Web-Mining Agents

Multi-Relational Latent Semantic Analysis

Prof. Dr. Ralf Möller
Universität zu Lübeck
Institut für Informationssysteme

Tanya Braun (Übungen)

Acknowledgements

Slides by: Scott Wen-tau Yih

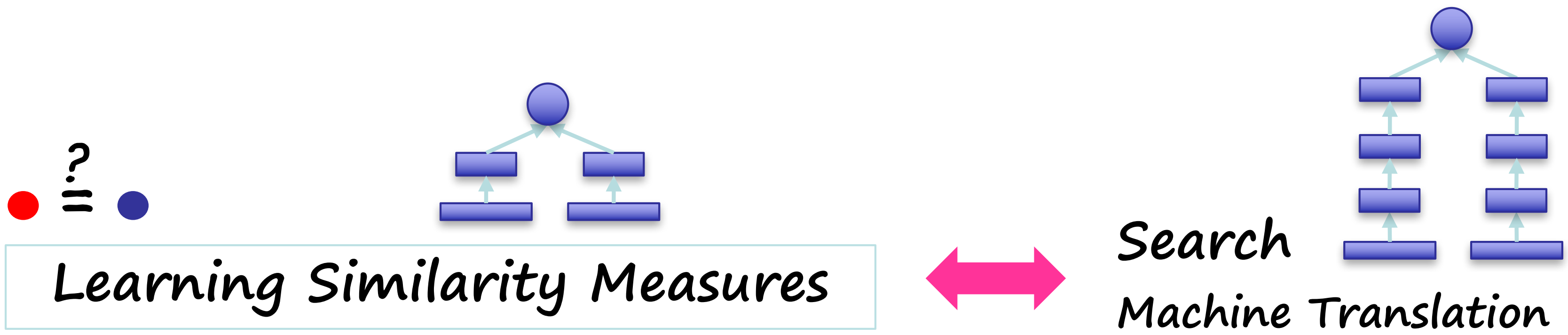
Describing joint work of Scott Wen-tau Yih with
Kai-Wei Chang, Bishan Yang,
Chris Meek, Geoff Zweig, John Platt

Microsoft Research

Continuous Semantic Representations

Text entities are represented as vectors

- *Words, phrases, sentences, or documents*
- *Learned via neural networks*
  or matrix/tensor decomposition methods
- *Relations are estimated by functions in the vector space*



S2Net [CoNLL-11, SIGIR-11]

DSSM [CIKM-13, ACL-14]

DSSM Deep Structured Semantic Model,
or more general, Deep Semantic Similarity Model

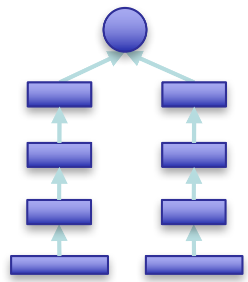
Continuous Semantic Representations

Learning Similarity Measures

S2Net [CoNLL-11, SIGIR-11]

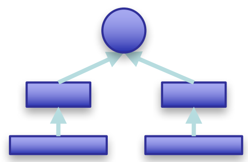
Search

Machine Translation



DSSM [CIKM-13, ACL-14]

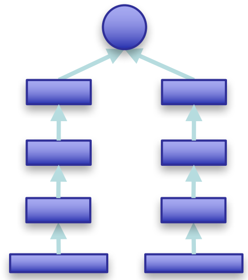
Continuous Semantic Representations



Learning Similarity Measures

Search

Machine Translation

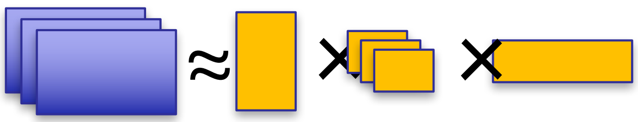


S2Net [CoNLL-11, SIGIR-11]

DSSM [CIKM-13*, ACL-14]

$f_{rel}(\bullet, \bullet)$

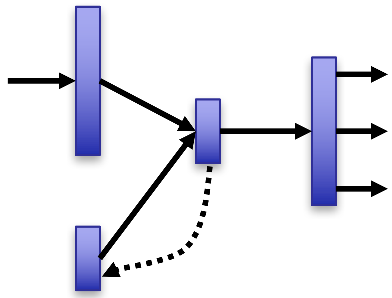
Multi-Relational LSA



Word Relation [EMNLP-12, EMNLP-13]
Knowledge Base Embedding [EMNLP-14]

Relational Similarity

Word Analogy
[NAACL-13 x2]



Open-Domain Question Answering

Fulfill user's information need with direct answers

- Answer Sentence Selection [ACL-13]

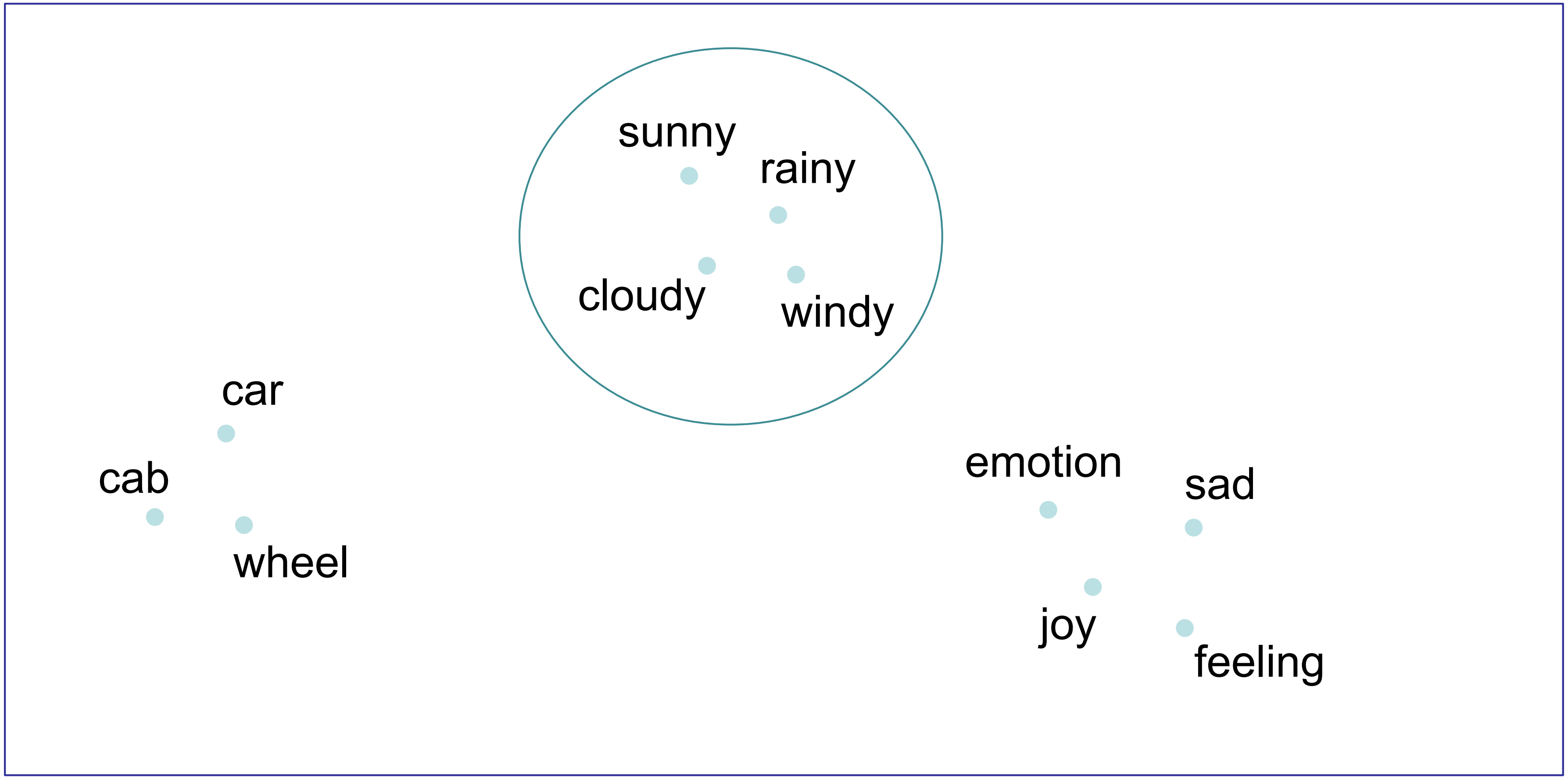
Q: Who won the best actor Oscar in 1973?
S_1: Jack Lemmon was awarded the Best Actor Oscar for Save the Tiger (1973).
S_2: Academy award winner Kevin Spacey said that Jack Lemmon is remembered as always making time for others.

