Using Joint Tensor Decomposition on RDF Graphs

Michael Hoffmann

AKSW Group, Leipzig, Germany
michoffmann.potsdam@gmail.com

Abstract. The decomposition of tensors has on multiple occasions shown state of the art performance on classical Semantic Web tasks such as link prediction. However, the structure of the LOD cloud has not been taken into account in these developments so far. In particular, the decentralized architecture underlying Linked Data suggests that the incorporation of distributed information into these decompositions might lead to better factorization results. The goal of this thesis is hence to develop and evaluate models for joint tensor factorization on RDF graphs that arise from parameter estimation in statistic models or algebraic approaches. In this paper, we focus on presenting the problem we address formally and on discussing preliminary experiments on semi-synthetic data. Our experiments yield promising results inspite of being derived from an ad-hoc approach.

1 Problem Statement

The Semantic Web landscape is populated by large interlinked RDF graphs containing partly redundant and often complementary knowledge. Since many knowledge graphs contain subgraphs of information on the same entities, we argue that it would be desireable to obtain a more complete picture for the relations between these resources while performing typical semantic tasks like knowledge extraction, entity matching or link prediction. Embedding the entities of those graphs into latent spaces, that consider relations over multiple knowledge bases, opens up new avenues for improvement on established methods for this family of semantic tasks, while simultaniously lending itself to new tasks like the generation of synthetic knowledge graphs for benchmarking or the detection of graph patterns. On the other hand, latent feature models have shown state-of-the-art performance on the aforementioned tasks, especially when processing very large graphs [13][7]. However, the state of the art focuses on one graph at a time and commonly does not take the topology of the Linked Data Web into consideration.

The aim of this thesis is to *develop factorization approaches that can efficiently process groups of graphs by performing the decompositions of their adjacency tensors jointly* and *apply these decompositions in knowledge extraction and enrichment tasks*. The intuition underlying this work is that by extracting latent features with respect to several graphs concurrently, we allow implicitly for information to permeate the boundaries of single graphs and thus achieve more accurate factorizations. To achieve this goal and jointly factorize even the largest knowledge bases, we aim to develop scalable and time-efficient approaches that are able to incorporate meta-information like links between entities in different knowledge bases and literals, while also hailing from sensible

and interpretable models. Thus, this thesis will focus on the research of joint decompositions of knowledge base adjacency tensors and their applications on semantic tasks like the detection of graph patterns to achieve better results on real world applications like query answering systems.

2 Relevancy

The results of this research could lead to latent feature models that incorporate information that is spread out globally over multible RDF graphs. A large number of tasks (especially tasks which require abstract representations of resources) can profit from these results, including (1) question answering, (2) structured machine learning, (3) named entity linking and (4) relation extraction. For example, question answering frameworks could use the latent features to map slots in a natural-language query to entries in a SPARQL query. In relation extraction, the latent features of resources could potentially improve the selection of sentences for extracting RDF triples from text. Named entity linking could profit from the latent features we generate by providing numeric hints towards entity mentions in natural language.

The runtime and accuracy of the provided solutions will be the deciding factors pertaining to their adoption by the Semantic Web community and the corresponding industry. Hence, we aim to develop time-efficient methods and apply them to the tasks aforementioned. Furthermore, to reason about these algorithms it would be desireable to derive them explicitly from model hypotheses on the structure of real-world knowedge bases. Thus, we will develop factorization models that abide by the characteristics of Linked Data. We hope that our results will play a key role towards a new generation of tools for processing large amounts of interlinked data.

3 Related Work

There is a wealth of knowledge on tensor factorization in the literature. For the seminal survey on this topic, see Kolda et al. [6]. On the topic of single tensor factorization, most models are based on either the DEDICOM [2] or the CP [3] family of decompositions. Nickel et al. proposed a relaxed DEDICOM model, refered to as RESCAL [12], that is shown to be highly scalable [13], while still exhibiting state of the art performance on smaller datasets. Building on this London et al. [10] suggested a similar model, adding a bias term and using the learned model as input for a neural network. Kolda et al. [5] proposed a CP Model, to identify citation communitys and measure document similarity. Kuleshov et al. [8] proposed to find the CP factors of their model, via simultaneous matrix diagonalization.

On the topic of joint tensor factorization, Khan. et al. [4] proposed a bayesian model for the problem of multi tensor factorization. They relaxed the constraints of the problem to use normal distributions for inference of the CP factors. Also the joint factorization of matrices and tensors has shown improvements over singular approaches, see [1].

