

SciNeM: A Scalable Data Science Tool for Heterogeneous Network Mining

Serafeim Chatzopoulos
Univ. of the Peloponnese &
“Athena” RC
schatzop@uop.gr

Thanasis Vergoulis
“Athena” RC
vergoulis@athenarc.gr

Panagiotis Deligiannis
“Athena” RC
deligianp@athenarc.gr

Dimitrios Skoutas
“Athena” RC
dskoutas@athenarc.gr

Theodore Dalamagas
“Athena” RC
dalamag@athenarc.gr

Christos Tryfonopoulos
Univ. of the Peloponnese
trifon@uop.gr

ABSTRACT

Heterogeneous Information Networks (HINs) provide a natural way to represent various relationships between entities of different types, thus they are valuable in many domains. Extracting knowledge from HINs typically relies on the concept of metapaths, which are paths in the network schema denoting relations of different semantics among entities. Moreover, real-world HINs are often extremely large, containing millions of nodes and edges. Thus, exploring HINs not only requires interdisciplinary expertise, being able both to interpret and select appropriate metapaths in the network, but also to run the analysis in an efficient and scalable manner. Since there is a lack of tools to facilitate this task, we present SciNeM, an open source, publicly available, scalable analysis tool for metapath-based knowledge discovery in HINs.

1 INTRODUCTION

Many modern applications rely on analysing large amounts of data that comprise multiple types of entities and relationships between them. For instance, *data-driven science*, which has become a very popular and effective paradigm for scientific research, is based on computationally exploring large heterogeneous datasets. Also, the foundations of the *Fourth Industrial Revolution* heavily rely on *data science* techniques for data-driven decision making based on large heterogeneous datasets from multiple sources.

Heterogeneous Information Networks (HINs) provide a way to represent such complex information. They are graphs comprising multiple types of nodes and relationships between them [7]. An example HIN is illustrated in Figure 1, representing the interactions of genes (*G*) with a class of biomolecules called miRNAs (*M*) and their relationship with particular biological processes (*P*) and diseases (*D*)¹. It contains 4 distinct node types (*G*, *M*, *P*, *D*) and 3 distinct types of (bidirectional) relationships (*GM*, *GP*, *GD*).

Various data science methods to analyse HINs and facilitate knowledge discovery from them have been proposed [6, 8, 11]. These typically rely on the concept of *metapaths*: paths in the HIN schema that represent types of entity relationships with particular semantics. For instance, two interesting metapaths in the HIN of Figure 1 are *GPG* and *MGDGM*. The former connects genes based on the processes they are involved in (i.e., strongly connected genes based on it may share common functionalities).

¹Note that some edges have been added for presentation purposes and may not reflect real relationships.

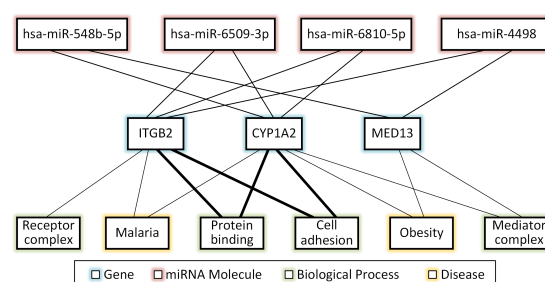


Figure 1: An example HIN.

The latter links miRNAs based on the diseases which relate to the genes with which they interact.

Many HIN analysis algorithms use metapaths as input; the metapath-based connectivity can be used to define a measure for node *similarity search* [8] or *similarity join* [11] or to rank nodes based on their centrality in a metapath-defined network [6]. In the previous example, using the metapath *GPG* for similarity join could reveal that genes *ITGB2* and *CYP1A2* are similar since they are involved in two common processes. Moreover, to further elaborate the analysis, it is often useful to apply constraints to a given metapath (e.g., in the previous example, to consider only metapath instances involving Cell adhesion).

Despite the wide applicability of HINs and the plethora of proposed algorithms in the literature, there is still a lack of (a) open-source, scalable implementations of these methods, and (b) tools to facilitate their use by non-experts. Also, implementing metapath-based analysis of HINs on top of a graph database, such as Neo4j, requires significant programming skills and familiarity with the system’s native query language; also, certain important features of Neo4j, including distributed execution, are only available in the Enterprise Edition. As a first attempt to fill this gap, we have recently developed SPHINX [2]; however, SPHINX is mainly tailored to similarity search and does not offer parallel and distributed execution that is required to scale to larger HINs.

