Nucleic Acids Research, 2005, Vol. 33, Database issue ***D289–D293***
doi:10.1093/nar/gki137

Biomedical term mapping databases

Jonathan D. Wren*, Jeffrey T. Chang[1], James Pustejovsky[2], Eytan Adar[3], Harold R. Garner[4] and Russ B. Altman[5]

Advanced Center for Genome Technology, Department of Botany and Microbiology, The University of Oklahoma, 101 David L. Boren Blvd, Rm 2025, Norman, OK 73019, USA, [1]Department of Molecular Genetics and Microbiology, Duke University Medical Center Computational and Applied Genomics Program, Duke Institute for Genome Sciences and Policy, Durham, NC 237710-0001, USA, [2]Department of Computer Science, Brandeis University, Waltham, MA 02454-9110, USA, [3]Information Dynamics Lab, Hewlett-Packard Laboratories, Palo Alto, CA 94304, USA, [4]McDermott Center for Human Growth and Development and the Center for Biomedical Inventions, University of Texas Southwestern Medical Center, TX 75390, USA and [5]Department of Genetics, Stanford University School of Medicine, CA 94305-5120, USA

Received August 15, 2004; Revised and Accepted October 29, 2004

ABSTRACT

Longer words and phrases are frequently mapped onto a shorter form such as abbreviations or acronyms for efficiency of communication. These abbreviations are pervasive in all aspects of biology and medicine and as the amount of biomedical literature grows, so does the number of abbreviations and the average number of definitions per abbreviation. Even more confusing, different authors will often abbreviate the same word/phrase differently. This ambiguity impedes our ability to retrieve information, integrate databases and mine textual databases for content. Efforts to standardize nomenclature, especially those doing so retrospectively, need to be aware of different abbreviatory mappings and spelling variations. To address this problem, there have been several efforts to develop computer algorithms to identify the mapping of terms between short and long form within a large body of literature. To date, four such algorithms have been applied to create online databases that comprehensively map biomedical terms and abbreviations within MEDLINE: ARGH (http://lethargy.swmed.edu/ARGH/argh.asp), the Stanford Biomedical Abbreviation Server (http://bionlp.stanford.edu/abbreviation/), AcroMed (http://medstract.med.tufts.edu/acro1.1/index.htm) and SaRAD (http://www.hpl.hp.com/research/idl/projects/abbrev.html). In addition to serving as useful computational tools, these databases serve as valuable references that help biologists keep up with an ever-expanding vocabulary of terms.

INTRODUCTION

The majority of databases published in this issue are referred to using their abbreviated forms, which is no different from most names within biology. But a problem arises when the same abbreviation is used to refer to different entities, also known as polynymy. On the surface, this seems more like a computer science problem than a biological one. Biomedical research, however, increasingly includes high-throughput and data-intensive experimental methods with the number of studied entities (e.g. genes, diseases and chemicals) growing steadily. In each of these fields, there are ongoing needs to be able to accurately identify these entities within the text (1–4). In this issue, for example, the Database of Interacting Proteins (DIP) (5) bolsters experimental entries with previously published interactions (6). And during the construction of PubGene, a human genetic network (7), the authors noted that one of the biggest problems in constructing a genetic network from text was the prevalence of polynyms, or acronyms with multiple definitions. An important part of any high-throughput effort to tie experimental findings to published knowledge within the scientific literature involves acronym resolution.

Similarly, named entity recognition is becoming increasingly important with several long-standing conferences such as the Text Retrieval Conference (TREC), Message Understanding Conference (MUC) and competitions such as Critical Assessment of Information Extraction systems in Biology (Biocreative) dedicated to the task. Term mapping databases provide the additional benefit of expanding named entity and synonym recognition. For example, in a text-mining application designed to recognize disease names (among other named entities) (2,8), the ARGH database described herein was used to identify disease names, symbols and spelling variants not found in OMIM (inherited diseases) or

*To whom correspondence should be addressed. Tel: +1 405 325 3415; Fax: +1 405 325 3442; Email: Jonathan.Wren@OU.edu

MeSH (inherited + epidemiological), expanding the number of recognized names by 2029.