- Word-alignment based approaches with enhanced lexical semantic models

Continuous Semantic Representations

- A lot of popular methods for creating word vectors!
  - Vector Space Model [Salton & McGill 83]
  - Latent Semantic Analysis [Deerwester+ 90]
  - Latent Dirichlet Allocation [Blei+ 01]
  - DNN [Collobert & Weston 08]
    - Chunking, POS, NER, SRL, (modeling long-distance modeling long-distance dependencies with time-delay networks)
  - Word2Vec [Mikolov+ 13]
- Encode term co-occurrence information
- Measure semantic similarity well

Continuous Semantic Representations



Semantics Needs More Than Similarity

Tomorrow will be *rainy*.

Tomorrow will be *sunny*.

~~*similar*(rainy, sunny)?~~

antonym(rainy, sunny)?

Leverage Linguistic Knowledge Bases

- Can't we just use the existing linguistic KBs?
  - Knowledge in these resources is never complete
  - Often lack of degree of relations
- Create a continuous semantic representation that
  - Leverages existing rich linguistic knowledge bases
  - Discovers new relations
  - Enables us to measure the degree of multiple relations (not just similarity)

Roadmap

- Two opposite relations:
  Polarity Inducing Latent Semantic Analysis (PILSA)
- More relations:
  Multi-Relational Latent Semantic Analysis (MRLSA)
- Relational domain knowledge:
  Typed MRLSA (TRESCAL)

- Yih, Zweig & Platt. *Polarity Inducing Latent Semantic Analysis*. In EMNLP-CoNLL-12.
- Chang, Yih & Meek. *Multi-Relational Latent Semantic Analysis*. In EMNLP-13.
- Chang, Yih, Yang & Meek. *Typed Tensor Decomposition of Knowledge Bases for Relation Extraction*. In EMNLP-14.

EMNLP: Empirical Methods in Natural Language Processing
CoNLL: Computational Natural Language Learning
ACL; Annual Meeting of the Association for Computational Linguistics

LSA, word2vec, and friends

- Can cope with homonyms due to word context

Problem: Handling Two Opposite Relations

Synonyms & Antonyms

- LSA cannot distinguish antonyms [Landauer 2002]
  - "***Distinguishing synonyms and antonyms is still perceived as a difficult open problem.***" *[Poon & Domingos 09]*

- Idea #1: Change the data representation

Polarity Inducing LSA [Yih, Zweig & Platt 2012]

- Data representation
  - Encode two opposite relations in a matrix using “polarity”
    - Synonyms & antonyms (e.g., from a thesaurus)
- Factorization
  - Apply SVD to the matrix to find latent components
- Measuring degree of relation
  - Cosine of latent vectors

Encode Synonyms & Antonyms in Matrix

- Joyfulness: joy, gladden; sorrow, sadden
- Sad: sorrow, sadden; joy, gladden

Target word: row-vector

	joy	gladden	sorrow	sadden	goodwill
Group 1: "joyfulness"	1	1	1	1	0
Group 2: "sad"	1	1	1	1	0
Group 3: "affection"	0	0	0	0	1

Encode Synonyms & Antonyms in Matrix

- Joyfulness: joy, gladden; sorrow, sadden
- Sad: sorrow, sadden; joy, gladden

Target word: row-vector

Inducing polarity

	joy	gladden	sorrow	sadden	goodwill
Group 1: "joyfulness"	1	1	-1	-1	0
Group 2: "sad"	-1	-1	1	1	0
Group 3: "affection"	0	0	0	0	1

Cosine Score: $+$ *Synonyms*

Encode Synonyms & Antonyms in Matrix

- Joyfulness: joy, gladden; sorrow, sadden
- Sad: sorrow, sadden; joy, gladden

Target word: row-vector

Inducing polarity

	joy	gladden	sorrow	sadden	goodwill
Group 1: "joyfulness"	1	1	-1	-1	0
Group 2: "sad"	-1	-1	1	1	0
Group 3: "affection"	0	0	0	0	1

Cosine Score: $-Antonyms$

Problem: How to Handle More Relations?

- Limitation of the matrix representation
  - Each entry captures a particular type of relation between two entities, or
  - Two opposite relations with the polarity trick
- Encoding other binary relations
  - Is-A (hyponym) – ostrich *is a* bird
  - Part-whole – engine is a *part of* car

Idea #2:
Encode multiple relations in a 3-way tensor (3-dim array)!

