Extracting Protein-Reaction Information from Tables of Unpredictable Format and Content in the Molecular Biology Literature

Sam Sloate, Vincent Hsiao, Nina Charness, Ethan Lowman, Chris Maxey, Guannan Ren, Nathan Fields, and **Leora Morgenstern**
Corresponding author: leora.morgenstern@leidos.com
Leidos
Arlington, VA 22203

Abstract

Tables in technical papers often provide much useful information that is not present in text. This paper focuses on the specific problem of *automated table reading (ATR)*: automating the interpretation and extraction of information from protein-reaction information (PRI) tables in the molecular biology literature (MBL), in the context of a general knowledge-based approach to automating table reading.We report on the results of our system evaluation, which demonstrated a precision of greater than .9 in identifying relevant tables and .8 in mapping tables to correct relational schema.

1 Introduction

The work reported in this paper was motivated by and performed for the DARPA Big Mechanism research project, which aims to build a large model of human cancer signaling pathways that could potentially be used for hypothesizing novel causal relationships that might inform cancer treatments. An initial step in constructing the model is collecting all human molecular pathway fragments that have been published in the literature; these fragments would then be assembled into larger pathways relative to such considerations as the context of the experiment in which the pathway fragments were observed, and the strength of the evidence.

The largest existing human cancer signaling pathway database, Pathway Commons [Cerami *et al.*, 2011], contains information about tens of thousands of manually curated pathway fragments (PFs) and associated protein reactions (PRs) that are imported from databases like Reactome [Croft *et al.*, 2014]. Pathway Commons is estimated to contain only 1% of the human PFs and protein reaction information (PRI) that exist in the molecular biological literature (MBL). Given the ever-increasing number of papers published containing information about PFs and PRs (more than 10K per year), it is likely that the gap between pathway databases like PC and the body of information published in MBL will continue to grow, unless reading these pathways and PRI is to some extent automated.

Although much PRI can be retrieved from the text of MBL papers [Valenzuela-Escarcega *et al.*, 2015], tables are a particularly valuable source of PRI. While PRI tables are not common, when they exist they can contain many times the amount of information as text — sometimes thousands or tens of thousands of PFs — most of which is not found in text. Thus, not reading tables would result in missing large amounts of PRI. Moreover, tables often give richer information than is present in text, e.g., details about the site of a reaction and measurements of increase or decrease of phosphates in the substrate.

This paper describes a general approach for automating the extraction of information for tables based on automating the mapping of set of columns of a pre-determined relational schema to sets of columns in a table, and our implementation of this method for automating the extraction of PRI from tables. Although reading tables avoids many of the difficulties of natural language processing (NLP), there are many other difficulties to be solved, most saliently that
(1) There is no standard schema for representing PRI in tables. Indeed, we have discovered hundreds of schemas in MBL tables. This means that (2) It is difficult to determine whether an MBL table is a PRI table. (3) There is no standard representation used for specific types of information, e.g., types of phosphorylation and site information. (4) Much of the information in PRI tables, including the participants of a reaction (the relevant entities of a pathway fragment), is not explicit in text, but must be derived from other sources, including surrounding context. Some of these issues have been noted, though not solved, by previous research to automate table reading [Pimplikar and Sarawagi, 2012; Cafarella *et al.*, 2008; 2009]; some arise from the specific domain studied, though as we point out in the conclusion, analogous problems exist for tables in other highly technical or specific domains.

1.1 Scope

Much research on automated table reading (ATR) [Hurst, 2000; Wang and Hu, 2002] has focused on the problem of table detection and on labeling rows and columns, since performing these tasks is generally a prerequisite to automated table reading. This is not the focus of this paper. All tables in this study were found in papers retrieved from the PubMed PMC website at http://www.ncbi.nlm.nih.gov/pmc/, and we were therefore able to take advantage of the format in which articles and tables are represented on that site. For articles in that website, we have developed methods for reading both

tables included in the body of an article (in either HTML or NXML) and tables included in supplementary material (these are generally in Excel format). Although there are interesting technical issues which we needed to solve to do this work – e.g., determining row and column content in tables in which there are subheaders spanning multiple columns; and determining table extent, particularly for Excel tables in which cells adjacent to tables are used to represent other sorts of information – these are not within this paper's scope. A separate document describes these results.

