AI Driven Knowledge Graph for Drug Discovery

Internship Report

Submitted By:

Saransh Gupta
(17QM30005)

Under the Guidance of

Academic Supervisor
Prof. Balagopal G Menon
(Assistant Professor)

Industrial and Systems Engineering
Indian Institute of Technology Kharagpur

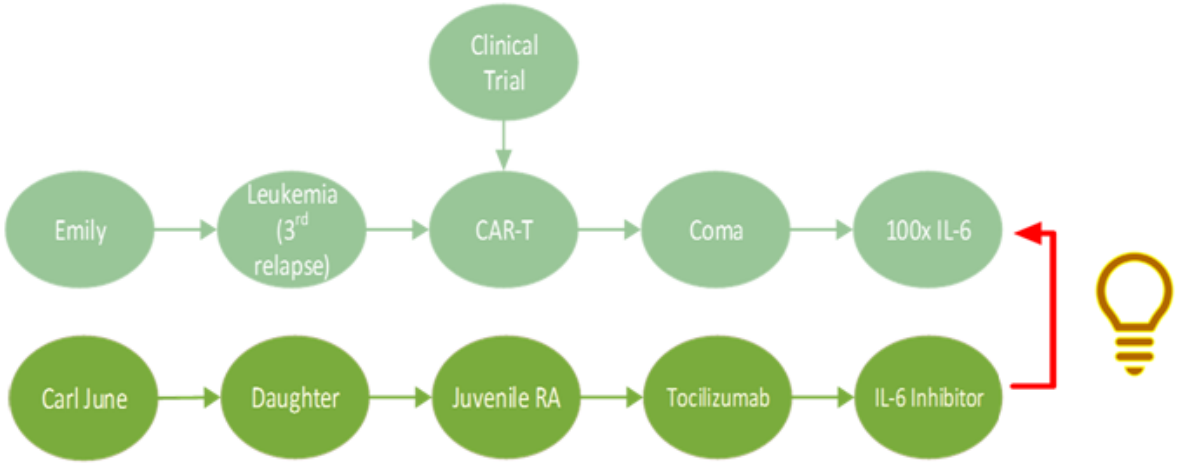
Internship Supervisor
Mr. Sravanth Ganta
(Data Science Manager)

Advanced Data Science
ZS Associates Inc

Contents

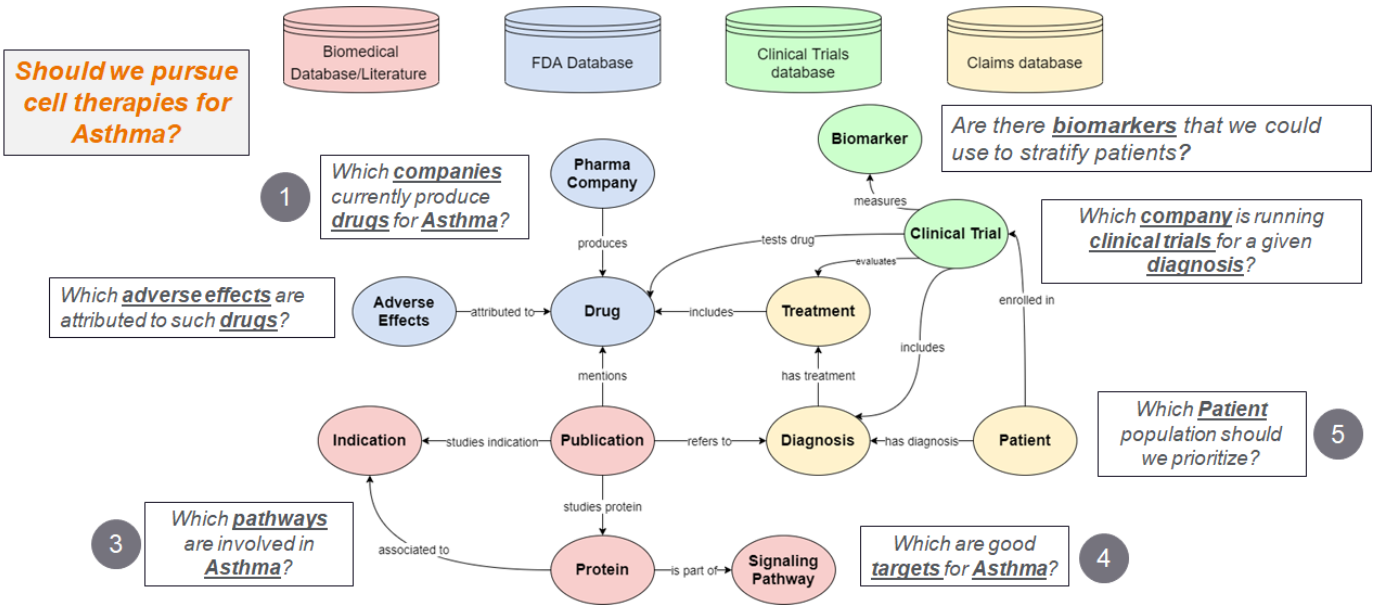
Introduction:

