Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

# The BioGrakn Disease Network

A. Messina, U. Maniscalco, P. Storniolo

***Rapporto Tecnico N.:***
**RT-ICAR-PA-2019-02** **Dicembre 2019**

Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR) –
Sede di Cosenza, Via P. Bucci 41C, 87036 Rende, Italy, URL: *www.icar.cnr.it*
– Sede di Napoli, Via P. Castellino 111, 80131 Napoli, URL: *www.na.icar.cnr.it*
– Sede di Palermo, Viale delle Scienze, 90128 Palermo, URL: *www.pa.icar.cnr.it*

Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

# The BioGrakn Disease Network

A. Messina[1], U. Maniscalco[1], P. Storniolo[1]

***Rapporto Tecnico N.:***
**RT-ICAR-PA-2019-02** **Dicembre 2019**

[1] Istituto di Calcolo e Reti ad Alte Prestazioni, ICAR-CNR, Sede di Palermo, Via Ugo La Malfa n. 153, 90146 Palermo.

# Index

# 1 Abstract

The field of systems biology is characterized by a huge amount of heterogeneous data, hard to integrate due to their complex nature and rich semantics.

One of the key goals in this scope is understanding the complex relationships among these biological data and, certainly, we need solutions to speed up their integration and querying.

Anyhow, analyzing large volumes of biological data through traditional database systems is troublesome and challenging.

In this work, we demonstrate how using a semantic knowledge graph for complex biological relationships, such as *BioGrakn Disease Network* (BioGraknDN), would accelerate the knowledge discovery process.

# 2 The BioGrakn Disease Network (BioGraknDN)

## 2.1 Motivation and significance

Nowadays, the availability of many analytical tools in biomedical science has produced a lot of information about all sorts of biological components (tissues, diseases, cells, proteins, drugs, pathways, etc.) and their functions.

Of course, these components are important individually, but we also need to understand their biological characteristics in relation to the potential interactions they have with each other, and this requires the integration of vast amounts of heterogeneous, highly complex and semantically rich data.

For example, some genes may be linked to multiple diseases, encoding many proteins, and could include some source information. Recognizing the relationships between such entities can be decisive in providing the biological context for hypothesis generation and validation.

As shown in Figure 1, before producing any insight (e.g., drug discovery), the first step typically implicates the integration of different biomedical data. These are often available from public or proprietary data sets, while others can be extracted through text mining methods from sources, such as PubMed articles.

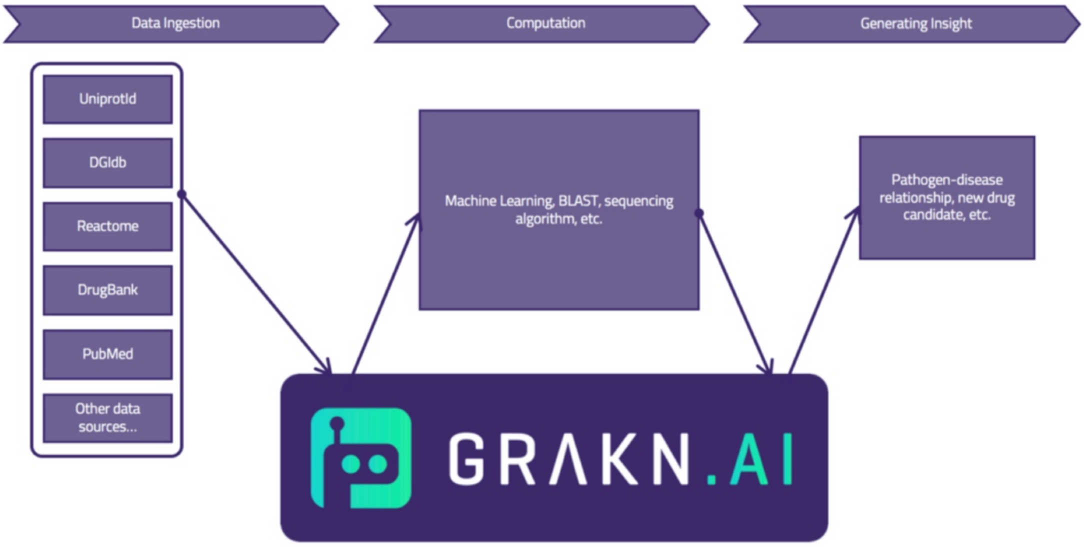


*Figure 1 - A general work-flow: from data ingestion to insights generation*

Data sets like UniProt, ENSEMBL, Drug Bank, etc., usually come in textual formats (TSV/CSV) and, as this is flat data, we need to connect these data sets so that the networks can be established.