2 Motivating Examples, Technical Challenges

We define the general automated table reading problem as follows: Given a target relational schema $R(x1 \ldots xn)$ and a table with columns $T(c1 \ldots cm)$, can we determine a a mapping between subsets of $x1 \ldots xn$ and subsets of $c1 \ldots cm$, and extract information corresponding to the mapped subsets? Note first, that we are looking to map subsets onto subsets rather than individual columns because a table can spread out information over several columns or compress several columns into one; and second, that this definition can be generalized to multiple relational schema: most tables will map to at most one relational schema, but some tables can map to several schemas.

For the particular problem of reading PRI tables, this problem can be stated in the following domain-specific way: From a given table, can we extract relation instances of the form *R(A,B,I,M,S,Q,N)* where *A* and *B* are Participants A (a protein or chemical) and B (a protein, the substrate) in some interaction; *I* is the reaction or interaction type itself (e.g., increases or decreases activity); *M* is the type of modification (e.g., phosphorylation or acetylation); *S* is the site at which the reaction takes place; *Q* is quantified information, e.g., the increase in molecules in the substrate, most often represented as a multiplier (fold change) or ratio, or the log of the ratio; and *N* tells whether or not the information is negative. Other teams on the project, whose systems read text rather than tables, looked for all these fields except for *Q*, which is generally not present in text. *Q* gives important information on the magnitude of the modification induced by a reaction; this can often be used to tell if the modification is considered to be significant. E.g., if the ratio is between .6 and 1.6, the reaction is considered by many researchers to have little effect. One can interpret the corresponding line of the table as containing negative information about the interaction; that is, it has not been shown to have a significant outcome.

It is rare that all information for all fields is present within a single table, though often much information can be inferred.

The technical challenges we faced are best understood by examining several sample tables. Consider Figure 1, which shows a fragment from one of the simplest of the PRI tables in the MBL. There are four columns shown in this fragment. [1] The first two columns are synonyms – the first gives a

[1] In the full table, there are six columns. The fifth column contains information similar to the fourth column but for a different cell line. The sixth column gives information about whether a substrate is an src-inhibitor. Both provide useful data, but such information is out of scope for this paper.

Table 1. Peptides with CSF-1R induced tyrosine phosphorylation in MCF-10A c

Protein Name	UniProt#	NCBI Site	Fold change* (CA-CSF-1R/MCF-10A)
Transferrin receptor	P02786	Y20	313.67
APLP2	Q06481	Y750	73.37
D-Prohibitin	Q99623	Y248	>55.42
PAR3	Q8TEW0	Y489	>34.42
PKM2	P14618-2	Y105	29.01
PIK3R1	P27986	Y470	>24.59

Figure 1: Table 1 fragment from PMCID 2962495

name for a protein, while the second gives the corresponding Uniprot identifier – for Participant B. (If these columns were not synonymous, it would be reasonable to hypothesize, subject to some checks, that the two columns represented Participants A and B. The third column gives site information (*S*), and the fourth column gives quantified information (*Q*).

This example demonstrates one of the difficulties of automated table reading: many desired pieces of information are not explicit in tables, but must be inferred and/or extracted from other sources. In this example, Participant A can be read from the table title, which gives CSF-1R as the protein acting on the substrate shown in columns 1 and 2. The table title also gives *M*, the type of modification, in this case (tyrosine) phosphorylation. This information can also be read from the values in the third column: a subset of the letters (S,T,Y) (corresponding to three common types of residue in phosphorylation), or the letter P, in a column that gives site information, are two common ways of indicating that the modification is phosphorylation. *N* can be deduced from the values in the *Q* column: in this case, the fold value is always high; thus all lines in the table give positive information. One can also infer the reaction type *I*, in this case increases activity, from the fold change.