A Knowledge Graph is a tool that can be used to computationally capture relationships between entities in the real world. It represents a collection of interlinked descriptions of entities – objects, events or concepts. Knowledge graphs put data in context via linking and semantic metadata and this way provide a framework for data integration, unification, analytics and sharing.



Knowledge graphs can enable data-driven decisions by integrating complex/heterogeneous data.

An Enterprise Knowledge Graph is a representation of an organization's knowledge, understood by both humans and machines.

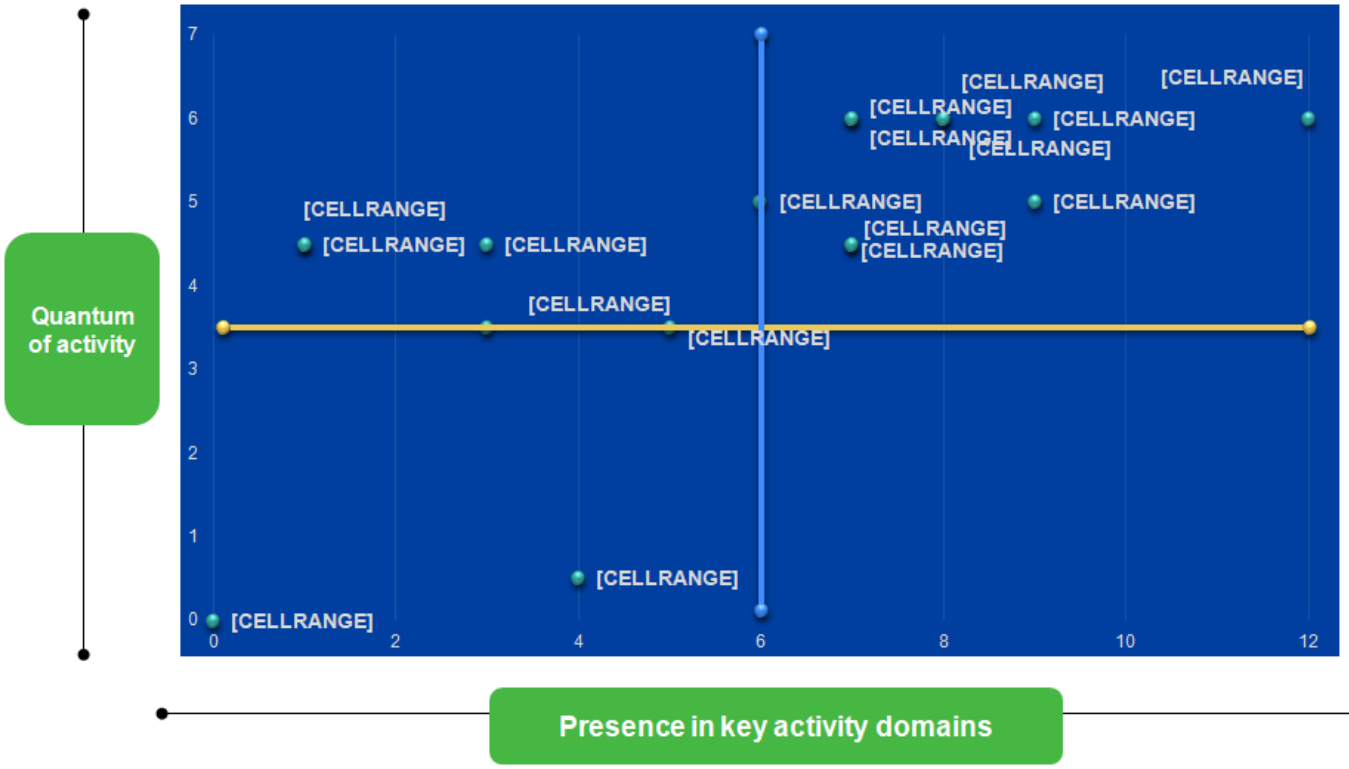


Problem Statement:

We want to create a Knowledge Graph that helps answer R&D Questions in Drug Discovery.

