YJBIN 1072 ARTICLE IN PRESS No. of pages: 9
DISK / 22/4/03 / Chitra(CE)/ Prince(TE) DTD 4.3.1DTD / SPS

ACADEMIC PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Biomedical Informatics xxx (2003) xxx–xxx

Journal of Biomedical Informatics

www.elsevier.com/locate/yjbin

# Automatically identifying gene/protein terms in MEDLINE abstracts

Hong Yu,[a,*] Vasileios Hatzivassiloglou,[a,1] Andrey Rzhetsky,[b] and W. John Wilbur[c]

[a] *Department of Computer Science, Columbia University, 1214 Amsterdam Avenue, New York, NY 10027, USA*
[b] *Department of Medical Informatics, Columbia Genome Center, Columbia University, 622 W, 168th St., VC-5, New York, NY 10032, USA*
[c] *National Center for Biotechnology Information, National Library of Medicine, NIH, Building 38A, Room 5S506, 8600 Rockville Pike, Bethesda, MD 20894, USA*

Received 4 January 2003

**Abstract**

*Motivation.* Natural language processing (NLP) techniques are used to extract information automatically from computer-readable literature. In biology, the identification of terms corresponding to biological substances (e.g., genes and proteins) is a necessary step that precedes the application of other NLP systems that extract biological information (e.g., protein–protein interactions, gene regulation events, and biochemical pathways). We have developed GPmarkup (for "gene/protein-full name mark up"), a software system that automatically identifies gene/protein terms (i.e., symbols or full names) in MEDLINE abstracts. As a part of marking up process, we also generated automatically a knowledge source of paired gene/protein symbols and full names (e.g., *LARD* for *lymphocyte associated receptor of death*) from MEDLINE. We found that many of the pairs in our knowledge source do not appear in the current GenBank database. Therefore our methods may also be used for automatic lexicon generation.
*Results.* GPmarkup has 73% recall and 93% precision in identifying and marking up gene/protein terms in MEDLINE abstracts.
*Availability:* A random sample of gene/protein symbols and full names and a sample set of marked up abstracts can be viewed at http://www.cpmc.columbia.edu/homepages/yuh9001/GPmarkup/. *Contact.* hy52@columbia.edu. Voice: 718-796-2985; fax: 212-939-7028.


*Keywords:* Automatic term recognition; Synonym; Mark up; Information extraction; Knowledge acquisition; Natural language processing

## 1. Introduction

The current MEDLINE database includes over 12 million computer-readable records in the biomedical domain and is expanding rapidly; it is a rich resource for biological knowledge including protein–protein interactions [1], gene regulation events [2], sub-cellular locations of proteins [3], and pathway discovery [4]. One way to automatically extract information stored in MEDLINE is to apply an information extraction system such as a natural language processing (NLP) parser [5]. Identifying gene/protein terms in MEDLINE abstracts is a necessary step towards an information extraction system.

Genes and proteins are usually represented by symbols and names in literature. The names usually are the long forms of their symbols and describe the functions of the genes or proteins. We hypothesize that authors define gene/protein symbols in their articles when the meanings are new in literature and the definitions can be captured by a computer program. We also hypothesize that if not all of the gene/protein symbols appearing in an abstract are defined, the definition may appear in other abstracts. Therefore literature redundancy (e.g., the same genes or proteins are represented by different authors in different articles) makes it plausible that we may obtain automatically a relatively exhaustive gene/protein symbol and full name table from all of MEDLINE. In this study, we empirically tested all of the above hypotheses.

* Corresponding author. Present address: Department of Computer Science, Columbia University, Mudd 471, 116th, New York, NY 10029, USA. Fax: +212-666-0140.
*E-mail addresses:* Hongyu@cs.columbia.edu (H. Yu), vh@cs.columbia.edu (V. Hatzivassiloglou), Andrey.Rzhesky@dmi.columbia.edu (A. Rzhetsky), wilbur@ncbi.nlm.nih.gov (W. John Wilbur).
[1] Present address: Department of Computer Science, Columbia University, 1214 Amsterdam Avenue, New York, NY 10027, USA.

1532-0464/03/$ - see front matter © 2003 Published by Elsevier Science (USA).
doi:10.1016/S1532-0464(03)00032-7

This study presents an algorithm and its implementation for automatic identification of gene and protein terms (i.e., symbols or full names) in MEDLINE abstracts. As a part of the algorithm, we also present a method for automatically generating a knowledge source of paired gene/protein symbols (e.g., *LARD*) and full names (e.g., *lymphocyte associated receptor of death*) from MEDLINE. Our results show that a large number of the pairs in our knowledge source do not appear in LocusLink, a public database of gene/protein symbols and corresponding full names [6,7].

A key step in our marking up methodology is to pair gene/protein symbols to their names, so that we can use biological function keywords (e.g., kinase) to differentiate the symbols from other technical terms. For example, by mapping abbreviation *PKA* to full name *protein kinase A*, not to full form *path of the kinematic axis*, we are able to identify *PKA* is a protein term since keywords *protein* and *kinase* appear in the full form of *PKA*.

