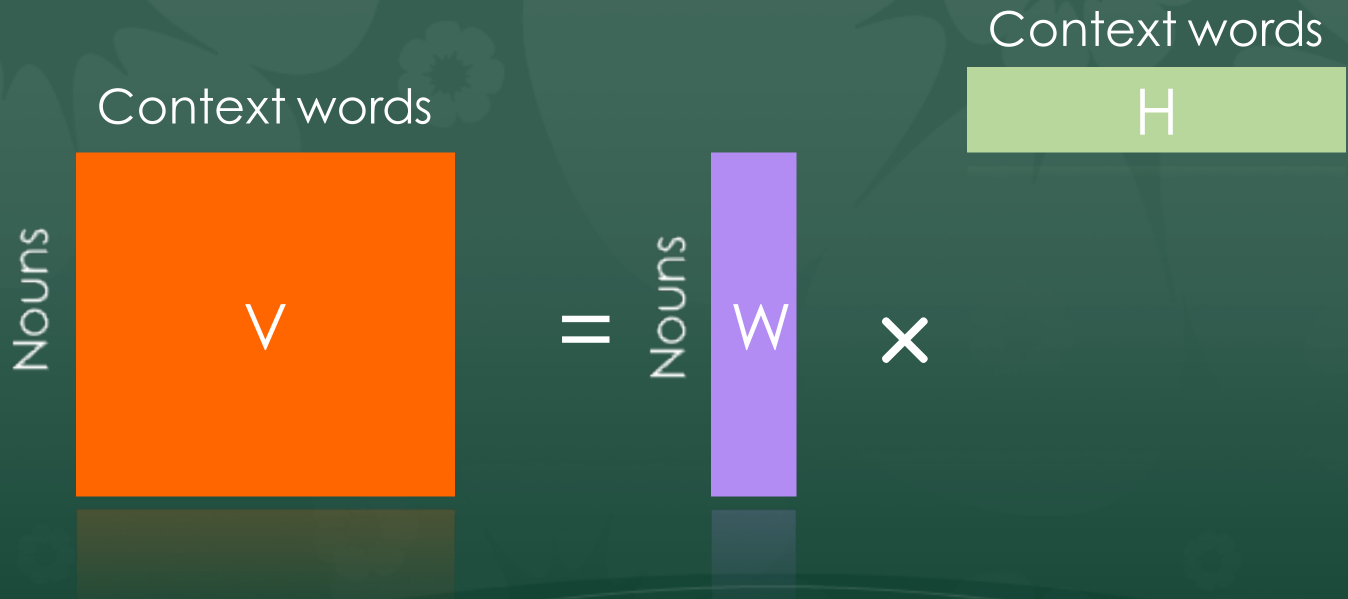
Overview of compositional semantics

	input	target	operation
Mitchell and Lapata (2008)	Vector	Noun-verb	Add & mul
Baroni and Zamparelli (2010)	Vector	Adjective & noun	Linear transformation (matrix mul)
Coecke et al. (2010), Grefenstette and Sadrzadeh (2011)	Vector	Sentence	Krochecker product
Socher et al. (2010)	Vector + matrix	Sentence	Vector & matrix mul