In this work, we introduce SciNeM² (Data Science tool for heterogeneous Network Mining), an open-source³ tool that offers a wide range of functionalities for exploring and analysing HINs and utilises Apache Spark for scaling out through parallel and distributed computation. SciNeM provides an intuitive, Web-based user interface to build and execute complex constrained metapath-based queries and to explore and visualise the corresponding results. Under the hood, all the supported state-of-the-art HIN analysis types have been implemented in a scalable

²<http://scinem.imsi.athenarc.gr>

³<https://github.com/schatzopoulos/SciNeM>

manner supporting the distributed execution of analysis tasks on computational clusters. SciNeM has a modular architecture making it easy to extend it with additional algorithms and functionalities. Currently, it supports the following operations, given a user-specified metapath: ranking entities using a random walk mode, retrieving the top- k most similar pairs of entities, finding the most similar entities to a query entity, and discovering entity communities.

2 SYSTEM OVERVIEW

2.1 Architecture & Functionalities

Figure 2 illustrates the key components of SciNeM’s architecture, as well as the data flow between them. All (back-end) components have been implemented on top of Apache Spark to allow scalable execution on computational clusters. In the following paragraphs, we elaborate on their functionality and implementation.

2.1.1 Distributed HIN Storage. This is SciNeM’s main storage layer. It is responsible for the storage of all HIN data and it is based on a Hadoop Distributed File System (HDFS) hosted on the storage media of the underlying computational cluster. Each HIN consists of a set of files including (a) a *schema file*, that describes the HIN node types and the types of relationships between them (compatible with Cytoscape’s Elements JSON format⁴), (b) *node files* in TSV format containing data attributes for the nodes of each type, and (c) *relationship files* that define the edges of the network. User-created HINs can be uploaded to this storage layer via the Web front-end.

2.1.2 HIN Transformation. Most metapath-based analysis types, like those discussed in Section 2.1.3, require a common pre-processing step that transforms the initial heterogeneous network to a homogeneous (or bipartite) one. This network is essentially a *view* of the HIN containing only the nodes of the first (or the first and last, respectively) entity type in the metapath and having one edge for each metapath instance connecting these entities. Further analysis is performed on the aforementioned HIN view.

The HIN Transformation component implements this pre-processing step. It takes as input a user-defined metapath and a set of constraints and identifies all pairs of nodes that are connected based on this constrained metapath. For each pair, it also captures the number of metapath instances that connect the corresponding nodes.

Since the calculation of the metapath-based view is a computationally intensive task, special care was taken for the efficient implementation of this component. The core of transformation is calculated using matrix multiplication between the adjacency matrices defined by the relations of the given metapath. Specifically, our approach is based on the work in [6] but extends it by utilising sparse matrix representations. Since the order of multiplications significantly affects the performance of the whole processing, we adopt a dynamic programming approach that estimates the optimal ordering taking into consideration the computational cost of sparse matrix multiplications introduced in [4]. This modification offers significant speedups in many cases. In addition, the implementation of this component utilises Apache Spark, thus taking advantage of parallel and distributed computing.

2.1.3 Metapath-based analysis. This component implements a range of metapath-based mining tasks for HINs. In particular, state-of-the-art methods for entity ranking, similarity join,

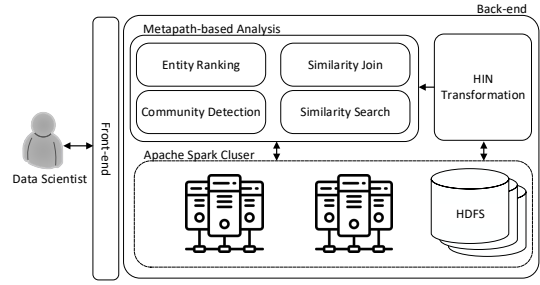


Figure 2: Architecture of SciNeM.

similarity search, and community detection are implemented, as explained below.

