Relational learning with many relations

Guillaume Obozinski

Laboratoire d’Informatique Gaspard Monge
École des Ponts - ParisTech

Joint work with Rodolphe Jenatton, Nicolas Le Roux and Antoine Bordes.

Labex Bézout - Huawei Seminar - April 3rd, 2015
Modelling relations between pairs of entities

Triplets:

Term 1 - Relation - Term 2
Modelling relations between pairs of entities

Triplets:

Term 1 - Relation - Term 2

Single relation

- Collaborative filtering
- Link prediction
- Modeling of social networks
Modelling relations between pairs of entities

Triplets:

Term 1 - Relation - Term 2

Single relation

- Collaborative filtering
- Link prediction
- Modeling of social networks

Multiple relations

- Collective classification
- Modelling in relational knowledge databases
- Proteins-protein and protein-ligand interactions
- Natural language semantics (and semantic role labelling)
Our motivation: Learning the semantic value of verbs

Model triplets:

Subject \( S_i \)  Verb \( R_j \)  Object \( O_k \)
Our motivation: Learning the semantic value of verbs

Model triplets:

\[
\begin{array}{ccc}
\text{Subject} & \text{Verb} & \text{Object} \\
S_i & R_j & O_k
\end{array}
\]

View this as the relation:

\[
R_j(S_i, O_k) = 1
\]
Different kinds of relational learning

Learn to predict relations from object attributes:
- Binary classification from pairs of feature vectors

Markov Logic Networks (Kok and Domingos, 2007)

Idea: relations derive from unobserved latent attributes.

Relational learning from intrinsic latent attributes

Relational learning with many relations
Different kinds of relational learning

Learn to predict relations from object attributes:
- Binary classification from pairs of feature vectors

Exploit logical properties of relations: transitivity, implication, mutual exclusion, etc
- Markov Logic Networks (Kok and Domingos, 2007)
Different kinds of relational learning

Learn to predict relations from object attributes:
- Binary classification from pairs of feature vectors

Exploit logical properties of relations: transitivity, implication, mutual exclusion, etc
- Markov Logic Networks (Kok and Domingos, 2007)

Predict relations from some observed relations
Different kinds of relational learning

Learn to predict relations from object attributes:
- Binary classification from pairs of feature vectors

Exploit logical properties of relations: transitivity, implication, mutual exclusion, etc
- Markov Logic Networks (Kok and Domingos, 2007)

Predict relations from some observed relations
- Idea: relations derive from unobserved latent attributes.
- Relational learning from *intrinsic latent attributes*
Stochastic Block Model

Wang and Wong (1987); Nowicki and Snijders (2001)
Stochastic Block Model

Wang and Wong (1987); Nowicki and Snijders (2001)

\[
P(Z_{ik} = 1) = \sum_{c, c'} P(Z_{ik} = 1 \mid C_i = c, C_k = c') P(C_i = c) P(C_k = c')
\]
Stochastic Block Model

Wang and Wong (1987); Nowicki and Snijders (2001)

\[
P(Z_{ik} = 1) = \sum_{c,c'} P(Z_{ik} = 1 \mid C_i = c, C_k = c') P(C_i = c) P(C_k = c')
\]

\[
P_{ik} = \sum_{c,c'} R_{cc'} S_{ci} O_{c'k} = (s^i)^\top R o^k
\]
Stochastic Block Model
Wang and Wong (1987); Nowicki and Snijders (2001)

\[ P(Z_{ik} = 1) = \sum_{c, c'} P(Z_{ik} = 1 \mid C_i = c, C_k = c') P(C_i = c) P(C_k = c') \]

\[ P_{ik} = \sum_{c,c'} R_{cc'} S_{ci} O_{c'k} = (s^i)\top R o^k \]

\[ P = S\top R O \]
A matrix factorization problem

\[ P = S^T R O \]

- \( 0 \leq R_{ik} \leq 1 \)
- \( o^k \in \triangle, \quad s^i \in \triangle \) with \( \triangle = \{ x \in \mathbb{R}^p_+ \mid \|x\|_1 = 1 \} \)
Stochastic Block Model for several relation types