An added complexity is that often information about Participant A is bundled with *Q*. For example, in Figure 2, the rightmost four columns give quantified information: the subheader expresses the fact that these columns gives the log of the ratio of the change of molecules in the substrate, comparing dDAVP (desmopressin) to a control. That is, dDAVP is Participant A. (dDAVP's role as Participant A can also be read from the table title, not shown; but this is not generally the case when it also coupled with *Q* information.) Note also that the single column type / argument *Q* is given over four columns, as a time-series. Of special interest are cases (e.g., 2nd line) where dDAVP initially *inhibits* the reaction, although after a few minutes, it *activates* the reaction. Should this be considered an activation or an inhibition? Our approach is to check for the greatest absolute value of change and to assign the reaction type based on that value, but it is clear that extracting the desired information from the table is not trivial.

Figure 3 shows a table with positive information, negative information, and for some lines of the table, a *lack* of information. Note also that in this table, a single column contains

RefSeq	Gene symbol	Peptide sequence	Phosphosite	Average $\log_2$(dDAVP/control)			
				0.5 min	2 min	5 min	15 min
NP_203501	*Plcb3*	NNS#ISEAK	Ser-1107	0.41	0.68[b]	0.65[b]	0.67
NP_036914	*Bckdha*	IGHHS#TSDDSSAYR	Ser-338[a]	−0.16	−0.26	−0.07	1.00[b]
NP_476476	*Pfkfb3*	NSVTPLAS#PEPTK	Ser-496	−0.26	0.76[b]	−0.08	0.40
NP_001094371	*Snta1*	NSAGGTSVGWDS#PPASPLQR	Ser-183[a]	−0.09	0.56	−0.33	1.12[b]
NP_001019489	*Asb4*	S#LPLSLK	Ser-408	1.40[b]	1.96[b]	1.20	1.24[b]

Figure 2: Table 1 fragment from PMCID 3277771

both site and kinase information (these must be teased apart in order to populate relation instances) and columns containing free text may appear in the table.

Figure 4 shows a typical Excel table; such tables are often found in a paper's supplementary materials. While Excel tables often have titles or other information in an area of the spreadsheet outside the bounds of the table, or on the tab of the spreadsheet, this file has neither. The link in the main paper pointing to this spreadsheet contains some hints about the table's contents. However, it is extremely difficult even for a human to understand this table without reading much of the paper. E.g., how could a human find any evidence of Participant A? Note also that there appear to be multiple experiments, or at least multiple modes of measurement that are reported in this table.

3 Technical Approach, Architecture

In general, automating the extraction of information from tables relative to a particular schema requires succeeding at two tasks: 1) Automating the identification of relevant tables and 2) Automating the correct extraction of information from tables marked as relevant.

Our research and experimentation has shown that only a small fraction of tables collected from the PMC site are PRI tables. We estimate that 2-3% of papers filtered against a set of search terms corresponding to protein reactions contain PRI tables, and that 1% of papers on the PMC site are PRI tables. For example, clinical tables or tables reporting on cell functionality would be unlikely to have PRI even if the papers themselves make some mention of protein reactions. Such rareness does not negate the usefulness of automating PRI table reading: when PRI tables are found, they often have hundreds or even tens of thousands of lines of data that are not present in text.

However, it did suggest that statistical classification approaches, as used by [Pimplikar and Sarawagi, 2012] in querying tables, were unlikely to be useful as an initial strategy: it would be too difficult to find (even using the help of Elasticsearch) and label even the relatively small training sets required by simple Support Vector Machines. Instead, we opted to develop a rule-based approach. In any case, we opted to use an approach consistent with that of University of Arizona [Valenzuela-Escarcega *et al.*, 2015], another performer in Big Mechanism. They have used a rule-based system which has excelled in precision and throughput in extracting PRI from MBL texts.

We note that if successful, a rule-based system could be used to automate the finding and labeling of training data; that is, it would enable distant-supervision of learning.

The system's architecture is shown in Figure 5.

3.1 Determining relevance

To be relevant, a table does not have to contain *all* information needed for a complete relation instance of *R(A,B,I,M,S,Q,N)*. For PRI, a table should at least contain the following: (i) evidence of two non-synonymous proteins, (ii) evidence of a protein reaction (such as a post-translational modification), and (iii) some quantified information concerning the reaction. As discussed, these pieces of information can often be combined to obtain further elements of the relation instance, but even if that cannot be done, the combination of such evidence makes the likelihood that a table is PRI-relevant quite high.

