

NDlib: A Python Library to Model and Analyze Diffusion Processes over Complex Networks

Giulio Rossetti
KDD Lab. ISTI-CNR
Pisa, Italy
giulio.rossetti@isti.cnr.it

Letizia Milli
University of Pisa
Pisa, Italy
letizia.milli@di.unipi.it

Salvatore Rinzivillo
KDD Lab. ISTI-CNR
Pisa, Italy
salvatore.rinzivillo@isti.cnr.it

ABSTRACT

Nowadays the analysis of dynamics of and on networks represents a hot topic in the Social Network Analysis playground. To support students, teachers, developers and researchers we introduced a novel framework, named NDlib, an environment designed to describe diffusion simulations. NDlib is designed to be a multi-level ecosystem that can be fruitfully used by different user segments. Upon NDlib, we designed a simulation server that allows remote execution of experiments as well as an online visualization tool that abstracts its programmatic interface and makes available the simulation platform to non-technicians.

KEYWORDS

spreading phenomena simulation; diffusion models' library; simulation platform

ACM Reference Format:

Giulio Rossetti, Letizia Milli, and Salvatore Rinzivillo. 2018. NDlib: A Python Library to Model and Analyze Diffusion Processes over Complex Networks. In *WWW '18 Companion: The 2018 Web Conference Companion, April 23–27, 2018, Lyon, France*. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3184558.3186974>

1 INTRODUCTION

In the last decades Social Network Analysis (SNA) has received increasing attention from several, heterogeneous fields of research. Undoubtedly, such pervasiveness has produced an amplification in the visibility of network analysis studies thus making this complex and interesting field one of the most widespread among higher education centers, universities and academies. Given the exponential diffusion reached by SNA, several tools were developed to make it approachable to the wider audience possible.

Although being a very active field of research per se, SNA is often used as a tool to analyze complex phenomena such as the unfolding of human interactions, the spreading of epidemic and diffusion of opinions, ideas, innovations. Even for such peculiar applications, we have witnessed during the last years the appearance of dedicated tools and libraries: however, the plethora of resources available often discourage the final users making hard and time-consuming the identification of the right tool for the specific task and level of expertise.

This paper is published under the Creative Commons Attribution 4.0 International (CC BY 4.0) license. Authors reserve their rights to disseminate the work on their personal and corporate Web sites with the appropriate attribution.

WWW '18 Companion, April 23–27, 2018, Lyon, France

© 2018 IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC BY 4.0 License.

ACM ISBN 978-1-4503-5640-4/18/04.

<https://doi.org/10.1145/3184558.3186974>

To cope with such issue, we introduced a novel framework designed to model, simulate and study diffusive phenomena over complex networks [1, 2].

NDlib represents a multi-level solution targeted to epidemic spreading simulations.

One of the main contribution of the proposed tool regards the heterogeneous audience it is intended to serve. Unlike its competitors, NDlib is designed to provide access to diffusion models to both technicians (e.g., programmers or researchers coming from quantitative disciplines) and non-technicians (e.g., students, social science researchers, analysts). To fulfil such goal it offers, aside from the standard programmatic interface, a web based visual one able to abstract from the low level description and definition of diffusion simulation. Moreover, NDlib comes with an experiment server platform that allows to easily set up controlled exploratories and to decouple experiment definition and execution.

NDlib is intended to overcome some limitations identified in existing libraries (e.g., providing support for dynamic network topologies) while reducing the overall usage complexity.

2 NDlib: NETWORK DIFFUSION LIBRARY

We organized our framework in three incremental modules: the NDlib core library (written in Python), a remote REST-ful experiment server accessible through API calls and, finally, a web oriented visual interface.

In this section we will describe and discuss the major characteristics of our library, as implemented in v3.x.

2.1 Library Rationale

At the core of our tool there is NDlib, acronym for “(N)etwork (D)iffusion Library”, a Python package built upon the network facilities offered by NetworkX¹. The library, available for Python 2.7.x and 3.x, is currently hosted on GitHub², on pypi³ and has its online documentation on ReadTheDocs⁴. Moreover, NDlib is also made available through the SoBigData.eu catalog⁵.

Currently, NDlib offers to its users 16 models coming both from epidemics and opinion dynamics literature. Moreover, it provides support to novel diffusion models definition and testing.

