Modeling multi-relational data from knowledge bases with embeddings

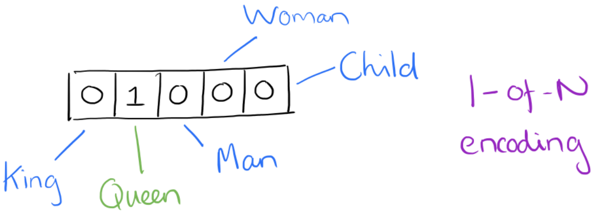
Dat Quoc Nguyen

Department of Computing
Macquarie University
Sydney, Australia

December 2016

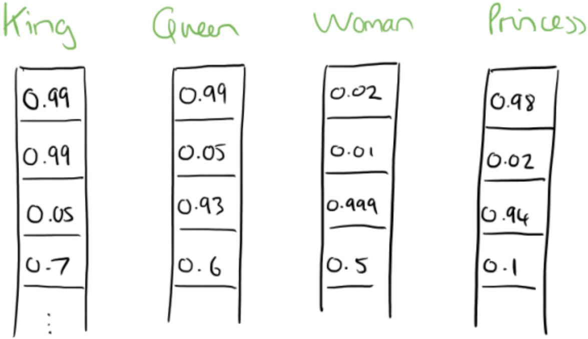
Vector representations/embeddings

- One-hot representation: high-dimensional and sparse vector
  - Vocabulary size: N
  - N-dimensional vector: filled with 0s, except for a 1 at the position associated with word index

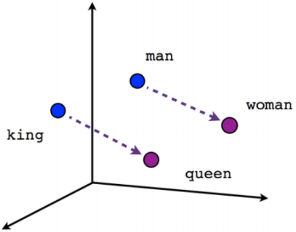


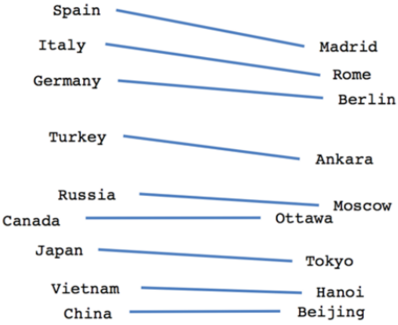
Vector representations/embeddings

- Deep learning evolution: most neural network toolkits do not play well with one-hot representations
  - Dense vector: low-dimensional distributed vector representation
  - Vector size $k \ll N$, for example: $k = 100$, Vocabulary size $N = 100000$



Vector representations/embeddings: "distance" results





$$\boldsymbol{v}_{king} - \boldsymbol{v}_{queen} \approx \boldsymbol{v}_{man} - \boldsymbol{v}_{woman}$$

$$\boldsymbol{v}_{Vietnam} - \boldsymbol{v}_{Hanoi}$$
$$\approx \boldsymbol{v}_{Japan} - \boldsymbol{v}_{Tokyo}$$
$$\approx \boldsymbol{v}_{Germany} - \boldsymbol{v}_{Berlin}$$
$$\approx \boldsymbol{v}_{some\_relationship}, \ saying: \ is\_capital\_of$$

$$\boldsymbol{v}_{Hanoi} + \boldsymbol{v}_{is\_capital\_of} \approx \boldsymbol{v}_{Vietnam}$$
$$\boldsymbol{v}_{Tokyo} + \boldsymbol{v}_{is\_capital\_of} \approx \boldsymbol{v}_{Japan}$$
$$\boldsymbol{v}_{Berlin} + \boldsymbol{v}_{is\_capital\_of} \approx \boldsymbol{v}_{Germany}$$

Triple (*head entity*, *relation*, *tail entity*)

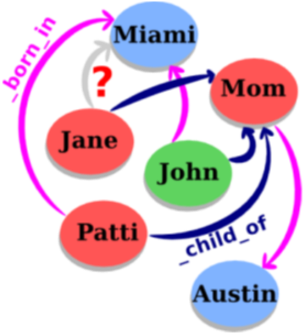
$$\boldsymbol{v}_h + \boldsymbol{v}_r \approx \boldsymbol{v}_t$$

Link prediction in knowledge bases?