Unfortunately, source biological data is not always uniform. For example, we can have some well-annotated proteins because they have been studied experimentally, but there also are ‘hypothetical proteins’ having little to no information.

Furthermore, since biological research gives us unpredictable results, new studies are constantly available and their outcomes could be integrated into our work-flow. In other words, we are dealing with a challenging, time-consuming, and difficult-to-scale process.

Once our data sources have been integrated somehow, a computational layer can produce some form of insight. For instance, we could predict a particular pattern using machine learning, or a sequencing algorithm could help us to find sequence similarities between genes or proteins.

Even though this insight may extend the previously ingested data, without a complex integration process, we would still have an incomplete biological context. For example, if a new drug candidate for a disease is suggested, we can't know its interactions with other biological entities. Also, if data are stored into a traditional relational database, we need to use too much complex and expensive join queries to pull out interesting results.

*BioGraknDN* can let us improve two areas of intervention:

- Data integration and ingestion: a hard-coded script to integrate flat data, no matter if in memory or into a relational database, is rigid and time-consuming, particularly when we are going to add new data sources. Furthermore, navigating and analyzing data can be computationally too expensive due to its inherent complexity.

- Biologically contextualizing newly generated insights: it will be hard to associate the data produced by, for example, machine learning algorithm or a sequencing, because the data integration process is characterized by an inflexible nature. Therefore, interactions with other biological components will lose their biological contextuality.

The problems above are addressed by using Grakn (1), which is an intelligent database that can organize complex networks of data in the form of a knowledge graph.

Entities and relationships are the concepts of the systems, while rules can be used to perform automated reasoning.

The principles of knowledge representation and reasoning are implemented in a type system schema, a more expressive and useful system than traditional relational and NoSQL databases when we have to manage large-scale linked data.

## 2.2 Software description

*BioGraknDN* is a Grakn knowledge graph, derived from (2), that integrates the following data sources:

- UniProt KnowledgeBase (UniprotKB) (3): annotated functional information on human proteins;

- Reactome (4): validated human metabolic pathways, annotated as a set of biological events and linked to proteins;

- Drug Gene Interaction Database (DGIdb) (5): approved drug compounds and their links with proteins;

- DisGeNET (6): curated subset for diseases and mapped to UniProt identifiers using gene names;

- HPA-Tissue (7): data on gene-expression-tissue enhanced associations;

- EBI IntAct (8): protein-protein interaction data;

- Gene Expression Omnibus (9): Three studies were integrated and differentially expressed genes (DEGs) were identified between asthma subtype/control cohorts using the limma Bioconductor package;

- TissueNet (10): associations between human tissues and protein-protein interactions;

Figure 2 shows a simple schema that represents the above data sets. The Grakn Python client was used to load this data from their TSVs and CSVs into *BioGraknDN*.

Having created BioGraknDN, there are several reasons that make this process so easy to do with Grakn. These include:

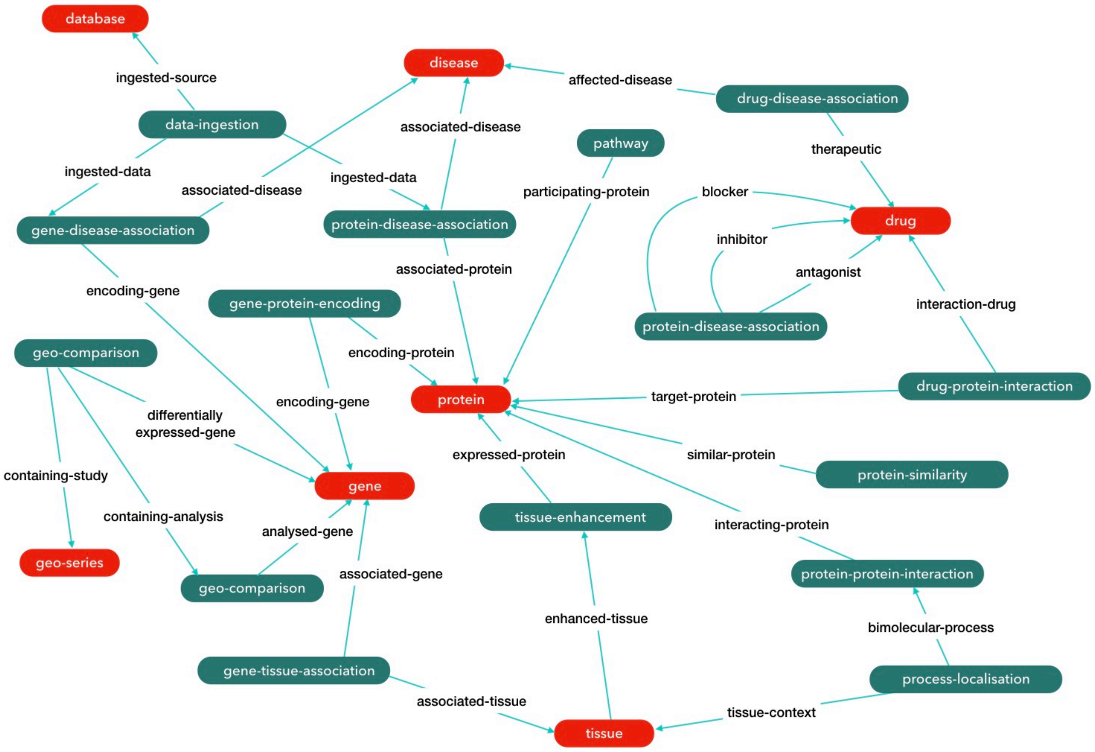
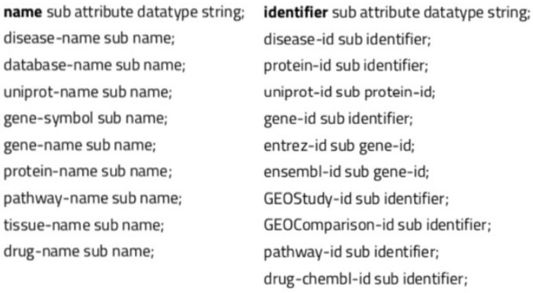


*Figure 2 - The database schema used in BioGrakn Disease Network*

1. *Graql*, the Grakn's flexible and expressive language, allows us to modify the schema as new datasets are ingested. This meant, for example, that when we need to insert drug-gene relationships from DGIdb, we could change the schema and create different roles (inhibitor, antagonist, and blocker) for drugs when interacting with a gene, and this avoids from having to create new relationship types for each role. If we were using a traditional relational database, implementing such a change would have meant a re-design of the schema, which can be a costly and complicated process.

2. Thanks to the type system implemented in Grakn, relationships types and attributes can be hierarchically modeled, to enable easier querying afterward. For example, the attribute's identifier and name are created as the parent type of all other identifiers and names. Figure 3 shows how they look like.



```
name sub attribute datatype string;
disease-name sub name;
database-name sub name;
uniprot-name sub name;
gene-symbol sub name;
gene-name sub name;
protein-name sub name;
pathway-name sub name;
tissue-name sub name;
drug-name sub name;

identifier sub attribute datatype string;
disease-id sub identifier;
protein-id sub identifier;
uniprot-id sub protein-id;
gene-id sub identifier;
entrez-id sub gene-id;
ensembl-id sub gene-id;
GEOStudy-id sub identifier;
GEOComparison-id sub identifier;
pathway-id sub identifier;
drug-chembl-id sub identifier;
```