We previously have developed a method that automatically maps biomedical abbreviations to full forms. In this study, we incorporated biological domain knowledge into the method of mapping abbreviations to full forms to enhance the mapping between gene/protein symbols and full names. The biological domain knowledge was obtained from manually reviewing published guidelines of the nomenclature of genes and proteins. We then developed a method to differentiate paired gene/protein symbols and full names from other biomedical abbreviations and full forms.

To mark up gene/protein terms in MEDLINE abstracts, we first mark up gene/protein symbols and full names when the full names are defined. We then look up a knowledge source to mark up the remaining gene/protein terms. We generate the knowledge source by extracting all pairs of gene/protein symbols and full names from over eleven million MEDLINE records (year 1966–2001).

## 2. Background

A number of rule-based, linguistic, statistical, machine-learning, and hybrid approaches have been developed to mark up gene/protein terms automatically in biological text. For example, Fukuda et al. (1998) applied morphological cues to identify protein terms (e.g., if a word contains uppercase letter(s) and special character(s), the word is a protein term). Gaizauskas et al. (2000) identified protein terms through suffixes such as –*ase*. Proux et al. (1998) identified non-English words as gene terms. Linguistic approaches have mainly applied part-of-speech tagging [8] or shallow parsing [9] to identify noun phrases, from which gene/protein terms were obtained. Hybrid approaches have combined linguistic with rule-based approaches for multi-word gene/protein term recognition. For example [8], applied Brill's tagger [10] in combination with rules such as "connect non-adjacent annotations if every word between them is either noun, adjective, or a numeral" to identify multi-word protein terms such as *ras guanine nucleotide exchange factor SOS*. Tanabe and Wilbur [11] retrained Brill's tagger on the biomedical domain for gene/protein name-identification. Statistical approaches have clustered abstracts for keyword identification [12]. Machine-learning approaches have applied naïve Bayes [9], Hidden Markov Models [13], and decision trees [14], to classify gene/protein terms. Other approaches include lookup in knowledge sources such as GenBank and SWISSPROT [15].

Our method of marking up gene/protein names is a mixture of pattern-recognition and knowledge-based approaches. We first map gene/protein symbols to full names when the full names are defined. Those gene/protein terms are then marked up. The rest of gene/protein terms are identified from the gene/protein symbol and full name knowledge source which we extracted automatically from MEDLINE.

### 2.1. Systems that automatically map gene and protein symbols to full names

A number of systems have been developed for automatic mapping between abbreviations and full names [16–23]. Those systems applied a variety of approaches including linguistic, rule, and statistical methods and reported precisions from 70–97%. Most of those systems tend to be domain independent and therefore may not perform ideally in a restricted domain such as biology. For example, most of pattern-recognition approaches [18,19] do not capture *ryk* (for *receptor tyrosine kinase related gene*) since *y* represents *tyrosine* and *y* is not the first letter of *tyrosine*. In addition, most of the systems do not differentiate gene/protein symbols from other abbreviations and full names.

A system that was developed specifically for mapping protein symbols to full names is PNAD-CSS (for "protein full name abbreviation dictionary construction support system") [24]. PNAD-CSS used morphological features to recognize proper nouns as protein terms in biological abstracts [8]. Knowing a phrase may contain a protein symbol and full name, PNAD-CSS recognized parentheses and determined whether the parenthetical phrase was an abbreviation of the outer phrase. To map a protein symbol to its name, PNAD-CSS broke up words of the preceding phrase, and determined whether the parenthetical abbreviation candidate maps to the initial letters of the broken-up phrase. For example, consider the phrase "*megestrol acetate* (*megace*)." PNAD-CSS parsed "*megestrol acetate*" as "*meges trol ac etate*," which is then matched to "*megace*." For example,

Table 1
Guidelines that are useful for applying computational approaches to map a gene or a protein symbol to its full name

1. A gene symbol should stand for a description of a phenotype, a gene product or a gene function [26].
2. A gene symbol shall be short (between three to six characters) [26–32].
3. A gene symbol is an abbreviation of its full name [28].
4. If the symbol of a gene contains a character or property for which there is a recognized abbreviation, the abbreviation should be used; for example, the single-letter abbreviation for amino acids used in aminoacyl residues or approved biochemical Abbreviations such as GLC for glucose, GSH for glutathione [31] and *Bp* for *binding protein* [32].
5. The initial character should always be a letter [29–33].
6. All Greek symbols should be changed to letters in the Latin alphabet [31].
7. Amino acids have their special symbols [34].
8. The protein symbol is the same as the gene symbol [33].
9. The creator of a gene full name shall follow the guidelines and get consultation from curator of the guideline before journal publication [26].
10. Gene full names should be included in the abstracts of any relevant papers [26].

"*meg*," "*ac*," and "*e*" in "*megace*" match the initial letter(s) of "*meges*," "*ac*," and "*etate*," respectively.

We find that PNAD-CSS has some limitations: it applies morphological cues for protein term recognition and the morphological cues may falsely identify as protein symbols other substances (e.g., *LSD-25* for *lysergic acid diethylamide*), cell types (e.g., *BHK-21* for *baby-hamster kidney-cell line*), procedures (e.g., *PCR* for *polymerase chain reaction*) as well as clinical syndromes and diseases (e.g., *CHF* for *congestive heart failure*). This is because many abbreviations that are not gene/protein symbols consist of upper-case letters and numbers. The PNAD-CSS' pattern-matching rules also did not contain special rules for protein names (for example, *y* represents *tyrosine*).