4 Research Question

To adress the issues stated above, we plan to analyze and generalize existing approaches and to use the insights gained there to develop new techniques and benchmark their performance on knowledge extraction and integration tasks against the current state of the art. As such, the research questions and corresponding subquestions at the heart of our work are as follows:

- *"What existing approaches can be applied to joint factorization?"*
  - What existing approaches can be interpreted as stemming from statistical models? By increasing the sample size one could generalize them to multiple knowledge bases.
  - Can we derive update rules that are computable on large datasets by combining different cost functions or in a distributed manner?
- *"How can we use the resulting embeddings to improve upon existing tasks and what new avenues can be explored?"*
  - Can one improve the performance of the translation of natural-language queries into formal query languages (SPARQL) by using embeddings that respect the spread of RDF data on the web?
  - Can we use these latent embeddings for the generation of synthetic graphs which still capture the structure of RDF Data on the web? This will be useful for benchmarking various querying systems and the like.

5 Hypotheses

The basic hypothesis behind our work is that using the architecture underlying the Linked Data Web can lead to the computation of better embeddings latent embeddings. We hence aim to develop novel methods for tensor factorizatiion that make use of distributed information to compute embeddings. The approaches we develop will aim to be time-efficient and easy to distribute so as to scale to billions of triples. Moreover, we will consider datasets that are updated regularly and aim for deriving updates rules or modularization techniques that will ensure a efficient update of latent features.

6 Preliminary results

To begin testing the potential of a simple variant of joint decomposition, we chose to use the proven RESCAL-ALS [12] method on a data dump of the Side Effect Resource SIDER.[1] From that data, we built an adjacency tensor that does not differentiate between literals and entities. Ee arrived at the full tensor with a shape of $11 \times 30378 \times 30378$ and models the complete knowledge contained in the RDF graph. To model delocalized information, we produced two more tensors X, Y of the same shape by deleting a percentage p of entities from the full tensor twice, taking care that entities deleted in

[1] http://linkeddatacatalog.dws.informatik.uni-mannheim.de/dataset/fu-berlin-sider/resource/ee6c1709-f2a6-4790-8549-152232dd9d93

X would be present in Y and vice versa[2].
The task at hand was to reconstruct the global knowledge contained in the full tensor based exclusively on the partial knowledge contained in X and Y. To that end, we chose to search for solutions of the "RESCAL"-form, that solve the problem

$$(\tilde{R}, \tilde{A}) = \arg\min_{(R,A)} \left(\|ARA^t - X\|_2^2 + \|ARA^t - Y\|_2^2 \right) + reg(A, R) \tag{1}$$

with $A \in \mathbb{R}^{r \times 30378}$, $R \in \mathbb{R}^{11 \times r \times r}$ for some $r < 30378$ and

$$reg(A, R) = \lambda \left(\|A\|_2^2 + \|R\|_2^2 \right)$$

Take note that for all solutions of (1), one also has[3]

$$(\tilde{R}, \tilde{A}) = \arg\min_{(R,A)} \|ARA^t - \frac{X+Y}{2}\|_2^2 + reg(A, R) \tag{2}$$

This allows us to approximate solutions of (1), by using the RESCALS-ALS algorithm, developed in [12]. To measure the goodness of fit, we first chose a rounding treshhold t and rounded the entries of ARA^t in comparison with that. After that we calculated the F1-measure (F1), precision (P) and recall (R) of ARA^t with respect to the full tensor and compared results for different choices of t, r and p. We wrote the algorithm in Python and all experiments were performed on an Intel Core i5-6500 CPU @ 3.20GHz $\times$ 4 with 16GB of RAM. As a baseline we chose the best performing RESCAL fit of the incomplete tensors X and Y. The following table holds the best results of our baseline tests[4]

Table 1. Rescal Baseline

p	r	t	F1	P	R	r	F1	P	R	r	F1	P	R	r	F1	P	R
10%	5	.2	0.41	0.45	0.39	10	0.45	0.49	0.41	15	0.48	0.51	0.45	20	0.49	0.52	0.46
20%		.2	0.37	0.46	0.31		0.40	0.49	0.34		0.42	0.63	0.31		0.40	0.51	0.33
30%		.2	0.32	0.46	0.25		0.34	0.49	0.26		0.36	0.52	0.28		0.37	0.53	0.29

To enable a fine grained comparison we include all values of t in the table that holds the values of the joint approach As one can see, the joint approach outperforms the baseline by as much as 25% on the parameter pair $(p = 30\%, r = 15)$. The large

[2] This ensures that no global information is being deleted

[3] This can be seen by summing over the identity

$$2\left((ARA^t)_{ijk} - \frac{X_{ijk} + Y_{ijk}}{2}\right)^2 = \left((ARA^t)_{ijk} - X_{ijk}\right)^2 + \left((ARA^t)_{ijk} - Y_{ijk}\right)^2 + \frac{1}{2}\left(X_{ijk} - Y_{ijk}\right)^2$$

[4] We rounded all values to two decimal places.

