GRAKN.AI

Precision Medicine Knowledge Graphs

One of the biggest challenges in our current state of medicine is to provide relevant, personalised and precise diagnoses and treatments. Rather than treating all patients the same, the goal is to fully take into account a person's demographics and genetic profile while treating or diagnosing them.

In a nutshell, the current problem is that a large number of drugs and treatments prescribed to patients do not treat the individual patient but the generic disease. This is something doctors are well aware of — not all treatments affect every patient in the same way. Yet for decades, the strategy of trial and error is still being used to a large extent to treat and diagnose. Not the most reassuring of thoughts.

However, there is hope. Due to recent advancements in bioinformatics and genomic sequencing, precision medicine (also known as personalised medicine) is slowly becoming a reality rather than a dream.

Precision medicine is a nascent field focused towards disease prevention and treatment while taking into account individual variability in genes, environment and lifestyle for each person. It aims at integrating vast data sets to build predictive and preventive models of complex versatile diseases. Precision medicine may simply mean prescribing the right drug to the right patient, with the right dose at the right time. Or, it may entail matching patients to the relevant therapeutic treatments which are relevant to their individual biological make up.

The benefits that this brings us mean that precision medicine is getting us much closer to e.g., curing cancer and predicting & preventing chronic diseases. In doing so, we may also be saving trillions of dollars to the health industry.

Though promising, there are still many problems with Precision Medicine. Some of these are related to the handling of data needed to personalise medicine. Let us look at these problems and see how they can be solved using Grakn.

Challenges with Precision Medicine

When doing precision medicine, we are faced with a myriad of challenges — right from obtaining the correct and relevant data sources, all the way to validating our treatments and diagnoses. With this in mind, I have been particularly interested in how precision medicine data should be handled. And having looked at the state of the art, I observed the following challenges:

1. **Integrating heterogeneous biological data is difficult**
   The first challenge starts with the raw data. This data, which pertains to biological concepts and relationships, is scattered all over the world and is produced at an unprecedented rate by a multitude of institutions in various formats and standards. This is also known as the big data disruption paradigm, where high throughput data pipelines are creating bottlenecks to the analysis and processing of that data. It is extremely difficult to ingest and integrate these multi-format and disparate data sets.

2. **Normalising raw complex biological data is difficult**
   The second challenge stems from the fact that the raw data contained in these data sets have no structure. This lack of structure makes it difficult to maintain and assure integrity, accuracy and consistency over this data. It also causes a lack of control over the validity of data when integrating such heterogeneous data sources.
3. **Discovering new and valuable insights is difficult**
   Finally, due to the heterogeneous nature of the biological data necessary for precision medicine, it becomes extremely tedious to generate or investigate insights in a scalable way. Of course, one valuable insight might be discovered manually for a single instance of a disease or patient, but such an approach is impossible to scale across thousands of patients. Moreover, in many cases, a manual approach may actually be impossible. What do we do then?

What are the Solutions these Challenges?

With this in mind, we can think of potential solutions that address these challenges. Based on my research, the below is what I suggest:

1. **Integrating heterogeneous biological data into one single database**
   To solve the disparateness of heterogeneous data, we need a method to easily accumulate patient profiles and relate biological data into one collection —in other words, we need a knowledge graph.
2. **Normalising biological data using a contextual structure**
   To enable the intelligent analysis and integration of such data — while maintaining data integrity — we need to impose an explicit structure on the concepts contained in that data. This will not only help to contextualise the concepts themselves, but also the relationships between them. This translates to having some sort of high-level data model to encompass the various types of data and consolidate their presence in the knowledge graph. This will also allow us to validate the data at the time of ingestion.
3. **Discovering new insights using automated reasoning**
   In order to extract or infer as much information as possible from our knowledge graph, we need some sort of automated reasoning tool to propagate our domain expertise throughout the entirety of the data. This will enable us to ask questions from our knowledge graph and get the right answers — while other traditional methods would fail.

Having identified a template to the solutions of the previously listed challenges, I wondered whether there was any one technology out there, that encompassed all three points?

Well, to my luck, ***Grakn*** solves all of these.

