CSE 576: TOPICS IN NLP

Project Phase 2- Automated Dataset Creation

Team:

Venkat Krishna Sai Kowkuntla (1218482044)

Pavan Kalyan Reddy Thota (1218436466)

Rajashekar Reddy Pinreddy (1218539543)

Sai Teja Vishal Janagala (1218332037)

Supreet Palabatla (1218516039)

Task: Relation Extraction Via Question Answering on Bio-Medical Corpora

In this phase of the project we automated the creation of new datasets using 4 different techniques. We are utilising the pre-existing datasets like CHEMPROT Dataset, MIMIC Dataset to create new ones.

The final dataset we have created is of csv format and will be consisting of the following structure.

DOC ID	Relation ID	Relation Type	Relation Name	Term 1	Term 2	abstract
10047461	CPR:3	Y	ACTIVATOR	Arg1:T13	Arg2:T57	Tomudex is ACTIVATOR of kinase
10047461	CPR:7	Y	NOT DOWNREGULATOR	Arg1:T13	Arg2:T57	Tomudex is NOT DOWNREGULATOR of kinase
10047461	CPR:7	Y	NOT INHIBITOR	Arg1:T13	Arg2:T57	Tomudex is NOT INHIBITOR of kinase
10047461	CPR:7	Y	NOT INDIRECT-DOWNREGULATOR	Arg1:T13	Arg2:T57	Tomudex is NOT INDIRECT-DOWNREGULATOR of kinase
10047461	CPR:8	Y	NOT AGONIST	Arg1:T13	Arg2:T57	Tomudex is NOT AGONIST of kinase
10047461	CPR:8	Y	NOT AGONIST-ACTIVATOR	Arg1:T13	Arg2:T57	Tomudex is NOT AGONIST-ACTIVATOR of kinase
10047461	CPR:8	Y	NOT AGONIST-INHIBITOR	Arg1:T13	Arg2:T57	Tomudex is NOT AGONIST-INHIBITOR of kinase
10047461	CPR:9	Y	NOT ANTAGONIST	Arg1:T13	Arg2:T57	Tomudex is NOT ANTAGONIST of kinase
10047461	CPR:3	Y	ACTIVATOR	Arg1:T7	Arg2:T39	Tomudex is ACTIVATOR of cyclin E
10047461	CPR:7	Y	NOT DOWNREGULATOR	Arg1:T7	Arg2:T39	Tomudex is NOT DOWNREGULATOR of cyclin E
10047461	CPR:7	Y	NOT INHIBITOR	Arg1:T7	Arg2:T39	Tomudex is NOT INHIBITOR of cyclin E
10047461	CPR:7	Y	NOT INDIRECT-DOWNREGULATOR	Arg1:T7	Arg2:T39	Tomudex is NOT INDIRECT-DOWNREGULATOR of cyclin E
10047461	CPR:8	Y	NOT AGONIST	Arg1:T7	Arg2:T39	Tomudex is NOT AGONIST of cyclin E
10047461	CPR:8	Y	NOT AGONIST-ACTIVATOR	Arg1:T7	Arg2:T39	Tomudex is NOT AGONIST-ACTIVATOR of cyclin E
10047461	CPR:8	Y	NOT AGONIST-INHIBITOR	Arg1:T7	Arg2:T39	Tomudex is NOT AGONIST-INHIBITOR of cyclin E

Columns in the Final Automated Dataset:

1. **DOC ID:** The DOC ID refers to the document ID/Text ID of the CHEMPROT Dataset on which we extract the relations between the entities.
   We have opted for the use of the same column for our automated dataset as we extract the new relations using the same text from the CHEMPROT Dataset.

2. **Relation ID:** The Relation ID refers to the Relation ID/Number as mentioned in the CHEMPROT Dataset which describes the relation group between two entities.
   We will be using the relations that are evaluated as 'Y' and create new relations from those.

3. **Relation Type:** The Relation Type refers to the Evaluation of the particular relation as given in the CHEMPROT Dataset.
   We have created a new dataset using only the Evaluation 'Y' of old relations.