*Figure 3 - Definition of parent types for other identifiers and names*

This means that any type of identifier can be queried for without having to explicitly state if, for example, an entrez-id or ensembl-id exists. Then, to query for entrez-id *29851* and to ask to be returned its gene symbol:

```
match $g isa gene, has identifier "29851", has gene-symbol $gs;
  get $gs; {$gs val "ICOS" isa gene-symbol;}
```

3. Furthermore, when we need to insert tissue-PPI relationships from TissueNet, we can use *hyper-relationships* to express this concept. The tissue entity is modeled as having a process-localization relationship with the protein-protein-interaction relationship, that is a relationship inside another relationship. Such level of expressivity is quite useful as it means the model can be designed more closely guided by the needs of its application.

Once the data has been integrated into *BioGraknDN*, we will be able to do some form of complex computation, such as sequencing or machine learning algorithm.

The next section shows how, thanks to *BioGraknDN*, the insights generated in these computations can be linked to a biological context.

## 2.3 Illustrative examples: Bringing biological context to newly generated insight

If we run a sequencing or ML algorithm, we can create a new type of insight, that can be extremely valuable. But to go further in the knowledge discovery process, we should also provide a biological context to that insight, integrating it into *BioGraknDN*.

The benefits can be outlined as follows:

1. A sequencing algorithm can be used to found similarities between sequences of proteins, and these can be inserted as sequence similarity relationships between two protein entities. For example, Figure 4 shows how we would model a sequence similarity between proteins with uniprotID *P09238* and *P39900*.

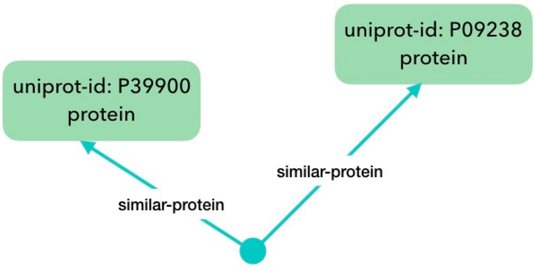


*Figure 4 - Example of sequence similarity between proteins*

2. At this point, Grakn rules can be defined in the schema to find out new insights from the data. The following sample rule will create a new drug-disease-association relationship when two proteins with a sequence similarity are found, where one protein is a target for a disease, and the other relates to a drug:

```
when {
    $di isa disease;
    $pr isa protein;
    $pr2 isa protein;
    $pr != $pr2;
    $dr isa drug;
    (associated-disease: $di, associated-protein: $pr)
        isa protein-disease-association;
    (similar-protein: $pr, similar-protein: $pr2)
        isa protein-similarity;
    (target-protein: $pr2, interacted-drug: $dr)
        isa drug-protein-interaction;
} then {
    (affected-disease: $di, therapeutic: $dr)
        isa drug-disease-association;
};
```

Thanks to this rule, we can now immediately query for drug-disease-association relationships, even without any inserted drug-disease data. The above rule lets Grakn infer for us and find candidate drugs and such a query would look like as:

```
match
  $di isa disease, has disease-name "Asthma";
  $dr isa drug;
  $r (affected-disease: $di, therapeutic: $dr)
    isa drug-disease-association;
  get;
```

Figure 5 shows the result returned when we look for potential candidate drugs against the disease *Asthma*, and we see that the drug *PHENYTOIN* may be a potential candidate against *Asthma*. No direct association exists in the data, actually, because these relationships were inferred by Grakn engine.

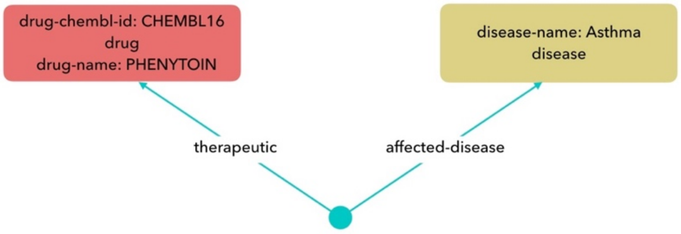


*Figure 5 - A potential candidate drug related to a disease*

If we double click on the relationship, the graph expands as shown in Figure 6, where we see that protein *Q969D9* is associated to *Asthma* and has a sequence similarity with *P01889*, which has a relationship with the drug *PHENYTOIN*.