Motivation:

- Answers to questions pertaining to Drug Discovery are embedded in literature research which are text documents and hence not analytics-ready.
- Research needs to curate thousands of papers and disparate databases. This is currently done by SME and hence can be prone to error or limited to SME's knowledge.
- The current solution is effort intensive, and it takes weeks to find and do literature review on a single entity such as a gene or disease.
- Most top pharma companies are investing significantly in knowledge graph development.



Key Project Objectives:

- Efficiently map information from across different research paper documents and get information of relation between entities such as gene, disease and protein.
- An automated pipeline that can ingest new articles for a gene and update Knowledge Graph.
- Leverage novel Machine Learning techniques both unsupervised & supervised learning to address this problem.

Expected Project Impact:

- Prospective of ML techniques to help provide effective / efficient / scalable solution for complex problems
- Significant reduction in manual effort
- Expansion of inferences beyond human intervention

Methodology:

For Phase-1 of data science experimentation we focused on discovering connections between genes and diseases.

1. Named Entity Recognition:

To Identify the relationship between a Gene and a Disease, at first we should have the data which contains the sentences, their PMIDs and the entities (Disease / Genes). For that purpose we have used pubtator[1] based taxonomy which is a published **API** widely used for identifying biological entities in the PubMed articles.

Fig 1: Bio-Medical Entity Recognition using PubTator

Once we get the entities, we save them to a dataframe which will then act as an initial dataset to classify where there exists a relationship or not between the entities.

Following are the columns extracted after running the NER model:

- **PMID:** PMID of the research article where the sentence belongs to
- **Entity1:** Genes are saved as entity1
- **Entity1_id:** Unique IDs for the genes extracted directly from pubtator
- **Entity2:** Disease are saved as entity2
- **Entity2_id:** Unique IDs for the diseases extracted directly from pubtator
- **Sentene_entity_combination:** Unique ID representing a unique sentence with a unique set of genes and disease
- **Sentence_ID**

Now, we have completed our first step which was to create a dataset with entity1 and entity2 identified.

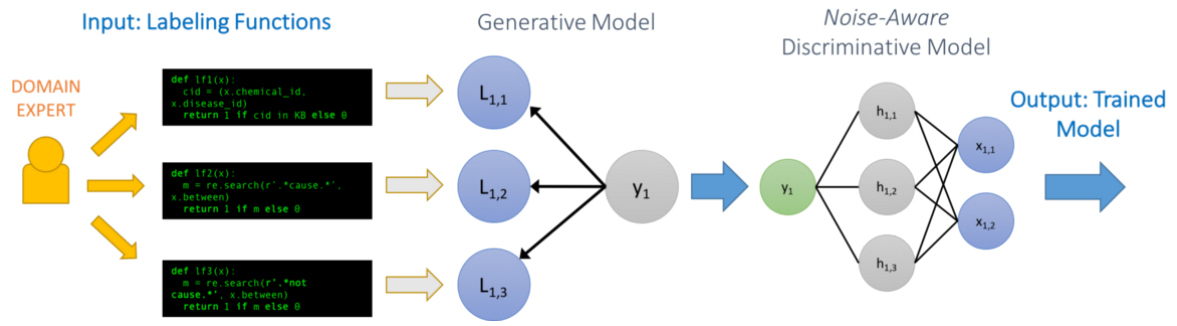
2. Classify Relationships Across The Entities Extracted:

Here, we have the dataset which contains the information about the sentences and the entities present inside them, but we lack the information about whether there exists a relationship between the entities or not.

We have more than **2 million rows** for the data and manually curating the complete dataset could have taken much more time and would not be a very efficient way. So in Order to extract the relationship, we use the weak supervision learning based methods which leverage some rule based approach to check whether there exists a relationship between the entities or not.

Hence, we use **Snorkel**[2] for that purpose:

Snorkel takes multiple labelling functions to generate the labelled data using weak supervision learning.



Let us look at some labelling functions[3] used to generate the data:

- **Labelling Function 1 (LF Association):**
  - From the Subject Matter Experts Team, we got a list of bioverbs, which if present between entity1 and entity2 must represent a relationship between the two entities.
- **Labelling Function 2 (LF Dependency Path[4]):**
  - Many times, it was observed that the bioverbs were not present between the entities, but it still represented some type of relationship.
  - Later on we found that the bioverbs are still connected with the entity1 and entity2 using a dependency graph.
  - Hence, we used the concepts of graph theory, built dependency parsing graphs for the different phrases and checked whether any entity links directly or indirectly to the bioverb or not.