If you're unfamiliar with this technology, Grakn is an intelligent database in the form of a knowledge graph to organise complex networks of data. It contains a knowledge representation system based on hyper-graphs; enabling the modelling of every complex biological relationship. This knowledge representation system is then interpreted by an automated reasoning engine, which performs reasoning in real-time. This software gets exposed to the user in the form of a flexible and easily understood query language — **Graql**.

GRAKN.AI

How do we Build a Precision Medicine Knowledge Graph?

But how do we actually go about building a precision medicine knowledge graph using Grakn?

Identifying the Right Data

Every data driven system needs to start with the data itself. As such, the first step is to identify the right types of data sources we need in order to go about personalising medicine.

There are two types of data we need. First, the data related to the personalised aspect, which may include the demographic and medical profile of the patients we plan on examining. These may include:

1. Genes
2. Variants
3. Medical History
4. Age
5. Gender
6. Ethnicity
7. Electronic Health Records

But in order to investigate how the patient's data correlates with medicine we also need other types of biomedical data:

1. Diseases
2. Drugs
3. Clinical Trials
4. Medical Literature

Once we have the raw data we want for our application, we need to find reliable sources where we can retrieve this data from. The following list exhibits some of the sources that may be used to download the raw data pertaining to precision medicine:

1. NCBI
2. DisGeNet
3. OncoKB
4. DrugBank
5. PubMed
6. ClinicalTrails.gov
7. ClinGen
8. PHARMGKB
9. ClinVar
10. Drugs@FDA

Normalising Raw Data

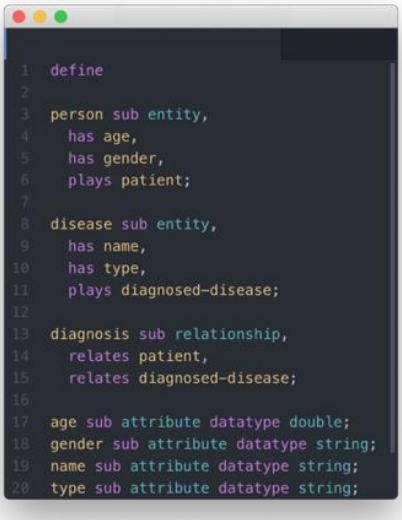
Now that we have raw data, we need to tackle the second problem; that of normalisation. To this end, Grakn utilises the entity-relationship model to group each concept into either an entity, attribute, or relationship. This means that all we have to do is to map each concept to a schema concept type, and

recognise the relationships between them. Let us look at an example to demonstrate how we would go about doing this:

First, let us assume we the following two raw data sets representing instances of people in one data set, and another about diseases:

In the “People” dataset, we can see that the fourth column refers to “disease”, giving us a diagnosis about the diseases a person has. In other words, the raw data shows a relationship between a person, who is linked with a disease through a diagnosis:

This structure can be represented using Graql as follows:



```
define

person sub entity,
  has age,
  has gender,
  plays patient;

disease sub entity,
  has name,
  has type,
  plays diagnosed-disease;

diagnosis sub relationship,
  relates patient,
  relates diagnosed-disease;

age sub attribute datatype double;
gender sub attribute datatype string;
name sub attribute datatype string;
type sub attribute datatype string;
```

