

# Using Natural Language Processing and the Gene Ontology to Populate a Structured Pathway Database

David Dehoney, Rachel Harte, Yan Lu, and Daniel Chin

PPD Discovery, Inc

{david.dehoney, rachel.harte, yan.lu, daniel.chin}@menlo.ppd.com

## Abstract

*Reading literature is one of the most time consuming tasks a busy scientist has to contend with. As the volume of literature continues to grow there is a need to sort through this information in a more efficient manner. Mapping the pathways of genes and proteins of interest is one goal that requires frequent reference to the literature. Pathway databases can help here and scientists currently have a choice between buying access to externally curated pathway databases or building their own in house. However such databases are either expensive to license or slow to populate manually. Building upon easily available, open-source tools we have developed a pipeline to automate the collection, structuring and storage of gene and protein interaction data from the literature. As a team of both biologists and computer scientists we integrated our natural language processing (NLP) software with the Gene Ontology (GO) to collect and translate unstructured text data into structured interaction data. For NLP we used a machine learning approach with a rule induction program, RAPIER (<http://www.cs.utexas.edu/users/ml/rapier.html>). RAPIER was modified to learn rules from tagged documents, and then it was trained on a corpus tagged by expert curators. The resulting rules were used to extract information from a test corpus automatically. Extracted Genes and Proteins were mapped onto Locustlink, and extracted interactions were mapped onto GO. Once information was structured in this way it was stored in a pathway database and this formal structure allowed us to perform advanced data mining and visualization..*

## 1. Introduction

The motivation for this tool was to speed the process of parsing interaction data from the literature and populating it into our in-house pathway database. Initially we tried co-occurrence [1]. While useful for discovering information it was inappropriate for populating our formal database. We decided to use an NLP approach to increase precision and get more details about interactions.

Based on work by Bunescu et al [2] we modeled each relationship as having two necessary components:

Interactors and Interactees. We also added one optional component: Interactions.

## 2. Tagging

Three expert biologists marked up a training corpus of 70 abstracts, tagging gene-gene- gene-protein, and protein-gene relationships. For each relationship the Interactors, Interactees and Interactions were tagged. Eg.

“In vitro, <Interactor> <protein> MRCKalpha </protein> <Interactor> <Interaction type = phosphorylation> phosphorylates </Interaction> the protein kinase domain of <Interactee> <protein> <Gene> LIM </Gene> kinases </protein> </Interactee>”

Figure 1: An example of a tagged relationship

## 3. Machine Learning

Rapier is a machine learning tool that learns information extraction rules from a set of documents and associated templates [3]. In ML parlance this is called **grammar rule induction**. We modified Rapier to work on tagged documents directly instead of templates. We ran this instance of Rapier on our manually tagged training corpus to produce a set of grammar rules. Three kinds of rules exist for our application: Interactor rules, Interactee rules and Interaction rules. An example follows in Figure 2:

POS: Noun phrase; Semantic: Protein  
Word: 'is'  
POS: Verb past participle  
Word: 'by'  
**POS: Noun phrase; Semantic: Protein**

Figure 2: Interactor extraction rule, 4 context lines followed by one extraction line (bolded)

This rule would correctly extract the Interactor from sentences such as “Protein A is inhibited by Protein B” (in this case, Protein B). It is interesting to note that the rules operate on three levels: word, part-of-speech, and semantic class.

## 4. Information Extraction

Once a set of grammar rules was created we used it to extract information from a test corpus of abstracts that we had previously inspected manually. Sentences were read one at a time and for each sentence our rule file was applied to extract Interactors, Interactees and Interactions (for simplicity, relationships were assumed to be expressed in a single sentence). If a sentence contained at least one Interactor and at least one Interactee a relationship was called between these entities (Refer to [2] for details on handling multiple Interactors and Interactees). If any Interactions were extracted they were then attributed to the relationship.

## 5. Ontology Mapping

Before a relationship could be stored in the database it had to be structured. The Interactors and Interactees were all mapped onto Locuslink, and Interactions were mapped onto a slightly modified version of GO [4]. GO terms are commonly used to label the processes and functions a given gene product can participate in. We extended that idea by recording which particular processes or functions the gene products were involved in at the time of this relationship.

Mapping was done using a table lookup: extracted terms were matched against lookup tables to find a reference symbol. This reference symbol was then stored in the database entry for the relationship.

The Interactor/Interactee lookup table was created from the Locuslink alternate symbol table. The Interaction Synonym Table was initially created manually and then expanded using the abstracts tagged in step 2. For example, after parsing the sentence in Figure 1 the following entry would be added:

Table 1: Example entry in Interaction Synonym Table

Ref ID	Ref Symbol	Synonym
GO:0016310	Phosphorylation	phosphorylates

## 6. Results

From our training corpus of 70 abstracts we had 145 labeled Interactors, 169 labeled Interactees and 179 labeled Interactions. From that Rapier induced 71 grammar rules for Interactors, 79 rules for Interactees, and 66 rules for Interactions. These grammar rules were then applied to a testing corpus of 10 abstracts to tag and extract 13 Interactors, 12 Interactees and 16 Relationships. Results are displayed in Table 2.

Table 2: Results for Rules against Test Set

	Recall	Precision
Interactor	39%	85%
Interactee	27%	75%
Interaction	24%	50%

## 7. Discussion

Overall results for NLP were encouraging. Recall was low but over a large set of documents precision is more important. We were thus pleased with the results for Interactors and Interactees. Interaction precision was on the low end and we're looking into improvements. We also decided to follow the work of Donaldson et al [5] and introduce a human reviewer to approve data entry.

Ontology Mapping worked very well. We chose to use Locuslink for our Interactors/Interactees and GO for our Interactions because both are widespread and freely available. Unfortunately, not all genes are recorded in Locuslink and even for those that are some concepts, such as protein domains and alternative splicing, aren't supported in a formal way. Future versions of our tool will resolve these issues. Two limitations of GO required us to add our own symbols to the ontology. Firstly, GO is meant for healthy processes, but we were also interested in pathological processes. Secondly, there are gaps in the GO hierarchy. For example, while **activation of MAPK** and **activation of SoxR protein** both exist, there is no generic **activation of protein**. In future versions we will consider other ontologies such as Celera's PANTHER [6].

## 8. References

- [1] B.J. Stapley and G. Benoit. *Biobibliometrics: Information retrieval and visualization from co-occurrences of gene names in medline asbtracts*. PSB, 529-540, 2000.
- [2] R. Bunescu, R. Ge, R. Kate, E. Marcotte, R. Mooney, A. Ramani, Y. Wong. *Learning to Extract Proteins and their Interactions from Medline Abstracts*. Submitted for Publication. <http://www.cs.utexas.edu/users/ml/publication/ic-abstracts.html>
- [3] M. Califf and R. Mooney, *Relational Learning of Pattern-Match Rules for Information Extraction*, AAAI, 328-334, 1999.
- [4] <http://www.geneontology.org/>
- [5] I. Donaldson, J. Martin, B. Bruijn, C. Wolting, V. Lay, B. Tuekam, S. Zhang, B. Baskin, G. Bader, K. Michalickova, T. Pawson, C. Hogue. *PreBIND and Textomy - mining the biomedical literature for protein-protein interactions using a support vector machine*. BMC Bioinformatics 4:11, 2003.
- [6] <http://panther.celera.com/>