Knowledge Graph Embeddings for Recommender Systems

Enrico Palumbo[1,2,3]

[1] LINKS Foundation, Turin, Italy
[2] EURECOM, Biot, France
[3] Politecnico di Torino, Turin, Italy
enrico.palumbo@polito.it

1 Introduction

"Less is more" is the motto that the architect Ludwig Mies van der Rohe invented to describe the minimalist aesthetic in architecture. In the digital age, though, abundance is king: it is estimated that 90% of all the world's data has been created in the past two years[4]. This abundance of options to choose from within just one-click should make us happier and more satisfied with our decisions, allowing us to find what is just right for our needs. In fact, psychological studies show that humans are quite bad at choosing among many different options ("The Paradox of choice" [1]). Recommender Systems are software tools that create a personalized filter that help users in reducing the number of available options, pre-filtering a subset of items that are deemed relevant, and thus helping in addressing the paradox of choice. Knowledge graphs are an ideal data structure for recommender systems, as they allow to easily represent user-item interactions, and item and user properties as typed edges connecting pairs of entities. Recommender systems based on knowledge graphs have shown to generate high quality recommendations that are also easier to interpret and explain [2–4]. However, engineering features for recommender systems from a knowledge graph is a complex, time-consuming and domain-dependent endeavor.

My dissertation is located at the intersection between the Recommender Systems and the Semantic Web research fields and it shows how knowledge graph embeddings can be use to automate the process of feature engineering for a knowledge-aware recommender systems. This approach allows to leverage the advancements of Deep Learning in learning features for prediction problems and the wealth of data available on the Web in the structured form of Knowledge Graphs to learn high-quality features that lead to accurate, non-obvious and interpretable recommendations.

In the rest of the paper, we describe the three main contributions of the dissertation: entity2rec, STEM, Path Recommender. We conclude by discussing their value, limitations and future directions.

[4] https://bit.ly/2Yr6YSm

2 entity2rec

The crucial point to leverage knowledge graphs to generate item recommendations is to be able to define effective features for the recommendation problem. Knowledge graph embeddings learn a mapping from the knowledge graph to a feature space solving an optimization problem, minimizing the time-consuming endeavor of feature engineering and leading to higher quality features.
The first contribution of the dissertation is to show how existing knowledge graph embeddings algorithms such as translational models [5], node2vec [6] can be applied to the recommendation problem. Then, we have pointed out that none of the existing algorithms allowed to both encode structural properties of the graph and the semantics of the KG properties in the learned features and we have introduced entity2rec [7, 8]. entity2rec learns user-item relatedness for item recommendation through property-specific knowledge graph embeddings. It works by creating property-specific subgraphs, learning property-specific embeddings and relatedness scores that are finally combined by an aggregation function to create a personalized ranking of items for a specific user. entity2rec can be seen as a generalization of node2vec to knowledge graphs that leverages node2vec's flexibility in encoding graph-structure data into feature vectors and, at the same time, preserves the semantics of the KG properties in its recommendation model. We have described a common experimental setup, composed of three datasets (Movielens1M, LastFM, LibraryThing), a specific evaluation protocol and a set of metrics to evaluate the quality of the recommendations (precision, recall, serendipity, novelty). We have then compared knowledge graph embeddings systems among each other and with a set of state-of-the-art collaborative filtering systems. Results show that entity2rec outperforms competing systems in terms of precision, recall and serendipity, and has a good level novelty, especially when the dataset is strongly sparse and has a low popularity bias (LibraryThing). Also, entity2rec recommendation model is a linear combination of property-specific relatedness scores. The semantic information that is encoded in the entity2rec recommendation model can have several advantages: it can be used to interpret and explain recommendations, as well as for configuring recommendations to specific user requests ("Suggest me a movie with my favorite actors").
Finally, we have experimented with entity2rec in a cold-start scenario introducing Tinderbook [9], a web application that provides book recommendations given a single book that the user likes. Tinderbook shows how entity2rec can be used with new users through an item-item relatedness measure. Tinderbook highlights the value of using Semantic Web technologies in building recommender systems in an applied scenario as DBpedia is used not only to create connections among entities, but also to provide book titles and abstracts.