Multi-Relational LSA (MR-LSA)

- Data representation
  - Encode multiple relations in a tensor
    - Synonyms, antonyms, hyponyms (is-a), … (e.g., from a linguistic knowledge base)
- Factorization
  - Apply tensor decomposition to the tensor to find latent components (→ RESCAL)
- Measuring degree of relation
  - Cosine of latent vectors after projection

Encode Multiple Relations in Tensor

- Represent word relations using a tensor
  - Each slice encodes a relation between terms and target words.

	joy	gladden	sadden	feeling
joyfulness	1	1	0	0
gladden	1	1	0	0
sad	0	0	1	0
anger	0	0	0	0

Synonym layer

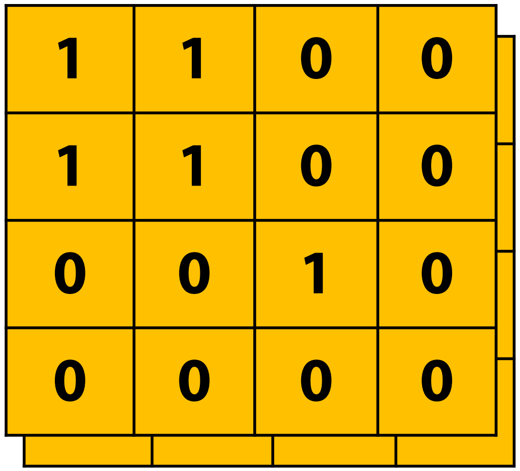
	joy	gladden	sadden	feeling
joyfulness	0	0	0	0
gladden	0	0	1	0
sad	1	0	0	0
anger	0	0	0	0

Antonym layer

Construct a tensor with two slices

Encode Multiple Relations in Tensor

- Can encode multiple relations in the tensor

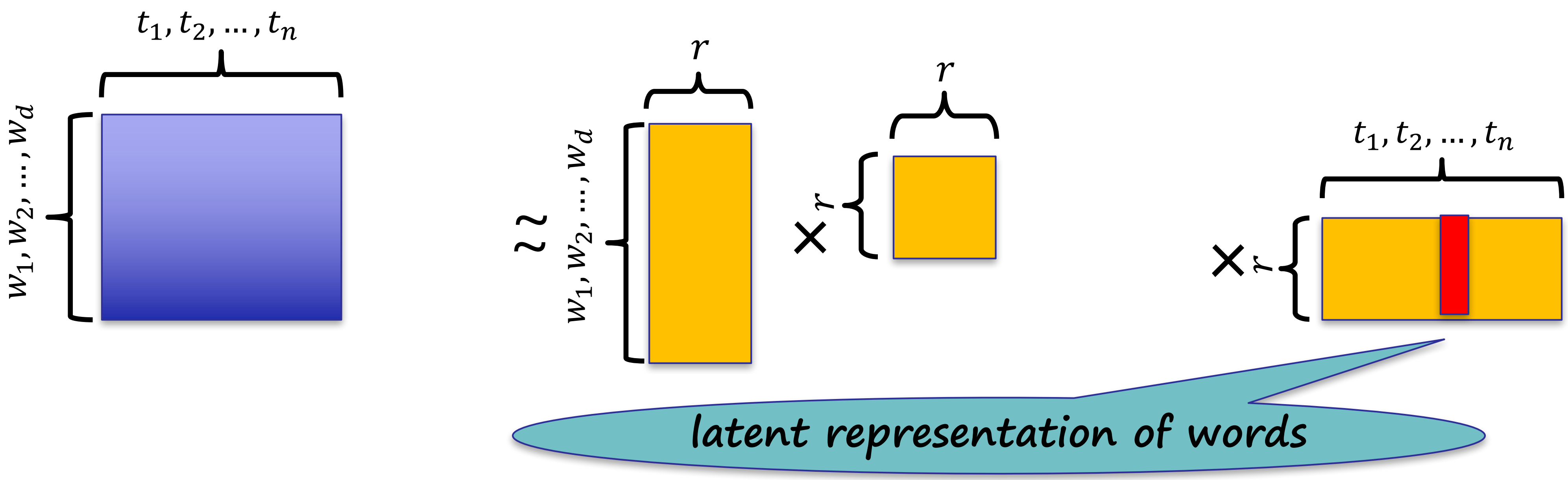


	joy	gladden	sadden	feeling
joyfulness	0	0	0	1
gladden	0	0	0	0
sad	0	0	0	1
anger	0	0	0	1

Hyponym layer

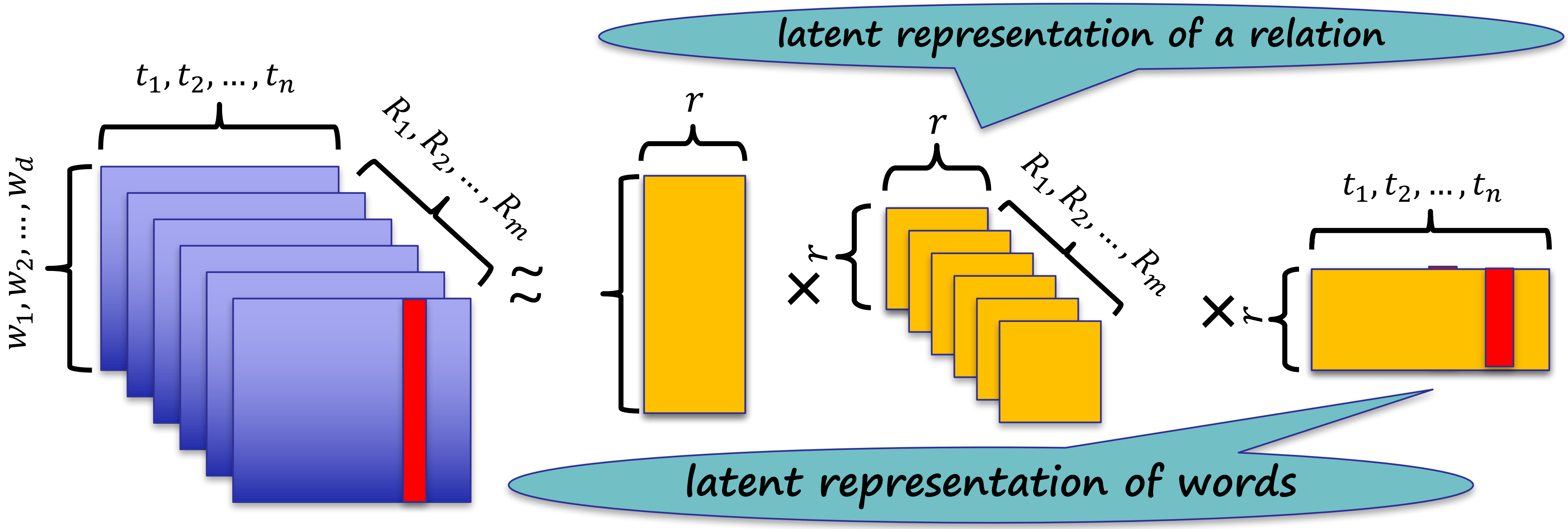
Tensor Decomposition – Analogy to SVD

- Derive a low-rank approximation to generalize the data and to discover unseen relations
- Apply Tucker decomposition and reformulate the results