Outline

Introduction

- Knowledge bases (KBs) of real-world triple facts (head entity, relation, tail entity) are useful resources for NLP tasks
- **Issue**: large KBs are still far from complete
- So it is useful to perform *link prediction in KBs* or *knowledge base completion* (KBC): predict which triples not in a knowledge base are likely to be true



Our embedding model STransE

- The TransE model (Bordes et al., 2013) represents each relation r by a translation vector $\boldsymbol{v}_r$, which is chosen so that $\boldsymbol{v}_h + \boldsymbol{v}_r \approx \boldsymbol{v}_t$
  - Good for 1-to-1 relationships, e.g: *is_capital_of*
  - Not good for 1-to-Many, Many-to-1 and Many-to-Many, e.g: *gender*
- STransE: a new embedding model for link prediction
  - Our STransE represents each entity as a low dimensional vector, and each relation by two matrices and a translation vector
  - STransE choose matrices $\mathbf{W}_{r,1}$ and $\mathbf{W}_{r,2}$, and vector $\boldsymbol{v}_r$ so that:
$$\mathbf{W}_{r,1}\boldsymbol{v}_h + \boldsymbol{v}_r \approx \mathbf{W}_{r,2}\boldsymbol{v}_t$$

Our embedding model STransE

- For each triple (h, r, t), STransE defines a score function $f(h, r, t)$ of its implausibility:

$$f(h, r, t) = \|\mathbf{W}_{r,1}\boldsymbol{v}_h + \boldsymbol{v}_r - \mathbf{W}_{r,2}\boldsymbol{v}_t\|_{\ell_{1/2}}$$

- To learn the vectors and matrices, we minimize the following margin-based objective function:

$$\mathcal{L} = \sum_{\substack{(h,r,t)\in\mathcal{G} \\ (h',r,t')\in\mathcal{G}'_{(h,r,t)}}} \max\Big(0, \gamma + f(h, r, t) - f(h', r, t')\Big)$$

$$\mathcal{G}'_{(h,r,t)} = \{(h', r, t) \mid h' \in \mathcal{E}, (h', r, t) \notin \mathcal{G}\} \cup \{(h, r, t') \mid t' \in \mathcal{E}, (h, r, t') \notin \mathcal{G}\}$$

  - γ: the margin hyper-parameter

Related work

Model	Score function $f(h,r,t)$
STransE	$\Vert \mathbf{W}_{r,1}\boldsymbol{v}_h + \boldsymbol{v}_r - \mathbf{W}_{r,2}\boldsymbol{v}_t \Vert_{\ell_{1/2}}$; $\mathbf{W}_{r,1}$, $\mathbf{W}_{r,2} \in \mathbb{R}^{k\times k}$; $\boldsymbol{v}_r \in \mathbb{R}^k$
TransE	$\Vert \boldsymbol{v}_h + \boldsymbol{v}_r - \boldsymbol{v}_t \Vert_{\ell_{1/2}}$; $\boldsymbol{v}_r \in \mathbb{R}^k$
TransH	$\Vert (\mathbf{I} - \boldsymbol{r}_p\boldsymbol{r}_p^\top)\boldsymbol{v}_h + \boldsymbol{v}_r - (\mathbf{I} - \boldsymbol{r}_p\boldsymbol{r}_p^\top)\boldsymbol{v}_t \Vert_{\ell_{1/2}}$ $\boldsymbol{r}_p$, $\boldsymbol{v}_r \in \mathbb{R}^k$; $\mathbf{I}$: Identity matrix size $k \times k$
TransD	$\Vert (\mathbf{I} + \boldsymbol{r}_p\boldsymbol{h}_p^\top)\boldsymbol{v}_h + \boldsymbol{v}_r - (\mathbf{I} + \boldsymbol{r}_p\boldsymbol{t}_p^\top)\boldsymbol{v}_t \Vert_{\ell_{1/2}}$ $\boldsymbol{r}_p$, $\boldsymbol{v}_r \in \mathbb{R}^n$; $\boldsymbol{h}_p, \boldsymbol{t}_p \in \mathbb{R}^k$; $\mathbf{I}$: Identity matrix size $n \times k$
TransR	$\Vert \mathbf{W}_r\boldsymbol{v}_h + \boldsymbol{v}_r - \mathbf{W}_r\boldsymbol{v}_t \Vert_{\ell_{1/2}}$; $\mathbf{W}_r \in \mathbb{R}^{n\times k}$; $\boldsymbol{v}_r \in \mathbb{R}^n$
NTN	$\boldsymbol{v}_r^\top tanh(\boldsymbol{v}_h^\top \mathbf{M}_r \boldsymbol{v}_t + \mathbf{W}_{r,1}\boldsymbol{v}_h + \mathbf{W}_{r,2}\boldsymbol{v}_t + \mathbf{b}_r)$ $\boldsymbol{v}_r$, $\mathbf{b}_r \in \mathbb{R}^n$; $\mathbf{M}_r \in \mathbb{R}^{k\times k\times n}$; $\mathbf{W}_{r,1}$, $\mathbf{W}_{r,2} \in \mathbb{R}^{n\times k}$
DISTMULT	$\boldsymbol{v}_h^\top \mathbf{W}_r \boldsymbol{v}_t$; $\mathbf{W}_r$ is a diagonal matrix $\in \mathbb{R}^{k\times k}$
Bilinear-COMP	$\boldsymbol{v}_h^\top \mathbf{W}_{r_1}\mathbf{W}_{r_2}...\mathbf{W}_{r_m}\boldsymbol{v}_t$; $\mathbf{W}_{r_1}, \mathbf{W}_{r_2}, ..., \mathbf{W}_{r_m} \in \mathbb{R}^{k\times k}$
TransE-COMP	$\Vert \boldsymbol{v}_h + \boldsymbol{v}_{r_1} + \boldsymbol{v}_{r_2} + ... + \boldsymbol{v}_{r_m} - \boldsymbol{v}_t \Vert_{\ell_{1/2}}$; $\boldsymbol{v}_{r_1}, \boldsymbol{v}_{r_2}, ..., \boldsymbol{v}_{r_m} \in \mathbb{R}^k$
TransE-NMM	$\Vert \boldsymbol{\vartheta}_{h,r} + \boldsymbol{v}_r - \boldsymbol{\vartheta}_{t,r^{-1}} \Vert_{\ell_{1/2}}$