Previously, we have developed a system, AbbRE (for "abbreviation and full name recognition and extraction," see [25]), that pairs biomedical abbreviations with full names. AbbRE first selected parenthetical expressions and the phrases preceding the parenthesis as candidate abbreviations and full names. It then applied a set of the pattern-matching rules to map abbreviations to full names. The rules were obtained from the common conventions authors use to create abbreviations. The following rules were included: (1) *the first letter of an abbreviation matches the first letter of a meaningful word of the full name;* (2) *the abbreviation matches the first letter of each word in the full name;* (3) *the abbreviation letter matches consecutive letters of a word in the full name* and (4) *the abbreviation letter matches a middle letter of a word in the full name if the first letter of the word matches the abbreviation*. AbbRE had 70% recall and 95% precision in identifying paired abbreviations and full names in biomedical articles.

Though AbbRE's pattern-matching rules did not contain special rules for protein names, AbbRE is robust and extensible. In this study (i.e., GPmarkup), we manually examined the published guidelines of the nomenclature of genes and proteins and added to AbbRE special rules to enhance its mapping gene/protein symbols to full names. In addition, we added in rules for differentiating gene/protein terms from other biomedical terms.

## 3. Methods and results

Our method section consists of six sub-sections: (1) Mapping gene/protein symbols to full names as well as abbreviations to full names. (2) Generating a knowledge source of paired abbreviations and full names from MEDLINE abstracts. (3) Filtering out other abbreviation-full name pairs to produce a knowledge source of paired gene/protein symbols and full names. (4) Marking up gene/protein terms in MEDLINE abstracts. (5) Evaluating GPmarkup. (6) Measuring the percentage of defined gene/protein symbols in MEDLINE abstracts.

### 3.1. Mapping gene/protein symbols to full names

To understand how gene/protein abbreviation-full name pairs are created in the first place, we examined a number of published guidelines for the nomenclature of genes and proteins. We found those guidelines are almost always species-specific (that is applicable only to genes and proteins from, say, yeast, and not rat). Species-specific may be caused by the fact that the committees for the nomenclature are formed by experts specializing on a particular model organism. Table 1 lists guidelines that were useful for mapping abbreviations to full forms.

Analysis of the published guidelines allowed us to identify some special abbreviations that are used for gene/protein nomenclature (see Table 2) and to develop the pattern-matching rules that map gene/protein symbols to names.

#### 3.1.1. Special abbreviations

see Table 2.

#### 3.1.2. Pattern-matching rules

GPmarkup applies a set of pattern-matching rules to map gene/protein symbols to full names when the full names are defined within the documents. The pattern-matching rules adapted AbbRE's (as described in Section 2.1) with the following modifications and extensions:

*Rule 1*: *Any number and special character is ignored for mapping gene/protein symbols to full names.*

We added in a rule to map letters only. We ignored numbers and special characters (e.g., "+") due to the following two reasons:

(1) Many numbers and special characters in a gene or a protein symbol do not appear in their full names. For example, *CYP2C19* for *cytochrome P450, subfamily IIC* (*mephenytoin 4-hydroxylase*), where "*19*" is not represented and "*2*" is represented by "*II*."
(2) Many numbers in gene or protein symbols order differently in their full names (e.g., *ALOX12* for *arachidonate 12-lipoxygenase*, where "*12*" in the symbol "*ALOX12*" is after "*LOX*" that represents *lipoxygenase*, but before "*lipoxygenase*" in the full name "*arachidonate 12-lipoxygenase*").

*Rule 2*: *Special abbreviation substitutions*

We substitute some nouns with their special abbreviations when we apply the pattern-matching rules. For example, instead of mapping *DYRK1A* to *dual-specificity tyrosine phosphorylation regulated kinase 1A*, we attempt to map *DYRK1A* to *dual-specificity Y phosphorylation regulated kinase 1A*, where *tyrosine* has been replaced by *Y*. After the mapping, we recover the original terms.

In reality, not all the authors use the special abbreviations (listed in Table 2) for their nomenclature. An example is *PTK2B* for *protein tyrosine kinase 2* β, where *tyrosine* is represented by its common abbreviation *T* instead of *Y*. Therefore, our algorithm considers both types of mapping (with and without substitution of a special noun with a shorthand) and selects the best matching version.

For example, we attempt to map *PTK2B* to both *protein tyrosine kinase 2* β and *protein Y kinase 2* β; we map *DYRK1A* to both *dual-specificity tyrosine phosphorylation regulated kinase 1A* and *dual-specificity Y phosphorylation regulated kinase 1A*.

When a full name has more than one word that has many abbreviations, we include all of the combinations for substitution. For example, in case of *NK AIF* for *sodium–potassium ATPase inhibitory factor*, we attempted to map *NKAIF* to *sodium–potassium ATPase inhibitory factor, Na–potassium ATPase inhibitory factor, sodium–K ATPase inhibitory factor*, and *Na–K ATPase inhibitory factor*. We found that *Na–K ATPase inhibitory factor* was mapped and we recovered the original full name.