Given this smaller set of required information, the difficulty reduces to finding and recognizing (i), (ii), and (iii). We need to look for evidence in (a)columns, (b) column headers, and (c) table titles/captions, as well as (d) possibly text outside the table as well. Together, (a) and (b) constitute *column identification*.

3.2 Column identification

Rules for identifying columns are based on domain knowledge. For PRI tables, they include the following:

Proteins

Columns of proteins, which almost always consist of instances of Participant B, are generally easy to identify. Typically protein columns are labeled with the name of the protein database whose nomenclature is being used; even without the column label, regular expressions can be used to identify entries as proteins. The task can be complicated by cell entries that consist of multiple proteins, or proteins along with other entities. Sometimes proteins are given as peptide subsequences; this often allows combining information about modification type, site, and Participant B.

Modification type

This information is often explicitly in column headers or table titles. Related terms may be used: e.g., "kinase" indicates that the modification is phosphorylation. The most common residues on which phosphorylation occurs, serine, tyrosine, and threonine, are represented by several common sets of abbreviations; their appearance in a column indicates phosphorylation. Marking the site at which a modification occurs by a differently-cased letter indicating the modification (e.g., *m* for methylation, *p* for phosphorylation) also gives useful information.

Site Information

Site information is often combined with information indicating modification, as in the NCBI site column in Figure 1 or the Phosphosite column in Figure 2. As the Site (putative kinase) column in Figure 3 shows, site information is often mixed with other specific information in a single table cell. In addition, multiple sites are often furnished within a single cell.

Table 1. IMCD phosphoproteins of potential relevance to vasopressin signaling/AQP2 trafficking

ID	Site (putative kinase)	Swiss-Prot	Spectrum	MS quant ratio
AQP2	***S256**** (PKA), S261* (p38), S264 (PKC), S269 (PKA)	P41181	MS2, MS3, MS4	2.67 ± 0.84**
AQP4	S321 (PKC)	P55087	MS2, MS3	
Arrestin, β1	***S412*** (ERK1/2)	P49407	MS2, MS3	1.46 ± 0.34
Bcl-2-associated transcription factor (Bclaf1)	*S177* (Cdk5)	Q9NYF8	MS2, MS3	0.33 ± 0.07**

Figure 3: Table 1 fragment from PMCID 1459033

No.	Proteins	Protein Names	Gene Names	Localization Prob	Score Diff	Number of Phospho (STY)	Amino Acid	Sequence Window	PEP	Mascot Score	PTM Score	Modified Sequence	Phospho (STY) Probabilities	Phospho (STY) Score Diffs	Position in peptide	Charge	m/z	Mass Error [ppm]	Ratio H/L	Ratio H/L Normalized	Ratio H/L rep1	Ratio H/L rep2
1	IPI:IPI001(	Probable u	Usp9x;Fafl	0.771469	10.0592	1	S	KVISSVSYY	0.00587	41.68	94.358	_VISSVS(ph)YYTHR_	VISSVS(0.771)Y(0.076)Y(0.076)T(0.076)HR	VIS(-43.31)S(-36.13)VS(10.06)Y(-10.06)Y(-10.06)T(-10.06)HR	6	3	464.55002	0.81605	1.0375	1.4098	1.0375	
2	IPI:IPI011	60S riboso	Rpl4;Rpl4;	0.602649	1.80909	1	S	PLISVYSEK	2.53E-2	61.18	176.78	_PLISVYS(ph	PLISVY(0.3	PLIS(-55.49)VY(	7	2	830.89512	-1.921	0.39693	0.49311	0.39693	
3	IPI:IPI011	Fatty acid	Fasn	0.5	0	1	S	GEGHLYSR	0.00150	14.26	81.177	_VAEVLAGEG	VAEVLAGE	VAEVLAGEGHLY	13	3	527.58713	-0.4300	0.77339	0.87191		0.77339
4	IPI:IPI0056	Focal adhe	Ptk2;Fadk	0.550761	0.896334	1	S	TANLDRSN	1.27E-1	31.58	170.84	_S(ph)NDKVY	S(0.551)N	S(0.9)NDKVY(-0	1	3	549.26595	-1.4132	1.167	1.2348		1.167
5	IPI:IPI0021	Dynactin s	Dctn2;Dct	0.676259	7.97037	1	S	KRTGYESG	3.17E-4	70.59	215.04	_RTGYES(ph)	RT(0.108)(	RT(-7.97)GY(-7	6	3	719.64505	-0.5130	0.37604	0.5185	0.37604	
6	IPI:IPI0085	PRP4 pre-(	Prpf4b;mC	0.961356	13.9589	1	S	ITPYLVSRF	2.88E-4	72.03	219.96	_LCDFGSASH	LCDFGSAS	LCDFGS(-152.0	21	2	1259.0594	-0.1533	1.1414	1.2011	1.1414	
7	IPI:IPI011	Tubulin be	Tubb5;Tub	0.653208	6.28245	1	S	GTYHGDSE	3.24E-2	47.33	171.67	_FWEVISDEH	FWEVISDE	FWEVIS(-41.82)	21	4	796.34893	0.5829	0.64727	0.67945	0.64727	
8	IPI:IPI011	Breast can	Bcar1;Cas	0.93433	12.3207	1	S	LSSSHHSV	8.63E-1	41.63	146.92	_GLLSSSHHS(	GLLS(0.00	GLLS(-20.19)S(	9	2	1038.4935	-0.9720	1.0477	1.2354		1.0477