STransE results for knowledge base completion

- We conducted experiments on two benchmark datasets WN18 and FB15k (Bordes et al., 2013)

Dataset	#E	#R	#Train	#Valid	#Test
WN18	40,943	18	141,442	5,000	5,000
FB15k	14,951	1,345	483,142	50,000	59,071

- **Entity prediction task**:
  - Predict h given $(?, r, t)$ or predict t given $(h, r, ?)$ where ? denotes the missing element
  - Corrupt each correct test triple (h, r, t) by replacing either h or t by each of the possible entities
  - Rank these candidates by their implausibility value
  - Metrics: mean rank and Hits@10

Method	**WN18**		**FB15k**	
	MR	H10	MR	H10
TransE	251	89.2	125	47.1
TransH	303	86.7	87	64.4
TransR	225	92.0	77	68.7
CTransR	218	92.3	75	70.2
KG2E	348	93.2	59	74.0
TransD	212	92.2	91	77.3
TATEC	-	-	**58**	76.7
Our STransE model	**206**	**93.4**	69	**79.7**
RTransE	-	-	**50**	76.2
PTransE	-	-	58	84.6

Applying STransE for search personalization

- Search engines play the most important roles in the Internet era
- Two users search using the same keywords, they are often looking for different information (i.e. difference due to the users' interests)
- Personalized search customizes results based on user's search history (i.e. submitted queries and clicked documents)
- Let (q, u, d) represent a triple $(\text{query}, \text{user}, \text{document})$
- The query q, user u and document d are represented by vector embeddings $\boldsymbol{v}_q$, $\boldsymbol{v}_u$ and $\boldsymbol{v}_d \in \mathbb{R}^k$, respectively

$$f(q, u, d) = \|\mathbf{W}_{u,1}\boldsymbol{v}_q + \boldsymbol{v}_u - \mathbf{W}_{u,2}\boldsymbol{v}_d\|_{\ell_{1/2}}$$

- Represent the profile for the user u by two matrices $\mathbf{W}_{u,1}$ and $\mathbf{W}_{u,2} \in \mathbb{R}^{k \times k}$ and a vector $\boldsymbol{v}_u$, which represents the user's topical interests.

Applying STransE for search personalization

$$f(q, u, d) = \|\mathbf{W}_{u,1}\boldsymbol{v}_q + \boldsymbol{v}_u - \mathbf{W}_{u,2}\boldsymbol{v}_d\|_{\ell_{1/2}}$$

- $\boldsymbol{v}_d$ and $\boldsymbol{v}_q$ are pre-determined by employing the LDA topic model
- To learn the user embeddings and matrices, we minimize the margin-based objective function:

$$\mathcal{L} = \sum_{\substack{(q,u,d) \in \mathcal{G} \\ (q',u,d') \in \mathcal{G}'_{(q,u,d)}}} \max\left(0, \gamma + f(q, u, d) - f(q', u, d')\right)$$

- Re-rank the original list of documents produced by a search engine:
  - Download the top 10 ranked documents given the input query q
  - For each ranked document d and query q, we apply a trained LDA model to infer the topic distribution $\boldsymbol{v}_d$ and topic distribution $\boldsymbol{v}_q$
  - For each triple (q, u, d), we calculate the value $f(q, u, d)$, and then sort the values f to achieve a new ranked list

