Building a Conference Recommender System Based on SciGraph and WikiCFP

Andreea Iana¹(✉), Steffen Jung¹, Philipp Naeser¹, Aliaksandr Birukou^{2,3},
Sven Hertling¹, and Heiko Paulheim¹

¹ Data and Web Science Group, University of Mannheim, Mannheim, Germany
andreeaiana@gmail.com, mail@jung.vision, pnaeser@mail.uni-mannheim.de,
{sven,heiko}@informatik.uni-mannheim.de

² Springer Nature, Heidelberg, Germany

Aliaksandr.Birukou@springer.com

³ Peoples' Friendship University of Russia (RUDN University), Moscow, Russia

Abstract. SciGraph is a Linked Open Data graph published by Springer Nature which contains information about conferences and conference publications. In this paper, we discuss how this dataset can be utilized to build a conference recommendation system, yielding a recall@10 of up to 0.665, and a MAP of up to 0.540, generating recommendations based on authors, abstracts, and keywords. Furthermore, we show how the dataset can be linked to WikiCFP to recommend upcoming conferences.

Keywords: Recommender system · SciGraph · Scientific publications

1 Introduction

Bibliographic datasets form a major topic in the Linked Open Data Cloud¹, accounting for a total of 12–13% of all datasets [15]. One of those datasets is SciGraph², which is published by Springer Nature and is the successor of Springer's Linked Open Data Conference Portal [3], comprising 7.2M articles and 240k books published by Springer Nature, and totaling to 1B triples.

In this paper, we aim at exploiting SciGraph to provide users with recommendations of conferences to submit their publications to, utilizing SciGraph for information on past conferences and publications, and WikiCfP for information on upcoming conferences.

¹ <https://lod-cloud.net/>.

² <https://www.springernature.com/de/researchers/scigraph>.

The publication has been prepared with the support of the “RUDN University Program 5-100”.

© The Author(s) 2019

M. Acosta et al. (Eds.): SEMANTICS 2019, LNCS 11702, pp. 117–123, 2019.

https://doi.org/10.1007/978-3-030-33220-4_9

2 Related Work

The idea of building recommender systems for scholarly content goes back almost 20 years [2, 7]. More recently, Linked Open Data has been recognized as a valuable source for building recommender systems. In particular, content-based recommender systems, which focus on the items to be recommended and their interrelations, can benefit strongly from detailed descriptions of those items in open datasets [4, 5].

Similar to the task in this paper, several approaches have been proposed for the recommendation of *research papers* (see [1] for a comprehensive survey). Although sharing the same domain, the setup is slightly different here – in our scenario, both the input data (i.e., authors, a textual abstract, keywords), and the prediction target (conferences instead of individual papers) are different.

3 Approach

3.1 Datasets

The main dataset used to train the recommender system is SciGraph. For training, we use publications from the years 2013–2015, whereas for evaluation, publications from the year 2016 are used. In total, SciGraph contains 240,396 books, however, only a fraction out of those correspond to the proceedings of a single conference. Moreover, it contains 3,987,480 individual book chapters, again, a fraction of which correspond to papers published at conferences. Additionally, SciGraph provides a taxonomy of research topics, called *Product Market Codes* (PMCs). In total, 1,465 of those PMCs are included in the hierarchy and assigned to books. Only 89 of those PMCs are related to computer science.

The second dataset we use is WikiCfP³, a website which publishes calls for papers. Since there is no downloadable version of the data (although the CC-BY-SA license allows for reusing the dataset), we built a crawler to create a dataset of CfPs, containing names, acronyms, dates, locations, and submission deadlines (which we consider mandatory attributes), as well as links to the conference page, the conference series, categorization in WikiCfP, and textual description (which we consider optional attributes). Overall, we crawled data for 65,714 CfPs in July 2018. The crawled data was linked to SciGraph using string similarity between the conference names. This leads to 53.1% of the CfPs linked to SciGraph.

3.2 Recommendation Techniques

We use three main families of recommendation techniques, i.e., recommendations based on authors, abstracts, and keywords. Furthermore, we also use an ensemble strategy. Generally, the recommendation strategies either exploit some notion of similarity (e.g., recommending conferences which contain publications with similar abstracts), or model the problem as a machine learning problem (i.e., since we have 742 conference series in our training set, we train a multi-label classifier for 742 classes).

³ <http://www.wikicfp.com>.

Author-based recommendations are computed based on the authors of an application. Essentially, we count the number of papers per conference series which share at least one author with the authors given in the abstract, and use that count as a score.⁴

Abstract-based recommendations compare the abstracts of publications in SciGraph with the abstract given by the user. Overall, we use two different approaches: the *max* strategy finds single publications with the highest abstract similarity and proposes the corresponding conference, while the *concat* strategy concatenates all abstracts related to a conference to a virtual document, and compares the given abstract to those virtual documents.