Given a particular constrained metapath, the *Entity Ranking* component estimates the significance of entities according to a random walk model applied to the corresponding HIN view [6]. In particular, the PageRank score of each node in the HIN view is calculated, and the corresponding entities are ranked based on these scores. The intuition is that this procedure brings as top-ranked results nodes which are well-connected inside the metapath-based view, i.e., nodes that correspond to entities which are important according to the semantics of the selected constrained metapath (this is why so many other nodes connect to them). To guarantee scalability, a high-performance Spark-based ranking component has been developed, allowing the analysis of very large HINs.

The *Similarity Join* component identifies the most similar pairs of nodes based on the way they are linked with other nodes when considering a particular (possibly constrained) metapath. As this type of analysis is computationally intensive, SciNeM leverages Locality Sensitive Hashing [3] (LSH) using Bucketed Random Projection to prune expensive similarity calculations. For each node, a feature vector is constructed based on its connectivity on the metapath-based HIN view. These vectors are then hashed into buckets, so that vectors that are similar end up in the same bucket with high probability. A similar approach is also followed by other relevant works (e.g., [11]).

The *Similarity Search* component detects nodes that are similar to a given query node. The notion of similarity used is the same as the one used by the Similarity Join component. In more details, SciNeM performs an approximate nearest neighbors search using the Euclidean Distance to determine (dis)similarity between nodes. Moreover, the same hashing technique as in Similarity Join is used to effectively prune the search space. Furthermore, it should be noted that, to improve scalability of the performed analyses, the Similarity Search and Join components have been implemented based on Apache Spark.

Finally, the *Community Detection* component identifies communities (i.e., clusters) of interacting nodes/entities based solely on the structural properties of the selected metapath-based HIN view, that is produced by the HIN Transformation component (see Section 2.1.2). The analysis is based on the Label Propagation Algorithm (LPA) [5], which is a popular community detection approach that requires no a priori knowledge about the network’s structure. It is based on propagating labels throughout the network and forming the communities following the intuition that labels will be trapped and become dominant in clusters of densely connected nodes. Although this type of analysis is less intensive

⁴<https://cytoscape.org/>

(a) Analysis task submission form.

(b) Constraint selection pop-up window.

Figure 3: Screenshots from submitting a Ranking analysis on the BIO dataset, using the MGDGM metapath with the constraint D.name=‘Colorectal Cancer’

than other approaches (e.g., Fast-Greedy, Infomap), for large networks it requires significant computational power and has a very large memory footprint. This is why the corresponding component of SciNeM takes advantage of a Spark-based, distributed implementation of the algorithm.

2.1.4 Web front-end. SciNeM’s Web UI supports determining and executing metapath-based analysis tasks. These can be executed on already available HINs or new ones uploaded by the user. A *visual wizard* used to determine the details of the desired analysis tasks lies at the core of this component (see also Section 2.2). The front-end was implemented using React⁵ JS library assisted with Redux⁶ state container for efficient state management. Graph visualisations (e.g., HIN schema visualisation for the query builder) were implemented using the Cytoscape JS library.

2.2 User Interface

Figure 3a presents a screenshot of SciNeM’s analysis task submission form. To perform a new analysis, the user first selects an existing HIN from the corresponding drop-down menu or uploads a new one. The latter requires uploading a single compressed file that contains the files described in Section 2.1.1⁷.

After selecting the input HIN, the user specifies the metapath to be used for the analysis and the desired constraints. To assist the user in selecting metapaths, an interactive version of the schema of the HIN is displayed in the submission form. The user can either click on the entity types (nodes) of the schema to incrementally build the desired metapath, or add extra entity types by selecting them from a drop-down list after clicking on the green button located at the end of the currently selected sequence. To select the desired constraints, the user can click on

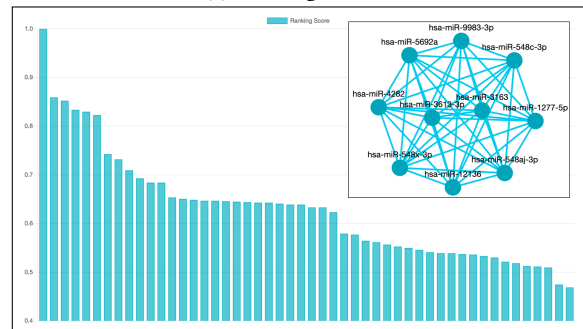
⁵<https://reactjs.org/>

⁶<https://redux.js.org/>

⁷Details can also be found in SciNeM’s dataset upload page.