Relational learning with many relations
Stochastic Block Model for several relation types

\[
\mathbb{P}(Z_{ik}^{(j)} = 1) = \sum_{c, c'} \mathbb{P}(Z_{ik}^{(j)} = 1 \mid C_i = c, C_k' = c') \mathbb{P}(C_i = c) \mathbb{P}(C_k' = c')
\]
Stochastic Block Model for several relation types

\[ P(Z_{ik}^{(j)} = 1) = \sum_{c,c'} P(Z_{ik}^{(j)} = 1 \mid C_i = c, C_k' = c') P(C_i = c) P(C_k' = c') \]

\[ P_{ik}^{(j)} = \sum_{c,c'} [R_j]_{cc'} S_{ci} O_{c'k} = (s_i^i)^{\top} R_j o^k \]
Stochastic Block Model for several relation types

\[ P(Z_{ik}^{(j)} = 1) = \sum_{c,c'} P(Z_{ik}^{(j)} = 1 \mid C_i = c, C'_k = c') P(C_i = c) P(C'_k = c') \]

\[ P_{ik}^{(j)} = \sum_{c,c'} [R_j]_{cc'} S_{ci} O_{c'k} = (s^i)^\top R_j o^k \]

\[ P_j = S^\top R_j O. \]
Collective matrix factorization

\[ P_j = S^\top R_j O \]

- \( 0 \leq [R_j]_{ik} \leq 1 \)
- \( o^k \in \triangle, s^i \in \triangle \) with \( \triangle = \{ x \in \mathbb{R}^p_+ | \|x\|_1 = 1 \} \)

Corresponds to the approach used in RESCAL (Nickel et al., 2012)
Collective matrix factorization

\[ P_j = S^T R_j O \]

- \( 0 \leq [R_j]_{ik} \leq 1 \)
- \( o^k \in \triangle, \ s^i \in \triangle \) with \( \triangle = \{ x \in \mathbb{R}_+^p | \| x \|_1 = 1 \} \)

Corresponds to the approach used in RESCAL (Nickel et al., 2012)

\[ \min_{S=O,R_j} \| Z_j - P_j \|_F^2 \]
A bilinear logistic model

$$Z_{ijk} = R_j(S_i, O_k)$$
A bilinear logistic model

\[ Z_{ijk} = R_j(S_i, O_k) \]

\[ P(R_j(S_i, O_k) = 1) = P_{ik} = (1 + \exp -\eta_{ik})^{-1} \]

with an “energy”

\[ \mathcal{E}(s^i, R_j, o^k) = \eta_{ik} = \langle s^i, R_j o^k \rangle \]
A bilinear logistic model

\[ Z_{ijk} = R_j(S_i, O_k) \]

\[ P(R_j(S_i, O_k) = 1) = P_{ik}^{(j)} = (1 + \exp(-\eta_{ik}^{(j)}))^{-1} \]

with an “energy”

\[ E(s^i, R_j, o^k) = \eta_{ik}^{(j)} = \langle s^i, R_j o^k \rangle \]

So that with

\[ H^{(j)} = (\eta_{ik}^{(j)})_{1 \leq i, k \leq n} \]

we have

\[ H^{(j)} = S^\top R_j O \]
Dealing with the number of parameters? related work

Clustering of Entities and Relations
Miller et al. (2009); Zhu (2012)
Bayesian Non-parametric clustering: Kemp et al. (2006); Sutskever et al. (2009)
Clustering in the context of Markov Logic Network: Kok and Domingos (2007)

Embeddings
Collective Matrix Factorization by (Nickel et al., 2012) (rescal)
Semantic Matching Energy (sme) model of Bordes et al. (2012): encodes relations as vectors for scalability.

Tensor factorization
CANDECOMP/PARAFAC Tucker (1966); Harshman and Lundy (1994)
Probabilistic formulation of Chu and Ghahramani (2009)
Dealing with the number of parameters? : related work

Clustering of Entities and Relations

- Miller et al. (2009); Zhu (2012)
- Bayesian Non-parametric clustering: Kemp et al. (2006); Sutskever et al. (2009)
- Clustering in the context of Markov Logic Network: Kok and Domingos (2007)
Dealing with the number of parameters? : related work

Clustering of Entities and Relations

- Miller et al. (2009); Zhu (2012)
- Bayesian Non-parametric clustering: Kemp et al. (2006); Sutskever et al. (2009)
- Clustering in the context of Markov Logic Network: Kok and Domingos (2007)

Embeddings

- Collective Matrix Factorization by (Nickel et al., 2012) (RESCAL)
- Semantic Matching Energy (SME) model of Bordes et al. (2012): encodes relations as vectors for scalability.
Dealing with the number of parameters? : related work

Clustering of Entities and Relations

- Miller et al. (2009); Zhu (2012)
- Bayesian Non-parametric clustering: Kemp et al. (2006); Sutskever et al. (2009)
- Clustering in the context of Markov Logic Network: Kok and Domingos (2007)

Embeddings

- Collective Matrix Factorization by (Nickel et al., 2012) (**RESCAL**)
- Semantic Matching Energy (**SME**) model of Bordes et al. (2012): encodes relations as vectors for scalability.

