New Datasets and a Benchmark of Document Network Embedding Methods for Scientific Expert Finding

Robin Brochier, Antoine Gourru, Adrien Guille, Julien Velcin

14.04.2020

Université Lyon 2, ERIC

Table of Contents

Introduction

Expert finding

- Main principle behind expertise retrieval: the search for expert candidates given a query.
- Many different kind of data can be used to tackle the expert finding task.
- In our case: a bipartite network of candidates and documents.

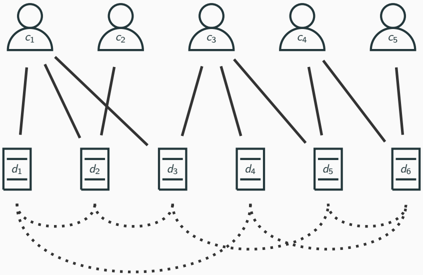

Figure 1: Example of a bipartite network composed of candidates and documents.

Contributions

- Propose 4 new datasets [1] for expert finding: 1 extracted from a scientific publication network (DBLP) and 3 from *question & answer* (*Q&A*) websites (Stack Exchange).
- Implement an evaluation methodology based on the ranking of candidates given a set of labeled document [Brochier et al., 2018].
- Explore the use of state-of-the-art document network embedding (DNE) algorithms for expert finding.
- Provide experiment results.

[1] get the data here: `https://github.com/brochier/expert_finding`

Related Works

The topic-query evaluation methodology

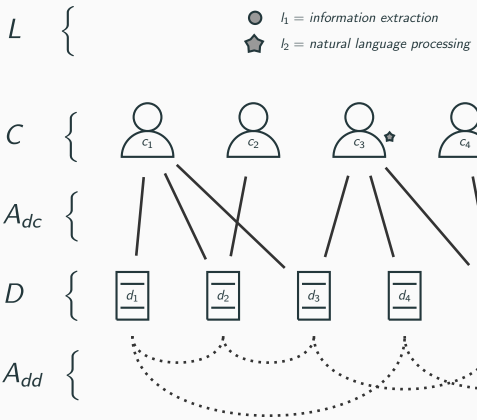

Figure 2: Hypothetical dataset for the topic-query evaluation methodology. The queries are the topic namings i.e. $q_1 = l_1$ and $q_2 = l_2$. A good algorithm would generate the following ranking: $q_1 \mapsto c_5 c_4 c_3 c_1 c_2$.

Algorithms for expert finding

- **P@noptic** Expert [Craswell et al., 2001] creates a meta-document for a candidate by concatenating the contents of all documents she is linked with.
- A **voting** model [Macdonald and Ounis, 2006] computes similarity scores between the query and the documents and aggregates them at the candidate level by using a fusion technique [Zhang et al., 2003].
- A **propagation** model [Serdyukov et al., 2008] uses random walks with restart [Page et al., 1999] to propagate the similarity scores between the query and the documents across the candidates.

Document network embedding: usual applications

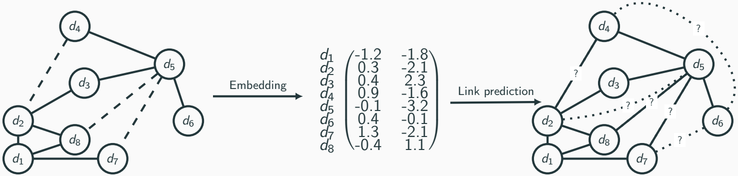

Figure 3: Link prediction with network embedding.

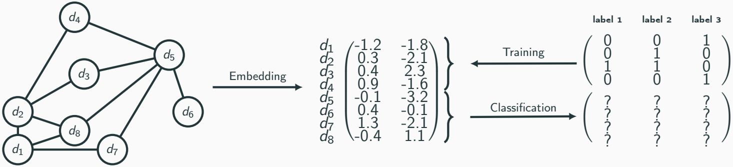

Figure 4: Node classification with network embedding.

DNE algotithms

- Text-Associated DeepWalk (**TADW**) [Yang et al., 2015] extends DeepWalk [Perozzi et al., 2014] to deal with textual attributes.
- Graph2Gauss (**G2G**) [Bojchevski and Günnemann, 2018] embeds each node as a Gaussian distribution instead of a vector.
- **GVNR-t** [Brochier et al., 2019] is a matrix factorization approach for document network embedding, inspired by GloVe [Pennington et al., 2014].
- **IDNE** [Brochier et al., 2020] uses a topic-word attention mechanism, trained from the connections of a document network.