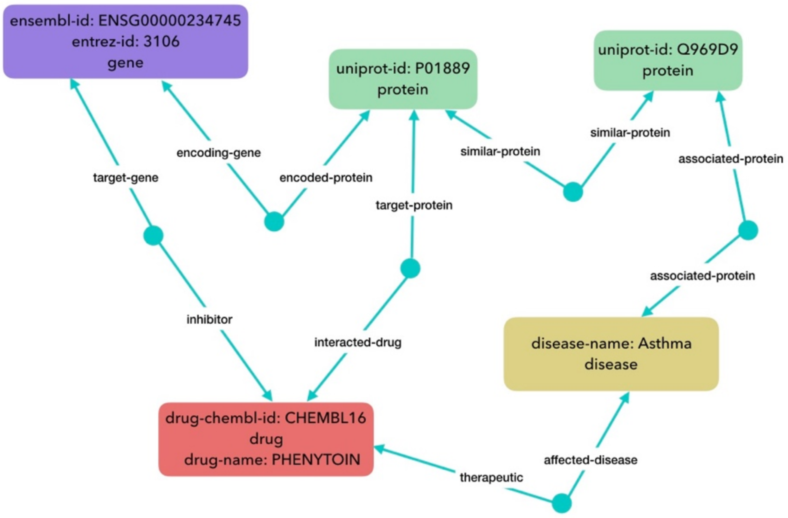


*Figure 6 - Expanded view of a relation between a potential candidate drug and a disease*

Because none of the original datasets included protein-drug associations, that relationship is also inferred through another rule, which states that: "*if a disease is associated with a gene, that disease should also be associated with the proteins which that gene encodes*". Therefore, as *PHENYTOIN* has been reported to be an inhibitor to the gene with entrez-id *3106*, the protein it encodes, *P01889*, gets also associated with it. Exploring such transitive relationships will be of great interest to drug development research.

3. If we want to explore the biological context of new insights and compare the network of neighborhoods of biological components, traversal type queries can help us a lot. These queries can reveal paths connecting components that may not have been at first anticipated. Such queries are done using Graql without difficulty, but would be computationally too expensive for a traditional relational database because multiple joins are needed. A Graql query (results in Figure 7) that asks for connections

between *Asthma*, the heart muscle, proteins, and drugs, could be:

```
match
  $di isa disease, has disease-name "Asthma";
  $ti isa tissue, has tissue-name "heart muscle";
  $dr isa drug; $pr isa protein;
  $pda (associated-disease: $di, associated-protein: $pr)
    isa protein-disease-association;
  $te (expressed-protein: $pr, enhanced-tissue: $ti)
    isa tissue-enhancement;
  $dpi (target-protein: $pr, interacted-drug: $dr)
    isa drug-protein-interaction;
limit 20;
get;
```