Tensor factorization

- CANDECOMP/PARAFAC Tucker (1966); Harshman and Lundy (1994)
- Probabilistic formulation of Chu and Ghahramani (2009)
Our solution: *Latent relational factors*

**Idea:** Modelling the relations between the relations...
**Our solution: Latent relational factors**

**Idea:** Modelling the relations between the relations...

\[
R_j = \sum_{r=1}^{d} \alpha^j_r \Theta_r, \quad \text{with} \quad \Theta_r = u_r v_r^\top
\]

for some sparse vector \( \alpha^j \in \mathbb{R}^d \).
Our solution: *Latent relational factors*

**Idea:** Modelling the relations between the relations...

\[
R_j = \sum_{r=1}^{d} \alpha_r^j \Theta_r, \quad \text{with} \quad \Theta_r = u_r v_r^T
\]

for some sparse vector \( \alpha^j \in \mathbb{R}^d \).

Given

- \( n_r \) number of relations
- \( p \) embedding dimension: \( R_j \in \mathbb{R}^{p \times p} \)
- \( d \) number of latent relational factors
- \( \bar{s} \leq \lambda d \) average number of non-zero \( \alpha \) coefficients
Our solution: *Latent relational factors*

**Idea:** Modelling the relations between the relations...

\[ R_j = \sum_{r=1}^{d} \alpha_r^j \Theta_r, \quad \text{with} \quad \Theta_r = u_r v_r^\top \]

for some sparse vector \( \alpha^j \in \mathbb{R}^d \).

Given

- \( n_r \) number of relations
- \( p \) embedding dimension: \( R_j \in \mathbb{R}^{p \times p} \)
- \( d \) number of latent relational factors
- \( \bar{s} \leq \lambda d \) average number of non-zero \( \alpha \) coefficients

\( \Rightarrow \) we reduce the \# of parameters from \( n_r p^2 \) to \( 2pd + \bar{sn}_r \)
Algorithmic approach

- Large scale $|\mathcal{P}| = 10^6$
Algorithmic approach

- Large scale $|\mathcal{P}| = 10^6$
- Stochastic projected block-coordinate gradient descent algorithm
Algorithmic approach

- Large scale $|\mathcal{P}| = 10^6$
- Stochastic projected block-coordinate gradient descent algorithm
- Mini-batches of 100 triplets
Algorithmic approach

- Large scale $|\mathcal{P}| = 10^6$
- Stochastic projected block-coordinate gradient descent algorithm
- Mini-batches of 100 triplets
- For each positive triplet $(i, j, k)$, sampling negative triplets $(i, j', k)$. 
Tensor factorization interpretation of our model

\[ \eta_{ik}^{(j)} = \langle s^i, R_j o^k \rangle = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]

\[ = \]
Tensor factorization interpretation of our model

\[ \eta_{ik}^{(j)} = \langle s^i, R_j o^k \rangle = (s^i)^\top \left[ \sum_{r=1}^{d} \alpha_r^j u_r v_r^\top \right] o^k \]
Tensor factorization interpretation of our model

\[ \eta^{(j)}_{ik} = \langle s^i, R_j o^k \rangle = (s^i)^\top \left[ \sum_{r=1}^{d} \alpha^j_r u_r v_r^\top \right] o^k \]

\[ = \sum_{r=1}^{d} \alpha^j_r ((s^i)^\top u_r)(v_r^\top o^k) \]
Tensor factorization interpretation of our model

\[ \eta^{(j)}_{ik} = \langle s^i, R_j o^k \rangle = (s^i)^\top \left[ \sum_{r=1}^{d} \alpha^j_r u_r v_r^\top \right] o^k \]

\[ = \sum_{r=1}^{d} \alpha^j_r (s^i)^\top u_r (v_r^\top o^k) \]

\[ = \sum_{r=1}^{d} \alpha^j_r \beta^i_r \gamma^k_r \quad \text{with} \quad \beta_r = S^\top u_r, \quad \gamma_r = O^\top v_r \]
Tensor factorization interpretation of our model

\[ \eta_{ik}^{(j)} = \langle s^i, R_j o^k \rangle = (s^i)^\top \left[ \sum_{r=1}^{d} \alpha_r^j u_r v_r^\top \right] o^k \]

\[ = \sum_{r=1}^{d} \alpha_r^j ((s^i)^\top u_r)(v_r^\top o^k) \]

\[ = \sum_{r=1}^{d} \alpha_r^j \beta_r^i \gamma_r^k \quad \text{with} \quad \beta_r = S^\top u_r, \quad \gamma_r = O^\top v_r \]

