CMPE561
Natural Language Processing Course
Extracting Protein-Protein Relationships Using Dependency Parsing and Machine Learning Techniques

Samet Atdag[1]

[1]*Department of Computer Engineering, Bogazici University, Istanbul*

(Dated: June 5, 2016)

Curating protein-protein relationships is not an easy task for humans since human curators are known to have low recall. Besides, typically research papers focus a research topic in detail, they also have hidden information about protein-protein relationships. G. Erkan et al. [1] extracted protein-protein relationship dataset processing Protein Interaction Pairs Sub-task 2 (IPS) and Protein Interaction Sentences Sub-task 3 (ISS) of BioCreAtIvE II (Critical Assessment for Information Extraction in Biology). In this study, I present my approach to identify interacting protein pairs using dependency parsing and machine learning techniques.

The code of the project is in Github account: https://github.com/sametatdag/CMPE561. (Refer to RelationExtraction directory on Github.)

Contents

I. INTRODUCTION

Protein-protein relationships have a key importance for predicting the protein function of a protein and drug ability of molecules. It is possible to extract protein-protein relationships on a small dataset by hand is an easy task, but in nature the size of the protein data to be processed is enormous and impossible to process by hand [2].

Faster processors, NLP and machine learning techniques allow us to process larger amount of data in a shorter amount of time. In this study, I present my approach of using dependency parse trees and machine learning techniques to extract protein relationships from the dataset by G. Erkan et al. [1].

II. SYSTEM DESCRIPTION

A. Dataset

The dataset contains sentences describing interacting proteins. Protein names have been replaced with the PROTX1, PROTX2, and PROTX0 keywords. PROTX1 and PROTX2 are the proteins that has the possibility of interaction. Sentences are binary labeled; 0 and 1 represents "Not interacting" and "Interacting" respectively. There are 4056 sentences in the dataset.

The dataset contains a short list of interaction verbs defining an interaction between proteins. Thanks to the authors of original paper, this list is already extracted.

B. Method

In this study two systems were developed: 1. A baseline system using only the words for detecting relationships 2. An advanced system using dependency parse trees for the same goal.

1. Baseline System

In the basic version of the system, a bag of words approach used with TF-IDF values. Each sentence is considered as a document and TF-IDF value of each word is used as a feature. Each document is represented as a feature vector of TF-IDF values.

2. Advanced System

G. Erkan et al. [1] describes new features that may be extracted via dependency parsing, such as: 1. Interaction words as features 2. Distances of proteins to interaction verb. 3. Interaction verbs that is places as a ancestor of a protein in the sentences.

To enrich the feature vector of each document and expectantly increase the accuracy of the results, two new feature sets are added to the baseline system. First, each interaction verb is added as a binary feature. Second, using dependency parsing, a binary feature representing an equal distance of PROTX1 and PROTX2 to the interacting verb.

1. Interaction Verbs Features

   In Table I, interaction verb feature sets of Sentence #1 and Sentence #2 is given. Detected interaction verbs are underlined.

   Sentence #1 : *In contrast, in melanoma cells, free E2F DNA binding activity (PROTX0 and PROTX0, to a lesser extent PROTX1, PROTX0, and occasionally PROTX2), was constitutively maintained at high levels independently of external melanocyte mitogens.*

   Sentence #2 : *Both PROTX0(+) and PROTX0(+) subsets of allospecific T lymphocytes are required: PROTX0(+) T cells drive the synthesis of pro - PROTX0 through PROTX00 engagement but have effects on pro - PROTX0 processing; PROTX0(+) T cells, unable to induce synthesis of pro - PROTX0 per se, are responsible for the generation of mature PROTX0 by pro - PROTX0 - producing DCs.*

TABLE I: Interaction verbs feature set

	Interaction Verbs									
	activity	binding	drive	effect	engagement	induce	producing	...	IV-n	0
Sentence #1	1	1	0	0	0	0	0	...	0	0
Sencente #2	0	0	1	1	1	1	1	...	0	0

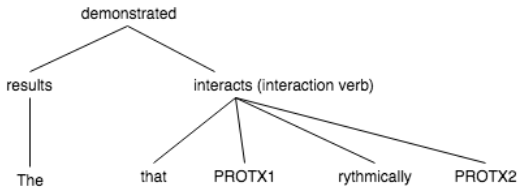
2. Dependency Parse Tree Features

   Each sentence is parsed using Stanford Parser [4] to obtain dependency parse tree of the sentences. Additional to baseline features, a new feature is extracted from dependency tree: if distances of PROTX1 and PROTX2 to the interaction verb in dependency tree is equal and equal to 1; this feature is set to 1, otherwise it is set to 0.

   In Fig. 1, a sample dependency tree of Sentence #3 is given.
   Let $\delta(n1, n2)$ defines the distance between $n1$ and $n2$. In Fig. 1, $\delta(PROTX1, InteractionVerb) = \delta(PROTX2, InteractionVerb) = 1$ so that, for this sentence, we set dependency-tree feature to 1.

   Sentence #3 : *The results demonstrated that PROTX1 interacts rhythmically with PROTX2.*

FIG. 1: Dependency tree of Sentence #3



3. Machine Learning Techniques In this task, the goal is to determine if a sentence hides a relationship or not, so that using a binary classifier is appropriate. As a strong binary classifier, SVM implementation of *scikit-learn* [3] Python library is used.