Diffusive phenomena are modeled as discrete-time agent-based processes. Given a network $G = (V, E)$, a diffusion model M , and the actual state of its nodes, S_i , the request of a diffusion iteration will

¹NetworkX: <https://goo.gl/PHXdnl>

²NDlib GitHub: <https://goo.gl/zC7p7b>

³NDlib pypi: <https://goo.gl/gc96xW>

⁴NDlib docs: <https://goo.gl/VLWtrn>

⁵SoBigData: <http://www.sobigdata.eu>

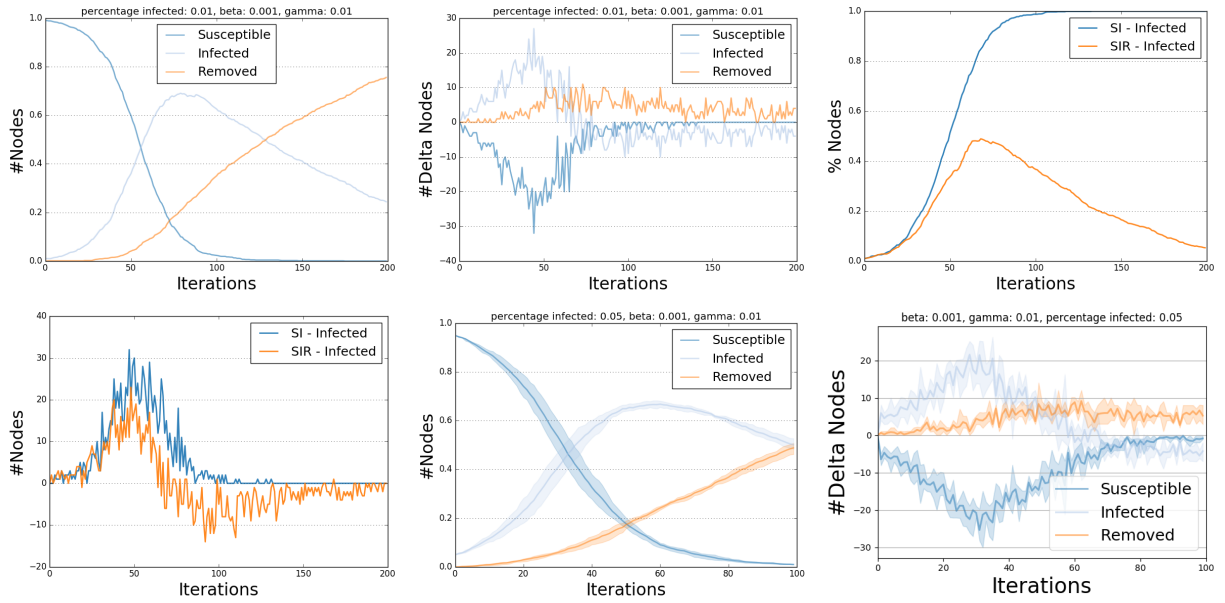


Figure 1: Visual Analysis. First row: (left) DiffusionTrend, (center) DiffusionPrevalence, (right) DiffusionTrendComparison plots. Second row: (left) DiffusionPrevalenceComparison, (center-right) Multiple Parallel execution plots.

return a novel nodes' state S_{i+1} obtained by applying the evolution rules specified by M to all the nodes in V .

To do so, the library model generic diffusion process as composed of three elements: (i) the graph G on which the process takes place; (ii) the diffusion model M ; (iii) the model configuration and initial infection state S_0 .

2.1.1 Visualization Facilities. To allow the final user to easily analyze simulation evolution, NDLIB exposes a set of predefined visual facilities.

It allows, for instance to describe the unfold of diffusive processes through Diffusion Trend plots (Figure 1, first row, left) as well as through Diffusion Prevalence ones (Figure 1, first row, right).

Moreover, NDLIB allows to easily compare the executions of different models: Figures 1, second row, illustrate examples of comparison plots built over the results of SI and SIR models for the *Infected* node class.

2.1.2 MODEL PARALLEL EXECUTIONS. NDLIB, with a single line of code, allows to execute in parallel, the simulation of the same model over a given network topology while varying the initial infection status to study its stability.