Table 2. Joint Rescal Factorization

p	t	r	F1	P	R	r	F1	P	R	r	F1	P	R	r	F1	P	R
10%	.8	5	0.13	0.96	0.07	10	0.19	0.96	0.10	15	0.21	0.97	0.12	20	0.22	0.97	0.13
	.7		0.16	0.93	0.09		0.23	0.93	0.13		0.26	0.94	0.15		0.27	0.94	0.16
	.6		0.21	0.88	0.12		0.29	0.88	0.17		0.31	0.89	0.19		0.32	0.90	0.20
	.5		0.26	0.82	0.16		0.34	0.82	0.22		0.37	0.83	0.24		0.39	0.85	0.25
	.4		0.34	0.74	0.21		0.41	0.74	0.29		0.44	0.76	0.31		0.46	0.78	0.33
	.3		0.40	0.63	0.30		0.46	0.65	0.36		0.49	0.67	0.39		0.51	0.69	0.41
	.2		0.44	0.49	0.40		0.48	0.53	0.45		0.51	0.55	0.48		0.53	0.56	0.50
20%	.8	5	0.05	0.95	0.03	10	0.08	0.95	0.04	15	0.09	0.96	0.05	20	0.10	0.96	0.05
	.7		0.08	0.92	0.04		0.09	0.91	0.06		0.12	0.93	0.06		0.14	0.93	0.07
	.6		0.11	0.87	0.06		0.14	0.86	0.08		0.17	0.88	0.09		0.18	0.89	0.10
	.5		0.16	0.81	0.09		0.20	0.81	0.12		0.23	0.84	0.13		0.25	0.85	0.15
	.4		0.25	0.76	0.15		0.31	0.77	0.19		0.35	0.80	0.22		0.35	0.81	0.23
	.3		0.32	0.67	0.21		0.38	0.68	0.27		0.42	0.72	0.30		0.44	0.73	0.32
	.2		0.40	0.54	0.31		0.45	0.56	0.37		0.48	0.60	0.40		0.49	0.61	0.42
30%	.8	5	0.04	0.96	0.02	10	0.05	0.96	0.03	15	0.06	0.96	0.03	20	0.07	0.97	0.04
	.7		0.05	0.93	0.03		0.07	0.93	0.04		0.08	0.93	0.04		0.09	0.94	0.05
	.6		0.08	0.90	0.04		0.10	0.89	0.05		0.11	0.90	0.06		0.12	0.91	0.06
	.5		0.11	0.85	0.06		0.14	0.85	0.08		0.17	0.87	0.09		0.18	0.88	0.10
	.4		0.18	0.81	0.10		0.24	0.83	0.15		0.28	0.85	0.17		0.30	0.86	0.19
	.3		0.25	0.72	0.15		0.32	0.74	0.20		0.37	0.78	0.24		0.40	0.79	0.26
	.2		0.35	0.60	0.25		0.41	0.63	0.30		0.45	0.64	0.34		0.46	0.67	0.35

changes of F1-measure in the joint approach at $t = 0.5$ can be explained by the fact that the algorithm tries to approximate the tensor $(X + Y)/2$. Take note that

$$\left(\frac{X+Y}{2}\right)_{ijk} = \begin{cases} 0 & \text{if } X_{ijk} = Y_{ijk} = 0 \\ 1 & \text{if } X_{ijk} = Y_{ijk} = 1 \\ \frac{1}{2} & \text{if } X_{ijk} \neq Y_{ijk} \end{cases} \tag{3}$$

Thus by rounding up at 0.5 one will begin to see the values that were correctly learned to be $1/2$, among others.
To remedy that situation we used the ad-hoc approach of learning the tensor $\max((X, Y))_{ijk} = \max(X_{ijk}, Y_{ijk})$[5] under the same quadratic loss function, using the same RESCAL-ALS algorithm. A sample of the results obtained are contained in the following table

Table 3. Rescal MAX

t	r	F1	P	R	t	r	F1	P	R	t	r	F1	P	R	t	r	F1	P	R
.2	5	0.46	0.46	0.46	.2	10	0.49	0.49	0.49	.3	15	0.53	0.63	0.45	.3	20	0.54	0.65	0.47

[5] take note that, by our synthetic construction of X and Y, max(X,Y) will represent the full knowledge graph, thus the parameter p is irrelevant here.

As one can see, the results improved compared with learning the arithmetic mean. This hints that there is improvement to be had by reweighting the entries in the joint decomposition, which we will explore in our future work.