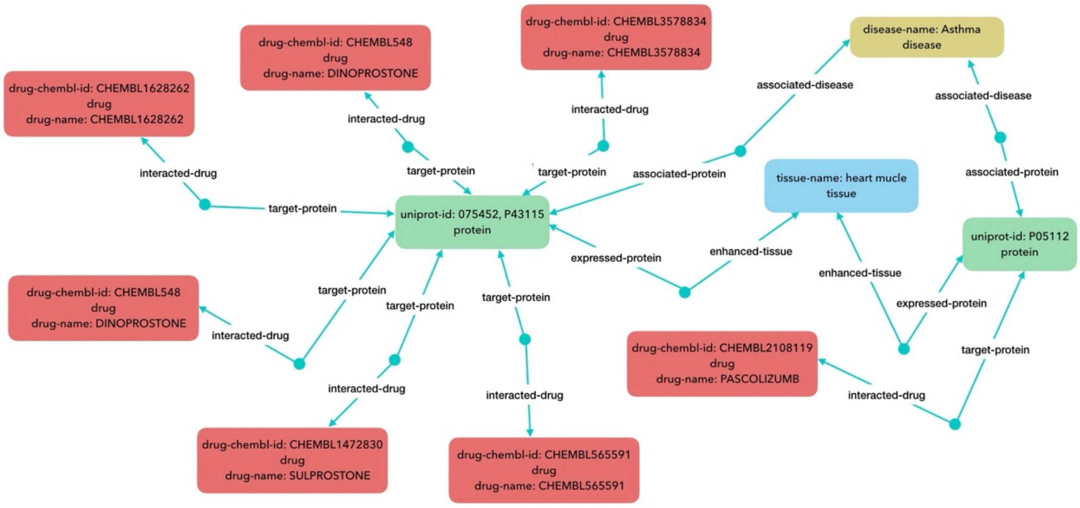


*Figure 7 - Visualization of the above query, where red nodes are drugs, greens are proteins, blue is the tissue heart muscle, and yellow represents the disease Asthma*

4. Also, we may want to query and identify proteins related to a disease from one particular study and explore how it relates to diseases from other studies. For example, to find proteins that are common to Asthma (11) and are also associated to other diseases, we can ask for all protein-disease-association relationship that are associated with the Kaneko database entity:

```
match
  $di isa disease, has disease-name "Asthma";
  $pr isa protein; $di2 isa disease;
  $db isa database, has database-name "Kaneko";
  $di2 != $di;
  $pda (associated-disease: $di, associated-protein: $pr)
    isa protein-disease-association;
  $di (ingested-source: $db, ingested-data: $r)
    isa data-ingestion;
  $pda2 (associated-disease: $di2, associated-protein: $pr)
    isa protein-disease-association;
limit 10;
get;
```

A possible result is shown in Figure 8.

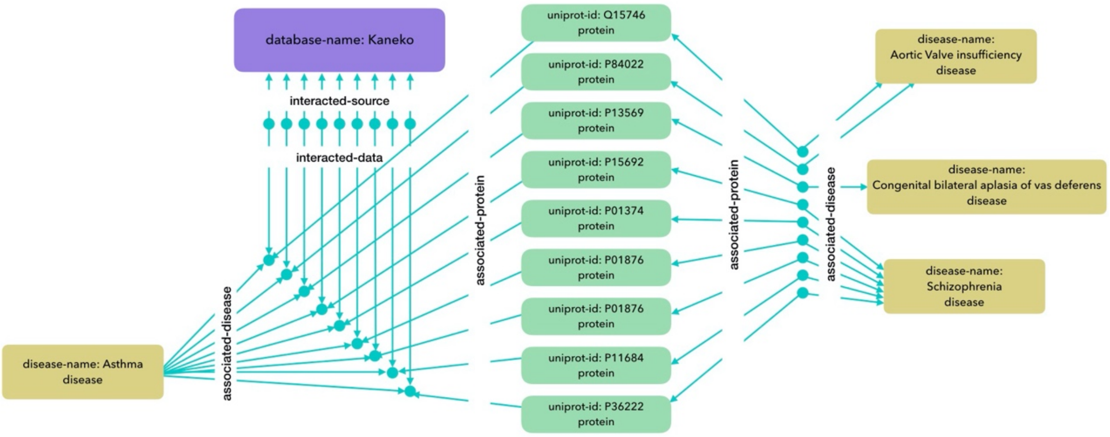


*Figure 8 - The purple node represents the Kaneko data scource, yellow nodes are diseases and greens are proteins*

5. In general, shortest path queries allow us to find the nearest connections between certain nodes. Here, we can find the nearest connections between biological components. For example, the shortest path between the protein P38398 and Asthma is displayed in Figure 9.