Methodology

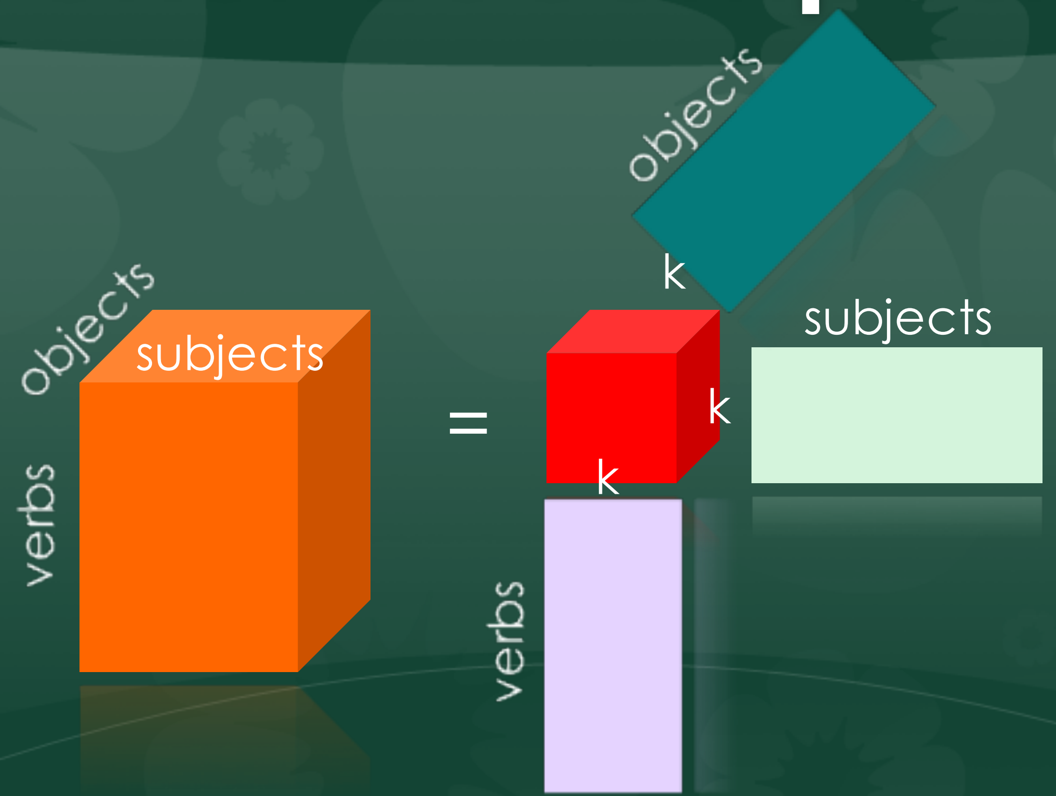
The composition of SVO triples

Construction of latent noun factors



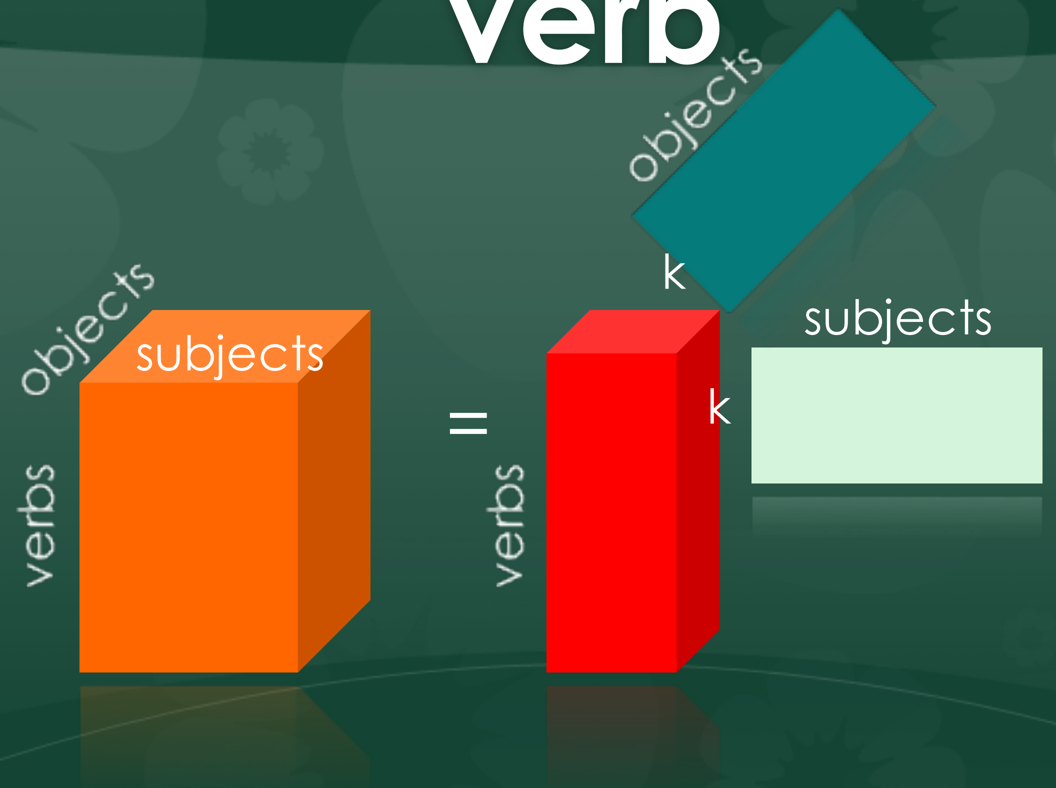
- Non-negative matrix factorization (NMF)
- Minimizes KL divergence between an original matrix $V_{I \times J}$ and $W_{I \times K} H_{K \times J}$ s.t. all values of the in the three matrices be non-negative