Figure 4: Table 3 fragment (Excel), PMCID 3229182

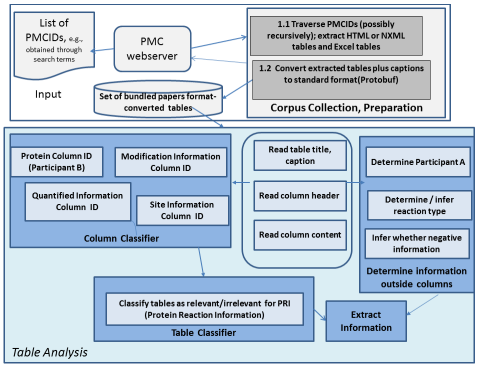

Figure 5: System Architecture

Quantified Information

Quantified information is displayed as columns of real numbers or real-numbered intervals. It is often difficult to tell from inspection of the column content that these real numbers describe increases or decreases in molecular activity in substrates relative to a control. This information must almost always be obtained from the column header itself. What makes obtaining this information difficult is that there appear to be scores of ways of labeling the header of this sort of column. Some examples include KO/WT (knock-out / wild-type), indicating that one is testing how knocking out a gene affects a modification relative to the wild-type control, KD/WT (knockdown/wild type), H/V (hormone / vehicle), ratio, fold-value, multiplier, and many column headers beginning with "log." Recognizing all instances of such column headers could still be improved. In more recent work, we have been exploring the use of clustering to help identify related column headers.

3.3 Discovering Participant A

It is almost never the case that Participant A is included in its own column in a table. This is mostly because of the nature of experiments in this domain: one catalyst is tested on multiple substrates. We use several strategies to find Participant A: 1. Look for mention of a suitable entity in the table title or caption. Assuming, however, that any protein or chemical mentioned in the title must be Participant A leads to lowered precision. 2. Look for mention of a suitable entity after a "log" term in a column header. This strategy is also error prone. What follows the "log" is often a short phrase that describes in some way the nature of the experiment; mention of Participant A is not the only possibility. 3. Scan the paper title, introductory paragraph of the paper, and/or first paragraph of the methods and material section for mention of suitable entities. This strategy is also error prone, since there will typically be multiple proteins mentioned in such paragraphs.

These strategies are best understood as ways of generating hypotheses about Participant A; corroborating evidence is supplied if other strategies also lead to the same Participant A. Once a Participant A is hypothesized, the system further checks its hypothesis by iterating on the contents of the column that has been identified as containing instances of Participant B. It checks the text of the paper for sentences of the form *A [modifies] B* or *B is [modified] by A*. Any such instances further raise the confidence metric that Participant A

has been hypothesized correctly.

3.4 Other Information and Inference

Often the nature of a reaction – e.g., whether it increases or inhibits activity – is mentioned directly in a table title. Such information can also be inferred from Q, e.g., by noting whether a ratio is greater or less than 1.