Different variants for generating recommendations are used. We utilize standard TF-IDF, as well as TF-IDF based on word n-grams, LSA and LSA based on word n-grams [10], and pLSA [6]. Furthermore, we utilize similarity based on word embeddings, based on word2vec [11], GloVe [13], and FastText [8], using both pre-trained embeddings, as well as embeddings trained on the SciGraph collection of abstracts. While all those approaches are based on similarities, we also tried directly predicting the conferences using a convolutional neural network (CNN) approach, which takes the self-trained word2vec embeddings as representations for words, as discussed in [9].

Keyword-based recommendations are based on *Product Market Codes* in SciGraph. Such product market codes are defined by Springer Nature and resemble other categorization systems in computer science, such as the ACM computing classification system. A second keyword-based model uses a script to identify Computer Science Ontology (CSO) [14] terms in the abstract entered by the user.

4 Evaluation

As sketched above, publication data from 2013–2015 were used as training data for the recommender system, whereas publications from 2016 were used for testing. For each publication in the test set, we try to predict the conference at which it has been published, and compare the results to the gold standard (i.e., the conference in which it has actually been published). We create 10 recommendations with each technique⁵, and report recall@10 and mean average precision (MAP).

Table 1 shows some basic statistics of the training and test set. In total, the recommender system is trained on 742 conference series and 555,798 papers written by 110,831 authors. As far as the abstracts are concerned, only little

⁴ We do not disambiguate authors here, since no further clues for disambiguation, such as organizations, or unique IDs, such as ORCID, are present in SciGraph.

⁵ The only exception are recommendations based on authors, which may create shorter lists in cases where all authors altogether have published at less than 10 conferences contained in SciGraph.

Table 1. Characteristics of the training and test set

	Training (2013–2015)	Test (2016)	Overlap
Distinct conference series ID	742	526	405
Distinct author names	110,831	53,862	20,529
Product market codes	155	150	115
Papers	555,798	200,502	–
English abstracts	57,797	21,323	–

Table 2. Results of the best performing individual recommendation techniques. For each individual technique, we only report the results of the best performing strategy (max or concat).

Method	Recall@10	MAP
Author-based	0.372	0.284
Abstract-based TF-IDF (concat)	0.461	0.237
Abstract-based n-gram TF-IDF (concat) w/ cosine similarity	0.490	0.270
Abstract-based n-gram TF-IDF (concat) w/ Multinomial Naive Bayes	0.494	0.273
Abstract-based LSA (concat)	0.461	0.237
Abstract-based n-gram LSA (concat)	0.490	0.270
Abstract-based pLSA (concat)	0.369	0.172
Abstract-based Glove pre-trained (max)	0.229	0.097
Abstract-based word2vec self-trained (max)	0.346	0.154
Abstract-based word2vec plus CNN (concat)	0.405	0.201
Abstract-based doc2vec (concat)	0.352	0.164
Keyword-based SciGraph market codes (max)	0.665	0.522
Keyword-based CSO (max)	0.201	0.081
Ensemble TF-IDF & word2vec plus CNN (10)	0.498	0.250
Ensemble TF-IDF & word2vec plus CNN & SciGraph market codes (10)	0.648	0.509
Ensemble TF-IDF & word2vec plus CNN & SciGraph market codes (100)	0.662	0.539
Ensemble TF-IDF & word2vec plus CNN & SciGraph market codes (1,000)	0.661	0.540

more than 10% of all the papers have an English language abstract.⁶ The average length of an abstract is 136 words.

Table 2 summarizes the results of the best performing models for recommendations based on authors, abstracts, and keywords. Generally, abstracts work better than authors, and keywords work better than abstracts. For abstracts, TF-IDF using single tokens yields a recall@10 of 0.461 and a MAP of 0.237. For using TF-IDF with n-grams, we explored different variants: we varied the upper limit for n between 2 and 5, and evaluated the approach with the 500k and 1M most frequent n-grams, as well as with all n-grams. The best results were obtained when using the 1M most frequent n-grams of size 1 to 4, outperforming the standard TF-IDF approach.

⁶ For a larger fraction of papers in SciGraph, no abstract is contained in the dataset.

In addition, we also evaluated a few ensemble setups. These were built by combining recommendation lists of length 10, 100, and 1,000, given by different base recommenders, and using a logistic regression as a meta learner [16] to generate a recommendation list of length 10 as in the setups above. We can observe that combining two abstract-based techniques (TF-IDF and word2vec plus CNN, which were very diverse in their predictions), outperforms the two individual techniques in both recall@10 and MAP.

Building ensembles incorporating SciGraph market codes yields no significantly better results than using keywords alone, demonstrating that those keywords are in fact the most suitable indicator for recommending conferences. Generally, extending the base recommendation lists beyond 100 elements does not change the results much, because conferences predicted on a position higher than 100 are unlikely to be considered in the final result list of size 10.