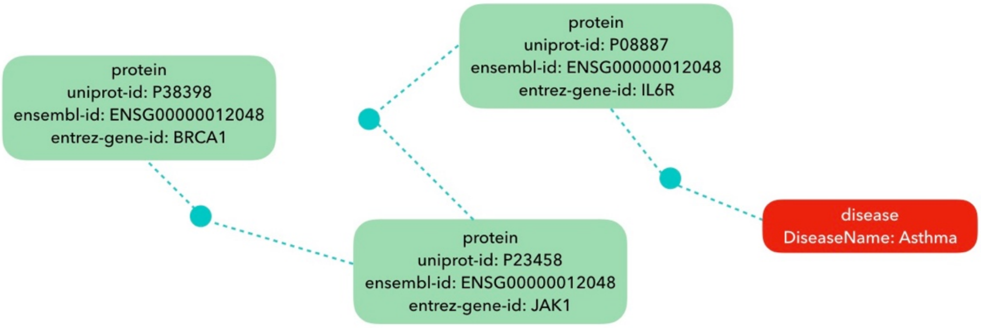


*Figure 9 - The shortest path between the protein P38398 and Asthma*

## 2.4 Impact

In summary, there are two areas where *BioGraknDN* and Grakn can help in bioinformatics:

- Ingesting and integrating biomedical data: *BioGraknDN* and Grakn facilitate the quick ingestion and integration of new data types that may come out due to the unpredictable nature of biomedical research and the dynamic and constantly changing requirements of biomedical communities. The available schema language, the hierarchical model, and hyper relationship provide us a level of expressivity to model not uniform biomedical data.

- Bringing biological context to newly generated insight: *BioGraknDN* and Grakn also allow an easy ingestion of new insights and data generated through sequencing or machine learning algorithms, so they can be understood in their biological context. New candidate drugs may be suggested by inferred relationships between unconnected biological components.

## 2.5 Conclusions

Since recent improvements in omics technologies have created a lot of genome-wide scanning data, bioinformatics must continue innovating to find new techniques for effective and scalable analysis of biological data. In this paper, we presented *BioGraknDN* and demonstrated how we could accelerate the knowledge discovery process in biomedical research thanks to a Grakn knowledge graph.

## 2.6 Notes

For a better readability, figures from 4 to 9 show graphic results of queries which were hand-drawn by replicating the true graphical results generated by the Grakn workbench. You can find examples of typical graphic results in the repository page of the project, at

https://github.com/crss-lab/biograkn/tree/master/diseasenetwork

# 3 References

1. **Grakn Labs Ltd.** Grakn. [Online] https://grakn.ai.

2. **Messina, Antonio, et al.** A Knowledge Graph-Based Semantic Database for Biomedical Sciences. *Advances in Intelligent Systems and Computing.* s.l. : Springer, 2017, pp. 299-309.

3. *UniProt: a hub for protein information.* **The UniProt Consortium.** D1, 2014, Nucleic Acids Research, Vol. 43, pp. D204-D212.

4. *The Reactome pathway Knowledgebase.* **Fabregat, Antonio, Sidiropoulos, Konstantinos and Garapati, Phani.** D1, 2015, Nucleic Acids Research, Vol. 44, pp. D481-D487.

5. *DGIdb 3.0: a redesign and expansion of the drug–gene interaction database.* **Cotto, Kelsy, et al.** D1, 2017, Nucleic Acids Research, Vol. 46, pp. D1068-D1073.

6. *DisGeNET: a comprehensive platform integrating information on human disease-associated genes and variants.* **Pinero, Janet, et al.** D1, 2016, Nucleic Acids Research, Vol. 45, pp. D833-D839.

7. *Tissue-based map of the human proteome.* **Uhlen, Mathias, et al.** 6220, 2015, Science, Vol. 347.

8. *The MIntAct project--IntAct as a common curation platform for 11 molecular interaction databases.* **Orchard, Sandra, et al.** Database issue, 2014, Nucleic acids research, Vol. 42, pp. D358-D363.

9. *NCBI GEO: archive for functional genomics data sets—update.* **Barret, Tanya, et al.** D1, 2012, Nucleic Acids Research, Vol. 41, pp. D991-D995.

10. *The TissueNet v.2 database: A quantitative view of protein-protein interactions across human tissues.* **Basha, Omer, et al.** D1, 2016, Nucleic Acids Research, Vol. 45, pp. D427-D431.

11. *The search for common pathways underlying asthma and COPD.* **Kaneko, Y., et al.** 2013, International journal of chronic obstructive pulmonary disease, Vol. 8, pp. 65-78.