Evaluation Methodology and Document Network Embedding

The document-query evaluation methodology

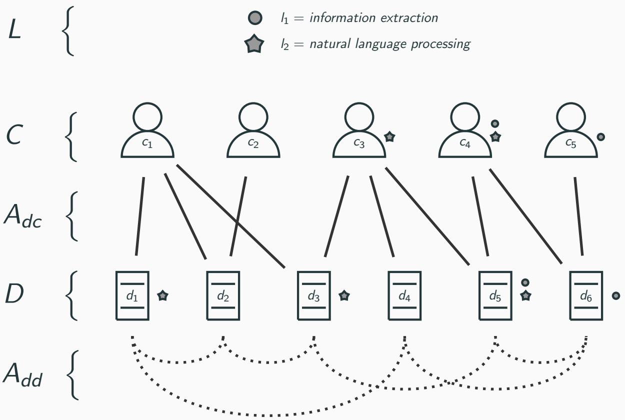

Figure 5: Hypothetical dataset for the document-query evaluation methodology. The queries are the annotated documents i.e. $q_1 = d_1$, $q_2 = d_3$, ... $q_4 = d_6$. A good algorithm would generate the following ranking: $q_1 \mapsto c_4 c_3 c_1 c_5 c_2$.

New Datasets

- For DBLP, annotations for candidates are provided by [Zhang et al., 2007] and documents are annoted by 2 PhD students of our lab.
- For Stack Exchange, we keep questions and answers with more than 10 user votes. We use tags as expertise fields and keep those that were used more than 50 times.

Table 1: General properties of the datasets.

	# candidates	# documents	# labels	# experts	# queries	label example
DBLP	707	1641	7	199	114	'information_extraction'
Stats	5765	14834	59	5765	3966	'maximum-likelihood'
Academia	6030	20799	55	6030	4214	'recommendation-letter'
Math Overflow	7382	38532	98	7382	10614	'galois-representations'

Using DNE for expert finding algorithms

- Instead of using tf-idf representations of the documents for P@noptic, the voting and the propagation models, we can use DNE algorithms.
- The document network provided to the DNE algorithm has adjacency matrix $A_d = A_{dc}A_{dc}^{\intercal} + A_{dd}$.
- It means 2 documents are connected if they have a direct link between them in the bipartite network or if they have an undirect link i.e. they have an author (candidate) in common.
- Running a DNE algorithm with input A_d and D, we obtain new document representations.

Extending DNE algorithms for expert finding (1)

Pre-aggregation scheme:

- As in the P@noptic model, meta-documents are generated by aggregating the documents produced by each candidates.
- We compute a candidate network $A_c = A_{dc}^{\intercal} A_{dc}$ and a document network $A_d = A_{dc} A_{dc}^{\intercal} + A_{dd}$.
- A meta-network is constructed:

$$A = \begin{pmatrix} A_d & A_{dc} \\ A_{dc}^{\intercal} & A_c \end{pmatrix}$$

.

- The candidate and document representations are then generated by treating this meta-network as an ordinary instance of document network.
- The scores of the candidates are generated by cosine similarity between the representation of the document-query and the representations of the candidates.

Extending DNE algorithms for expert finding (2)

Post-aggregation scheme:

- We first train the DNE algorithm on the network of documents defined by $A_d = A_{dc}A_{dc}^{\intercal} + A_{dd}$.
- A representation for a candidate is computed by averaging the vectors of all documents associated to her.
- The scores are then computed by cosine similarity.

Experiment Results

Experiment Results on DBLP

Table 2: Mean scores with their standard deviations on DBLP

	AUC	P@10	AP
random	49.47 (09.80)	05.00 (06.66)	07.09 (03.81)
panoptic (tf-idf)	74.06(12.94)	22.37 (16.35)	23.24 (12.55)
voting (tf-idf)	78.60 (11.97)	26.05 (15.76)	28.24 (13.92)
propagation (tf-idf)	79.26 (13.09)	33.07 (19.61)	34.66 (18.21)
pre-agg TADW	65.84 (12.94)	15.61 (11.63)	17.26 (08.78)
pre-agg GVNR-t	76.90 (11.46)	19.04 (11.70)	21.39 (09.61)
pre-agg G2G	72.87 (12.75)	15.70 (11.62)	18.53 (09.37)
pre-agg IDNE	78.08 (11.27)	20.18 (11.85)	22.00 (09.87)
post-agg TADW	68.01 (13.37)	16.32 (11.57)	18.01 (08.97
post-agg GVNR-t	73.91 (13.93)	18.86 (12.19)	20.57 (10.33)
post-agg G2G	68.94 (15.23)	16.23 (12.02)	18.21 (09.76)
post-agg IDNE	76.87 (13.36)	19.04 (14.57)	21.57 (10.96)
voting (IDNE)	82.23 (11.08)	34.82 (18.46)	37.27 (16.16)
propagation (IDNE)	**82.44** (16.14)	**44.47** (22.91)	**47.01** (22.06)