### 3.1.3. Parenthetic pattern

Prior to pattern-matching rules, GPmarkup selects candidate abbreviations and full names. For this task, GPmarkup recognizes special patterns such as "<abbreviation>(<full name>)" or "<full name>(<abbreviation>)". Recall AbbRE also recognized these patterns. However, AbbRE can not recognize gene/protein terms that incorporate nested parentheses. For example, AbbRE fails to map *acyl-coenzyme A* (*acyl-CoA*) *dehydrogenases* to *ACD* from the following string extracted from [35] *the expression of various acyl-coenzyme A* (*acyl-CoA*) *dehydrogenases* (*ACD*) since it parses into the following two components:

*the expression of various acyl-coenzyme A* (*acyl-CoA*) and *dehydrogenases* (*ACD*)

To correct for this shortcoming, we introduced into the newer algorithm (GPmarkup) an additional rule to recognize gene/protein full names that incorporate parentheses. It then parses the above string into the following two components:

*the expression of various acyl-coenzyme A* (*acyl-CoA*) *and the expression of various acyl-coenzyme A* (*acyl-CoA*) *dehydrogenases* (*ACD*)

where the phrases preceding and within the parentheses in each component incorporate candidate abbreviations and full names, to which GPmarkup further applies its pattern-matching rules to map abbreviations to full names.

### 3.2. Generating a knowledge source of paired abbreviations/full names from MEDLINE abstracts

We applied GPmarkup to 11 million MEDLINE records (1966–2001), which contain the same number of titles and over six million abstracts (note that not all MEDLINE records contain abstracts). We obtained a knowledge source that consisted of 574,327 unique pairs of abbreviations and full names. The most frequently defined abbreviations were *PCR* (*polymerase chain reaction*, which appeared in 7988 abstracts) and *NO* (*nitric oxide*, which appeared in 7855 abstracts).

Table 2
Special abbreviations that are used in gene/protein nomenclature

| Type | |
|---|---|
| Amino acids | We use all one letter codes where these differ from the first letter of the amino acid. For example, *tyrosine—Y* (*SYK* for *spleen tyrosine kinase*) |
| Two chemical symbols used | *Sodium–Na, potassium–K* (*NKAIF* for *sodium–potassium ATPase inhibitory factor*) |
| Three other symbols used | *Inhibitor—N or NH, box—X* (*CDKN1A* for *cyclin-dependent kinase inhibitor 1A* (*p21*, *Cip1*), CDX1 for *caudal type homeo box transcription factor 1*) |

### 3.3. Filtering out other abbreviation-full name pairs to produce a knowledge source of paired gene/protein symbols and full names

The algorithm outlined above also identifies a large number of general abbreviations that are not gene/protein symbols and full names. We therefore developed a rule-based approach to partition our knowledge source of abbreviation-full name pairs into gene/protein symbol-full name pairs and other abbreviation-full name pairs.

Our rule-based approach combines morphological cues, functional keywords, and position-functional keywords to filter out non-gene/protein terms. The approach is described as follows:

If an abbreviation contains a number, the abbreviation and full name is a gene/protein symbol-full name pair only if the full name contains one or more of the following keywords (denoted as set K1): *protein*(*s*), *gene*(*s*), *peptide*(*s*), *molecule*(*s*), *enzyme*(*s*), *ligand*(*s*), *compound*(*s*), *receptor*(*s*), *channel*(*s*), *transcriptor*(*s*), *regulator*(*s*), *inhibitor*(*s*), *antibody*, *antibodies*, *globulin*(*s*), *factor*(*s*), *motif*, *domain*(*s*), *compound*(*s*), *segment*(*s*), *subunit*(*s*), *locus*, *loci*, *cassette*(*s*), *chain*, *complex*(*es*), *homeobox*(*es*), *box*(*es*), *member*(*s*), *deletion*, *axon*, *family*, *families*, *chromosome*(*s*), *sequence*, $\alpha$, $\beta$, $\gamma$, *interleukin* and any words except for *disease* that ends in *–ase.*

If an abbreviation does not contain a number, the abbreviation and full name is gene/protein symbol-full name pair only if the last word of the full name is a keyword in set K1.

We obtained functional keywords by manually examining all of the entries in LocusLink. Note that some keywords (e.g., "gene") in set K1 can appear as both the last word or the middle word of a gene/protein term (e.g., *Btg4* for *B-cell translocation gene 4* and *AFG3L1* for *AFG3* (*ATPase family gene 3, yeast*)*-like 1*). On the other hand, some keywords (e.g., "chromosome") do not appear as the last word of, but only within a gene/protein term (e.g., *C10ORF2* for *chromosome 10 open reading frame 2*).

We applied the rules to abbreviations and full names and generated a knowledge source of 86,767 unique pairs of gene/protein symbols and full names. The most frequently defined gene/protein symbols included *egf* (for *epidermal growth factor*, appears in 2023 abstracts), *il* (for *interleukin*, appears in 2183 abstracts), and *ldl* (for *low density lipoprotein*, appears in 2673 abstracts).

### 3.4. Marking up gene/protein terms in MEDLINE abstracts

We further developed and implemented an algorithm to mark up gene/protein terms in MEDLINE abstracts. GPmarkup first maps abbreviations to full names and then performs the markup for any abbreviation with an identified full name (details in Sections 3.2 and 3.3). For the remaining terms in abstracts, we looked up the knowledge sources of paired abbreviations and full names and paired gene/protein symbols and names. As an effort to achieve a higher precision, we only looked up multi-word gene/protein terms, since a single word term could be ambiguous (for example, *aap* denotes *antiarrhythmic peptide* or *automatic action potential*, the former is a protein name, and the latter is not).