Tensor Decomposition – Analogy to SVD

- Derive a low-rank approximation to generalize the data and to discover unseen relations
- Apply decomposition and reformulate the results



Measure Degree of Relation

- Similarity
  - Cosine of the latent vectors

- Other relation (both symmetric and asymmetric)
  - Take the latent matrix of the *pivot* relation (synonym)
  - Take the latent matrix of the relation
  - Cosine of the latent vectors after projection

Measure Degree of Relation: Raw Representation

- $ant(\text{joy}, \text{sadden}) = \cos\left(\boldsymbol{w}_{:,\text{joy},syn}, \boldsymbol{w}_{:,\text{sadden},ant}\right)$

	joy	gladden	sadden	felling
joyfulness	1	1	0	0
gladden	1	1	0	0
sad	0	0	1	0
anger	0	0	0	0

Synonym layer

	joy	gladden	sadden	felling
joyfulness	0	0	0	0
gladden	0	0	1	0
sad	1	0	0	0
anger	0	0	0	0

Antonym layer

Measure Degree of Relation: Raw Representation

- $ant(\text{joy}, \text{sadden}) = \cos(\boldsymbol{w}_{:,\text{joy},syn}, \boldsymbol{w}_{:,\text{sadden},ant})$

	joy	gladden	sadden	felling
joyfulness	1	1	0	0
gladden	1	1	0	0
sad	0	0	1	0
anger	0	0	0	0

Synonym layer

Ω

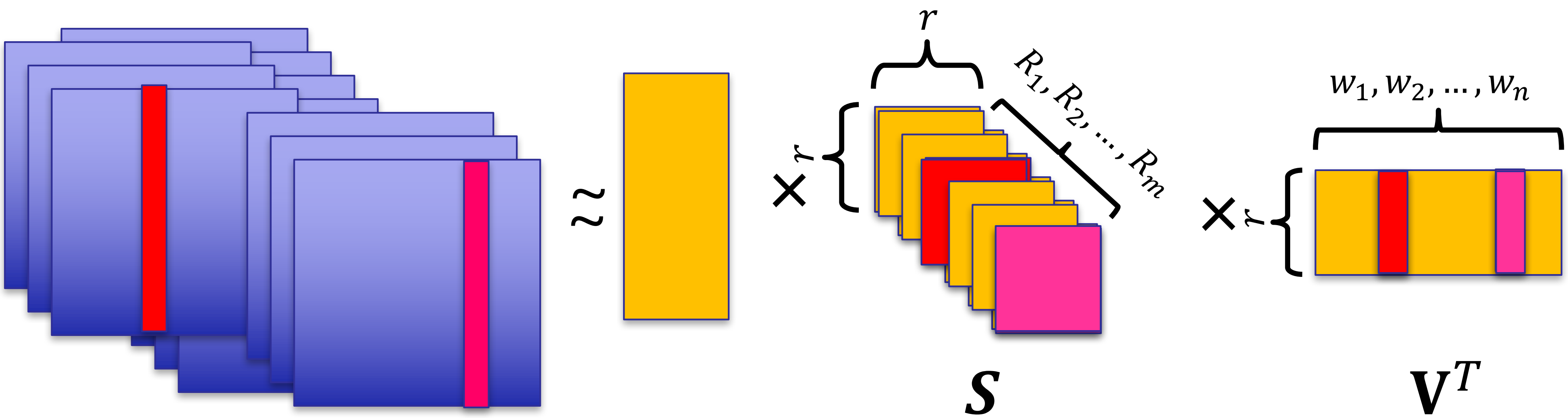
	joy	gladden	sadden	felling
joyfulness	0	0	0	0
gladden	0	0	1	0
sad	1	0	0	0
anger	0	0	0	0

Antonym layer

Measure Degree of Relation: Latent Representation

- $rel(\mathrm{w}_i, \mathrm{w}_j) = \cos(\boldsymbol{S}_{:,:,syn}\mathbf{V}^T_{i,:}, \boldsymbol{S}_{:,:,rel}\mathbf{V}^T_{j,:})$

$Cos \; (\quad \times \quad , \quad \times \quad)$



Problem: Use Relational Domain Knowledge

- Relational domain knowledge – the entity type
  - Relation can only hold between the right types of entities
    - Words having *is-a* relation have the same part-of-speech
    - For relation *born-in*, the entity types are: (person, location)
- Leverage type information to improve MRLSA
- Idea #3: Change the objective function

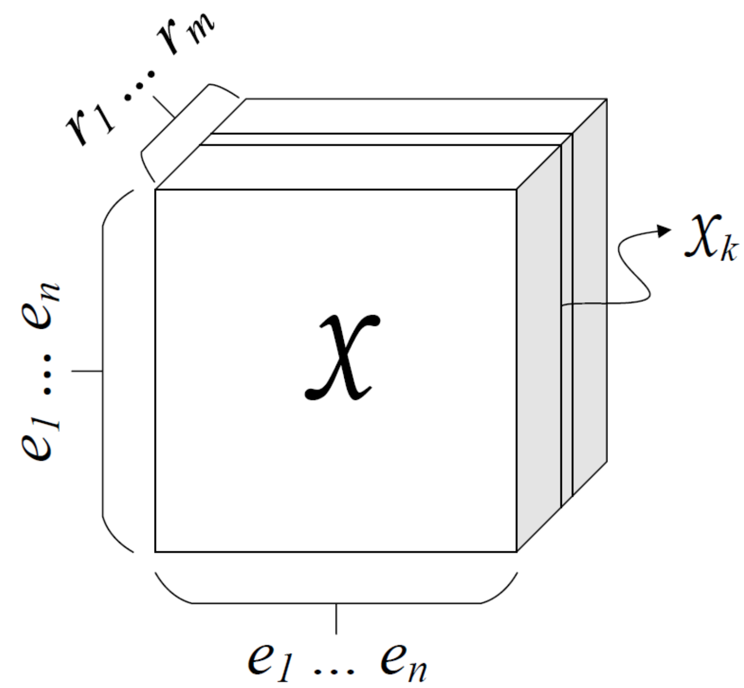
Typed Multi-Relational LSA (TRESCAL)

- Only legitimate entities are included in the objective function of tensor decomposition
- Benefits of leveraging the type information
  - Faster model training time
  - Higher prediction accuracy
- Experiments conducted using knowledge base
  - Application to Relation Extraction

Knowledge Base Representation (1/2)