3 STEM

In addition to defining effective features, a crucial element for the quality of knowledge-aware recommender systems is the quality of the knowledge graph itself. Typically, when building a knowledge graph from a set of heterogeneous data sources, duplicates are a major source of noise in the data. The problem of deciding whether two records refer to the same real world entity is called 'entity matching'. Threshold-based classifiers are simple and very much in-use algorithms to solve the entity matching problem. The idea is that two records refer to the same-world entity if their similarity exceeds a given threshold. However, this threshold introduces a trade-off between precision and recall: a high threshold leads to high precision and low recall, and a low threshold to the opposite. In the dissertation, we introduce "STEM: Stacked Threshold-based Entity Matching" [10]. STEM is a machine learning layer that can be 'stacked' upon existing threshold-based classifiers to improve their precision and recall combining the predictions of several thresholds. STEM has been tested on three datasets from different domains (finance, music) using two different threshold-based classifiers (linear [11] and Naive Bayes[5]). We have shown that stacking breaks the trade-off between precision and recall, significantly enhancing the F-score (from +13% up to +43%), using two different threshold-based classifiers and three different datasets. We have also shown that STEM is less dependent on the amount of training than a system that performs machine learning from 'scratch', i.e. using directly property similarity values. Finally, we have described how STEM has been applied in a real use-case: the construction of the 3cixty knowledge graph [12]. We have described the aim of the 3cixty project and shown how STEM has been an important element of the process of knowledge graph generation, improving the data reconciliation process and leading to a higher quality of the graph.

4 Path Recommender

The third topic that I addressed in the dissertation is the extension of the recommendation problem to temporal sequences, i.e. Sequence-Aware Recommender Systems (SARS). Specific attention is devoted to the problem of learning to recommend tourist paths, sequences of tourist activities that can be of interest for a user. First, we have created a new dataset of sequences of tourist activities collecting Foursquare check-ins enriched with semantic metadata that we have publicly released as the Semantic Trails Dataset[13]. We have then introduced the Path Recommender[14], a RNN-based recommender systems that learns to generate tourist paths from the sequences of check-ins. The Path Recommender is based on the assumption that sequence-aware recommendation can be seen as a language modelling problem and it is trained to predict the next user check-in given the previous ones. To evaluate the Path Recommender we have introduced and published Sequeval [15], an evaluation framework that includes protocols, metrics, and baselines for the evaluation of SARS. The experimental results have shown that the Path Recommender generates accurate and non-obvious recommendations with respect to traditional sequence-aware algorithms. Finally, we have extended the Path Recommender model to the music playlist continuation

[5] https://github.com/larsga/Duke

task in the RecSys2018 challenge, introducing additional data sources such as music lyrics, and achieving the the 14th position out of 33 participants in the creative track and the 36th position out of 113 participants in the main track.

5 Significance and future directions

The first contribution of this dissertation is entity2rec. entity2rec has been publicly released on Github for the community to use[6] and represents a state-of-the-art recommender systems based on knowledge graph embeddings that also provides interpretability and explainability of the recommendations. Furthermore, the application of entity2rec in Tinderbook[7] is an excellent showcase of the use of Semantic Web technologies in an online recommender systems: users generally give positive feedback about the app, saying that it's fun to use and gives accurate suggestions. Some users have complaint a lack of diversity and as a future work we would like to investigate on hybrid optimization algorithms that can account for this metric. More on the theoretical level, we have shown that the recommendation problem can be seen as a link prediction problem on a knowledge graph and that any existing knowledge graph embedding algorithm can be applied to the recommendation problem. The major limitation of entity2rec is that it currently does not support incremental training. As a future direction, the locality of the random walks can be exploited to update embedding models when new data is added without retraining from scratch.

For what concerns the entity matching problem in the generation of knowledge graphs, we release STEM on Github[8]. STEM readily comes with a support for Silk and Duke, but can be easily adapted to any threshold-based classifier by any researcher or practitioner looking to improve the F-score of the entity matching process. STEM currently uses a naive sequential implementation that can easily be parallelized to save computational time.