Whether information is negative must also often be inferred, especially when time-series information is given,

3.5 Extracting information

The end goal of our system is extracting information on protein reactions that can be used by knowledge bases like Pathway Commons. To that end, when a table is determined to be relevant, all information corresponding to fields in the target schema is extracted into a BioPAX-consistent format [Rodchenkov *et al.*, 2013]. Proteins are converted to their equivalent in Uniprot whenever possible.

4 Evaluation

4.1 Training and Test Sets

We formed a training and test corpus of papers and tables by entering the search term "phosphoproteomics" into the PMC search engine at http://www.ncbi.nlm.nih.gov/pmc/ in May 2015. The term "phosphoproteomics" was chosen to increase the likelihood that papers returned would include PRI tables. Around 3-4% of papers returned contained at least one PRI table. Thus, the task of finding PRI-relevant tables was still very difficult; however, it made it somewhat easier for humans to create a gold standard with PRI-relevant tables.

Of the more than 2200 PMCIDs (corresponding to papers) that were returned, we used papers with trailing digits 1-6 as a training set. We reserved papers with trailing digits 8 and 9 for internal testing purposes, and reserved papers with trailing digits 7 and 0 for the test whose results would be reported to DARPA.(We consulted with PubMed and PMC to ascertain that trailing digits of papers are assigned randomly, so that no bias was introduced in thus creating our training and test sets.) We did not touch any of the papers in the reserved set, either processing them or manually inspecting them, until the time of the test. (Automated preprocessing to scrape and find tables can take several days, so this was done several days before the test was run.)

We were also provided with a set of 1000 PMCIDs from MITRE, the evaluation team for Big Mechanism, to be used as a test set. As with our own test sets, we did not process or inspect these until the time of the test.

The test aimed to answer two questions:

Question 1: How well could the system find relevant tables, that is, tables with protein reaction information? Specifically, could it be shown that the system performed statistically significantly better than random?

Question 2: How accurately could the system extract information? That is, given a table, would the system extract information correctly?

For Question 1, we aimed to compute precision, recall, and an F1 score. For Question 2, we focused on precision.

4.2 Evaluation of Question 1

Human Gold Standard Development

Two team members with knowledge of molecular biology and who had not been involved in development of the table reading system created a Human Gold Standard for Question 1. First, they were shown examples of relevant and irrelevant papers from the training corpus. Then, they were given the test set of "phosphoproteomics" papers, trailing digits 7 and 0. There were 515 papers in this test set and 977 tables. Using Elasticsearch, the Human Gold Standard developers searched for terms that were similar to those in the examples of relevant papers in the training corpus.

The two team members worked separately. Inter-annotator agreement was high, near 90%. The few tables that they did not agree on were discarded. The size of the Human Gold Standard was limited by the number of relevant tables that these team members could find (30). They had similar patterns of being able to find some relevant tables quickly (the equivalent of low-hanging fruit), and then slwoing down until they got to the point that it was too frustrating to continue. The Human Gold Standard thus consisted of 30 relevant tables and 30 irrelevant tables.

Results

For the Question 1 test, the system was input the 60 tables of the Human Gold Standard and labeled each table as relevant or irrelevant. It achieved precision of .93 and recall of .5, for an F1 score of .65. This is statistically significantly better than random.

The scores were consistent with (though a bit lower than) earlier testing on the internal test set, trailing digits 8 and 9. Precision was consistently excellent (usually perfect), while recall hovered between .5 and .6.

Evaluation of Question 2

The system ran on the "phosphoproteomics" corpus, trailing digits 7 and 0 (515 papers, 977 tables) and on the corpus supplied by the government evaluation team (1000 documents, 646 tables). The system labeled 30 tables from the first test set as relevant and extracted more than 12,000 protein reactions. Note that the system labeled 30 tables as relevant, even though of the 30 relevant tables in the Human Gold Standard, it only labeled 15 as relevant. The reason for this is that the system was able to find relevant tables that the humans were not able to find. (We checked to make sure that the system had indeed found relevant tables.) Given the small percentage of relevant tables in any corpus, and the difficulty of searching for such tables, even with Elasticsearch, this is not surprising. In other words, even given virtually unlimited time, human recall of (ability to find) relevant tables is no better than system recall.

The system labeled 11 tables from the second test set as relevant and extracted 585 protein reactions.