So, \( H \) is related to \( R \) via

\[ H = (I \otimes S^\top \otimes O^\top) R = \sum_{r=1}^{d} (I \alpha_r) \otimes (S^\top u_r) \otimes (O^\top v_r) \]

i.e. \( H \) is constrained to be the image of the lower dimensional tensor \( R \).
Experiments
Learning semantic representation for verbs

Data

- 2,000,000 Wikipedia articles
- POS-tagging + chunking + lemmatization + semantic role labelling using SENNA (Collobert et al., 2011)
- keeping sentences with syntax subject - verb - direct object
- with each term = a single word from the WordNet lexicon
Learning semantic representation for verbs

Data
- 2,000,000 Wikipedia articles
- POS-tagging + chunking + lemmatization + semantic role labelling using SENNA (Collobert et al., 2011)
- keeping sentences with syntax subject - verb - direct object
- with each term = a single word from the WordNet lexicon

Data Characteristics
- Dictionary of 30,605 words
- $n_r = 4,547$ relations
- Training set: 1,000,000 unique triplets
- Validation set: 50,000 unique triplets
- Testing set: 250,000 unique triplets
Learning semantic representation of verbs

Hyperparameters

- Embedding dimension $p \in \{25, 50, 100\}$
- Number of latent decompositions matrices $d \in \{50, 100, 200\}$
- Sparsity level as $\lambda \in \{0.01, 0.05, 0.1, 0.5, 1\} \times (n_r \times d)$

Weighting of negative triplets

Actual reduction of the number of parameters

"From $n_r^2$ parameters to $2pd + \bar{s}n_r$"

With $n_r = 4,547$, $p = 25$ and $d = 200$,

From 2,841,875 to 19,104.
Learning semantic representation of verbs

Hyperparameters

- Embedding dimension $p \in \{25, 50, 100\}$
- Number of latent decompositions matrices $d \in \{50, 100, 200\}$
- Sparsity level as $\lambda \in \{0.01, 0.05, 0.1, 0.5, 1\} \times (n_r \times d)$
- Weighting of negative triplets

Actual reduction of the number of parameters

"From $n_r^2$ parameters to $2pd + \bar{s}n_r$"

With $n_r = 4,547, p = 25$ and $d = 200$,

From 2,841,875 to 19,104.
Learning semantic representation of verbs

Hyperparameters

- Embedding dimension $p \in \{25, 50, 100\}$
- Number of latent decompositions matrices $d \in \{50, 100, 200\}$
- Sparsity level as $\lambda \in \{0.01, 0.05, 0.1, 0.5, 1\} \times (n_r \times d)$
- Weighting of negative triplets

Actual reduction of the number of parameters

“From $n_r p^2$ parameters to $2pd + \bar{s}n_r$”
Learning semantic representation of verbs

Hyperparameters

- Embedding dimension $p \in \{25, 50, 100\}$
- Number of latent decompositions matrices $d \in \{50, 100, 200\}$
- Sparsity level as $\lambda \in \{0.01, 0.05, 0.1, 0.5, 1\} \times (n_r \times d)$
- Weighting of negative triplets

Actual reduction of the number of parameters

“From $n_r p^2$ parameters to $2pd + \bar{s}n_r$”

With $n_r = 4,547$, $p = 25$ and $d = 200$,

From 2,841,875 to 19,104.
Verb prediction

<table>
<thead>
<tr>
<th>Method</th>
<th>Median Rank</th>
<th>Mean Rank</th>
<th>p@5</th>
<th>p@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our approach</td>
<td>19/96.7</td>
<td>96.7</td>
<td>0.89</td>
<td>0.98</td>
</tr>
<tr>
<td>SME Bordes et al.</td>
<td>19/99.2</td>
<td>99.2</td>
<td>0.89</td>
<td>0.98</td>
</tr>
<tr>
<td>Bigram</td>
<td>17/157.7</td>
<td>517.4</td>
<td>0.87</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Best synonyms considered
Verb prediction

- Rank of the correct verb
- Fraction of examples where the correct verb is in the top $z\%$ (average Recall at precision $(100 - z)\%$)