STransE results for search personalization

- Use a dataset of query logs of of 106 anonymous users in 15 days from 01 July 2012 to 15 July 2012
- Separate the last log entries within search sessions into a test set and a validation set; Use the remaining log entities for training
- After pre-processing, the training set consists of 5,658 correct triples, the test and validation sets contain 1,210 and 1,184 correct triples

#days	#users	#distinct queries	#clicked docs	#sessions	#distinct docs
15	106	6,632	8,052	2,394	33,591

- Results:

Metric	**SE**	**SP**	**STransE**	**TransE**
MRR	0.559	$0.631_{+12.9\%}$	$\mathbf{0.656}_{+17.3\%}$	$0.645_{+15.4\%}$
P@1	0.385	$0.452_{+17.4\%}$	$\mathbf{0.501}_{+30.3\%}$	$0.481_{+24.9\%}$

  - **SE**: The original rank from the search engine; **SP**: The SOTA search personalization method with a learning-to-rank framework
  - The subscripts denote the relative improvement over the baseline SE

Summary

- A new embedding model named STransE for link prediction in KBs (i.e. for KB completion)
- STransE uses a low-dimensional vector and two projection matrices to represent each relation
  - Produce highly competitive results on standard link prediction evaluations
  - Extend STransE to exploit relation path information in knowledge bases
- When applying to a search personalization task, STransE helps to improve the ranking quality significantly

Outline

Introduction

- Embedding models for KBC:
  - Associate entities and/or relations with dense feature vectors or matrices
  - Obtain SOTA performance and generalize to large KBs
- Most embedding models for KBC learn only from triples
- Recent works show that the relation paths between entities in KBs provide useful information and improve KBC

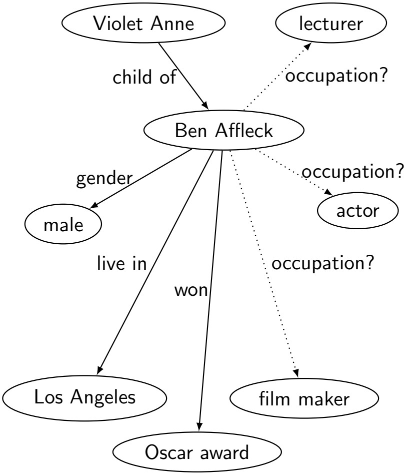
$$
\begin{aligned}
&(\text{Harrison Ford}, \textbf{born\_in\_hospital}/\text{r}_1, \text{Swedish Covenant Hospital})\\
\Rightarrow&(\text{Swedish Covenant Hospital}, \textbf{located\_in\_city}/\text{r}_2, \text{Chicago})\\
\Rightarrow&(\text{Chicago}, \textbf{city\_in\_country}/\text{r}_3, \text{United States})
\end{aligned}
$$

Relation path $\text{p} = \{\text{r}_1, \text{r}_2, \text{r}_3\}$ is useful for predicting the relationship "*nationality*" between the head and tail entities

Introduction

- **Our motivation**: neighborhoods could provide lots of useful information for predicting the relationship between the entities

$$
\begin{aligned}
&\text{Ben\_Affleck} \\
&= \omega_{r,1}(\text{Violet\_Anne}, \text{child\_of}) \\
&\quad + \omega_{r,2}(\text{male}, \text{gender}^{-1}) \\
&\quad + \omega_{r,3}(\text{Los\_Angeles}, \text{live\_in}^{-1}) \\
&\quad + \omega_{r,4}(\text{Oscar\_award}, \text{won}^{-1})
\end{aligned}
$$

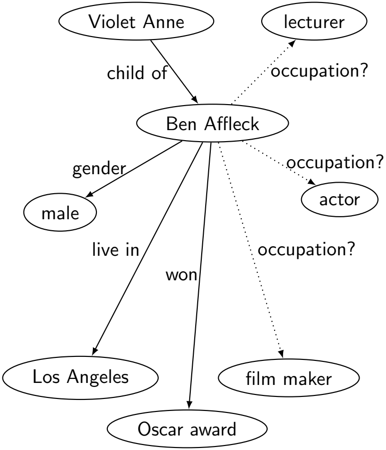


Our neighbor-based entity representation

$$\mathcal{E} = \{\text{Ben\_Affleck}, \text{Los\_Angeles}, \ldots\}$$
$$\mathcal{R} = \{\text{live\_in}, \text{won}, \text{child\_of}, \text{gender}, \ldots\}$$
$$\begin{aligned}\mathcal{G} = \{&(\text{Violet\_Anne}, \text{child\_of}, \text{Ben\_Affleck}),\\ &(\text{Ben\_Affleck}, \text{won}, \text{Oscar\_award}),\\ &(\text{Ben\_Affleck}, \text{live\_in}, \text{Los\_Angeles}), \ldots\}\end{aligned}$$