   In both of baseline and advanced systems, dataset is split into two parts; train set contains 3000 documents and test set contains the rest 1056 sentences.

   For the baseline system, SVM is trained with only TF-IDF values of training set. A linear kernel is used with SVM. Then trained SVM is used for prediction of test set. Prediction is run for 10 times, then results are averaged.

   For the advanced system, two separate smaller utility programs are written: a Python program two extract interaction verb feature set and a Java application to extract dependency tree distance feature. Before training the SVM, baseline feature set is enhanced with these two feature sets, then SVM is trained and prediction is run.

   For both of the systems, Accuracy, F-Measure, Precision and Recall values are calculated.

III. RESULTS AND OBSERVATIONS

Baseline system has an average accuracy of 76.23% and additional features increased the average accuracy to 85.32%. Precision, Recall and F-Measure values are given below.

A. Baseline System

TABLE II: Baseline System Results

	Precision	Recall	F-measure
Class Label: 0	0.77	0.66	0.71
Class Label: 1	0.76	0.84	0.80
Avg / Total	0.76	0.76	0.76

B. Advanced System

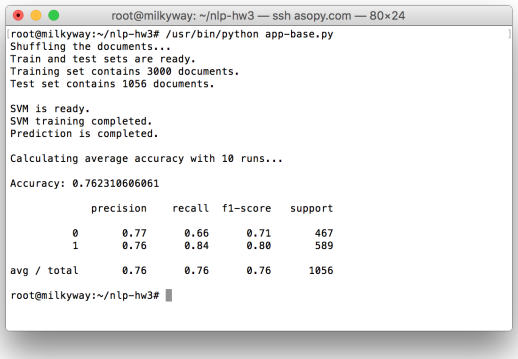
TABLE III: Advanced System Results

	Precision	Recall	F-measure
Class Label: 0	0.86	0.82	0.84
Class Label: 1	0.85	0.88	0.87
Avg / Total	0.85	0.85	0.85

IV. SCREENSHOTS OF THE PROGRAMS

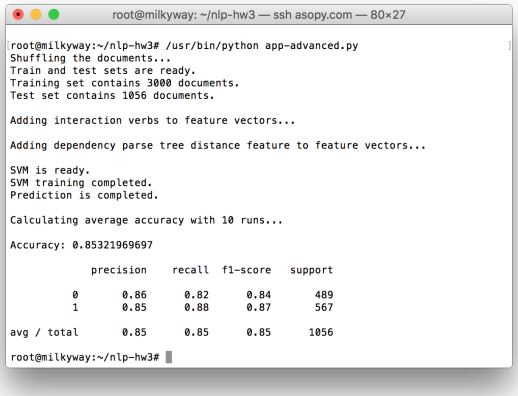
A. Baseline System

FIG. 2: Baseline System - Sample Run



B. Advanced System

FIG. 3: Advanced System - Sample Run



V. COMMENTS ON THE RESULTS

There are two important observations in this experiment. First of all, using TF-IDF with SVM creates a quite strong classifier for binary classification systems. Using only the words in the sentences are capable of classification ratio of 76.23%. Secondly, additional features including interaction verbs and dependency parse tree features increased the overall ratio to 85.32%.

On notable point on the results is follows: in the baseline system, Recall value of Class Label-0 is significantly lower than Recall value of Class Label-1 (0.66 to 0.84). In the advanced system, recall-0 is still lower but the gap is closed (0.82 to 0.88). Hence, additional features have an important effect on finding more evidence for the classifier to classify the document. This is supportive for my proposal of adding more features to feature set.

VI. NOTES AND FUTURE WORK

As it is seen that enriched feature set constitutes better accuracies, we can enrich the feature set with other extra features.

It is worth to mention that using proteins with equal distance to interaction verb creates a subset of all documents in the dataset. So that, other features from dependency parse trees such as interaction verbs with distance 2, or immediate parent nodes of each protein in the tree even it is not an interaction verb will expectantly increase overall accuracy.

VII. CONCLUSION

To conclude, in this study, protein-protein relationships are extracted using BoW-TF-IDF method with an SVM classifier. It is observed that additional features contribute to overall accuracy. Since only 2 new feature sets are added and they increased the accuracy, with adding other dependency parse tree features, overall accuracy can be improved considerably.

[1] G. Erkan, A. Ozgur, DR. Radev "Extracting interacting protein pairs and evidence sentences by using dependency parsing and machine learning techniques.," Proceedings of the Second BioCreative Challenge Workshop, 2007.

[2] C. Blaschke, M. A. Andrade, C. A. Ouzounis and A. Valencia "Automatic extraction of biologicalinformation from scientific text: Protein-protein interactions." InProceedings of the AAAI Conference on IntelligentSystems for Molecular Biology (ISMB 1999), pages 60-67.

[3] Pedregosa et al. "Scikit-learn: Machine Learning in Python," JMLR 12, pp. 2825-2830, 2011.

[4] D. Klein, C. D. Manning "Accurate Unlexicalized Parsing." Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423-430. 2003.