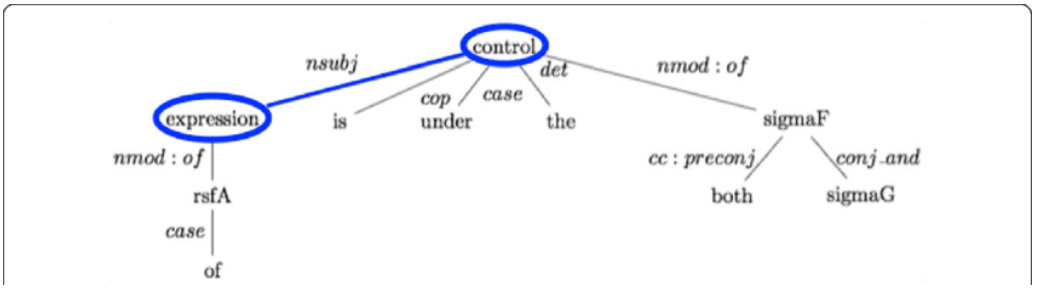
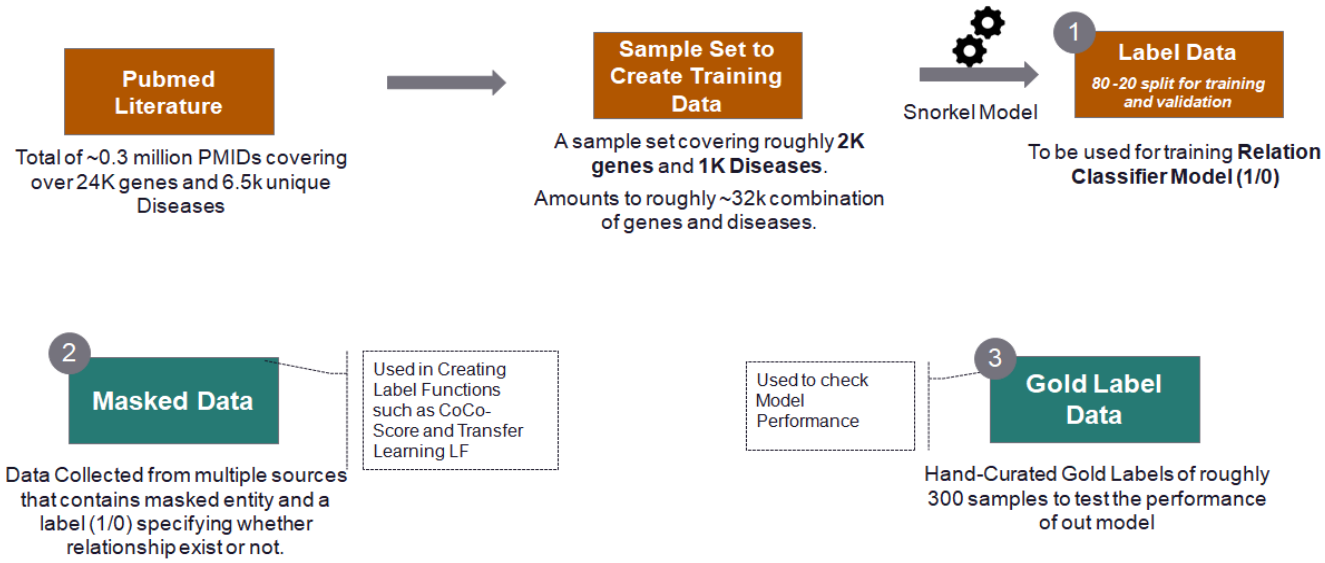


Fig: Example of dependency parsing

- **Labelling Function 3 (LF CoCo Score[5]):**
  - Using **Co Occurrence (CoCo)** Score of entity1 and entity2.
  - CoCo Score checks for the probability of occurring an entity1 and an entity2 together. A higher CoCo score represents that the entity1 and the entity2 are highly likely to appear together in a sentence hence, they must contain a relationship between them.
- **Labelling Function 4 (LF Masked Model):**
  - Using **EU-ADR[8], GAD[7] and ChemProt[6]** data we trained a discriminative model on masked sentences, so that we have a label function that can leverage knowledge from pre-trained transformer models in a generalized way.

Once we get the predictions out of the three labelling functions, we then use a weak supervision[9] method to generate final labels by combining all the labels.