$\mathcal{N}_e$ is the set of all entity and relation pairs that are neighbors for entity e

$$\begin{aligned}\mathcal{N}_{\text{Ben\_Affleck}} = \{&(\text{Violet\_Anne}, \text{child\_of}),\\ &(\text{male}, \text{gender}^{-1}),\\ &(\text{Los\_Angeles}, \text{live\_in}^{-1}),\\ &(\text{Oscar\_award}, \text{won}^{-1})\}\end{aligned}$$



Our neighbor-based entity representation

- $\boldsymbol{v}_e \in \mathbb{R}^k$: k-dimensional "base" vector associated with entity e
- $\boldsymbol{u}_{e,r} \in \mathbb{R}^k$: relation-specific entity vector, $e \in \mathcal{E},\ r \in \mathcal{R} \cup \mathcal{R}^{-1}$
- The neighborhood-based entity representation $\boldsymbol{\vartheta}_{e,r}$ for an entity e for predicting the relation r is defined as follows:

$$\boldsymbol{\vartheta}_{e,r} = a_e \boldsymbol{v}_e + \sum_{(e',r') \in \mathcal{N}_e} b_{r,r'} \boldsymbol{u}_{e',r'} \tag{1}$$

a_e and $b_{r,r'}$ are the mixture weights that are constrained to sum to 1:

$$a_e \propto \delta + \exp \alpha_e \tag{2}$$

$$b_{r,r'} \propto \exp \beta_{r,r'} \tag{3}$$

$\delta \geqslant 0$: hyper-parameter

α_e, $\beta_{r,r'}$: learnable exponential mixture parameters

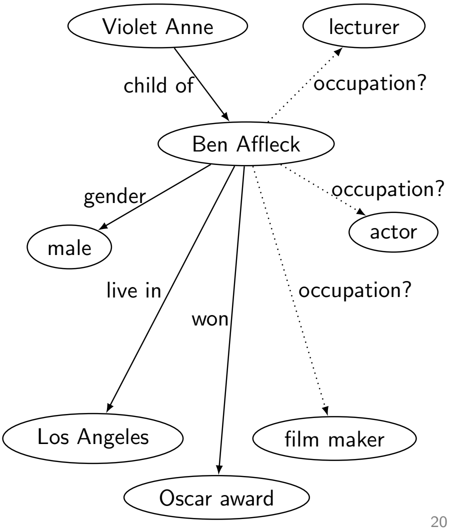
Our neighbor-based entity representation

$$\boldsymbol{\vartheta}_{e,r} = a_e \boldsymbol{v}_e + \sum_{(e',r') \in \mathcal{N}_e} b_{r,r'} \boldsymbol{u}_{e',r'}$$

$$e = \text{Ben\_Affleck}$$

$$r = \text{occupation}$$

$$\mathcal{N}_e = \{(\text{Violet\_Anne}, \text{child\_of}), (\text{male}, \text{gender}^{-1}), (\text{Los\_Angeles}, \text{live\_in}^{-1}), (\text{Oscar\_award}, \text{won}^{-1})\}$$



Our new embedding model TransE-NMM for KBC

- Embedding models define for each triple $(h, r, t) \in \mathcal{G}$, a *score function* $f(h, r, t)$ that measures its implausibility
- **Goal**: choose f such that the score $f(h, r, t)$ of a plausible triple (h, r, t) is smaller than the score $f(h', r', t')$ of an implausible triple (h', r', t').
- Entity e and relation r are represented with vectors $\boldsymbol{v}_e \in \mathbb{R}^k$ and $\boldsymbol{v}_r \in \mathbb{R}^k$

$$f(h, r, t)_{\text{TransE}} = \|\boldsymbol{v}_h + \boldsymbol{v}_r - \boldsymbol{v}_t\|_{\ell_{1/2}}$$

- The score function of our new model TransE-NMM is defined as follows:

$$f(h, r, t) = \|\boldsymbol{\vartheta}_{h,r} + \boldsymbol{v}_r - \boldsymbol{\vartheta}_{t,r^{-1}}\|_{\ell_{1/2}} \tag{4}$$

$$\boldsymbol{\vartheta}_{e,r} = a_e \boldsymbol{v}_e + \sum_{(e',r') \in \mathcal{N}_e} b_{r,r'} \boldsymbol{u}_{e',r'}$$