<table>
<thead>
<tr>
<th>Method</th>
<th>Median/Mean Rank</th>
<th>p@5</th>
<th>p@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bigram</td>
<td>48/517.4</td>
<td>0.72</td>
<td>0.83</td>
</tr>
<tr>
<td>SME Bordes et al. (2012)</td>
<td>56/199.6</td>
<td>0.77</td>
<td>0.95</td>
</tr>
<tr>
<td>Our approach</td>
<td>17/157.7</td>
<td>0.87</td>
<td>0.95</td>
</tr>
<tr>
<td>Best synonyms considered</td>
<td>19/96.7</td>
<td>0.89</td>
<td>0.98</td>
</tr>
<tr>
<td>SME Bordes et al. (2012)</td>
<td>19/99.2</td>
<td>0.89</td>
<td>0.98</td>
</tr>
</tbody>
</table>
Verb prediction

- Rank of the correct verb
- Fraction of examples where the correct verb is in the top \( z \)% (average Recall at precision \( (100 - z) \)%)

<table>
<thead>
<tr>
<th></th>
<th>synonyms not considered</th>
<th>median/mean rank</th>
<th>p@5</th>
<th>p@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our approach</td>
<td>50 / <strong>195.0</strong></td>
<td>0.78</td>
<td>0.95</td>
<td></td>
</tr>
<tr>
<td>SME Bordes et al. (2012)</td>
<td>56 / 199.6</td>
<td>0.77</td>
<td><strong>0.95</strong></td>
<td></td>
</tr>
<tr>
<td>Bigram</td>
<td><strong>48</strong> / 517.4</td>
<td>0.72</td>
<td>0.83</td>
<td></td>
</tr>
</tbody>
</table>
Verb prediction

- Rank of the correct verb
- Fraction of examples where the correct verb is in the top $z\%$ (average Recall at precision $(100 - z)\%$)

<table>
<thead>
<tr>
<th></th>
<th>synonyms not considered</th>
<th>best synonyms considered</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>median/mean rank</td>
<td>median/mean rank</td>
</tr>
<tr>
<td></td>
<td>p@5</td>
<td>p@20</td>
</tr>
<tr>
<td>Our approach</td>
<td>50 / 195.0</td>
<td>0.78</td>
</tr>
<tr>
<td>SME Bordes et al. (2012)</td>
<td>56 / 199.6</td>
<td>0.77</td>
</tr>
<tr>
<td>Bigram</td>
<td>48 / 517.4</td>
<td>0.72</td>
</tr>
</tbody>
</table>
Lexical Similarity Classification

Given two verbs are they similar semantically or not?

Data (Yang and Powers, 2006)
- 130 pairs of verbs
- labeled with score in \{0, 1, 2, 3, 4\}
- Ex:
  - (divide, split) score 4
  - (postpone, show) score 0
Lexical Similarity prediction results: PR curves

Similarity measures between verbs from

- our approach,
- SME Bordes et al. (2012),
- Collobert et al. (2011)
- the best (out of three) WordNet similarity measure (counting the number of nodes along the shortest path in the “is-a” hierarchy).
Conclusions

- Highly multi-relational data is worth modelling
- Relational learning from *intrinsic latent attributes*
- Matrix factorization models arising from variants on the stochastic block model
Conclusions

- Highly multi-relational data is worth modelling
- Relational learning from *intrinsic latent attributes*
- Matrix factorization models arising from variants on the stochastic block model
- Our approach ties or beats existing approaches on benchmark datasets
- Scales to
  - almost 5000 relations
  - more than 30,000 entities
  - 1,000,000 training triplets
- Trigram modeling
  - crucial in benchmark relational learning datasets
  - marginal in the NLP experiment
References


Formulation of the optimization problem

\[
\min_{s, o, \{\alpha^j\}, \{\Theta_r\}, y, y', z, z'} \sum_{(i,j,k) \in \mathcal{P}} \eta_{ik}^{(j)} - \sum_{(i,j,k) \in \mathcal{P} \cup \mathcal{N}} \log(1 + \exp(\eta_{ik}^{(j)})),
\]

s.t. \( \eta_{ik}^{(j)} = \mathcal{E}(s^i, R_j, o^k) \),

\[
R_j = \sum_{r=1}^{d} \alpha_r^j u_r \cdot v_r^\top, \quad ||\alpha^j||_1 \leq \lambda,
\]

\( O = S, \quad z = z' \),

\( s^j, o^k, y, y', z, u_r \) and \( v_r \) in the ball \( \{w; ||w||_2 \leq 1\} \)