Sneak Peak Into Different Datasets



3. Using roBERTa - BASE[10] model for relation classification:

By this point of step, we have generated a dataset which contains the sentences, with the entities inside them, along with the information whether there exists a relationship between entity1 and entity2.

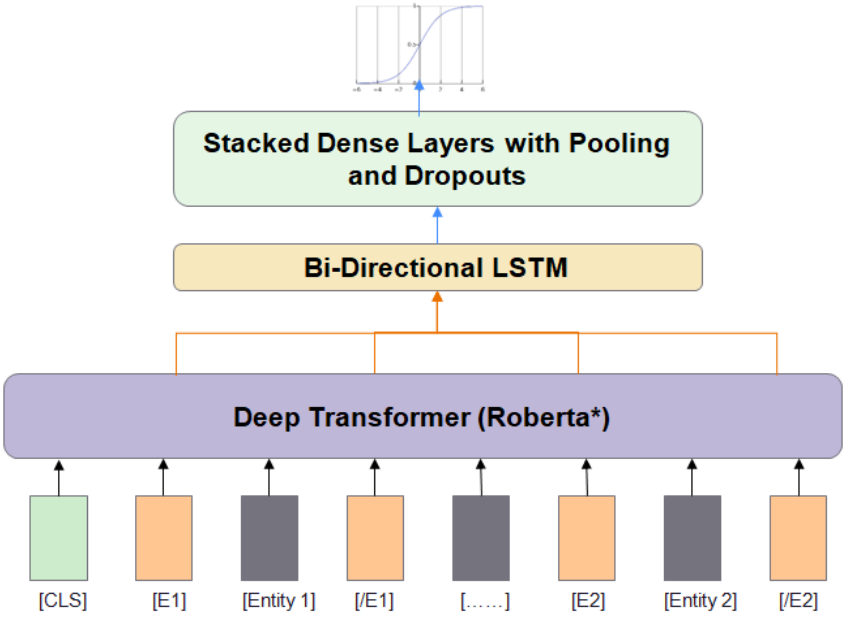


Fig: Model Architecture of the relations classifier

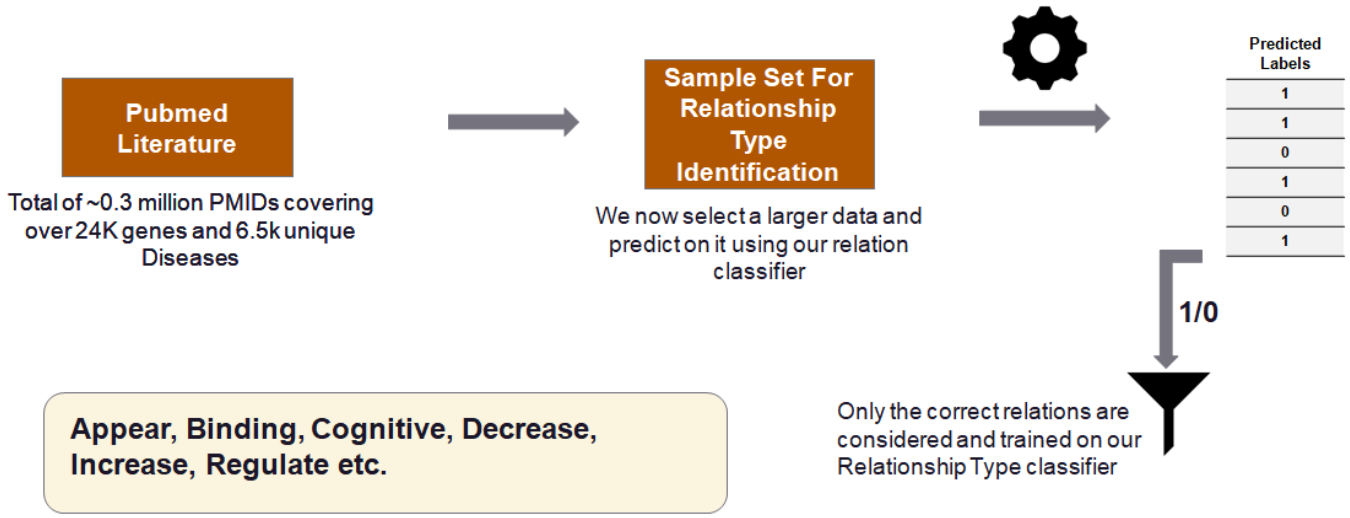
Since weak supervision methods might give us a lot of noise in the dataset, so to reduce the noise and for the better generalizability of the relation classifier, we use roBERTa - base, a BERT[11] based architecture which learns relationships well.