Tucker decomposition



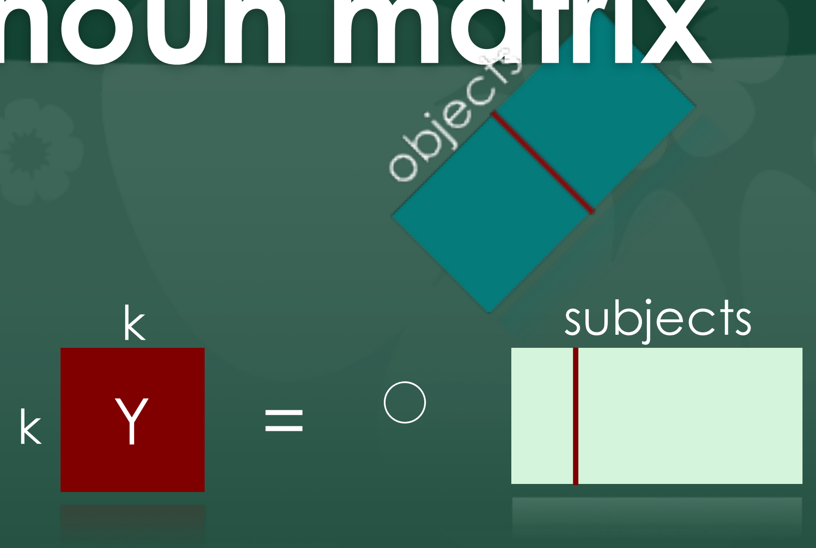
- Generalization of the SVD
- Decompose a tensor into a core tensor, multiplied by a matrix along each mode

Decomposition w/o the latent verb



- Only the subject and object mode are represented by latent factors (to be able to efficiently compute the similarity of verbs)

Extract the latent vectors from noun matrix



$$\mathbf{Y}_{<athlete,race>} = \mathbf{w}_{athlete} \circ \mathbf{w}_{race}$$

The athlete runs a race.

- Compute the outer product (○) of subject and object.

Capturing the latent interactions with verb matrix

$$Z_{run,<athlete,race>} = G_{<athelete,race>} * Y$$

- Take the Hadamard product (*) of matrix Y with verb matrix G, which yields our final matrix Z.

Examples & Evaluation

Semantic features of the subject combine with semantic features of the object

factors	subject	object	value
$\langle 195, 119\rangle$	*people* (.008), *child* (.008), *adolescent* (.007)	*cup* (.007), *championship* (.006), *final* (.005)	.007
$\langle 25, 119\rangle$	*hockey* (.007), *poker* (.007), *tennis* (.006)	*cup* (.007), *championship* (.006), *final* (.005)	.004
$\langle 90, 119\rangle$	*professionalism* (.007), *teamwork* (.007), *confidence* (.006)	*cup* (.007), *championship* (.006), *final* (.005)	.003
$\langle 28, 119\rangle$	*they* (.004), *pupil* (.003), *participant* (.003)	*cup* (.007), *championship* (.006), *final* (.005)	.003