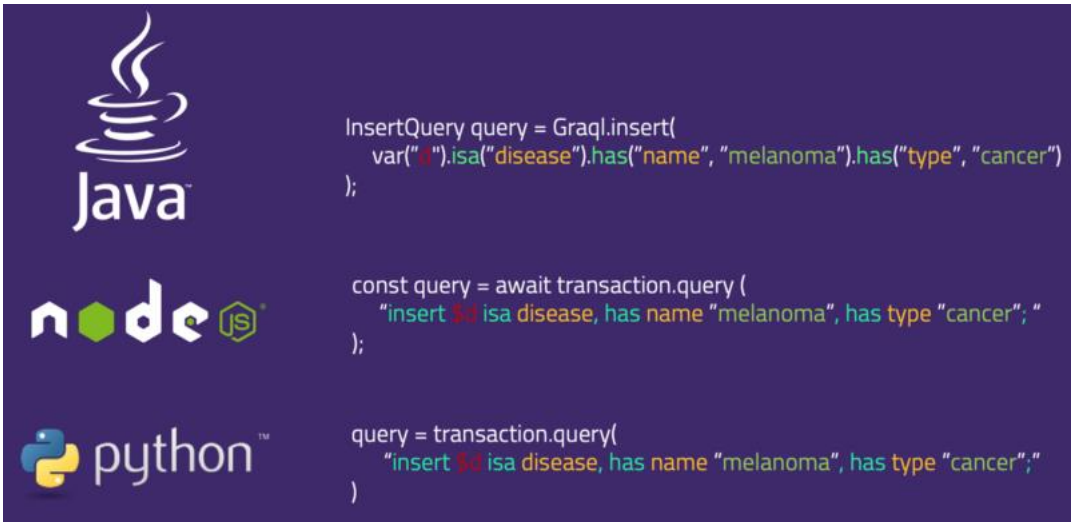
We recognised a person and a disease as an entity, having certain attributes. Then, we defined a relationship between the disease and person that we called diagnosis: where the role-players in the relationships are the patient and diagnosed-disease.

In order to constrain the attributes of each concept we also need to denote what data type they adhere to. In this case, we defined four attributes: one attribute with datatype double, and three of string type.

Migrating the Raw Data into Grakn

Now that we have the data, and a structure imposed on this data, the next step is to migrate this into Grakn. Please note there are many different ways to do migration, but here I would like to specifically touch on how we would go about using Java, NodeJS and python.

For this, we can easily use any of these languages to read/parse the raw data file and iterate over each entry within those files. The image below depicts how to insert a single instance of a disease with a name and type into Grakn using any of these three languages:



To learn more about migrating data into Grakn, make sure to read this article: Modelling & Migrating Big Biological Data with Grakn.

GRAKN.AI

Discovering a New Relevant Therapy for Patients

After migration, we can start to discover new insights. Discovering insights refers to finding new data that may be valuable to what we are trying to accomplish. In order to do that, we need to first look or ask for something. In other words, we start with a question — the questions we ask to find answers to in precision medicine. These questions can range from asking if a particular individual's profile is susceptible to any disease, or asking what treatments are relevant to an individual's biological make-up.

Let us look at an example, and see how our precision medicine knowledge graph may provide answers to them:

Question: Given a person suffering from melanoma; what clinical trial could be recommended to her/him?

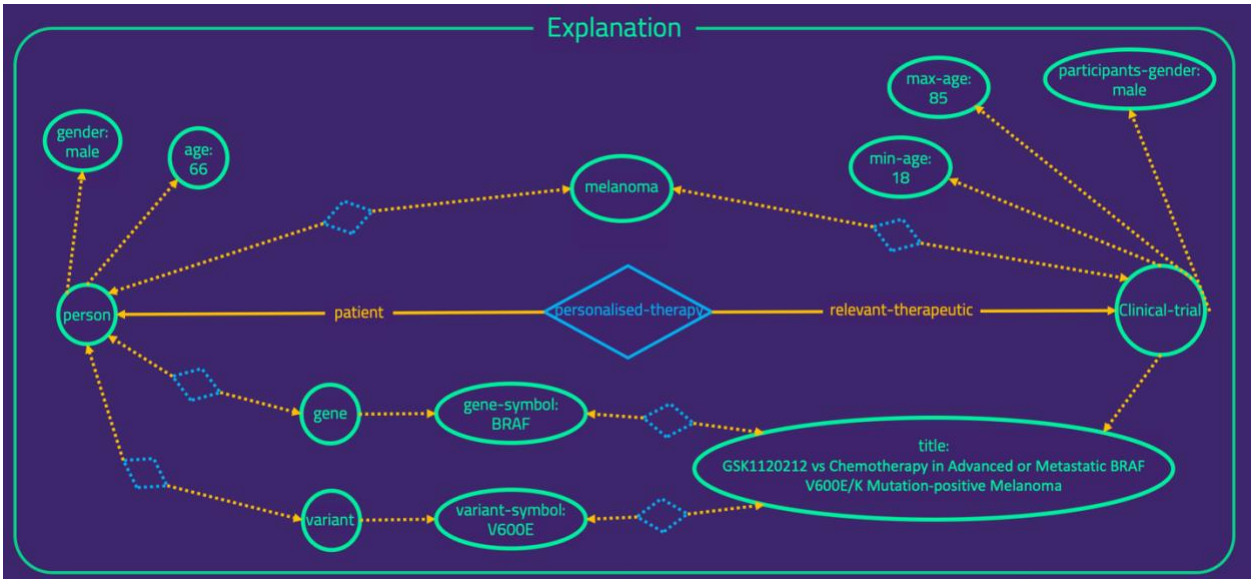
```
match $p isa person;
$d isa disease has disease-name "melanoma";
($p, $d) isa diagnosis;
$c isa clinical-trial;
$r ($p, $c) isa personalised-therapy;
get; limit 1;
```