We examined the 11 tables labeled as relevant. 10 were labeled correctly, but one of the tables was irrelevant, giving a precision score of just under .91. To score the precision of the information extracted, we followed the rubric that the government evaluation team was using for text-reading systems. An entirely correct schema mapping received 1 full point. Half

a point was deducted for an error (e.g., getting Participant A wrong or missing Participant A if it existed in the table (even in a caption or column header), getting Participant B wrong, or getting the modification type wrong.) Thus, any more than one error resulted in a score of 0.

We examined each table to determine whether the system had correctly mapped the subset of columns it selected onto the desired relational schema. We randomly selected three lines of each table for inspection. In all cases, the three rows agreed with one another and appeared to fairly represent the table. We scored .8 on precision of correct mapping to relational schema. Most of our errors resulted from missing Participant A when it was in a caption or column header, or in getting Participant A wrong. Aside from Participant A, we achieved a precision of .95.

4.3 Subsequent Progress and Evaluations

Following the evaluation, we revamped the system with multiple objectives:

1. Making the code more efficient, so that we could run test-and-fix cycles more efficiently.
2. Fixing bugs
3. Improving recall by recognizing different types of post-translational modifications, as well as recognizing unusual choices on the part of table designers. For example, in the same way that authors often stuffed two types of information into one column (e.g., Participant A and quantified information), they also separated into multiple columns information that might be expected to stay together, such as phosphorylation site and phosphorylation residue.

After updating our system, we conducted an expanded search and collected more than 3500 PMCIDs, yielding more than 400 relevant tables from 91 PMCIDs. We extracted more than 120,000 protein interactions from these tables. We then selected 9 tables from new PMCIDs that we had not previously inspected or processed in prior evaluations, and evaluated precision. This time, we graded individual rows, and inspected more rows per table to make sure that we were not overlooking possible errors.

Our precision was 1.0 for Participant B, .89 for modification type, .93 for site information, and 1.0 for negative information. We continued to do poorly in recognizing Participant A, however, and this remains an area of current researh.

5 Related Work

Much work on table reading has focused on detecting tables or elements of tables, such as columns, rows, headers, and stubs. [Fang *et al.*, 2012; Hurst, 2000; Wang and Hu, 2002]. While such work is clearly important to the general problem of table reading, it is not very relevant to our current work for two reasons. First, for the large PMC corpus (containing millions of papers) on which we focus, we have solved the problem of finding and extracting the physical elements of tables. Second, we are mainly concerned with the semantics of the tables, and these papers do not focus much on semantics.

[Wong, 2008] studies the extraction of information from biomedical tables. However, he limits himself to extracting named entities. We extract entities but also focus on schema mapping, relation recognition, and extraction.

The work of [Pimplikar and Sarawagi, 2012; Cafarella *et al.*, 2008; 2009] shows a direction that we would like to pursue: using statistical classification techniques to understand table relations. We are currently exploring several clustering methods for this purpose.

The work of [Mulwad *et al.*, 2014] is of particular interest. The authors aim to do meta-analysis of medical tables on the web. Central to their approach is a mapping of the categories that are found (e.g., in table metadata) to well-known ontologies such as DBPedia and SNOMED. We plan to model future research on this work, but note two ways ways in which the authors' work is different from ours. Most saliently, in order to interpret PRI tables, the system is not primarily concerned with medical ontologies such as SNOMED; rather, it needs to process concepts of experimentation and measurement, as well as detailed concepts in molecular biology. Mulwad et al.'s work seems most suited to meta-analysis of clinical papers. Second, we focus on extraction, something that is absent in Mulwad et al.'s work. Despite these differences, we would be interested in exploring connections between these approaches.

6 Current and Future Work; Generalization to Other Domains

We are improving on and extending this work in several directions. First, we are working on improving recognition of Participant A by integrating this work with biomedical NLP systems so that we can read more of the text that is relevant to Participant A. Merging text and table-reading systems could have other advantages, including allowing cross-checking between different systems on closely related data.

Second, we are running the automated table reading system on larger corpora. In recent work, we have run the table reading system on 13K papers, yielding several dozen tables and 42K complete protein reactions. This not only shows that our system is not just a toy system, but will afford us much data to analyze so that we can improve performance. The extracted information has the potential to significantly increase the size of the Pathway Commons KB.