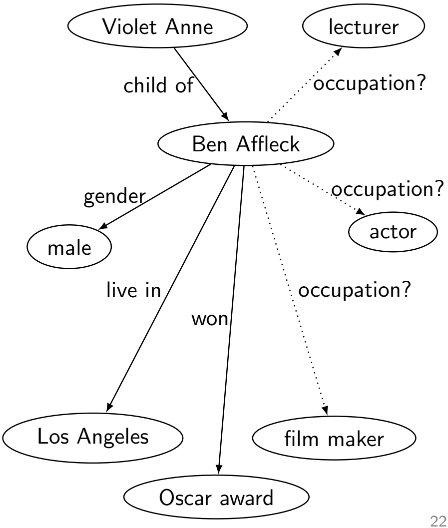
$$\boldsymbol{u}_{e,r} = \boldsymbol{v}_e + \boldsymbol{v}_r \tag{5}$$

$$\boldsymbol{v}_{r^{-1}} = -\boldsymbol{v}_r \tag{6}$$

Our new embedding model TransE-NMM for KBC

$$\boldsymbol{\vartheta}_{e,r} = a_e \boldsymbol{v}_e + \sum_{(e',r') \in \mathcal{N}_e} b_{r,r'} \left(\boldsymbol{v}_{e'} + \boldsymbol{v}_{r'} \right)$$

$$\begin{aligned} e &= \text{Ben\_Affleck} \\ r &= \text{occupation} \\ \mathcal{N}_e &= \{(\text{Violet\_Anne}, \text{child\_of}), \\ &\quad (\text{male}, \text{gender}^{-1}), \\ &\quad (\text{Los\_Angeles}, \text{live\_in}^{-1}), \\ &\quad (\text{Oscar\_award}, \text{won}^{-1})\} \end{aligned}$$



Parameter optimization

- Model parameters:
  - Entity vectors $\boldsymbol{v}_e$
  - Relation type vectors $\boldsymbol{v}_r$
  - $\boldsymbol{\alpha} = \{\alpha_e | e \in \mathcal{E}\}$: entity-specific weights
  - $\boldsymbol{\beta} = \{\beta_{r,r'} | r, r' \in \mathcal{R} \cup \mathcal{R}^{-1}\}$: relation-specific weights
- Minimize the L_2-regularized margin-based objective function:

$$\mathcal{L} = \sum_{\substack{(h,r,t) \in \mathcal{G} \\ (h',r,t') \in \mathcal{G}'_{(h,r,t)}}} [\gamma + f(h,r,t) - f(h',r,t')]_+ + \frac{\lambda}{2}\left(\|\boldsymbol{\alpha}\|_2^2 + \|\boldsymbol{\beta}\|_2^2\right)$$

$$\begin{aligned}\mathcal{G}'_{(h,r,t)} = &\{(h',r,t) \mid h' \in \mathcal{E}, (h',r,t) \notin \mathcal{G}\} \\ &\cup \{(h,r,t') \mid t' \in \mathcal{E}, (h,r,t') \notin \mathcal{G}\}\end{aligned}$$

  - $[x]_+ = \max(0, x)$
  - γ: the margin hyper-parameter
  - λ: the L_2 regularization parameter