When a string can be mapped to several terms stored in our knowledge sources, GPmarkup favors longer term mapping and markup. It does not mark up a term which is used as a modifier of entity other than genes and proteins. For example, GPmarkup does not markup the protein term *amyloid β protein* in a string of *cerebral amyloid β protein angiopathy*, because the protein name is used as a modifier for the disease term *angiopath*.

GPmarkup applies direct matching (i.e., the string in text exactly appears in our knowledge sources) except that GPmarkup includes a word that immediately follows a gene or a protein symbol or full name if the word either consists of a number or is a functional keyword including "gene," "protein," "homologue," and "receptor." For example, knowing a β and *il12 p40* as gene or protein symbols, GPmarkup also identifies a β*40* and *il12 p40 homologue.*

### 3.5. GPmarkup evaluation

We performed evaluation in the following three steps: (1) mapping abbreviations to full names, (2) filtering out other terms to produce a knowledge source of paired gene/protein symbols and names, and (3) marking up gene/protein terms in MEDLINE abstracts. We therefore evaluate GPmarkup phase by phase. We also compared the knowledge source of paired gene/protein symbols and full names with the ones in LocusLink. We evaluated by recall (i.e., number of correct answers identified by our system divided the total number of correct answers) and precision (i.e., number of correct answers divided by the total number of answers specified by our system). We estimated confidence intervals for these measures based on the binomial distribution.

#### 3.5.1. Mapping abbreviations to full names

We randomly (by time of publication) selected 30 MEDLINE abstracts and asked three biomedical experts (all with PhD or MD) to map abbreviations to full names when the full names are defined within the abstracts. The gold standard was determined by a majority vote of experts. GPmarkup correctly mapped 56 abbreviations and full names out of a total of 59 pairs that were determined by experts. GPmarkup wrongly iden-

tified one pair that was not an abbreviation and full name. GPmarkup's recall and precision in identifying and extracting abbreviations and full names were, with 95% confidence intervals, 0.95 (0.86–0.99) and 0.98 (0.91–1.00), respectively.

### 3.5.2. Filtering out other terms

We then evaluated our rule-based approach for partitioning the knowledge source of abbreviation-full name pairs into gene/protein symbol-full name pairs and other abbreviation-full name pairs. We randomly selected 1000 pairs of gene/protein symbols and full names and 1000 pairs of other abbreviations and full names partitioned by GPmarkup and evaluated recall and precision of the partitioning. We asked experts (see 3.5.1) for help in defining a gold standard. Table 3 lists the results of the evaluation. Note that GPmarkup included some incomplete-matches of abbreviations and full names (e.g., {*il-6, interleukin*}). Since the ratio of gene/protein symbol-names to other abbreviation-full name pairs was 1:5.6 (86,767/[574,327–86,767]); the numbers were described in Sections 3.2 and 3.3), GPmarkup had an accuracy of $0.95 \pm 0.02$, with 95% confidence. The figure 0.95 comes from the ratio $(982 + 949 * 5.6)/(1000 + 1000 * 5.6)$ which is based on the numbers in Table 3 and their relative frequencies as just computed.

### 3.5.3. Marking up gene/protein terms in MEDLINE abstracts

We then evaluated GPmarkup in marking up gene/protein terms in MEDLINE abstracts. We randomly (by time of publication) selected 50 MEDLINE abstracts, which consists of a total of 539 sentences (including the title). Some selected abstracts did not cover biological domain and therefore did not have gene/protein terms at all. Therefore, we did not select only biological abstracts for evaluation because we judge a false markup is as bad as a missing markup. We therefore judged that a random selection of abstracts best reflects our system's recall and precision.

Table 4 lists the evaluation results of the 50 abstracts. GPmarkup applies XML format for term mark up. For example, the tag "phr"(for "phrase") has attributes including "sem" (for "semantic category") that has value "gp" (for "gene and protein terms") and "t" (for "target") that represents gene/protein full names. We count any appearance of gene/protein terms. For example, if protein "*amyloid β protein*" appears three times in the abstract, we count three instead of one for this case. We posted a sample set of marked up abstracts at http://www.cpmc.columbia.edu/homepages/yuh9001/GPmarkup/).

From Table 4, if we count a partial-matching as a match, the recall and the precision of GPmarkup were, with 95% confidence, $0.73 \pm 0.05$ $(222 + 15)/(222 + 15 + 88)$ and $0.93 \pm 0.03$ $(222 + 15)/(222 + 15 + 17)$, respectively. We found all partial matches represent valid proteins. However, if we do not include a partial-matching as a match, the recall and precision of GPmarkup were, with 95% confidence, $0.68 \pm 0.05$ $222/(222 + 15 + 88)$ and $0.87 \pm 0.04$ $(222/(222 + 15 + 17)$, respectively.