4. **Relation Name:** The Relation Name refers to the Name of the relation between two entities **Term 1** AND **Term 2**.
   We create new relations using different techniques and thereby generate new abstracts.

5. **Term 1:** The Term 1 refers to the entity A which is related to entity B(Term 2) with Relation R(Relation Name)

6. **Term 2:** The Term 2 refers to the entity B which is related to the entity A(Term 1) with Relation R(Relation Name)

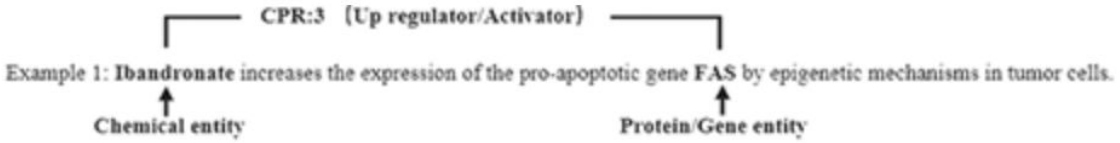
7. **Abstract:** Abstract refers to the summarized text generated by combining Term 1 with Term 2 with Relation R

Techniques Used to Generate New Dataset:

1. **Paraphrasing Relation Names with similar meaning (Relations from Same Group):**
   Description:

Group	Eval	CHEMPROT Relations
CPR:3	Y	UPREGULATOR|ACTIVATOR|INDIRECT-UPREGULATOR
CPR:4	Y	DOWNREGULATOR|INHIBITOR|INDIRECT-DOWNREGULATOR
CPR:5	Y	AGONIST|AGONIST-ACTIVATOR|AGONIST-INHIBITOR
CPR:6	Y	ANTAGONIST

   These are the Relation Groups that are Evaluated 'Y' according to the CHEMPROT Dataset.
   We are creating new data by substituting the old relation name with the substitution word from the same relation group.



   In the above example from CHEMPROT, we are mentioning that the relation type is CPR:3 (UPREGULATOR| ACTIVATOR| INDIRECT-UPREGULATOR)

The relation given in the Dataset is:
Example1 CPR:3 Y ACTIVATOR Arg1:T12 Arg2:20

So, we generate two additional relations by replacing Relation ACTIVATOR with two other relations (UPREGULATOR,INDIRECT-UPREGULATOR) in the same group CPR:3

New Dataset formed :
Example1 CPR:3 Y UPREGULATOR Arg1:T12 Arg2:20
Example1 CPR:3 Y INDIRECT-UPREGULATOR Arg1:T12 Arg2:20

This relation holds true as the relation is satisfied when substituted by its synonyms.

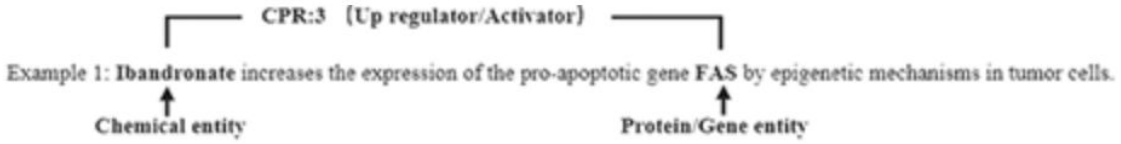
Task Evaluation:

Task completed	-	**Venkat, Vishal and Pavan**
Time to Complete the task	-	**6 Hours**
Code Written in	-	**Python**
Number of Rows Generated	-	**16000**
Tools Used	-	**Spyder, Excel**

2. **Creating new Relations by replacing Relation name with Negating Relations (Relations from other relation group):**
Description:

Group	Eval	CHEMPROT Relations
CPR:3	Y	UPREGULATOR|ACTIVATOR|INDIRECT-UPREGULATOR
CPR:4	Y	DOWNREGULATOR|INHIBITOR|INDIRECT-DOWNREGULATOR
CPR:5	Y	AGONIST|AGONIST-ACTIVATOR|AGONIST-INHIBITOR
CPR:6	Y	ANTAGONIST

These are the Relation Groups that are Evaluated ‘Y’ according to the CHEMPROT Dataset.
We are creating new data by substituting the old relation name with the substitution word from the same relation group.