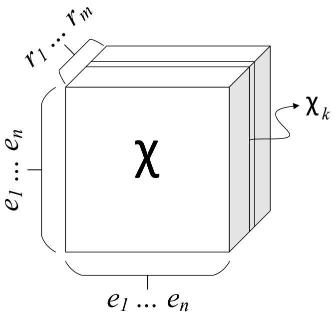
- Collection of subj-pred-obj triples – (e_1, r, e_2)

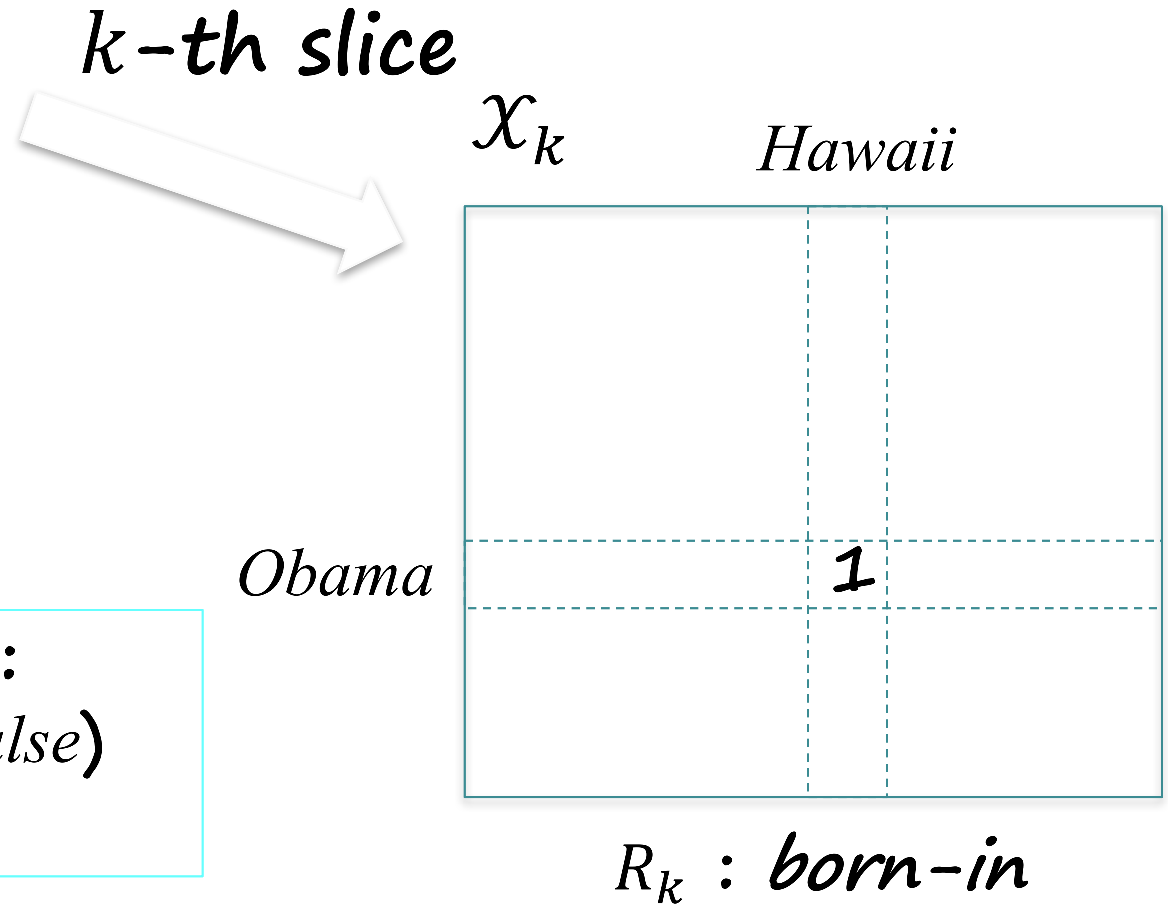
Subject	Predicate	Object
Obama	Born-in	Hawaii
Bill Gates	Nationality	USA
Bill Clinton	Spouse-of	Hillary Clinton
Satya Nadella	Work-at	Microsoft
…	…	…



n: # entities, m: # relations

Knowledge Base Representation (2/2)





A 0 entry means:

- Incorrect (*false*)
- Unknown

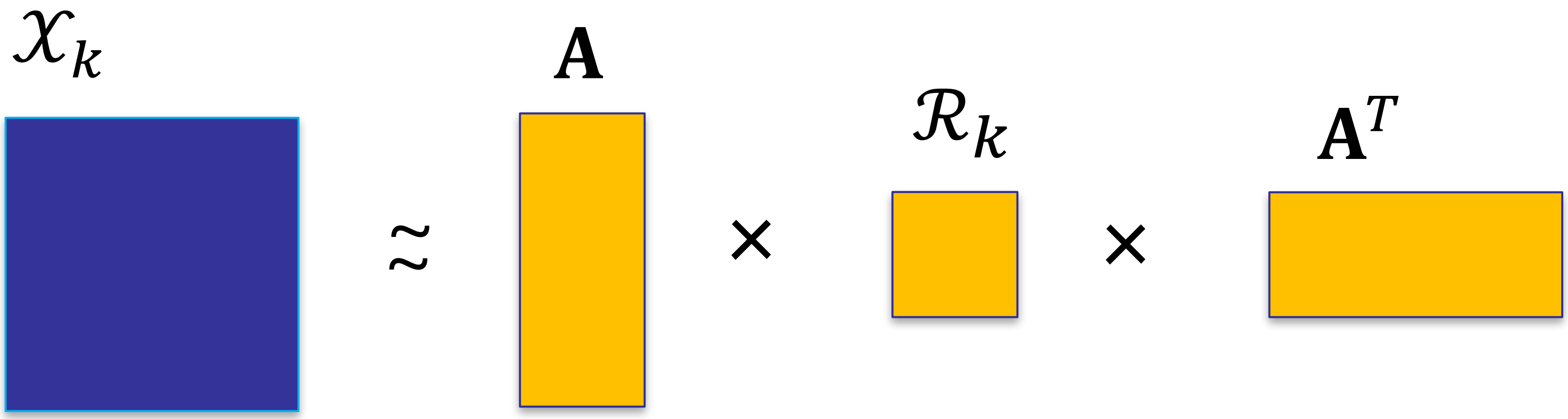
Knowledge Base Embedding

- Each entity in a KB is represented by an R^d vector
- Predict whether (e_1, r, e_2) is true by $f_r\left(v_{e_1}, v_{e_2}\right)$

- Related Work
  - RESCAL [Nickel+, ICML-11]
  - SME [Bordes+, AISTATS-12]
  - NTN [Socher+, NIPS-13]
  - TransE [Bordes+, NIPS-13]
  - TransH [Wang+, AAAI-14]