### 3.5.4. Comparing gene/protein symbols and full names extracted from MEDLINE with LocusLink

We downloaded the knowledge source of paired gene/protein symbols and full names from LocusLink [36]. LocusLink is maintained by the National Center for Biotechnology Information. It presents information on official nomenclature of genes and lists a total of 115,890 manually annotated paired gene symbols and full

Table 3
Evaluation results of GPmarkup in filtering the knowledge source of paired abbreviations and full names to produce a knowledge source of paired gene/protein symbols and full names

| Evaluation cases | Expert judgments | | |
|---|---|---|---|
| | Number of gene/protein symbol-full name pairs | Number of other abbreviation-full name pairs | Number of non abbreviation-full name pairs |
| 1000 pairs of gene/protein symbols and full names as identified by GPmarkup | 982 | 9 (e.g., *srg* for *spent restaurant grease*) | 9 (e.g., *gene* for *genes*) |
| 1000 pairs of other abbreviations and full names as identified by GPmarkup | 1 (i.e., *A-Igg* for *Anti-human Igg*) | 949 | 50 (e.g., *ph2* for *phages*) |

Table 4
Evaluation results of GPmarkup

| Type of category | GPmarkup identified |
|---|---|
| Complete-matching (e.g., *<phr sem="gp" t="signaling lymphocyte activation molecule">slam</phr>*) | 222 |
| Partial-matching[a] (e.g., *<phr sem="gp">interleukin 1</phr> receptor ii*) | 15 |
| Missing (e.g., *2b4*) | 88 |
| False-matching[b] (e.g., *<phr sem="gp">acupuncture points and channels</phr>*) | 17 |

[a] The correct full name is "interleukin 1 receptor ii".
[b] False-matching includes those non-gene and non-protein terms that are identified by GPmarkup.

names, though we found that only 65,987 entries have both gene/protein symbols and full names.

We randomly selected 100 entries that incorporate both symbols and full names from the LocusLink and manually identify their existence in our knowledge source of paired gene/protein symbols and full names. We also randomly selected 100 unique gene/protein symbol and full name pairs from our knowledge source and manually identified their existence in LocusLink.

We found that 62 out of 100 selected pairs in our knowledge source did not appear in LocusLink. Examples included {*ACY1-ACP, acyl-acyl carrier protein*}, {*GCDFP, gross cyst disease fluid protein*}, {*CCK-OP, cholecystokinin octopeptide*} and {*l-PK, l pyruvate kinase*} though some of the missing pairs represent protein products instead of direct genes. For example, {*l-PK, l pyruvate kinase*} is a spliced product of its gene {*PKLR, pyruvate kinase*},[2] which appears in LocusLink and there is no gene for {*CCK-OP, cholecystokinin octopeptide*}.[3] Eight pairs partially matched to LocusLink. For example, *PPI, peptide prolyl cis trans isomerase* appears in our knowledge source. In LocusLink, we found {*PPIa, peptidylprolyl isomerase a* (*cyclophilin a*)}."

On the other hand, we found that only 40 LocusLink entries could be found in our knowledge source (16 of them have variations). We judged that four of those 60 failed entries are not gene/protein symbols and full names (e.g., {*shs, sutherland-haan x-linked mental retardation syndrome*}). To find whether the remaining 56 entries exist in MEDLINE, we searched 12 million MEDLINE records (1966–2002). We applied direct matching (case insensitive) and manually analyzed abstracts that contained either the symbol or the full name of those 56 failed entries. We failed to find the existence of 50 of them in MEDLINE, either symbols or full names. Examples include {*2700088m22rik, riken cdna 2700088m22 gene*} and {*atp5bl1, atp synthase, h+ transporting, mitochondrial f1 complex, β polypeptide-like 1*}. Of the rest of six entries, we could find symbols in MEDLINE, but failed to find full names. Examples include {*aspa*, *aspartoacylase* (*aminoacylase 2*, *canavan disease*)} and {*assp6*, *argininosuccinate synthetase pseudogene 6*}, for the former we found the full name with variations, for the latter we found that the full name did not exist in the MEDLINE record where the symbol appeared.

### 3.6. The percentage of undefined gene/protein symbols and full names

If all the gene/protein symbols and full names were defined in MEDLINE abstracts, then GPmarkup would also serve the purpose for disambiguation by assigning full names to symbols. However, not all the gene/protein symbols are defined in the abstracts.

We measured the percentage of defined gene/protein symbols in MEDLINE abstracts. We randomly selected 100 abstracts (according to the time of publication) from a total of 782,560 MEDLINE abstracts (1966–2001) that were retrieved by the keyword "protein." Those abstracts contain 1069 sentences (including titles). We measured the percentage of undefined gene/protein symbols. We counted unique appearance of gene/protein symbols within abstracts. Based on the authors' judgment, the numbers of defined and undefined gene/protein symbols were 92 and 27, respectively. The percentage of defined gene/protein symbols and full names was, with 95% confidence, $0.77 \pm 0.08$.

## 4. Discussion

Many public databases such as GenBank have gene/protein synonym knowledge sources. However, the databases are largely maintained manually and therefore are not always up to date. GPmarkup can generate automatically a knowledge source of paired gene/protein symbols and full names from MEDLINE abstracts. The automated fashion may reduce manual efforts. In addition, GPmarkup may capture the most up-to-date gene/protein symbols and full names if the full names are defined in abstracts and follow the guidelines of nomenclature of genes and proteins.