In the above example from CHEMPROT, we are mentioning that the relation type is CPR:3 (UPREGULATOR| ACTIVATOR| INDIRECT-UPREGULATOR)

The relation given in the Dataset is:
Example1 CPR:3 Y ACTIVATOR Arg1:T12 Arg2:20

So, the negating relation for the above example is formed from the relations of the other group.
We create a new Negating Relations group using the Python code that is obtained by negating the other groups of relations and save them to the relations csv.

Later, we use the relations csv to generate new relations by replacing the relation

So, we generate three additional relations by replacing Relation ACTIVATOR with two other relations (NOT DOWNREGULATOR,NOT INHIBITOR,NOT INDIRECT-DOWNREGULATOR) in the different group CPR:4

So, Negating relations here for the given Example1 Relation are:
Example1 CPR:7 Y NOT DOWNREGULATOR Arg1:T12 Arg2:20
Example1 CPR:7 Y NOT INHIBITOR Arg1:T12 Arg2:20
Example1 CPR:7 Y NOT INDIRECT-DOWNREGULATOR Arg1:T12 Arg2:20

This relation holds true as the relation is satisfied when substituted by its antonyms(Relations with not same meaning).

Task Evaluation:

Task completed	-	**Venkat, Pavan and Rajashekar**
Time to Complete the task	-	**6 Hours**
Code Written in	-	**Python**
Number of Rows Generated	-	**69000**
Tools Used	-	**Spyder, Excel**

3. **Summarizing Text using Knowledge from the Dataset:**
   Description:
   Dataset gives us the knowledge of relations between two entities present in the abstract. We can summarize the abstract into short sentences using a template and later use the template to train the data to improve accuracy.

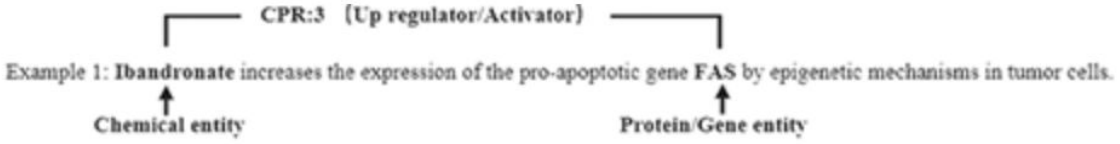
   The Knowledge of entities and relations from the training dataset is given as follows:

10047461	CPR:3	Y	ACTIVATOR	Arg1:T13	Arg2:T57
10047461	CPR:3	Y	ACTIVATOR	Arg1:T7	Arg2:T39
10047461	CPR:3	Y	ACTIVATOR	Arg1:T7	Arg2:T40
10047461	CPR:3	Y	ACTIVATOR	Arg1:T7	Arg2:T41
10047461	CPR:3	Y	INDIRECT-UPREGULATOR	Arg1:T13	Arg2:T55
10047461	CPR:3	Y	INDIRECT-UPREGULATOR	Arg1:T13	Arg2:T56
10047461	CPR:3	Y	INDIRECT-UPREGULATOR	Arg1:T2	Arg2:T30
10047461	CPR:3	Y	INDIRECT-UPREGULATOR	Arg1:T2	Arg2:T31
10047461	CPR:3	Y	INDIRECT-UPREGULATOR	Arg1:T2	Arg2:T32
10047461	CPR:3	Y	INDIRECT-UPREGULATOR	Arg1:T2	Arg2:T33
10047461	CPR:4	Y	INDIRECT-DOWNREGULATOR	Arg1:T13	Arg2:T54
10047461	CPR:4	Y	INDIRECT-DOWNREGULATOR	Arg1:T7	Arg2:T38

We can use the knowledge between Term 1 and Term 2 and generate a new abstract using a template.