Tensor Decomposition Objective

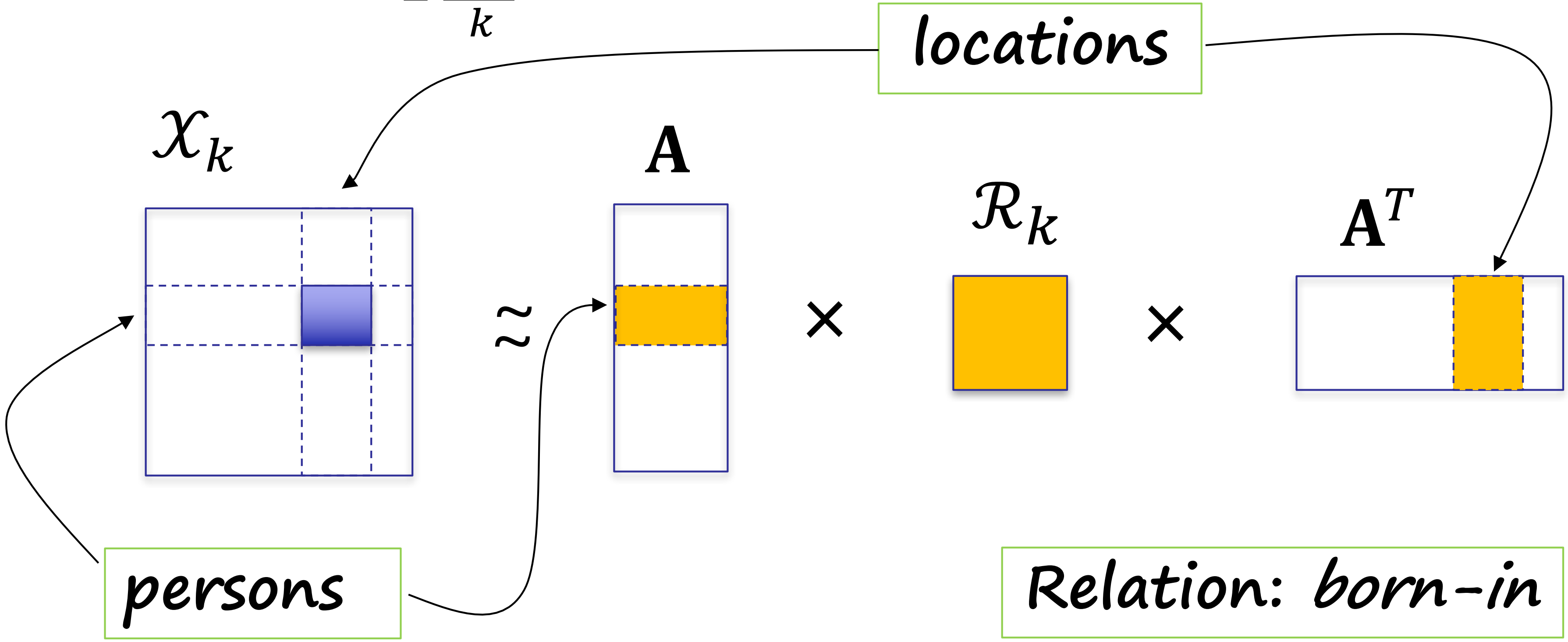
- Objective: $\frac{1}{2}\sum_k \|\mathcal{X}_k - \mathbf{A}\mathcal{R}_k\mathbf{A}^T\|_F^2$



RESCAL [Nickel+, ICML-11]

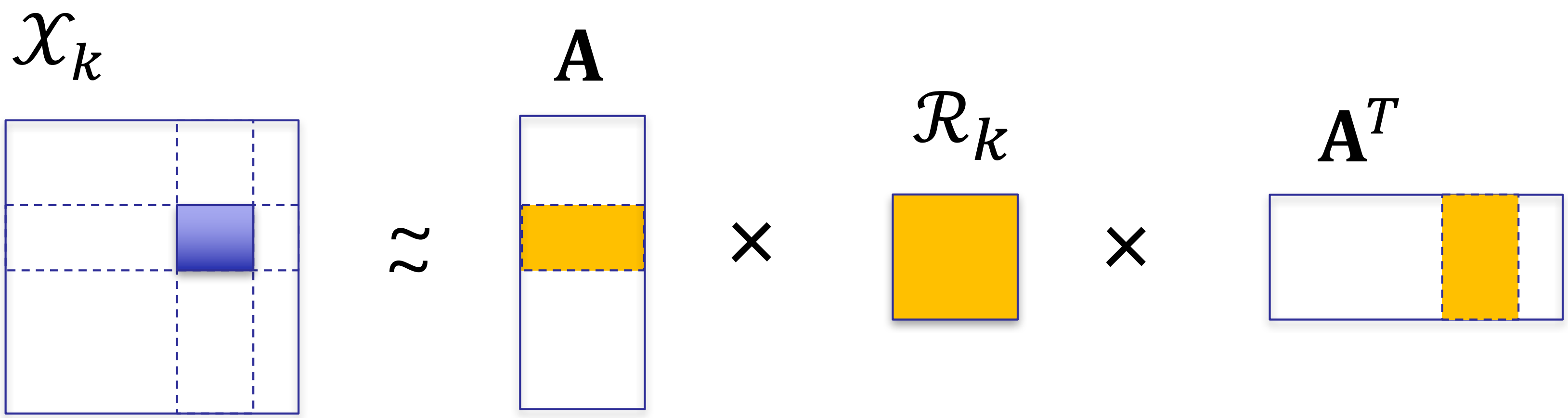
Typed Tensor Decomposition Objective

- Objective: $\frac{1}{2}\sum_k \|\mathcal{X}'_k - \mathbf{A}_l \mathcal{R}_k \mathbf{A}_r^T\|_F^2$



Typed Tensor Decomposition Objective

- Objective: $\frac{1}{2}\sum_{k} \|\mathcal{X}'_k - \mathbf{A}_l \mathcal{R}_k \mathbf{A}_r^T\|_F^2$



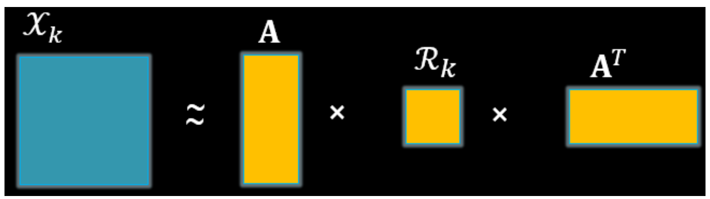
Experiments – KB Completion

- KB – Never Ending Language Learning (NELL)
  - Training: version 165
  - Developing: new facts between v.166 and v.533
  - Testing: new facts between v.534 and v.745