Third, we are working on generalizing the system to work on different domains:

(1) Tables containing other information about proteins, including interactions between proteins and biological processes and molecular functions; and expression of proteins in tumor samples.

(2) Tables containing information about climate conditions and crop yields, to populate crop-forecasting models.

(3) Tables containing information relevant to weapons development, to be used by intelligence analysts. Much of the data that analysts use is in tables. The amount of data is much too large for analysts to read; tools that would allow them to query and retrieve answers would help these analysts work more efficiently.

7 Acknowledgements

We gratefully acknowledge helpful comments and advice from Ron Ferguson, Ron Keesing, Ryan Murphy, Tifani O'Brien, Ibrahim Shafi, Mark Williams, Mark Clark, Ernie

Davis, Emek Demir, Danny Powell, and Ted Senator. This work was supported by DARPA under contract W911NF-14-C-0119.

References

[Cafarella *et al.*, 2008] Michael J. Cafarella, Alon Y. Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang. Webtables: exploring the power of tables on the web. *PVLDB*, 1(1):538–549, 2008.

[Cafarella *et al.*, 2009] Michael J. Cafarella, Alon Y. Halevy, and Nodira Khoussainova. Data integration for the relational web. *PVLDB*, 2(1):1090–1101, 2009.

[Cerami *et al.*, 2011] Ethan G. Cerami, Benjamin E. Gross, Emek Demir, Igor Rodchenkov, Özgün Babur, Nadia Anwar, Nikolaus Schultz, Gary D. Bader, and Chris Sander. Pathway commons, a web resource for biological pathway data. *Nucleic Acids Research*, 39(Database-Issue):685–690, 2011.

[Croft *et al.*, 2014] David Croft, Antonio Fabregat Mundo, Robin Haw, Marija Milacic, Joel Weiser, Guanming Wu, Michael Caudy, Phani Garapati, Marc Gillespie, Maulik R. Kamdar, Bijay Jassal, Steven Jupe, Lisa Matthews, Bruce May, Stanislav Palatnik, Karen Rothfels, Veronica Shamovsky, Heeyeon Song, Mark Williams, Ewan Birney, Henning Hermjakob, Lincoln Stein, and Peter D'Eustachio. The reactome pathway knowledgebase. *Nucleic Acids Research*, 42(Database-Issue):472–477, 2014.

[Fang *et al.*, 2012] Jing Fang, Prasenjit Mitra, Zhi Tang, and C. Lee Giles. Table header detection and classification. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, 2012.

[Hurst, 2000] Matthew Francis Hurst. The interpretation of tables in text, 2000. Ph.D. thesis, University of Edinburgh.

[Mulwad *et al.*, 2014] Varish Mulwad, Tim Finin, and Anupam Joshi. Interpreting medical tables as linked data for generating meta-analysis reports. In *Proceedings of the 15th IEEE International Conference on Information Reuse and Integration, IRI 2014, Redwood City, CA, USA, August 13-15, 2014*, pages 677–686, 2014.

[Pimplikar and Sarawagi, 2012] Rakesh Pimplikar and Sunita Sarawagi. Answering table queries on the web using column keywords. *PVLDB*, 5(10):908–919, 2012.

[Rodchenkov *et al.*, 2013] Igor Rodchenkov, Emek Demir, Chris Sander, and Gary D. Bader. The biopax validator. *Bioinformatics*, 29(20):2659–2660, 2013.

[Valenzuela-Escarcega *et al.*, 2015] Marco Antonio Valenzuela-Escarcega, Gustave Hahn-Powell, Mihai Surdeanu, and Thomas Hicks. A domain-independent rule-based framework for event extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, System Demonstrations*, pages 127–132, 2015.

[Wang and Hu, 2002] Yalin Wang and Jianying Hu. Detecting tables in HTML documents. In *Document Analysis Systems V, 5th International Workshop, DAS 2002, Princeton, NJ, USA, August 19-21, 2002, Proceedings*, pages 249–260, 2002.

[Wong, 2008] Wli Wong. Extracting named entities from tables in biomedical literature, 2008. Honour's thesis, University of Melbourne.