Answer:

The answer that Grakn returns is a relationship called personalised-therapy, which connects a person and a clinical-trial. The clinical trial returned has information about the trial which we ingested in the graph; including its title (GSK1120212 vs Chemotherapy in Advanced or Metastatic BRAF V600E/K Mutation-positive Melanoma), intervention type (drug), and also a URL to the trial page.

Even though Grakn gives us a correct answer to our question, this data was actually never ingested into Grakn — no connections exist between persons and clinical trials. So, how did we get this relevant answer?

In short — Grakn's automated reasoner created this answer for us through automated reasoning. As this type of reasoning is fully explainable, we can interpret any inferred concept to understand how it was inferred/created. Below you can see how this explanation looks like. In the next section, I will dive deeper into how we created the logic and rules that allowed Grakn to infer these relationships.



Forming Rules to Propagate Reasoning Over the Graph

```
define

personalised-patient-therapy sub rule
when {
    ($person, $trial) isa eligible-trial-participant;
    ($person, $trial) isa relevant-trial-participant;
}
then {
    ($person, $trial) isa personalised-therapy;
};
```

```
trial-participant-eligibility sub rule,
when {
    $person isa person, has age $age, has gender $gender;
    $trial isa clinical-trial,
      has min-age <= $age,
      has max-age >= $age,
      has gender == $gender;
    $disease isa disease; ($disease, $person); ($disease, $trial);
},
then {
    ($person, $trial) isa eligible-trial-participant;
};
```

The inferred relationship between a person and a clinical-trial was called a personalised-therapy, as is shown in rule 1: personalised-patient-therapy. This gets created when two other conditions (represented by relationships) are met:

First, the person must be eligible to take part in the clinical-trial (eligible-trial-participant). This is evaluated by another rule (rule 2: trial-participant-eligibility) which takes into consideration the patient's age, gender and diagnosis, and compares it with the clinical-trial.

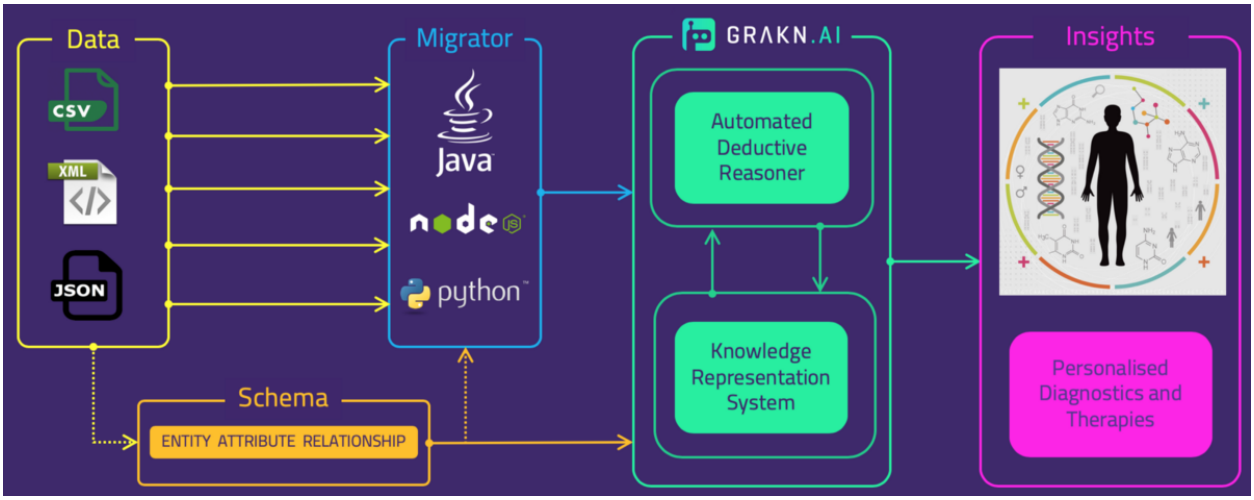
```
trial-participant-relevance sub rule,
when {
    $person isa person;
    ($person, $gene); $gene isa gene, has symbol $gs;
    ($person, $variant); $variant isa variant, has symbol $vs;
    $trial isa clinical-trial, has title contains $gs, has title contains $vs;
},
then {
    ($person, $trial) isa relevant-trial-participant;
};
```