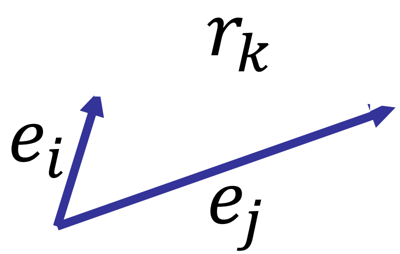
# Entities	753k
# Relation Types	229
# Entity Types	300
# Entity-Relation Triples	1.8M

Tasks & Baselines

- Entity Retrieval: $(e_i, r_k, ?\,)$
  - One positive entity with 100 negative entities
- Relation Retrieval: $(e_i, ?\,, e_j)$
  - Positive entity pairs with equal number of negative pairs
- Baselines:

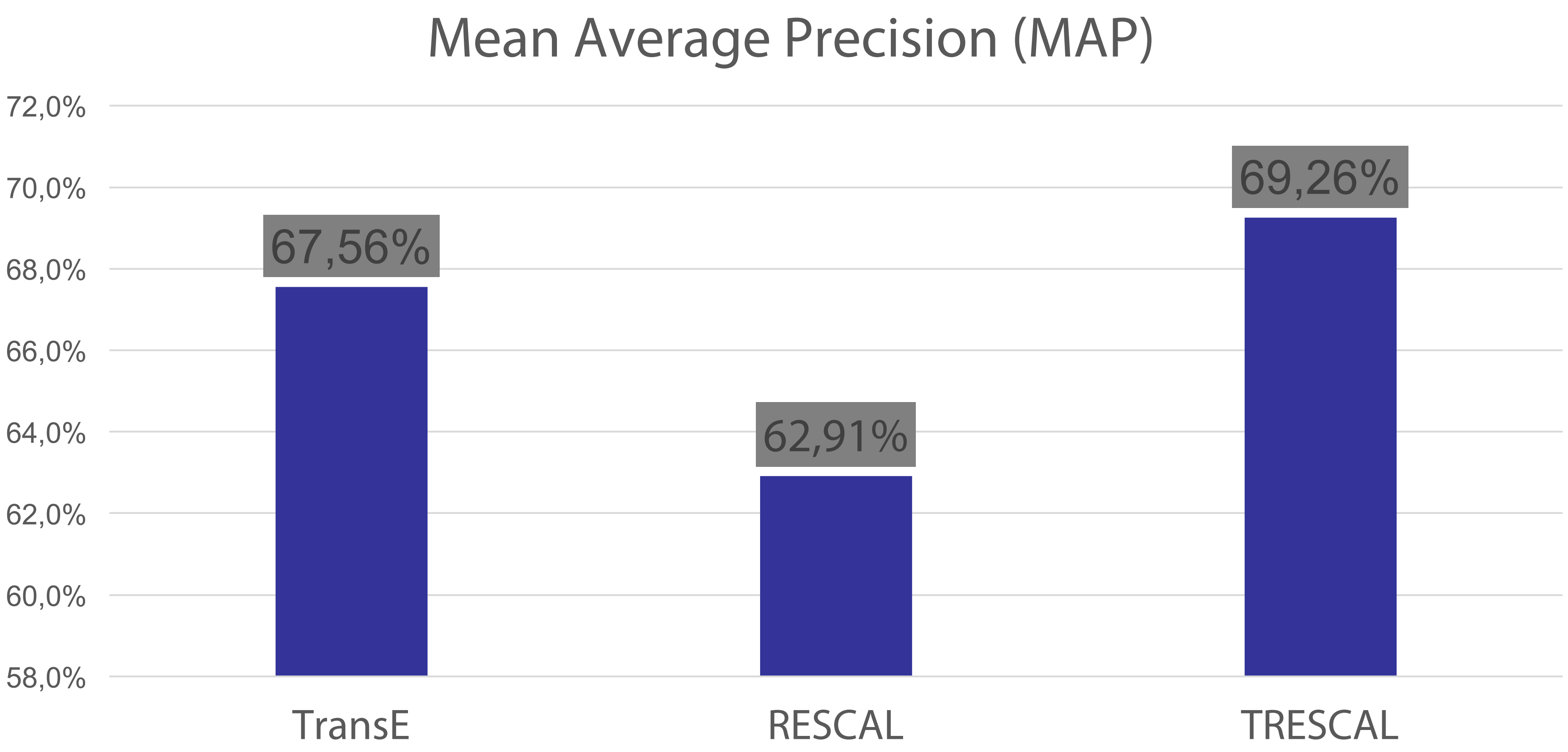


RESCAL

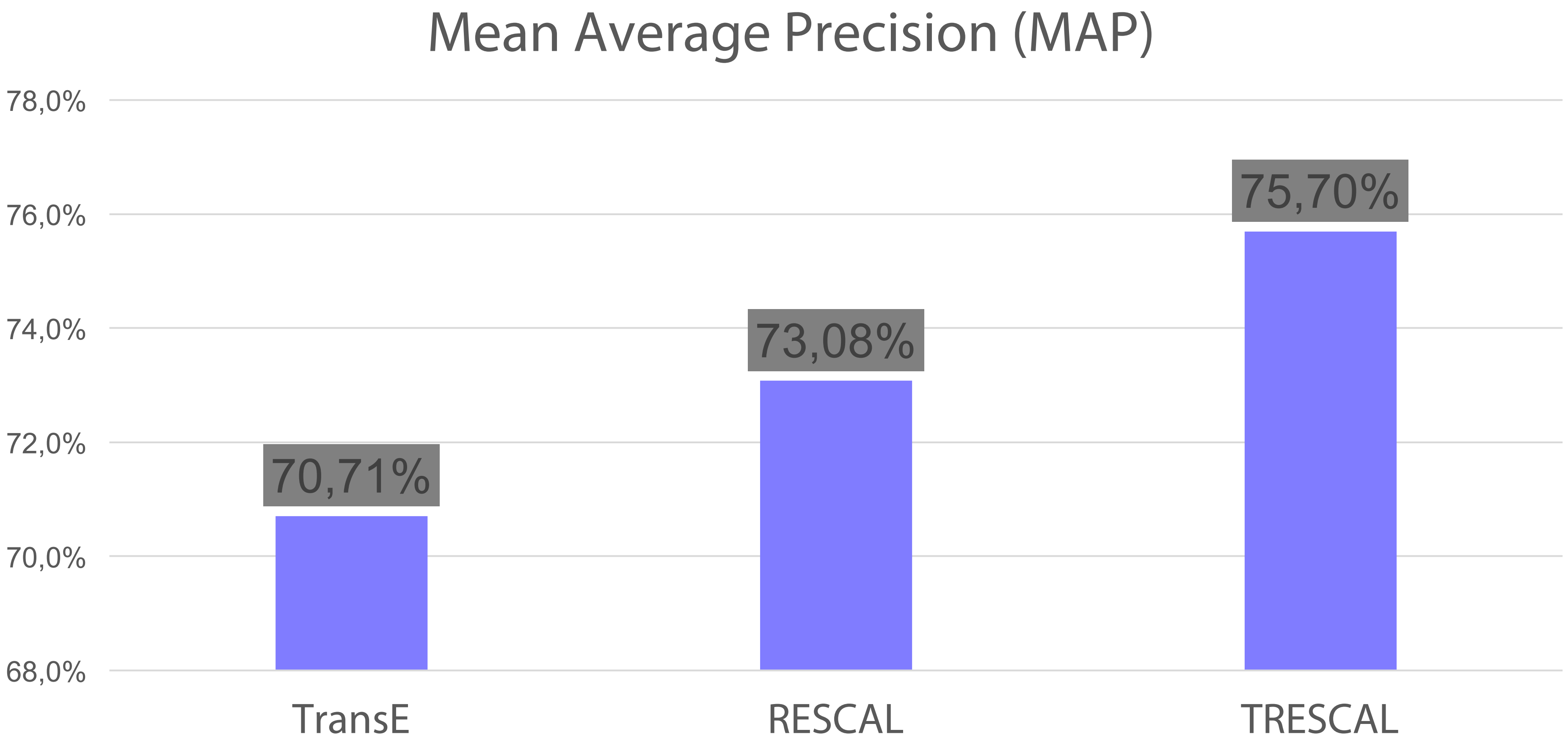


TransE

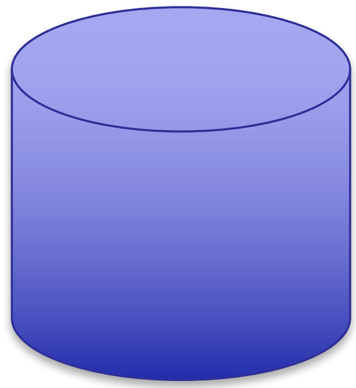
Entity Retrieval



Relation Retrieval



Experiments – Relation Extraction



Dan Roth is a *professor at* UIUC.

(Dan Roth, work-at, UIUC)

Relation Extraction as Matrix Factorization

[Riedel+ 13]

- Row: Entity Pair