The recall figures reported in Table 2 do not exceed 0.665, but this result should be considered in a broader context. In total, only 77% of all conferences in the test set are also contained in the training set, i.e., we do not have any training signals for the remaining conferences. Since we can only use previous publications of proceedings for generating training features, the approaches discussed in this paper can only recommend conferences known from the training set, i.e., the maximum recall we could reach with these methods would be 0.815.

In general, we can see that keyword-based models are the best performing ones. However, they are also the least user-friendly ones, since product market codes are assigned by editors at Springer Nature (more recently, using automated tools [12]). While end users might be able to assign them at a decent quality, the actual recommendation quality with user-assigned keywords might actually be lower than the one based on editor-assigned product market codes. Another possible issue is that by selecting up to seven keywords out of 1,465, one could easily create pseudo-keys for conferences (i.e., each conference can be uniquely identified by its keywords), so overfitting might also be an issue for those models.

Another observation we have made in our experiments is that there is a strong bias towards machine learning and neural networks related conferences. As the corpus is focused on computer science conferences, and the training dataset is from the past few years (an informal inspection of the data in SciGraph yielded that roughly half of the papers in the graph are related to artificial intelligence), this topic is over-represented in our training dataset. Hence, the system is likely to create more recommendations for such conferences.

5 Conclusion

In this paper, we have introduced a recommendation system for conferences, based on abstracts, authors, and keywords.⁷ The system can be used by authors searching for upcoming conferences to publish at. The recommendations are computed based on SciGraph, with submission deadlines added from WikiCfP.

⁷ A prototype is available at <http://confrec.dws.uni-mannheim.de/>.

We have observed that the best signal for creating recommendations are keywords, in particular market codes in SciGraph, which, however, are not often easy to select for laymen users. With those keywords, a recall@10 of up to 0.665 and a MAP of up to 0.522 can be reached. Recommendations based on authors (recall@10 of 0.372 and MAP of 0.284) and abstracts (recall@10 up to 0.494, MAP up to 0.273) are clearly inferior, where the best results for the latter are obtained with TF-IDF based on word n-grams. Moreover, the good results obtained with vector space embeddings pre-trained on other text categories (e.g., news articles or Wikipedia texts) could not be reproduced on a target corpus of abstracts of scientific texts from various research fields.

References

1. Beel, J., Langer, S., Genzmehr, M., Gipp, B., Breiteringer, C., Nürnberger, A.: Research paper recommender system evaluation: a quantitative literature survey. In: International Workshop on Reproducibility and Replication in Recommender Systems Evaluation, pp. 15–22. ACM (2013)
2. Birukou, A., Blanzieri, E., Giorgini, P.: A multi-agent system that facilitates scientific publications search. In: Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 265–272. ACM Press (2006)
3. Birukou, A., Bryl, V., Eckert, K., Gromyko, A., Kaendl, M.: Springer LOD conference portal. In: International Semantic Web Conference, Posters and Demos (2017)
4. Di Noia, T., Mirizzi, R., Ostuni, V.C., Romito, D., Zanker, M.: Linked open data to support content-based recommender systems. In: Proceedings of the 8th International Conference on Semantic Systems, pp. 1–8. ACM (2012)
5. Heitmann, B., Hayes, C.: Using linked data to build open, collaborative recommender systems. In: AAAI Spring Symposium (2010)
6. Hofmann, T.: Probabilistic latent semantic analysis. In: 15th Conference on Uncertainty in Artificial Intelligence, pp. 289–296. Morgan Kaufmann (1999)
7. Janssen, W.C., Popat, K.: UpLib: a universal personal digital library system. In: ACM Symposium on Document Engineering, pp. 234–242. ACM (2003)
8. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. arXiv preprint [arXiv:1607.01759](https://arxiv.org/abs/1607.01759) (2016)
9. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751 (2014)
10. Landauer, T.K., Foltz, P.W., Laham, D.: An introduction to latent semantic analysis. *Discourse Process.* **25**(2–3), 259–284 (1998)
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
12. Osborne, F., Salatino, A., Birukou, A., Motta, E.: Automatic Classification of Springer Nature Proceedings with Smart Topic Miner. In: Groth, P., et al. (eds.) ISWC 2016. LNCS, vol. 9982, pp. 383–399. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46547-0_33
13. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)

14. Salatino, A.A., Thanapalasingam, T., Mannocci, A., Osborne, F., Motta, E.: The computer science ontology: a large-scale taxonomy of research areas. In: Vrandečić, D., et al. (eds.) ISWC 2018. LNCS, vol. 11137, pp. 187–205. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00668-6_12
15. Schmachtenberg, M., Bizer, C., Paulheim, H.: Adoption of the linked data best practices in different topical domains. In: Mika, P., et al. (eds.) ISWC 2014. LNCS, vol. 8796, pp. 245–260. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11964-9_16
16. Wolpert, D.H.: Stacked generalization. *Neural Netw.* **5**(2), 241–259 (1992)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.