We also found that a majority of gene/protein symbols and full names extracted in our knowledge source did not appear in LocusLink. Recall LocusLink consists of a large number of mainly manually annotated paired gene/protein symbols and full names. In addition, we found a majority of pairs in LocusLink did not appear in our knowledge source either; most of those pairs did not even appear in MEDLINE by keyword search. The results suggest that there is a gap between LocusLink knowledge source and the actual text. This difference may make it difficult to apply LocusLink directly for looking up terms in MEDLINE. On the other hand, since our knowledge source of paired gene/protein symbols and names were directly extracted from MEDLINE, they may be more useful as a knowledge-based markup.

One limitation of GPmarkup is that not all the gene/protein symbols and full names are defined in the abstracts and therefore GPmarkup may not capture some gene/protein symbols and full names. However, two other factors alleviate this problem: authors are encouraged to define gene/protein full names in the abstracts of any relevant papers [26], and the literature is redundant. Therefore, applying GPmarkup to all of MEDLINE abstracts is likely to capture a majority of

[2] GenBank Accession No. U47654.
[3] For details see http://arbl.cvmbs.colostate.edu/hbooks/pathphys/endocrine/gi/cck.html).

gene/protein symbols and full names that appear in the text.

GPmarkup may also miss gene/protein symbols and full names when authors do not follow the guidelines for naming genes and proteins. To capture these gene/protein symbols and full names, we may integrate into GPmarkup statistical approaches such as Hisamitsu and Niwa's approach [18,20] of selecting phrases associated with parentheses that were statistically significant. In addition, GPmarkup may also miss abbreviations and full names that are introduced through syntactic patterns (e.g., appositions). In the near future we plan to utilize the approaches of [37] that enumerated syntactic patterns for abbreviation detection.

Other limitations include the ambiguity in usage of gene/protein terms. For example, we do not differentiate a gene term from a protein one. We do not differentiate a general gene/protein term (e.g., *growth factors*) from a specific one (e.g., *protein kinase A*). We also do not identify to which organism, tissue, cell type, and sublocation a gene/protein term refers. We propose to integrate the approach of [38] for disambiguating gene/protein terms. We also hope to develop statistical NLP approaches for further disambiguation.

Our study shows that many gene/protein symbols (77%) are defined within the abstracts, GPmarkup can map a majority of gene/protein symbols to full names. GPmarkup does not mark up undefined gene/protein symbols if the symbols have several full names in the knowledge source of abbreviation-full name pairs. For example, *aap* denotes *antiarrhythmic peptide, alkyl acceptor protein, alzheimer amyloid precursor protein, aminoantipyrine*, and *automatic action potential* in our knowledge source and GPmarkup thus does not mark up "aap" as a gene/protein term when it is not defined in the abstract. We therefore sacrifice GPmarkup's recall for high precision. In the future, we will integrate a disambiguation method that assigns the full names from our knowledge source to the ambiguous symbols. Once a symbol is assigned to its full name, we can apply our rule-based approach (see Section 3.3) determining whether the symbol is a gene/protein term.

Note that we recognized a gene/protein term if the term actually represents a gene/protein in the abstract. We described earlier that we did not mark up "*cerebral amyloid β protein angiopathy*" as a protein name even though "*cerebral amyloid β protein*" by itself is a protein name. Other researchers may do differently [11].

## 5. Conclusion

This study shows that GPmarkup is efficient (73% recall and 93% precision) in marking up gene/protein terms in MEDLINE abstracts. Our results may provide a useful supplement to manually curated resources such as LocusLink (GenBank). A method to more accurately identify the full names of undefined abbreviations would increase the recall of GPmarkup and enhance its usefulness.

## Acknowledgments

We want to thank Dr. Carol Friedman and Ivan Iossifov for valuable discussions. This research was supported in part by National Science Foundation Information Technology Research Grant EIA-0121687 and National Institutes of Health Grant RO1 GM61372-01A2.

## References

[1] Blaschke C et al. Automatic extraction of biological information from scientific text: protein–protein interactions. Proc Int Conf Intell Syst Mol Biol 1999:60–7.
[2] Stapley BJ, Benoit G. Biobibliometrics: information retrieval and visualization from co- occurrences of gene names in Medline abstracts. Pac Symp Biocomput 2000:529–40.
[3] Stapley BJ, Kelley LA, Sternberg MJE. Predicting the sub-cellular location of proteins from text using support vector machines. In: PSB, Hawaii, 2002.
[4] Ng SK, Wong M. Toward routine automatic pathway discovery from on-line scientific text abstracts. Genome Inform Ser Workshop Genome Inform 1999;10:104–12.
[5] Carol Friedman, P.K., Michael Krauthammer, Hong Yu, Andrey Rzhetsky. GENIES: A Natural-Language Processing System for the Extraction of Molecular Pathways from Complete Journal Articles. In: ISMB, 2001.
[6] Maglott DR et al. NCBI's LocusLink and RefSeq. Nucleic Acids Res 2000;28(1):126–8.
[7] Pruitt KD et al. Introducing RefSeq and LocusLink: curated human genome resources at the NCBI. Trends Genet 2000;16(1):44–7.
[8] Fukuda K et al. Toward information extraction: identifying protein names from biological papers. Pac Symp Biocomput 1998:707–18.
[9] Nobata C, C, CN, TJI. Automatic term identification and classificaiton in biology texts. In: Proceedings of the Natural Language Pacific Rim Symposium (NLPRS'99), 1999.
[10] Brill E. Transformation-based error-driven learning and natural language processing: a case study in part of speech tagging. Comput Linguistics 1995.
[11] Tanabe L, Wilbur WJ. Tagging gene and protein names in biomedical text. Bioinformatics 2002;18:1124–32.
[12] Andrade MA, Valencia A. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. Bioinformatics 1998;14(7):600–7.
[13] Collier NH, Nobata C, Tshjii J. Extracting the names of genes and gene products with a hidden markov model. In: Proceedings of the 18th International Conference on Computational Linguistics (COLING'2000), 2000, p. 201–7.
[14] Nobata C, Collier NH, Tsujii J. Comparison between tagged corpora for the named entity task. In: Proceedings of the workshop on comparing corpora (at ACL'2000), Kilgarriff, A., Berber Sardinha, 2000.