Related work

Model	Score function $f(h, r, t)$
STransE	$\|\mathbf{W}_{r,1}\boldsymbol{v}_h + \boldsymbol{v}_r - \mathbf{W}_{r,2}\boldsymbol{v}_t\|_{\ell_{1/2}}$; $\mathbf{W}_{r,1}$, $\mathbf{W}_{r,2} \in \mathbb{R}^{k\times k}$; $\boldsymbol{v}_r \in \mathbb{R}^k$
TransE	$\|\boldsymbol{v}_h + \boldsymbol{v}_r - \boldsymbol{v}_t\|_{\ell_{1/2}}$; $\boldsymbol{v}_r \in \mathbb{R}^k$
TransH	$\|(\mathbf{I} - \boldsymbol{r}_p\boldsymbol{r}_p^\top)\boldsymbol{v}_h + \boldsymbol{v}_r - (\mathbf{I} - \boldsymbol{r}_p\boldsymbol{r}_p^\top)\boldsymbol{v}_t\|_{\ell_{1/2}}$ $\boldsymbol{r}_p$, $\boldsymbol{v}_r \in \mathbb{R}^k$; $\mathbf{I}$: Identity matrix size $k \times k$
TransD	$\|(\mathbf{I} + \boldsymbol{r}_p\boldsymbol{h}_p^\top)\boldsymbol{v}_h + \boldsymbol{v}_r - (\mathbf{I} + \boldsymbol{r}_p\boldsymbol{t}_p^\top)\boldsymbol{v}_t\|_{\ell_{1/2}}$ $\boldsymbol{r}_p$, $\boldsymbol{v}_r \in \mathbb{R}^n$; $\boldsymbol{h}_p, \boldsymbol{t}_p \in \mathbb{R}^k$; $\mathbf{I}$: Identity matrix size $n \times k$
TransR	$\|\mathbf{W}_r\boldsymbol{v}_h + \boldsymbol{v}_r - \mathbf{W}_r\boldsymbol{v}_t\|_{\ell_{1/2}}$; $\mathbf{W}_r \in \mathbb{R}^{n\times k}$; $\boldsymbol{v}_r \in \mathbb{R}^n$
NTN	$\boldsymbol{v}_r^\top tanh(\boldsymbol{v}_h^\top\mathbf{M}_r\boldsymbol{v}_t + \mathbf{W}_{r,1}\boldsymbol{v}_h + \mathbf{W}_{r,2}\boldsymbol{v}_t + \mathbf{b}_r)$ $\boldsymbol{v}_r$, $\mathbf{b}_r \in \mathbb{R}^n$; $\mathbf{M}_r \in \mathbb{R}^{k\times k\times n}$; $\mathbf{W}_{r,1}$, $\mathbf{W}_{r,2} \in \mathbb{R}^{n\times k}$
DISTMULT	$\boldsymbol{v}_h^\top\mathbf{W}_r\boldsymbol{v}_t$; $\mathbf{W}_r$ is a diagonal matrix $\in \mathbb{R}^{k\times k}$
Bilinear-COMP	$\boldsymbol{v}_h^\top\mathbf{W}_{r_1}\mathbf{W}_{r_2}...\mathbf{W}_{r_m}\boldsymbol{v}_t$; $\mathbf{W}_{r_1}, \mathbf{W}_{r_2}, ..., \mathbf{W}_{r_m} \in \mathbb{R}^{k\times k}$
TransE-COMP	$\|\boldsymbol{v}_h + \boldsymbol{v}_{r_1} + \boldsymbol{v}_{r_2} + ... + \boldsymbol{v}_{r_m} - \boldsymbol{v}_t\|_{\ell_{1/2}}$; $\boldsymbol{v}_{r_1}, \boldsymbol{v}_{r_2}, ..., \boldsymbol{v}_{r_m} \in \mathbb{R}^k$
TransE-NMM	$\|\boldsymbol{\vartheta}_{h,r} + \boldsymbol{v}_r - \boldsymbol{\vartheta}_{t,r^{-1}}\|_{\ell_{1/2}}$

Evaluation: experimental setup

Dataset:	WN11	FB13	NELL186
#R	11	13	186
#E	38,696	75,043	14,463
#Train	112,581	316,232	31,134
#Valid	2,609	5,908	5,000
#Test	10,544	23,733	5,000

- #E: number of entities
- #R: number of relation types
- #Train, #Valid and #Test are the numbers of correct triples in the training, validation, and test sets, respectively
- Each validation and test set also contains the same number of incorrect triples as the number of correct triples

- **Triple classification task**:
  - Predict whether a triple (h, r, t) is correct or not
  - Set a relation-specific threshold θ_r for each relation type r
  - For an unseen test triple (h, r, t), if $f(h, r, t)$ is smaller than θ_r then the triple will be classified as correct, otherwise incorrect
  - Relation-specific thresholds are determined by maximizing the micro-averaged accuracy on the validation set

Evaluation: experimental setup

- **Entity prediction task**:
  - Predict h given $(?, r, t)$ or predict t given $(h, r, ?)$ where ? denotes the missing element
  - Corrupt each correct test triple (h, r, t) by replacing either h or t by each of the possible entities in turn
  - Rank these candidates in ascending order of their implausibility value compute by the score function
  - "Raw" and "Filtered" setting protocols in which "Filtered" setting is to filter out before ranking any corrupted triples that appear in the KB
  - Metrics: mean rank (MR), mean reciprocal rank (MRR) and Hits@10 (H10)
- **Relation prediction task**:
  - Predict r given $(h, ?, t)$ where ? denotes the missing element
  - Corrupt each correct test triple (h, r, t) by replacing r by each of the possible relations in turn