With the Path Recommender, we have shown that recommending sequences can be seen as a language modelling problem and that RNNs are quite effective at this task. We have published the Semantic Trails Dataset[9] and the Sequeval evaluation framework[10]. The public release of this dataset and the evaluation framework aim to foster research in the field, as well as to push towards shared evaluation standards that increase reproducibility and comparability of studies. Note that the Path Recommender currently does not use semantic information from knowledge graphs. A promising future work is to integrate the Path Recommender with entity2rec to create knowledge-aware sequential recommendations. Also, a promising future direction is the application of the new breakthroughs that transformer models have created in the NLP field to the challenges addressed in this dissertation [16]).

[6] https://github.com/D2KLab/entity2rec
[7] www.tinderbook.it
[8] https://github.com/enricopal/STEM
[9] https://github.com/D2KLab/semantic-trails
[10] https://github.com/D2KLab/sequeval

References

1. Barry Schwartz. The paradox of choice: Why more is less. Ecco New York, 2004.
2. Tommaso Di Noia, Vito Claudio Ostuni, Paolo Tomeo, and Eugenio Di Sciascio. Sprank: Semantic path-based ranking for top-n recommendations using linked open data. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(1):9, 2016.
3. Petar Ristoski, Jessica Rosati, Tommaso Di Noia, Renato De Leone, and Heiko Paulheim. Rdf2vec: Rdf graph embeddings and their applications. *Semantic Web*, (Preprint):1–32, 2018.
4. Rose Catherine, Kathryn Mazaitis, Maxine Eskenazi, and William Cohen. Explainable entity-based recommendations with knowledge graphs. *arXiv preprint arXiv:1707.05254*, 2017.
5. Enrico Palumbo, Giuseppe Rizzo, Raphaël Troncy, Elena Baralis, Michele Osella, and Enrico Ferro. Translational models for item recommendation. In *European Semantic Web Conference*, pages 478–490. Springer, 2018.
6. Enrico Palumbo, Giuseppe Rizzo, Raphaël Troncy, Elena Baralis, Michele Osella, and Enrico Ferro. Knowledge graph embeddings with node2vec for item recommendation. In *European Semantic Web Conference*, pages 117–120. Springer, 2018.
7. Enrico Palumbo, Giuseppe Rizzo, and Raphaël Troncy. Entity2rec: Learning user-item relatedness from knowledge graphs for top-n item recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 32–36. ACM, 2017.
8. Enrico Palumbo, Diego Monti, Giuseppe Rizzo, Raphaël Troncy, and Elena Baralis. entity2rec: Property-specific knowledge graph embeddings for item recommendation. *Expert Systems with Applications*, page 113235, 2020.
9. Enrico Palumbo, Alberto Buzio, Andrea Gaiardo, Giuseppe Rizzo, Raphael Troncy, and Elena Baralis. Tinderbook: Fall in love with culture. In *European Semantic Web Conference*, pages 590–605. Springer, 2019.
10. Enrico Palumbo, Giuseppe Rizzo, and Raphaël Troncy. Stem: Stacked threshold-based entity matching for knowledge base generation. *Semantic Web*, 10(1):117–137, 2019.
11. Robert Isele, Anja Jentzsch, and Christian Bizer. Silk server-adding missing links while consuming linked data. In *Proceedings of the First International Conference on Consuming Linked Data-Volume 665*, pages 85–96. CEUR-WS. org, 2010.
12. Raphael et al. Troncy. 3cixty: Building comprehensive knowledge bases for city exploration. *Web Semantics: Science, Services and Agents on the World Wide Web*, 46:2–13, 2017.
13. Diego Monti, Enrico Palumbo, Giuseppe Rizzo, Raphaël Troncy, and Maurizio Morisio. Semantic trails of city explorations: How do we live a city. *arXiv preprint arXiv:1812.04367*, 2018.
14. Enrico Palumbo, Giuseppe Rizzo, Raphaël Troncy, and Elena Baralis. Predicting your next stop-over from location-based social network data with recurrent neural networks, 2017.
15. Diego Monti, Enrico Palumbo, Giuseppe Rizzo, and Maurizio Morisio. Sequeval: An offline evaluation framework for sequence-based recommender systems. *Information*, 10(5):174, 2019.
16. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.