Knowledge from Training Dataset:
ACTIVATOR Arg1:T13 Arg2:T57

Abstract:



Template used to generate new abstracts:
Term 1 is an **Relation** of **Term 2**

New Abstract Created from the Knowledge:
We identified the mapping of Term 1 and Term 2 using the entities tables
Term 1 - Ibandronate
Term 2 - Gene FAS

We then create a new abstracts using the above given template

New Summarized Text:
Ibandronate is an Activator of Gene FAS

The Newly created Text is appended to each row of the Output Table and final Output is used for training the model to improve accuracy.

Task Evaluation:
Task completed - **Rajashekar, Pavan and Supreet**

Time to Complete the task - **6 Hours**
Code Written in - **Python**
Number of Rows Generated - **15000**
Tools Used - **Spyder, Excel**

4. **Generating new Text using already pre-existing datasets like MIMIC.**
   Description:
   We simulated the process of generating the synthetic data using code in Python.

   We will be using MIMIC Dataset to get the entities and relations and use it for text generation using a template.

   We demonstrated this technique by storing the entities and relations in the PostgresQL Tables and fetched the entities from the table using Python psycopg2 library.

   After fetching the entities and relations from the PostgresQL table, we will populate multiple texts (~10k samples) using substitution technique in a template.

   Example:

   Table in PostgresQL(Data is Fetch from MIMIC Dataset):

	DRUG_NAME_POE character varying	PROD_STRENGTH character varying	GSN character varying	STARTDATE character varying	ENDDATE character varying	DIAGNOSIS character varying
1	Revital	10mg	1003	09-09-2020	19-09-2020	Acidity
2	Dolo	650mg	1002	19-09-2020	25-09-2020	Headache
3	Levo	100mg	1005	15-09-2020	20-09-2020	Cold
4	Razo	10mg	1007	01-09-2020	15-09-2020	Stomach Ache
5	Augmintin	625mg	1008	09-09-2020	15-09-2020	Fever
6	Tylenol	25mg	1009	09-09-2020	15-09-2020	Fever
7	Aspirin	35mg	1010	09-09-2020	15-09-2020	Fever
8	Volini	15mg	1011	09-09-2020	15-09-2020	Leg Sprains
9	Crocin	25mg	1012	09-09-2020	15-09-2020	Body Pains

   Entities are fetched from the above table using psycopg2 and two entities are related using a relation table.

   Relation Table:
   R-1 Cures
   R-2 Alleviates
   R-3 Reduces
   R-4 Increases
   R-5 Worsens

   We create a new text by combining entities and relations from the relation Table and generate a new text

Example text generated between entities DRUG_NAME_POE and DIAGNOSIS with relation **R-1 Cures**.
Revital Cures Acidity
Dolo Cures HeadAche
Levo Cures Cold
Razo Cures Stomach Ache
Augmentin Cures Fever
Tylenol Cures Fever

Task Evaluation:

Task completed	-	**Vishal and Supreet**
Time to Complete the task	-	**6 Hours**
Code Written in	-	**Python**
Number of Rows Generated	-	**15000**
Tools Used	-	**Spyder, Excel, PostgresQL**

Instructions to Run the Code:

1. Download the code from the github Repository
2. Check the paths of the tsv and csv files according to their locations.
3. Run automated_dataset_creation_task1&2.py
4. Verify automated_dataset1 which consists of additional new relations created.
5. Run automated_dataset_creation_task3.py
6. Verify automated_dataset2 which consists of additional relations and abstracts created.
7. Configure an input table named as “MIMIC Table” in the postgresQL using the MIMIC Dataset as knowledge.(You can use the columns DRUG_NAME_POE, PROD_STRENGTH,GSN,STARTDATE,ENDDATE,DIAGNOSIS from the MIMIC Dataset to create MIMIC Table)
8. Configure the getOpenConnection method of the automated_dataset_creation_task4.py according to the username and password of your PostgresQL Table
9. Set the environment variables to include the library psycopg2 (npm install psycopg2)
10. Run automated_dataset_creation_task4.py
11. Verify automated_dataset3 which consists of additional abstracts created using template substitution.

Packages used - Pandas, psycopg2.