[15] Krauthammer M et al. Using BLAST for identifying gene and protein names in journal articles. Gene 2000;259(1–2):245–52.
[16] Pakhomov S. Semi-supervised maximum entropy based approach to acronym and abbreviation normalization in medical text. In: Proc. 40th annual meeting of the association for computational linguistics, Philadelphia, Pennsylvania, USA, 2002.
[17] Bowden PR, Eventt L, Halsted P. Automatic arconym acquisition in a knowledge extraction program. In: CompuTerm98, Montreal, Ontario, 1998.
[18] Hisamitsu T, Niwa Y. Extraction of useful terms from parenthetical expression by using simple rules and statistical measures. In: CompuTerm98, Montreal, Canada, 1998.
[19] Schwartz AS, Hearst MA. A simple algorithm for identifying abbreviation definitions in biomedical text. In: Pac Symp Biocomput, 2003.
[20] Liu H, Friedman C. Mining terminological knowledge in large biomedical corpora. In: Pac Symp Biocomput, 2003.
[21] Park Y, Byrd RJ. Hybrid text mining for finding abbreviations and their definitions. In: Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing, Pittsburgh, PA, 2001.
[22] Larkey LS, Ogilvie P, Price MA. Acrophile: an automated acronym extractor and server. In: Proceedings of the Fifth ACM International Conference on Digital Libraries, 2000.
[23] Wren JD, Garner HR. Heuristics for identification of acronym-definition patterns within text: towards an automated construction of comprehensive acronym-definition dictionaries. Methods Inf Med, 2002, in press.
[24] Yoshida M, Fukuda K, Takagi T. PNAD-CSS: a workbench for constructing a protein name abbreviation dictionary. Bioinformatics 2000;16(2):169–75.
[25] Yu H, Hripcsak G, Friedman C. Mapping abbreviations to full forms in biomedical articles. J Am Med Inform Assoc 2002;9(3):262–72.
[26] Kohli J. Genetic nomenclature and gene list of the fission yeast *Schizosaccharomyces pombe*. Curr Genet 1987;11(8):575–89.
[27] Nomenclature CPG. Chicken Poultry Genome Nomenclature Web site. Available at http://www.ri.bbsrc.ac.uk/chickmap/nomenclature.html. Accessed May 1, 2001.
[28] Zebrafish-Nomenclature, Zebrafish Nomenclature Committee and Guidelines. Available at http://zfin.org/zf_info/nomen_comm.html. Accessed May 1, 2001.
[29] Maltais LJ et al. Rules and guidelines for mouse gene nomenclature: a condensed version. International committee on standardized genetic nomenclature for mice. Genomics 1997;45(2):471–6.
[30] Antonarakis SE. Recommendations for a nomenclature system for human gene mutations. Nomenclature working group. Hum Mutat 1998;11(1):1–3.
[31] Chicken-nomenclature, Nomenclature for naming loci, alleles, linkage groups, and chromosomes to be used in poultry genome publications and databases. Available at http://www.ri.bbsrc.ac.uk/chickmap/nomenclature.html. Accessed May 1, 2001.
[32] Rat-nomenclature, Rat: Nomenclature Committee Guidelines. Available at http://ratmap.gen.gu.se/ratmap/WWWNomen/Brief.html. Accessed May 1, 2001.
[33] Horvitz HR et al. A uniform genetic nomenclature for the nematode *Caenorhabditis elegans*. Mol Gen Genet 1979;175(2):129–33.
[34] Aminoacid-nomenclature, Nomenclature and Symbolism for Amino Acids and Peptides. Available at http://www.chem.qmw.ac.uk/iupac/AminoAcid/. Accessed May 1, 2001.
[35] Kibayashi M, Nagao M, Chiba S. Influence of valproic acid on the expression of various acyl-CoA dehydrogenases in rats. Pediatr Int 1999;41(1):52–60.
[36] GenBank, http://www.ncbi.nlm.nih.gov/Genbank/GenbankOverview.html. Accessed August 1, 2002.
[37] Klavans J, Muresan S. Evaluation of the DEFINDER System for Fully Automatic Glossary Construction. In: Proceedings of the AMIA Symposium, 2001.
[38] Hatzivassiloglou V, Duboue PA, Rzhetsky A. Disambiguating proteins, genes, and RNA in text: a machine learning approach. Bioinformatics 2001;17(Suppl 1):S97–S106.