The sheer growth in published scientific literature precludes manual efforts of defining acronyms as being practical or cost-effective. Automated approaches to constructing acronym-definition databases enable simple and rapid updates at low cost, the domain of analysis to be clearly defined (e.g. MEDLINE covers the scientific biomedical literature) and comprehensively analyzed, allow the compilation of frequency information for users to assess both meaning and standard form, and are unbiased in their inclusion of entries.

STANDARD NOMENCLATURE AND INFORMATION RETRIEVAL

Historically, as the number of researchers publishing within a given field of study grows, there is an increase in the variability of naming. As a consequence, information retrieval and analysis become more difficult. Acronyms are known to be problematic when used for information retrieval (9,10), but full phrases can be as well. Naively, a biologist might believe that by typing a gene name into PubMed or Ovid's query engine, they will retrieve a complete list of articles ever published containing that gene name, but this is not the case. For example, the reader can attempt the following experiment by going to PubMed (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed) or Ovid (http://gateway.ovid.com), two different search engines that offer access to the MEDLINE database, and searching for the gene JNK using the search patterns shown in Table 1. As this table shows, for each database the number of results returned varies quite significantly depending upon the spelling used (database content between PubMed and Ovid does not precisely overlap and each uses its own search algorithm—the emphasis here is on the intra-database variation rather than inter-database variation). Retrieved terms are a function of how frequently the variant occurs within MEDLINE as well as the respective information retrieval algorithm used. Even when the major spelling variants are searched together (Table 1, pattern #6), the cumulative numbers still do not add up to the total found by searching on JNK—the symbol each spelling variant maps back to. JNK, however, is unusual in that it uniquely defines this gene within MEDLINE. Many eukaryotic gene acronyms such as calcitonin (CT), neurokinin (NK) and neutrophil migration (NM) are highly ambiguous (11).

Biologists may not care how many different ways a phrase might be spelled or what terms it maps onto within the literature, but when conducting literature searches it is certainly important to them that all relevant literature on a term has been retrieved. Thus, given that retrieved literature can be highly dependent upon the precise query term used, it would be useful to them to know how common that query term is among others that map onto the same concept. Term mapping efforts can help in establishing standard naming conventions by providing the most common spelling variants found within the literature to help guide conventions. For example, the Human Gene Nomenclature Committee (12), also in this issue, has long recognized the problems that ambiguity causes and helps to determine which gene names should be considered as the accepted standard. Finally, acronym-mapping efforts also provide a means to improve information retrieval.

Table 1. Number of results returned when searching either PubMed or Ovid using the phrases above typed in exactly as shown

Pattern	Search pattern	No. of results in PubMed	No. of results in Ovid
1	JNK	5477	7902
2	c-jun N-terminal kinase	3773	2912
3	c-jun NH_2-terminal kinase	503	731
4	c-jun amino-terminal kinase	3057	3039
5	jun N-terminal kinase	2451	3445
6	#2 OR #3 OR #4 OR #5	4487	5860
7	MAPK8 (official LocusLink name, ID#5599)	2	3
8	Mitogen activated protein kinase 8	381	382

The results were as of May 14, 2004. First, the gene name JNK is used as the query. Then its official name, according to LocusLink, is used (MAPK8). Notice the literature has more references to JNK, but the number retrieved depends upon how it is spelled. Retrieval numbers are more consistent with the standardized name.

OVERVIEW OF DATABASES

Several different approaches to mapping acronym-definition patterns have been undertaken by various groups for different purposes (13–20). However, efforts that can accurately resolve acronym definitions on a large scale (i.e. millions of records rather than thousands or hundreds) and have an online interface are a more recent phenomenon. To date, there are four databases: ARGH (21), the Stanford Biomedical Abbreviation Server (22), AcroMed (23) and SaRAD (24). Thus, this report presents an overview of each database as well as a statistical summary (Table 2) and a comprehensive comparison of the features and capabilities (Table 3).

