Optimising and expanding PREMONition: A tool for creating and analysing sparse functional association networks

Introduction

In some biological research projects, after uncovering a list of genes that are likely involved in causing a disease, it is often challenging to identify which genes to continue with in further research. There are many available methods and analyses that provide insights on the importance of genes, but not all are as relevant or valid to use in different contexts. Jasper Bosman created the PREMONition (PREdicting MOlecular Networks) tool in Python, which creates sparse protein-protein interaction networks, as part of his PhD research on circadian clocks in microbes [1]. The goal of this internship project was to optimise PREMONition and expand it, by including more methods and analyses as sources for creating functional association networks (FANs). In addition, a web application was also to be developed, making usage of the tool easy for everybody and to provide an interactive visualisation of the networks with information from automatically run analyses available.

Materials and Methods

PREMONition is written in Python 3.11.6 and uses the NetworkX package [2] version 3.2 for graph/network storage and interaction, Pandas version 2.1 is used for loading input files and filtering data. After loading and filtering the data, multiple preprocessing steps are run, removing a large part of nodes and edges in static unnecessary scenarios. When this is done the core part of PREMONition runs, which uses the PageRank algorithm [3] and scores based on different input sources (illustrated in Figure 1) to iteratively filter out nodes, until all the nodes that can be removed are removed and the network is sparse.

The accompanying web application is made in Python with Django version 4.2 as the backend, Tailwind CSS in combination with DaisyUI for frontend styling and ReactJS with Cytoscape.js for the interactive network visualisation. Executing the Python tool is done by first sending a task to Redis, which acts as a message broker, and then Celery detects and performs the task, running the tool and communicating back to Django.