In case of multiple executions, all NDLIB plotting facilities are augmented to show both average trends as well as their variations (as in Figure 1 second row, center and right) – a solution that allows for a better understanding of model stability.

2.2 Experiment Server

In some scenarios – e.g., due to limited computational resources or to the rising of other particular needs – it may be convenient to separate the machine on which the definition of the experiment is made from the one that actually executes the simulation. To satisfy

such needs, we developed a RESTful service, NDLIB-REST⁶, that builds upon NDLIB an experiment server accessible through API (Application Programming Interface) calls.

The simulation web service is designed around the concept of *experiment*. An experiment, identified by a unique identifier, is composed of two entities: (i) a network and (ii) one (or more) configured models. Experiments are used to keep track of the simulation definition, to return consecutive model iterations to the user and to store - locally on the experiment server - the current status of the diffusion process.

NDLIB-REST is shipped also as a Docker⁷ container so to make also it easier to setup. Moreover, the simulation server is, by default, executed within a Gunicorn⁸ instance allowing parallel executions of multiple experiments at the same time. NDLIB-REST is built using Flask⁹ and offers a standard online documentation page that can also be directly used to test the exposed endpoints both configuring and running experiments.

In order to provide a standard interface to the experiment server NDLIB-REST comes with a Python wrapper for the exposed API that allows to embed remote calls in analytical scripts.

2.3 Web Platform

Finally, on top of NDLIB-REST is built NDLIB-Viz¹⁰. NDLIB-Viz provides the same facility of the NDLIB library to those users that prefer to avoid writing code. It employs the same “Experiment” concept introduced by NDLIB-REST and allows to simulate several models