- Column: Relation

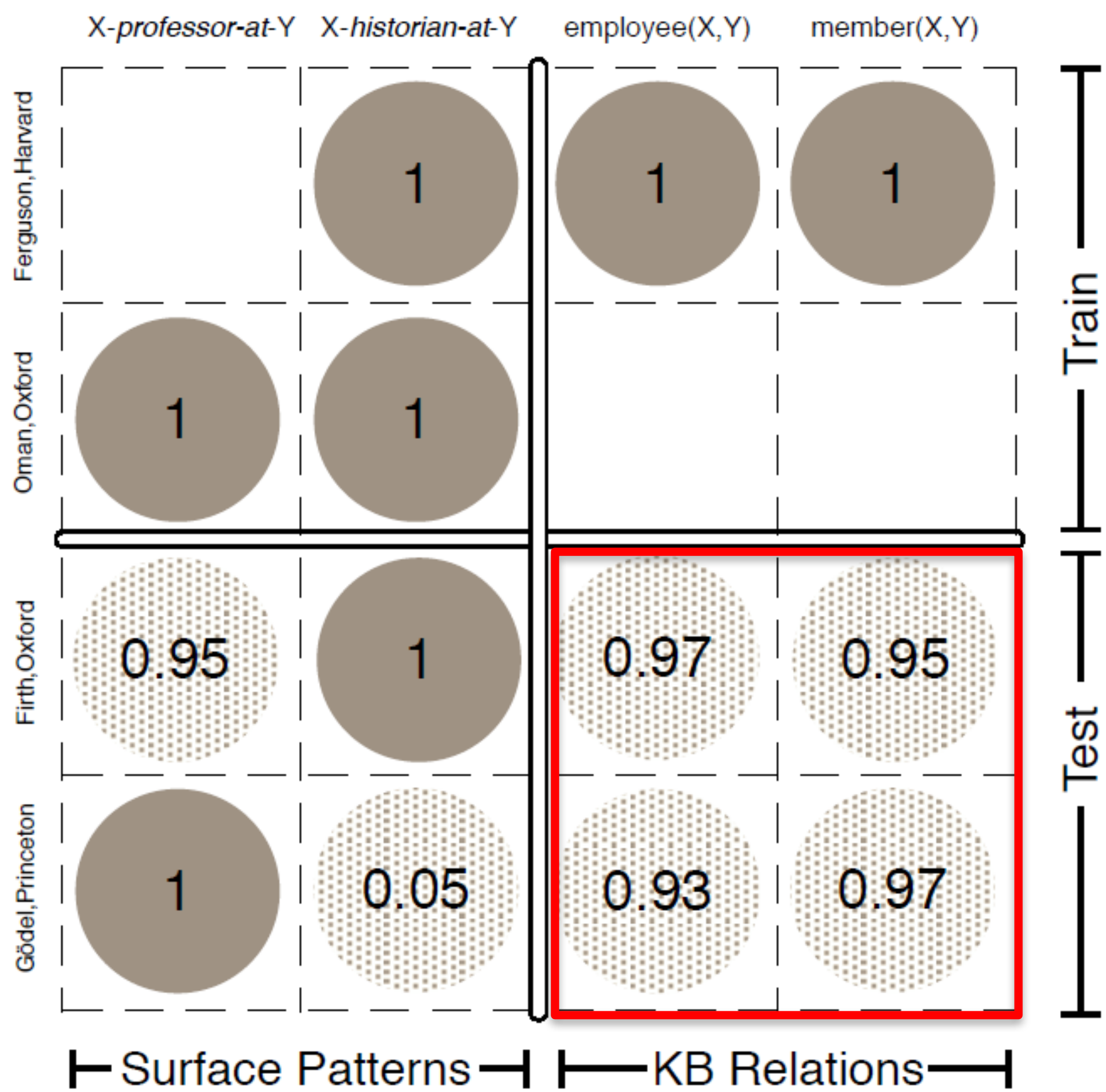


Fig.1 of [Riedel+ 13]

Conclusions

- Continuous semantic representation that
  - Leverages existing rich linguistic knowledge bases
  - Discovers new relations
  - Enables us to measure the degree of multiple relations
- Approaches
  - Better data representation
  - Matrix/Tensor decomposition
  - Relational domain knowledge
- Challenges & Future Work
  - Capture more types of knowledge in the model
  - Support more sophisticated inferential tasks

Acknowledgements again

Slides by: Scott Wen-tau Yih
(orignal presentation shortened)

Describing joint work of Scott Wen-tau Yih with
Kai-Wei Chang, Bishan Yang,
Chris Meek, Geoff Zweig, John Platt

Microsoft Research