Results

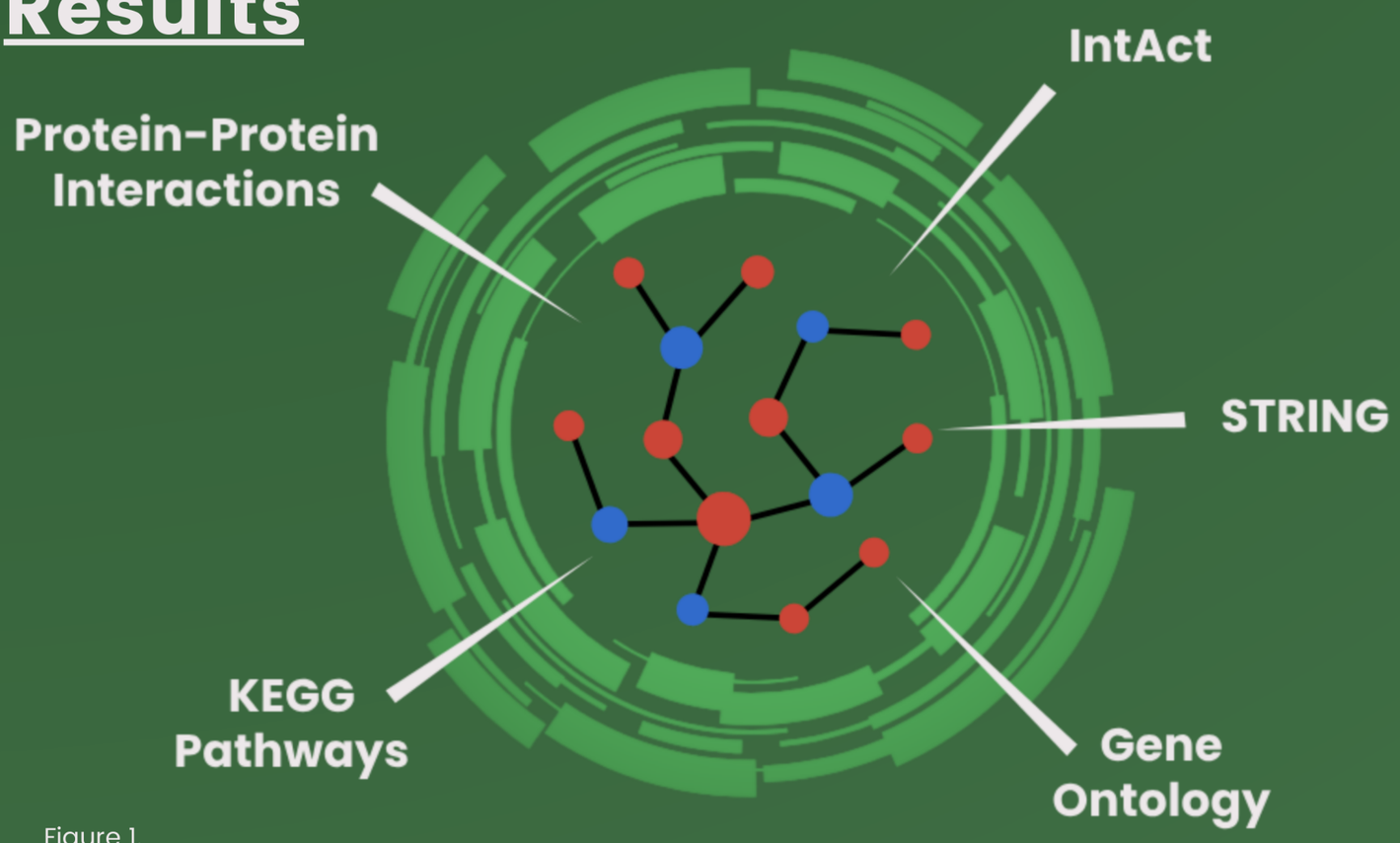
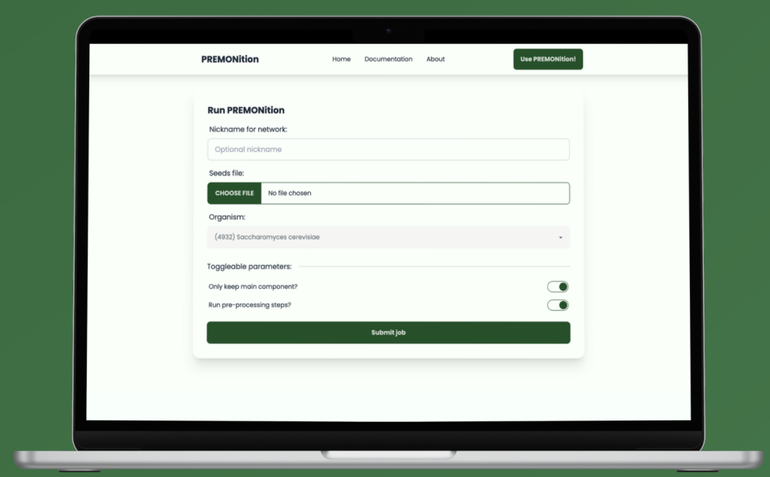


Figure 1.
Visual representation showing a simple functional association network and multiple sources they can be based on and created with. Blue nodes are proteins added by PREMONition in order to connect and include as many red nodes (input genes) in an as big as possible fully connected network.

The Python tool can be run from the command line, be imported as a python package or be used through the developed web application. Sparse FANs with only a single connecting node (protein) between input genes are created more than 10 times faster than the original PREMONition.



Discussion and Conclusion

This internship project primarily created a foundation to continue building further on. The core python code of PREMONition does not actually properly use multiple sources for scoring the nodes to keep yet. The web is also still rather simple, with only the functionality to submit a job to run the tool and the results page with the visualisation of the network.

Future Perspectives

For future development of the core python code, a dynamic scoring formula will be implemented that uses the multiple sources and which is also user-configurable. The web application will get functionality to overlay KEGG pathways and Gene Ontology terms, but also a user login system which will greatly improve user experience.

Hanze

Student
Vincent Talen
v.k.talen@st.hanze.nl
389015

Supervisor Research Centre
Jasper Bosman
Supervisor Hanze
Ronald Wedema

[1] Bosman, J. (2021). Discovering circadian clocks in microbes. [Thesis fully internal (DIV), University of Groningen]. University of Groningen. https://doi.org/10.33612/diss.170828314
[2] Hagberg, A., Swart, P., & S Chult, D. (2008). Exploring network structure, dynamics, and function using NetworkX.
[3] Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank Citation Ranking: Bringing Order to the Web. The Web Conference.