Second, the person must be relevant to the trial. This is evaluated by rule 3, trial-participant-relevance, which checks that the title of the clinical trials contains the gene and variant symbol belonging to the person. This would indicate relevance between the individual person and the clinical-trail.

It should be noted that the rules above are a demonstration of how automated reasoning can be used to create a high-level abstraction over complex insights which scale though the data. They by no means compare to the full knowledge of how a biologist would go about achieving such an insight.

How do All the Pieces Fit Together in one Architecture

Now, let us take a step back, and look at how all of the components of building a precision medicine knowledge graph piece together.

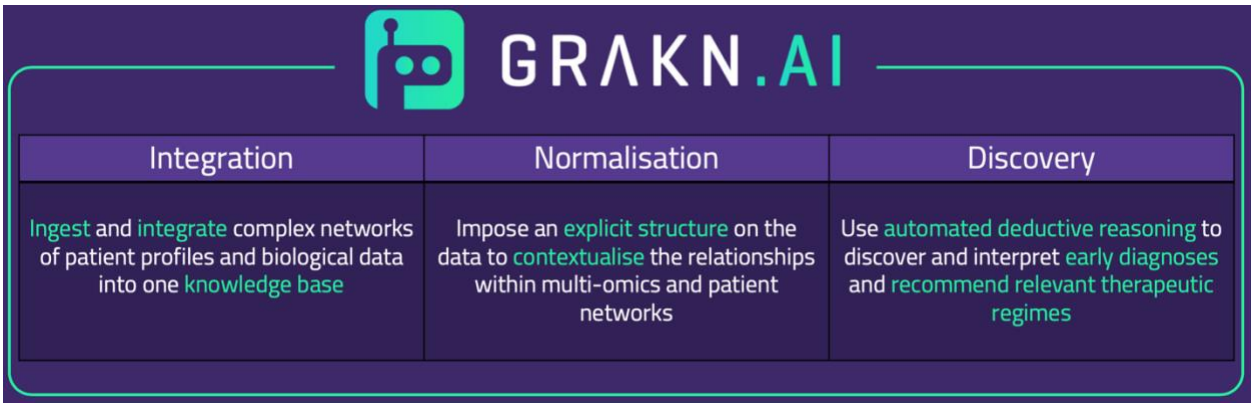


We start with the data which can come from multiple sources and in various formats. That raw data is used to create a schema (high level data model) to enforce a structure on the raw data. Once that is done, we use one of Grakn's clients to migrate the instances of data into Grakn making sure every insertion adheres to the schema. Grakn stores this in its knowledge representation system, which can be queried for insights to discover complex relationships or even test out hypotheses. These insights can already be in the graph or even be created at the time of query by the reasoning engine; for example, discovering personalised diagnoses and therapies.

Conclusion

So, we know that *Precision Medicine* is extremely promising in revolutionising global health care. We understand that there are barriers and bottle-necks associated with achieving it. I hope to have shown that Grakn can help to bring us many steps closer to having precision medicine as the standard for

health care. In summary, Grakn helps to solve the three key challenges in precision medicine when it comes to handling data:



For more information, please visit grakn.ai or contact us at enterprise@grakn.ai.