Rank	MIRNA name	Ranking Score
1	hsa-miR-548c-3p	1.0
2	hsa-miR-9983-3p	0.859094
3	hsa-miR-3613-3p	0.852244

(a) Ranking results.



(b) Visualisations of ranking results.

Figure 4: Screenshots from the results of the analysis of Figure 3.

the filter icon located below the involved entity type. A pop-up window will appear on the screen (see Figure 3b). The user selects the desired constraints and then hits the ‘Save’ button.

Finally, the user selects the types of analysis to be performed (multiple can be selected simultaneously) and clicks on the ‘Execute analysis’ button⁸. A progress bar appears in the screen (see at the bottom of Figure 3a) to monitor the status of the execution. Moreover, a unique identifier is assigned to each analysis so that the user can return to the analysis using the option ‘Reattach to analysis’ from SciNeM’s navigation bar.

After the analysis is completed, the results appear in a tabular form (see Figure 4a). The user can browse them or select to download all or part of them. She also has the option to select some of them to create a condition file, i.e. a special file in JSON format that encodes them into a set of constraints that can be used in a later analysis. In particular, after creating a condition file the user can provide it as input in a later analysis by clicking on the ‘Load from file’ button of the constraints pop-up window (see Figure 3b). Essentially this creates a mechanism to use the results of an analysis as input to a subsequent one.

Finally, apart from the tabular results, SciNeM also provides a set of visualisations. The user can click on the ‘Visualize’ button, located above the list of results, and select the visualization type to display (for the cases for which more than one visualization type is provided). Figure 4b displays examples of such visualisations, in particular a bar chart showing the distribution of ranking scores in the top ranking results of Figure 4a and a graph showing the part of the corresponding metapath-based HIN view that contains the top-10 results (the node sizes are based on the corresponding ranking scores).

⁸It should be noted that for all similarity search analysis tasks the user should also determine the search entity before starting the execution.

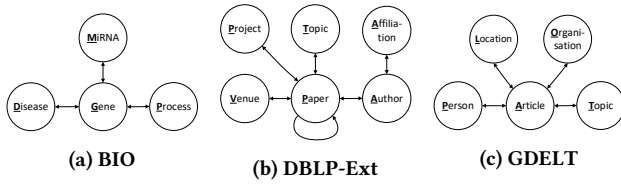


Figure 5: Schema definitions of pre-loaded HINs.

3 DEMONSTRATION

During the demonstration, the audience will have the opportunity to become familiar with the concepts of metapath-based analysis in HINs and to interact with SciNeM’s user interface exploring its functionalities. The members of the audience will be able to execute their own analysis tasks and, if needed, to upload their own HINs. However, to facilitate examining SciNeM’s capabilities, three datasets have been already prepared and made available:

- **BIO.** It contains data about the involvement of genes in biological processes and diseases (based on GeneOntology [1, 10] and DisGeNET⁹, respectively). It also contains data about the suppression of genes by miRNAs (provided by MR-microT¹⁰). It includes 4 entity types (see Figure 5a), containing a total of 61, 177 nodes and 4, 190, 808 edges.
- **GDELT.** It contains data for news articles and associated entities collected by the GDELT project¹¹. In particular, we have collected articles published in 2019 from BBC and CNN. GDELT consists of 5 entity types (see Figure 5c) totaling 245, 950 nodes and 6, 523, 924 relationships.
- **DBLP-Ext.** This HIN contains bibliographic data from the DBLP Citation Dataset of AMiner [9] enriched with european project data from the Cordis project¹². It contains 6 entity types with 12, 152, 816 nodes and 190, 998, 307 relationships. DBLP-Ext’s schema is presented in Figure 5b.

Based on these HINs, four indicative scenarios have been prepared for demonstration. Short descriptions of them follow:

Scenario 1: Important miRNAs for a disease (Ranking). Although the involvement of genes in biological processes and diseases is relatively well-studied, this is not the case for the role of miRNAs. Yet, it is possible to reveal a miRNA’s role based on the list of genes it suppresses. Using SciNeM on the BIO dataset, a member of the audience can reveal miRNAs having important role in ‘Colorectal Cancer’ by selecting to rank miRNAs based on the MGDGM metapath using the *D.name* = ‘Colorectal Cancer’ condition. Highly ranked entities have large centrality in the corresponding HIN view, thus they are highly connected through metapath instances that satisfy the determined condition about the disease of interest. A search in PubMed reveals that there are various articles mentioning in their abstract and/or title both the top retrieved miRNA (‘miR-548c’) and disease of interest.