Here are some of the results from the training:

From the graphs, we can interpret that there is a significant improvement in the results as compared to what we were getting from the weak supervision methods.

4. <u>What is the relationship between the entities:</u>

Now, we have trained a relationship classifier which tells us whether there exists a relationship between entity1 and entity2 or not, but that is just part-1 of the problem. In this part, we will address how do we predict if there exists a relationship between the entities, then what type of relationship it is?



Broad overview of the dataset obtained:

Text	Entity1	Entity2	Relationship (0/1)

Now we want to classify what type of relationship our entities have. So we filter out for the dataset where the relationship flag is coming out to be 1.
Then again we leverage the same weak supervision based approach using the similar labelling functions to generate a multi-class dataset.
We proposed three snorkel labelling functions:

- **Labelling Function 1 (LF Association):**
  - From the SME team, we got a list of bio-verbs, which has the bioverbs mapped to the particular category of their class.
    - For eg: increase,upregulate, boost, escalate, improve: all these keywords were mapped to a single class **“upregulates”**
    - Hence, if we state the relationship between entity1 and entity2, we would write the triplet like this:
      - *Entity1* **<u>Upregulates</u>** *Entity2*
    - These are the triplets which we want to feed inside the knowledge graphs, these will help us finding the drugs
  - We search for the keyword between entity1 and entity2 here.
- **Labelling Function 2 (LF Dependency Path):**
  - Similar to the part -1, here we search for the keywords in the dependency path
- **Labelling Function 3 (LF SRL+lsimp):**
  - This labelling function has two parts:

        - Isimp (sentence simplification for biomedical text)[12]:
            - Simplifies the complex sentence into simpler sentences using the simple grammar rules based approach.
        - SRL (semantic role labelling)[13]:
            - It is used for determining the latent predicate argument structure of a sentence and providing representations that can answer basic questions about sentence meaning, including who did what to whom, etc.
    - Using the two approaches, we determine the type of relationships between the phrases containing the entity1 and the entity2.

Once we get the predictions from the Labelling functions, we then use a weak supervision based approach to predict the relations between the entities.

5. **<u>Using the Machine Learning based Text classification methods:</u>**

Now we have the complete dataset which contains text, entities and their type of relationship. Now we can simply build a NLP based text classifier to classify the type of relationship between the entities.

Steps for the NLP text classifier:

- **Features Generation:**
    - Create tf-idf[15] features of the sentences
    - Extract sentences level embeddings and word level embeddings using ***BioWordVec***[14]
    - Extract word embeddings of the bio-verbs and the targets
    - For each label we took the average of sentence embeddings to make a contextual representation for each label
- **Feature Engineering:**
    - Take the dot product of the tf-idf[15] vectors with the target embeddings to get the feature similarity of tf-idf
    - Perform similar steps with verb-phrase embeddings and sentence level embeddings

- **Data Processing:**
  - Split the dataset into 80:20 (train : validation) dataset
- **Training:**
  - Train a series of machine learning classifiers such as CATBoost[16], XGBoost[17], LGBM[18].
  - Check their performances on the validation dataset
  - Apply grid-search[19] CV for hyper-parameter optimization and select the best model with best parameters