Table 1: Factor pairs with highest value for matrix $\mathbf{Y}_{\langle athlete, race\rangle}$

factors	subject	object	value
$\langle 7, 89\rangle$	*password* (.009), *login* (.007), *username* (.007)	*filename* (.007), *null* (.006), *integer* (.006)	.010
$\langle 40, 89\rangle$	*anyone* (.004), *reader* (.004), *anybody* (.003)	*filename* (.007), *null* (.006), *integer* (.006)	.007
$\langle 195, 89\rangle$	*people* (.008), *child* (.008), *adolescent* (.007)	*filename* (.007), *null* (.006), *integer* (.006)	.006
$\langle 45, 89\rangle$	*website* (.004), *Click* (.003), *site* (.003)	*filename* (.007), *null* (.006), *integer* (.006)	.006

Table 2: Factor pairs with highest value for matrix $\mathbf{Y}_{\langle user, command\rangle}$

Animacy: 28, 40, 195; Sport: 25; Sport event: 119; Tech: 7, 45, 89

Verb matrix contains the verb semantics computed over the complete corpus

factors	**subject**	**object**	**value**
$\langle 128, 181\rangle$	*Mathematics* (.004), *Science* (.004), *Economics* (.004)	*course* (.005), *tutorial* (.005), *seminar* (.005)	.058
$\langle 293, 181\rangle$	*organization* (.007), *association* (.007), *federation* (.006)	*course* (.005), *tutorial* (.005), *seminar* (.005)	.053
$\langle 60, 140\rangle$	*rail* (.011), *bus* (.009), *ferry* (.008)	*third* (.004), *decade* (.004), *hour* (.004)	.038
$\langle 268, 268\rangle$	*API* (.008), *Apache* (.007), *Unix* (.007)	*API* (.008), *Apache* (.007), *Unix* (.007)	.038

Table 3: Factor combinations for $\mathbf{G}_{run}$

'Organize' sense: <128, 181>; <293, 181>
'Transport' sense: <60, 140>
'Execute' sense: <268, 268>

Tensor G captures the semantics of the verb

- Most similar verbs from Z
  - $Z_{run,\langle athlete,race\rangle}$: finish (.29), attend (.27), win (.25)
  - $Z_{run\langle user,command\rangle}$: execute (.42), modify (.40), invoke (.39)
  - $Z_{damage,\langle man,car\rangle}$: crash (.43), drive (.35), ride (.35)
  - $Z_{damage,\langle car,man\rangle}$: scare(.26), kill (.23), hurt (.23)
- Similarity is calculated by measuring the cosine of the vectorized representation of the verb matrix
- Can distinguish word order

Transitive (SVO) sentence similarity task

- Extension of the similarity task (Mitchell and Lapata, ACL 2008)
  - http://www.cs.ox.ac.uk/activities/CompDistMeaning/GS2011data.txt
  - 2,500 similarity judgments
  - 25 participants

p	target	subject	object	landmark	sim
19	meet	system	criterion	visit	1
21	write	student	name	spell	6

Latent model outperforms previous models

model	contextualized	Non-contextualized
baseline	.23	
multiplicative	**.32**	.34
categorical	**.32**	.35
latent	**.32**	**.37**
Upper bound	.62	

- Multiplicative (Mitchell and Lapata, ACL-2008)
- Categorical (Grefenstette and Sadrzadeh, 2011)
- Upper bound = inter-annotator agreement (Grefenstette and Sadrzadeh, EMNLP 2011)

Conclusion

- Proposed a novel method for computation of compositionality within a distributional framework
  - Compute a latent factor model for nouns
  - The latent factors are used to induce a latent model of three-way (*subject*, *verb*, *object*) interactions, represented by a core tensor
- Evaluated on a similarity task for transitive phrases and exceeded the state of the art