ARGH (http://lethargy.swmed.edu/argh/argh.asp)

The Acronym Resolving General Heuristic (ARGH) program (21) uses a set of heuristic recognition and refinement rules for identifying acronyms and their definitions within scientific text. The advantage of using heuristics is that the rule set can be changed to fit whichever circumstance works best. The disadvantages are that rule changes require re-evaluation of efficiency (precision/recall), and changes of upstream rules (rules applied earlier than others) sometimes have unpredictable effects upon downstream efficiency.

ARGH proceeds from right to left after identifying a parenthetical phrase within text. If the parenthetical phrase is a single word it treats it as a potential acronym, attempting to match each acronym letter to letters within the words immediately to the left of it. If the parenthetical is multiple words

Table 2. Summary statistics for each of the databases

Database	Unique acronyms	Unique definitions	Total acronym-definition pairs	MEDLINE records processed	Last updated
ARGH	206 348	767 609	885 060	12 808 695	January 2004
Stanford	699 043	1 490 909	1 716 288	11 447 996	March 2002
AcroMed	211 000	703 924	481 531	11 000 000	December 2002
SaRAD	64 764	193 103	3 960 168	11 253 125	January 2002

Table 3. Comparison of the four databases described herein

Database	Base method[a]	Query method[b]	Stem[c]?	Terms normalized[d]?	Quality evaluation[e]?	Grouped[f]	Relative frequency[g]?	Concept mapping[h]?
ARGH	HR	P+W	N	N	N	N	Y	N
Stanford	DP	D	N	Y	Y	Y	N	N
AcroMed	NLP	D	Y	Y	N	Y	Y	Y
SARAD	HS	D	Y	Y	N	Y	N	Y

[a]HR, heuristic/rule-based; HS, heuristic/score-based; DP, Dynamic Programming (alignment) based; and NLP, Natural Language Processing.
[b]Query (search) method available to the user to find terms: P, precise match; D, degenerate match (e.g. a search on JNK also retrieves JNK-1); and W, wildcard matching.
[c]Stemming removes plural endings.
[d]Term normalization treats certain characters or patterns as equivalent (e.g. 'beta-carotene' and 'beta carotene' would be considered the same term).
[e]Quality evaluation provides a score of how confident the algorithm was in pairing short and long forms.
[f]Grouping clusters together long-form terms considered to be conceptually the same definition (e.g. by stemming/normalizing or some other means).
[g]Relative frequency indicates how common (% wise) one definition is over the others.
[h]Concept mapping associates extracted terms with higher-level concepts such as MeSH terms.

then ARGH treats the word immediately to the left as a potential acronym and the parenthetical as a potential definition. ARGH is capable of recognizing word patterns that are not in the same order as the acronym letters (e.g. 'Propelling Efficiency' as the definition for EP), but is not able to recognize purely symbolic acronyms (e.g. potassium when abbreviated as 'K' because of its latin root, kalium). ARGH has been used to provide acronym resolution for other literature-mining algorithms (2,8). The ARGH database includes lexical variations seen in the literature such as alternative hyphenation patterns, symbols, spelling, word order and word choice. Acronyms can be queried for their corresponding definitions and word patterns queried for any associated acronyms—this includes the ability to query using wildcard matches. Frequency of occurrence is given for each acronym-definition pair, to aid users in ascertaining which definition could be considered 'standard', at least by popular use. And to aid the user in determining context, each entry is linked to an example abstract within PubMed, where ARGH had identified the acronym-definition pair. As new records are added, the statistics are kept updated at http://lethargy.swmed.edu/argh/Statistics.htm. ARGH is updated annually.

Stanford Biomedical Abbreviation Database (http://abbreviation.stanford.edu/)

The Biomedical Abbreviation Database at Stanford contains all abbreviations found in the titles and abstracts of MEDLINE records by the Chang *et al*. algorithm (22). The algorithm looks for parentheses in the text and scores the probability that the word(s) inside the parentheses may be an abbreviation or long form, and that its counterpart precedes it immediately. Once found, the algorithm aligns the parenthetical word or phrase against the preceding text using a dynamic programming algorithm similar to that used to align protein sequences.