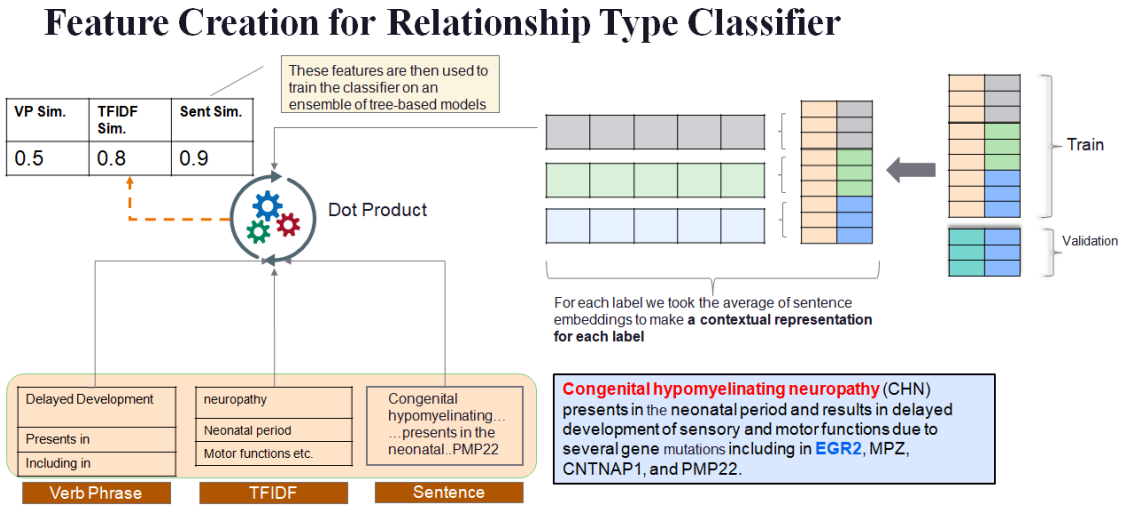
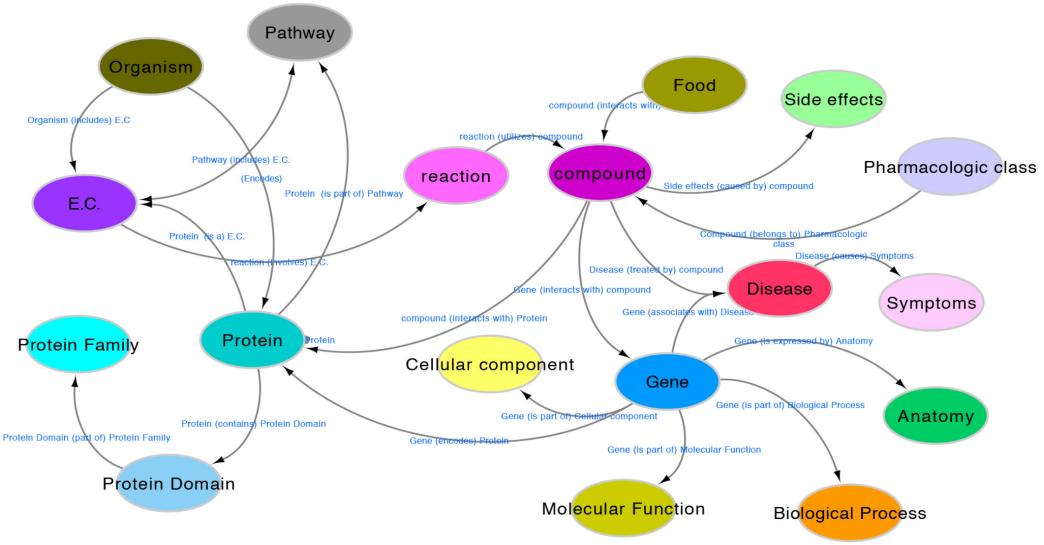


Fig: Overview of the Pipeline-2

6. Creation of Knowledge Graph[20]:

We have completed our Machine learning part which was to create the dataset, followed by predicting the relations and the type of relations between the entities. Now we have the triplets predicted and they are ready to be deployed to a Knowledge Graph.



Results:

Model performances on the part-1 of the pipeline:

1 : Relationship Exists 0 : No Relationship Exists				
Discriminative Model (80-20 split)*				
	Label	Precision	Recall	F1
Validation Data	0	70%	42%	53%
	1	51%	80%	63%
	weighted	61%	66%	58%
Gold Dataset	0	68%	39%	50%
	1	49%	88%	77%
	weighted	61%	68%	66%

Fig: Performance using weak supervision

1 : Relationship Exists 0 : No Relationship Exists				
Discriminative Model (80-20 split)*				
	Label	Precision	Recall	F1
Validation Data	0	80%	42%	55%
	1	61%	90%	73%
	weighted	71%	66%	64%
Gold Dataset	0	73%	49%	55%
	1	68%	84%	78%
	weighted	65%	63%	69%

Fig: Performance using roberta-base (BERT)

Model performances on the part-2 of the pipeline:

	precision	recall	f1-score	support
Appear	0.74	0.58	0.65	184
Binding	0.63	0.63	0.63	272
Cognitive	0.68	0.70	0.69	358
Decrease	0.55	0.52	0.53	413
Elevate Change Activity	0.71	0.66	0.69	173
Examine	0.63	0.63	0.63	145
Include	0.69	0.71	0.70	416
Increase	0.66	0.67	0.66	167
Indicate	0.72	0.76	0.74	226
Interact	0.68	0.73	0.70	823
Neutral Change Activity	0.63	0.49	0.55	85
Proceed	0.69	0.68	0.68	398
Regulate	0.70	0.73	0.72	433
Restrict Change Activity	0.64	0.64	0.64	391
Score	0.69	0.65	0.67	329
accuracy			0.67	4813
macro avg	0.67	0.65	0.66	4813
weighted avg	0.67	0.67	0.67	4813