⁶NDLIB-REST: <https://goo.gl/c6yHcY>

⁷Docker: <https://www.docker.com/>

⁸Gunicorn: <http://gunicorn.org/>

⁹Flask: <http://flask.pocoo.org/>

¹⁰Available, currently as a separate branch, at: <https://goo.gl/tYi480>

Network Diffusion Library

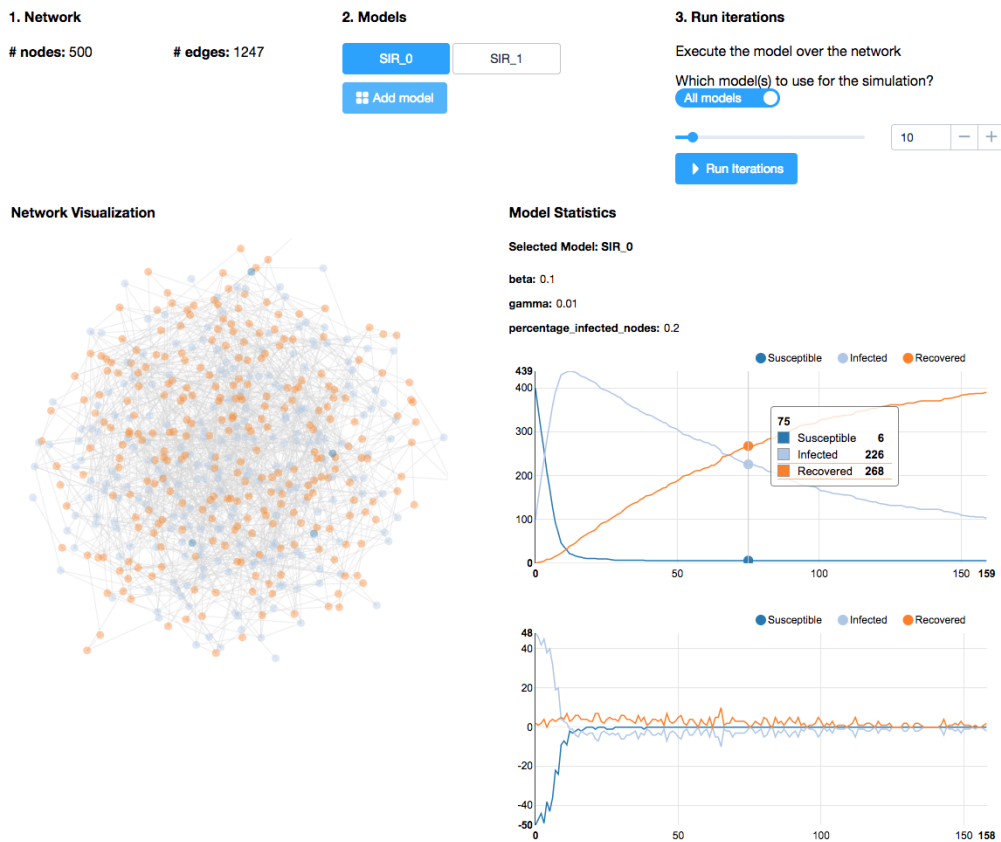


Figure 2: Web Platform. NDLIB Visualization Framework appearance during a simulation. The top toolbar presents a schematic workflow to execute a simulation. The left view presents a visualization of the status of each node. The right part presents a synthetic visualization of properties of the simulation. Mouse interaction allows the user to select a specific time instant of the simulation to update all the other views accordingly

(or different instantiation of the same model) at the same time comparing how they unfold over a given graph. As a first step, the user should create an experiment and a network. Once the network has been created, it is rendered on the screen in a viewport.

At the second step, the user may instantiate one or more diffusion models to attach to the network. Each model is simulated according to the specifics of NDlib-REST. After that the user can choose the number of iterations to execute. The left view (see Figure 2) presents a visualization of the status of each node. Each node is assigned a color to represent its status in a specific time instant of the simulation. The right part shows the result of simulation for each model. An aggregated visualization of each model is presented in a block containing the reference to the model and its parameters. Two charts to show DiffusionTrend and DiffusionPrevalence plots. Exploiting the web interface, the plots are interactive. Exploring the

plots with the mouse the user may receive additional information on specific time instant.

3 DEMO PRESENTATION

NDLIB is a modular framework designed to provide easy access to network diffusion simulation models to a broad user base. Such characteristic makes our library applicable to several analytical contexts: from educational purposes (e.g., employing NDLIB-Viz), diffusion model definition and testing (e.g., leveraging its core library), to the definition of a remote experiment on massive networks (e.g., setting up an experimental server on a dedicated machine).

In our demo we aim to introduce the NDLIB ecosystem and to discuss some of its main characteristics. We plan to deliver our demo through the live execution of some simple diffusion models exploiting the library, the remote interface and visual service. Our only hardware requirement are a projector and internet connection.

A succinct example of the contents of the demo is available as video on youtube¹¹.

In our demo we plan to introduce the context of network diffusion analysis, categorize the audience we want to reach and highlight the specificities of our library. Leveraging Jupyter Notebooks, we will show some simulations of diffusion models performed with NDLIB. In the process, we will discuss how to define, configure and execute an experiment as well as how to describe novel custom diffusive models.

As a second step, we will present the RESTful service, NDLIB-REST, simulating a remote execution of diffusion experiments. Finally, we will showcase NDLIB-Viz, the web based visual platform designed to abstract NDLIB technical complexity.

ACKNOWLEDGMENTS

This work is funded by the European Community's H2020 Program under the funding scheme "FETPROACT-1-2014: Global Systems Science (GSS)", grant agreement # 641191 CIMPLEX "Bringing Citizens, Models and Data together in Participatory, Interactive Social EXploratories"¹². This work is supported by the European Community's H2020 Program under the scheme "INFRAIA-1-2014-2015: Research Infrastructures", grant agreement #654024 "SoBigData: Social Mining & Big Data Ecosystem"¹³.

REFERENCES

- [1] Giulio Rossetti, Letizia Milli, Salvatore Rinzivillo, Alina Sirbu, Dino Pedreschi, and Fosca Giannotti. 2017. NDlib: a python library to model and analyze diffusion processes over complex networks. *International Journal of Data Science and Analytics* (20 Dec 2017). <https://doi.org/10.1007/s41060-017-0086-6>
- [2] Giulio Rossetti, Letizia Milli, Salvatore Rinzivillo, Alina Sirbu, Dino Pedreschi, and Fosca Giannotti. 2017. NDlib: Studying Network Diffusion Dynamics. In *IEEE International Conference on Data Science and Advanced Analytics, DSAA*. Tokyo.

¹¹<https://www.youtube.com/watch?v=tYHNOuKJwbE&feature=youtu.be>

¹²CIMPLEX: <https://www.cimplex-project.eu>

¹³SoBigData: <http://www.sobigdata.eu>