Evaluation: quantitative results

Data		Method	Triple class. Mic.	Triple class. Mac.	Entity prediction MR	Entity prediction MRR	Entity prediction H@10	Relation prediction MR	Relation prediction MRR	Relation prediction H@10
WN11	R	TransE	85.21	82.53	4324	0.102	19.21	2.37	0.679	**99.93**
		TransE-NMM	**86.82**	**84.37**	**3466**	**0.123**	**20.59**	**2.14**	**0.687**	99.92
	F	TransE			4304	0.122	21.86	2.37	0.679	**99.93**
		TransE-NMM			**3447**	**0.137**	**23.03**	**2.14**	**0.687**	99.92
FB13	R	TransE	87.57	86.66	9037	0.204	35.39	1.01	0.996	99.99
		TransE-NMM	**88.58**	**87.99**	**8289**	**0.258**	**35.53**	1.01	0.996	**100.0**
	F	TransE			5600	0.213	36.28	1.01	0.996	99.99
		TransE-NMM			**5018**	**0.267**	**36.36**	1.01	0.996	**100.0**
NELL186	R	TransE	92.13	88.96	309	0.192	36.55	8.43	0.580	77.18
		TransE-NMM	**94.57**	**90.95**	**238**	**0.221**	**37.55**	**6.15**	**0.677**	**82.16**
	F	TransE			279	0.268	47.13	8.32	0.602	77.26
		TransE-NMM			**214**	**0.292**	**47.82**	**6.08**	**0.690**	**82.20**

- **Mic.**: Micro-averaged accuracy; **Mac.**: Macro-averaged accuracy
- "R" and "F" denote the "Raw" and "Filtered" settings used in the entity prediction and relation prediction tasks, respectively
- Better results are in **bold**

Evaluation: quantitative results

Method	**W11**	**F13**
TransR	85.9	82.5
CTransR	85.7	-
TransD	<u>86.4</u>	**89.1**
TranSparse-S	<u>86.4</u>	88.2
TranSparse-US	**86.8**	87.5
NTN	70.6	87.2
TransH	78.8	83.3
SLogAn	75.3	85.3
KG2E	85.4	85.3
Bilinear-COMP	77.6	86.1
TransE-COMP	80.3	87.6
TransE	85.2	87.6
TransE-NMM	**86.8**	<u>88.6</u>

Micro-averaged accuracy for triple classification on WN11 and FB13

Results on the NELL186 test set:

Method	**Triple class.**		**Entity pred.**	
	Mic.	Mac.	MR	H@10
TransE-LLE	90.08	84.50	535	20.02
SME-LLE	93.64	89.39	<u>253</u>	37.14
SE-LLE	<u>93.95</u>	88.54	447	31.55
TransE-SkipG	85.33	80.06	385	30.52
SME-SkipG	92.86	<u>89.65</u>	293	**39.70**
SE-SkipG	93.07	87.98	412	31.12
TransE	92.13	88.96	309	36.55
TransE-NMM	**94.57**	**90.95**	**238**	<u>37.55</u>

The entity prediction results are in the "Raw" setting

Evaluation: qualitative results

- Take the relation-specific mixture weights from the learned TransE-NMM
- Extract neighbor relations with the largest mixture weights given a relation

Relation	**Top 3-neighbor relations**
has_instance (WN11)	type_of subordinate_instance_of domain_topic
nationality (FB13)	place_of_birth place_of_death location
CEOof (NELL186)	WorksFor TopMemberOfOrganization PersonLeadsOrganization

Summary

- We introduced a neighborhood mixture model for knowledge base completio by constructing neighbor-based vector representations for entities
- We demonstrated its effect by extending the state-of-the-art embedding model TransE with our neighborhood mixture model
- Our model significantly improves TransE and obtains better results than the other state-of-the-art embedding models on three evaluation tasks
- We plan to apply the neighborhood mixture model to the relation path models to combine the useful information from both relation paths and entity neighborhoods

Thank you for your attention!

- Dat Quoc Nguyen, Kairit Sirts, Lizhen Qu and Mark Johnson. STransE: a novel embedding model of entities and relationships in knowledge bases. In *Proceedings of NAACL-HLT*, 2016.
- Dat Quoc Nguyen, Kairit Sirts, Lizhen Qu and Mark Johnson. Neighborhood Mixture Model for Knowledge Base Completion. In *Proceedings of CoNLL*, 2016.
- Thanh Vu*, Dat Quoc Nguyen*, Mark Johnson, Dawei Song and Alistair Willis. Search Personalization with Embeddings. In *Proceedings of ECIR 2017*, to appear.