Experiment Results on Math Overflow

Table 3: Mean scores with standard deviations on Math Overflow

	AUC	P@10	AP
random	49.98 (01.62)	06.44 (08.28)	06.53 (03.06)
panoptic (tf-idf)	81.87 (04.46)	21.95 (19.15)	22.95 (07.54)
voting (tf-idf)	86.80 (03.23)	61.11 (18.68)	40.10 (08.27)
propagation (tf-idf)	88.08 (03.38)	**93.68** (12.16)	**49.58** (08.90)
pre-agg TADW	NA	NA	NA
pre-agg GVNR-t	65.34 (09.22)	44.02 (28.31)	16.88 (08.55)
pre-agg G2G	66.84 (08.99)	22.95 (17.81)	12.49 (05.70)
pre-agg IDNE	67.01 (09.26)	22.96 (17.84)	13.40 (06.02)
post-agg TADW	NA	NA	NA
post-agg GVNR-t	63.84 (07.59)	41.81 (22.68)	14.96 (06.25)
post-agg G2G	65.06 (09.09)	22.43 (16.94)	11.78 (05.51)
post-agg IDNE	66.74 (09.10)	21.92 (17.21)	13.11 (05.87)
voting (IDNE)	**88.71** (03.76)	68.46 (18.53)	43.53 (09.90)
propagation (IDNE)	69.38 (09.65)	92.35 (13.88)	39.62 (09.89)

Observations

- In general, the propagation model performs best.
- The aggregation schemes both perform poorly. Note that they achieve better performances on DBLP that on Stack Exchange.
- The voting model can benefit from DNE representations.
- The propagation model benefits from DNE representations only on DBLP. Its performance highly decreases on Stack Exchange data.

Conclusion

Conclusion

- The propagation model performs best since it takes fully benefit from the network. However, using document representations based on network embedding algorithms seems to corrupt the propagation.
- Neither DNE algorithms nor our aggregation schemes capture the autority of a node. This might explain their low performance.
- To bridge the gap between DNE algorithms and the expert finding task, one should (1) improve their handling of the heterogeneity of the network and (2) focus on their ability to capture the autority of the nodes.

Thank You !

Bojchevski, A. and Günnemann, S. (2018).
Deep gaussian embedding of graphs: Unsupervised inductive learning via ranking.
In *International Conference on Learning Representations*, pages 1–13.

Brochier, R., Guille, A., Rothan, B., and Velcin, J. (2018).
Impact of the query set on the evaluation of expert finding systems.
Proceedings of the 3rd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2018) co-located with the 41st International ACM SIGIR Conference.

Brochier, R., Guille, A., and Velcin, J. (2019).
Global vectors for node representations.
In *The World Wide Web Conference*, pages 2587–2593. ACM.

Brochier, R., Guille, A., and Velcin, J. (2020).
Inductive document network embedding with topic-word attention.
In *Proceedings of the 42nd European Conference on Information Retrieval Research*. Springer.

Craswell, N., Hawking, D., Vercoustre, A.-M., and Wilkins, P. (2001).
P@noptic expert: Searching for experts not just for documents.
In *Ausweb Poster Proceedings, Queensland, Australia*, volume 15, page 17.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990).
Indexing by latent semantic analysis.
Journal of the American Society for Information Science, 41(6):391–407.

Macdonald, C. and Ounis, I. (2006).
Voting for candidates: adapting data fusion techniques for an expert search task.
In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 387–396. ACM.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999).
The pagerank citation ranking: Bringing order to the web.
Technical report, Stanford InfoLab.

Pennington, J., Socher, R., and Manning, C. (2014).
Glove: Global vectors for word representation.
In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014).
Deepwalk: Online learning of social representations.
In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM.

Serdyukov, P., Rode, H., and Hiemstra, D. (2008).
Modeling multi-step relevance propagation for expert finding.
In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1133–1142. ACM.

Yang, C., Liu, Z., Zhao, D., Sun, M., and Chang, E. (2015).
Network representation learning with rich text information.
In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Zhang, J., Tang, J., and Li, J. (2007).
Expert finding in a social network.
In *International Conference on Database Systems for Advanced Applications*, pages 1066–1069. Springer.

Zhang, M., Song, R., Lin, C., Ma, S., Jiang, Z., Jin, Y., Liu, Y., Zhao, L., and Ma, S. (2003).
Expansion-based technologies in finding relevant and new information: Thu trec 2002: Novelty track experiments.
NIST SPECIAL PUBLICATION SP, 251:586–590.