7 Approach

To exemplify the general approach, consider the statistical model (S, p), with the measurable space

$$S := (\Omega, P(\Omega)), \text{ with } \Omega := \prod_m \{0,1\}^{k \times n \times n} \tag{4}$$

equipped with a propability measure

$$p : P(\Omega) \to [0,1]; \{X_1, \ldots, X_m\} \mapsto p(\{X_1, \ldots, X_m\}). \tag{5}$$

This is the general setting and by choosing different parametric families of propability measures, one can use inference methods to obtain estimates for the parameters.
Take note that the choice of S makes this approach novel, because the usual methods for statistical inference on semantic adjacency tensors use measurable spaces of the form $(\{0,1\}^{k \times n \times n}, P(\{0,1\}^n, p)$ to model the propabilities for links on a single tensor.
This is well known and appears frequently in the Semantic Web literature. For instance in [11], the authors use the propability function

$$p(X|A,R) := \prod_{i,j,k} \tilde{p}(x_{ijk}| \langle A_i, R_k A_j \rangle) \tag{6}$$

with $\tilde{p}(x_{ijk} = 1| \langle A_i, R_k A_j \rangle) := \sigma(\langle A_i, R_k A_j \rangle)$. Then they chose to estimate the parameters A and R by minimizing the log-likelihood. By extending the sample space to multiple tensors, one has a larger sample size for each individual entry and that will lead to better parameter estimations.
Furthermore, if one has to deal with multiple knowledge bases, the question of confidence arises. To solve the problem of observing a link in knowledge base A and not in knowledge base B, one would have to introduce confidence values to the model, to make informed predictions.
Another approach will be modelling the random events in our model as Markov processes. This would be appropriate to model situations where the source of our data are multiple versions of a knowledge graph, at different points in time. The ability to incorporate Markov processes of graphs into our model, is theoretical highly valuable, since they possess an unparalleled modelling power for real world processes.

8 Evaluation Plan

To test our hypotheses, we will apply our derived algorithms in three stages. Stage one will be a test on synthetic or semi-synthetic data, where we will build tensors from real-world data, according to synthetic rules to model the existence of a tensor that contains the global knowledge and measure the performance of the model using the

F1-measure[6].
Stage two will be a test on real world tensors which contain a large amount of related entities, e.g., YAGO [14] and DBPedia [9]. Thus, we are abstracting the existence of a global knowledge tensor, still keeping the same evaluation sheme as in stage one.
Finally in stage three, we will test the latent factors we have benchmarked in stage two to fields such as relation extraction, entity linking and question answering to boost their performances.

9 Reflections

As our preliminary results show, one can expect better latent embeddings by considering multiple sources of relational data. While there is a wealth of knowledge for the factorization of matrices or tensors, there is still relatively little work done on the joint factorization of tensors. Furthermore, by considering quite general propability functions, we can model sophisticated dependencies of different knowledge graphs.
This will be the start of this PHD thesis.

Acknowledgements This work has been supported by Eurostars projects DIESEL (E!9367) and QAMEL (E!9725) as well as the European Union's H2020 research and innovation action HOBBIT (GA 688227).
Also, I want to thank my advisors Dr. Axel-C. Ngonga Ngomo, Ricardo Usbeck and the whole team at AKSW for all their helpful comments and fruitful coffee-driven conversations.

References

1. E. Acar, M. A. Rasmussen, F. Savorani, T. Næs, and R. Bro. Understanding data fusion within the framework of coupled matrix and tensor factorizations. *Chemometrics and Intelligent Laboratory Systems*, 129:53–63, 2013.
2. R. A. Harshman. Models for analysis of asymmetrical relationships among n objects or stimuli. In *First Joint Meeting of the Psychometric Society and the Society for Mathematical Psychology, McMaster University, Hamilton, Ontario*, 1978.
3. F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys*, 6(1):164–189, 1927.
4. S. A. Khan, E. Leppäaho, and S. Kaski. Multi-tensor factorization. *arXiv preprint arXiv:1412.4679*, 2014.
5. T. Kolda. Multilinear algebra for analyzing data with multiple linkages.
6. T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, Aug. 2009.
7. D. Krompaß, M. Nickel, and V. Tresp. Large-scale factorization of type-constrained multi-relational data. In *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*, pages 18–24. IEEE, 2014.
8. V. Kuleshov, A. T. Chaganty, and P. Liang. Tensor factorization via matrix factorization. *CoRR*, abs/1501.07320, 2015.

[6] just as in our "Preliminary Results" section

9. J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6(2):167–195, 2015.
10. B. London, T. Rekatsinas, B. Huang, and L. Getoor. Multi-relational learning using weighted tensor decomposition with modular loss. *CoRR*, abs/1303.1733, 2013.
11. M. Nickel and V. Tresp. Logistic Tensor Factorization for Multi-Relational Data. *ArXiv e-prints*, June 2013.
12. M. Nickel, V. Tresp, and H.-P. Kriegel. A three-way model for collective learning on multi-relational data. In L. Getoor and T. Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 809–816, New York, NY, USA, June 2011. ACM.
13. M. Nickel, V. Tresp, and H.-P. Kriegel. Factorizing yago: Scalable machine learning for linked data. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 271–280, New York, NY, USA, 2012. ACM.
14. F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA, 2007. ACM.