Our System is currently achieving a validation set **F1 of 66%**

Conclusion:

We have described how to formulate the problem of knowledge graph identification: jointly inferring a knowledge graph from the noisy output of an information extraction system through a combined process of determining co-referent entities, predicting relational links, collectively classifying entity labels, and enforcing ontological constraints.
We illustrate how we can use weak supervision learning methods to create the dataset which actually doesn't exist on any of the platforms. Using the dataset to create a relation classifier and followed by building a relation type classifier, putting the triplets and the entities altogether and creating a knowledge graph.

This Project was an internal project which was successfully converted into a client project for **Bristol Myers Squibb.**

Acknowledgment:

I would like to thank **Prof. B.G. Menon (Academic supervisor)** and **Mr. Sravanth Ganta (Internship Supervisor)** for giving me this golden opportunity and offering enthusiastic assistance. This work was carried out at **ZS Associates Inc.**

The Team:

Helena Deus
Data Science Manager

Data Science

Sravanth Ganta
Data Science Manager

Anshik
Data Science Associate Consultant

Priyansh Jain
Data Science Associate

Adarsh Upadhyay
Data Science Associate Consultant

Saransh Gupta
Data Science Associate - Intern

Systems biology

Prateek Bajaj
Biomedical Research Consultant

Andreas Constantinou
Biomedical Research Associate Consultant

Shawntel Okonkwo
Biomedical Research Associate Consultant

Juan Sendoya
Biomedical Research Associate Consultant

Abhilash Gangadharan
Data Science Associate Consultant

References:

1. Wei CH, Allot A, Leaman R, Lu Z. PubTator central: automated concept annotation for biomedical full text articles. Nucleic Acids Res. 2019 Jul 2;47(W1):W587-W593. doi: 10.1093/nar/gkz389. PMID: 31114887; PMCID: PMC6602571.

2. Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, Christopher Ré, Machine Learning (cs.LG); Machine Learning (stat.ML), Proceedings of the VLDB Endowment, 11(3), 269-282, 2017, DOI: 1711.10160

3. Ratner A, Bach SH, Ehrenberg H, Fries J, Wu S, Ré C. Snorkel: Rapid Training Data Creation with Weak Supervision. Proceedings VLDB Endowment. 2017;11(3):269-282. doi:10.14778/3157794.3157797

4. Neumann, Mark, et al. "Scispacy: Fast and robust models for biomedical natural language processing." arXiv preprint arXiv:1902.07669 (2019).

5. Doddington, George. "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics." Proceedings of the second international conference on Human Language Technology Research. 2002.

6. Lee, Jinhyuk, et al. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining." Bioinformatics 36.4 (2020): 1234-1240.

7. Becker KG, Barnes KC, Bright TJ, Wang SA. The genetic association database. Nat Genet. 2004 May;36(5):431-2. doi: 10.1038/ng0504-431. PMID: 15118671.

8. Trifiro G, Fourrier-Reglat A, Sturkenboom MC, Díaz Acedo C, Van Der Lei J; EU-ADR Group. The EU-ADR project: preliminary results and perspective. Stud Health Technol Inform. 2009;148:43-9. PMID: 19745234.

9. Hoffmann, Raphael, et al. "Knowledge-based weak supervision for information extraction of overlapping relations." Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies. 2011.

10. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

11. Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
12. Peng, Yifan, et al. "iSimp: A sentence simplification system for biomedicail text." 2012 IEEE International Conference on Bioinformatics and Biomedicine. IEEE, 2012.

13. He, Luheng, et al. "Deep semantic role labeling: What works and what's next." Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017.

14. Zhang, Yijia, et al. "BioWordVec, improving biomedical word embeddings with subword information and MeSH." Scientific data 6.1 (2019): 1-9.

15. Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. No. 1. 2003.

16. Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2017). CatBoost: unbiased boosting with categorical features. arXiv preprint arXiv:1706.09516.

17. Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016.

18. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30, 3146-3154.

19. Liashchynskyi, Petro, and Pavlo Liashchynskyi. "Grid search, random search, genetic algorithm: a big comparison for NAS." arXiv preprint arXiv:1912.06059 (2019).

20. Wang, Z., Zhang, J., Feng, J., & Chen, Z. (2014, June). Knowledge graph embedding by translating on hyperplanes. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 28, No. 1).