Scenario 2: Similar venues to a given one based on the topics of their published papers (Similarity Search). A member of the audience is interested in finding similar venues to the “Very Large Data Bases” (VLDB) conference, according to the topics of their recent papers. As a result, she selects to perform a similarity search on the DBLP-Ext dataset using the *VPTPV* metapath and the constraint *P.year* > 2000. The top results include very relevant venues like the “Int’l Conference on Data Engineering”

(ICDE) in the first position and the “Int’l Conference on Management of Data” (SIGMOD) in the second position.

Scenario 3: Communities of organizations based on article mentions (Community Detection). A member of the audience is interested in revealing clusters of related organizations (e.g., governmental institutions, companies) based on their mentions in the news articles of an international network source like CNN. To do so, she chooses to perform community detection on the GDELT dataset using the *OAO* metapath with the *A.source* = “cnn” constraint. The results contain various interesting communities; e.g., the one with *id* = 111 consists of 3 institutions having an agenda related to climate change (“UN Intergovernmental Panel on Climate”, “European Union Copernicus Climate Change Programme”, “World Meteorological Organization”), whereas the one with *id* = 394 includes 7 institutions involved in politics in India.

4 CONCLUSION

We demonstrated SciNeM, an open source, high-performance and scalable online data science tool that facilitates metapath-based analysis of HINs. Its intuitive user interface aids non-experts to perform a variety of HIN analysis tasks such as metapath-based ranking, similarity join, similarity search, and community detection. Finally, SciNeM’s users may upload their own HIN datasets to analyse, however the tool also provides pre-loaded datasets for demonstration reasons.

ACKNOWLEDGMENTS

This work was partially funded by the EU H2020 project SmartDataLake (825041).

REFERENCES

- [1] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. 2000. Gene Ontology: tool for the unification of biology. *Nature Genetics* 25, 1 (2000), 25–29.
- [2] S. Chatzopoulos, K. Patroumpas, A. Zeakis, T. Vergoulis, and D. Skoutas. 2020. SPHINX: A System for Metapath-based Entity Exploration in Heterogeneous Information Networks. In *Proceedings of the 46th International Conference on Very Large Data Bases (VLDB 2020)*.
- [3] P. Indyk and R. Motwani. 1998. Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality. In *Proc. of the Thirtieth Annual ACM Symposium on Theory of Computing (STOC '98)*. 604–613.
- [4] D. Kernert, F. Köhler, and W. Lehner. 2015. SpMacho - Optimizing Sparse Linear Algebra Expressions with Probabilistic Density Estimation. In *Proc. of the 18th International Conference on Extending Database Technology, EDBT 2015, Brussels, Belgium*. 289–300.
- [5] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E* 76, 3 (2007), 036106.
- [6] C. Shi, Y. Li, P. S. Yu, and B. Wu. 2016. Constrained-meta-path-based ranking in heterogeneous information network. *Knowl. Inf. Syst.* 49, 2 (2016), 719–747.
- [7] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. S. Yu. 2017. A Survey of Heterogeneous Information Network Analysis. *IEEE TKDE* 29, 1 (2017), 17–37.
- [8] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. 2011. PathSim: Meta Path-Based Top-K Similarity Search in Heterogeneous Information Networks. *PVLDB* 4, 11 (2011), 992–1003.
- [9] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. 2008. ArnetMiner: Extraction and Mining of Academic Social Networks. In *Proceedings of the 14th ACM SIGKDD*. ACM, 990–998.
- [10] The Gene Ontology Consortium. 2018. The Gene Ontology Resource: 20 years and still GOing strong. *Nucleic Acids Research* 47, D1 (11 2018), D330–D338.
- [11] Y. Xiong, Y. Zhu, and P. S. Yu. 2015. Top-k Similarity Join in Heterogeneous Information Networks. *IEEE Trans. Knowl. Data Eng.* 27, 6 (2015), 1710–1723.

⁹<https://www.disgenet.org>

¹⁰<http://diana.imis.athena-innovation.gr/DianaTools/index.php?r=mrmicrot/>

¹¹<https://www.gdeltproject.org>

¹²<https://cordis.europa.eu>