The code presented aims to perform Relation Extraction on the Chemprot dataset, as a result by which we would be able to identify complex Chemical Protein interactions which would have a series of applications down the line in BioMedical domain. Through the course of the work, we have done a series of pre-processing tasks catered to specific models and their requirements. Two models to carry out the tasks were implemented: one is an SVM fed with a novel approach with Dependency Parsing and ELMO vectorised form while the other is a hybrid model being a confluence of three models CNN, Bi-LSTM and Bio-BERT on top of which selective weighted voting was performed.

Process to run the notebook/code and get results:

1. ***Running BioBERT***:

Execute the following steps to successfully build and run the model.

* Open a Terminal in source directory
* Create an environment: ***python3 -m venv biobertenv***
* Activate and use the environment: ***biobertenv/bin/activate***
* Install the required libraries in the environment created by running the requirements.txt file by using the command: ***pip install -r requirements.txt***
* To fine tune the model, run the python file BioBert\_Tuning.py by running the command: ***python BioBert\_Tuning.py* (Please keep in mind that this is an extremely computationally intensive and time consuming process (11 hrs to be precise)) OR you could skip this process on the whole and use the pretrained-tuned model that we have run, saved under the folders biobert\_tokenizer and biobert\_finetuned which is the output of the BioBert\_Tuning.py.** The python file also uses Merged.csv which is nothing but the combination of Train and Dev datasets. Hence it is included in the zip as well. If the models come out as compressed files please unzip them and then proceed to next steps. The model files can be found in <https://drive.google.com/drive/folders/1jCzGSbx9dU9Z45Pjkw07zgBZpDj_Caxq>.
* Now that we have the model, we can now run Test\_BioBert.py to run the model on top of the test dataset. To do this execute the command: ***python Test\_BioBert.py.*** This should take around 20 minutes maximum. The output of this gives us a file called Output.txt which saves the prediction made on the test dataset.

1. ***Running CNN and Bi-LSTM***: The notebook titled Voting covers the creation of our first model which is a hybrid model. It contains different processes namely import of specific libraries, Data Preprocessing, Exporting Pre-processed data for usage down the line **(Please do note that this is because run of the ELMO model is slightly computationally intensive and takes around 30 mins to run on Train and 15 mins on Test hence, we export it to the local such that the file is saved for future use of other models using the pre-processed train and test data),** model runs and evaluation**.** The python notebook can be run by opening it on Anaconda Jupyter Notebook or on Visual Studio Jupyter Notebooks. The notebook at first runs through the preprocessing phase after which it runs CNN, Bi-LSTM. Then it reads the output of the BioBERT generated in Output.txt. These 3 models are then passed into the Voting process. **(MAKE SURE THAT THE PYTHON NOTEBOOK IS BEING RUN IN THE SAME PATH AS THE BioBERT AS IT UTILISES THE OUTPUT FOR VOTING)**
2. ***Running SVM***: The notebook titled SVM covers the same template as above. All that needs to be done is to run the notebook titled SVM.ipynb and evaluate the results produced. **(MAKE SURE THAT THE PYTHON NOTEBOOK IS BEING RUN IN THE SAME PATH AS THE BioBERT and VOTING AS THE NOTEBOOK USES THE EXPORTED TRAIN AND TEST PREPROCESSED DATA TO REDUCE TIME COMPLEXITY)**