It was discovered that the alignments between correct abbreviation/long-form pairs are distinctive and can be distinguished from incorrect ones. Many abbreviations are formed using the first letters of words, the syllables, etc. In alignments from incorrect abbreviations, the letters may be unaligned or aligned on internal letters. Thus, quality of the abbreviation is scored by rewarding characteristics that indicate correct abbreviations (e.g. letter in abbreviation matches first character of word in long form), and penalizing those that do not (e.g. letter in abbreviation is missing in long form). Such a strategy can distinguish correct abbreviations from incorrect ones. Although the algorithm is tolerant to variation, correct pairings may be idiosyncratic. For example, numbers are often dropped in gene names (e.g. RB1 for retinoblastoma).

The Stanford Biomedical Abbreviation Database is available on the web. Users can search the database for an abbreviation or a word that occurs in the long form. Because there may be small syntactic variations in the abbreviation or long form (e.g. RB1 and RB-1), the database aggregates similar ones and presents only ones that differ significantly. The abbreviation search functionality is also available as an XML-RPC web service, so that users can incorporate the search into their own programs (http://bionlp.stanford.edu/webservices.html). Sample code in Perl, Python and Java is provided, although the service can be accessed in any computer language.

AcroMed (http://medstract.med.tufts.edu/acro1.1/index.htm)

The Brandeis–Tufts bio-acronym server, AcroMed, is an automatically generated searchable database of over 481 500 biomedical acronyms and their associated normalized long-forms extracted from 11 million Medline records. Every acronym is displayed with its corresponding set of senses. Each acronym-long-form pair in the database is linked to the abstracts in which it was discovered, and the set of equivalent long-forms corresponding to a single sense can be submitted directly to PubMed as searches, by a single click, as a query reformulation. Furthermore, AcroMed also attempts to classify each acronym-long-form pair by its semantic type, using an ontology composed of both UMLS and GO taxonomic terms. Aliases of named entities are presently being incorporated into the acronym server as well (e.g. WAF1 as alias of p21).

The AcroMed server was constructed using two strategies for extracting acronym-meaning (long-form) pairs from the Medline corpus. First, a pattern-matching algorithm identifies an acronym and then moves left in the input string to determine candidates for the long form of the acronym. The input text is a simple sequence of strings. This is basically the same strategy that was used by the works mentioned in the previous section. Regular expressions were designed to match potential

acronyms and look for its contextual meaning. Some subroutines convert the potential acronym into a regular expression. This regular expression is used to search in the close context from the position where the potential acronym was found. Strings matching potential acronyms are rated with a formula to compare how good the acronym is to a comparison or threshold measure. Then each of its composing characters is checked, to match as a prefix or infix of the words that compose the string. If there is a match (a suffix that starts with the same character/symbol in the acronym) it is assigned a specific score. If the score is below a defined threshold, the pair is accepted.

In the second strategy, the application of the pattern-matching machinery was constrained above after having performed a robust phrase-level parsing of the input string. Once the proper syntactic structure was assigned to the Noun Phrase within which a potential acronym might occur, the finite-state matching algorithm was applied with considerable precision for identifying the long form. Both the precision and recall of this technique are significantly greater than that achieved in previous works. The reason for this marked improvement is due to several factors. Conventional approaches to acronyms have conflated two computationally distinct problems:

(i) Determining the window size of the text within which the long form for the acronym lies.
(ii) Identifying the long form by matching, deleting and simplifying character strings relative to the acronym itself.

Much greater accuracy can be attained if these two problems are treated as separate computational tasks. Importantly, the first problem is solved by a constrained context-free parsing algorithm, developed independently for the automated interpretation and extraction of protein and gene descriptions and their relationships in biomedical text in our larger project called Medstract (http://www.medstract.org). AcroMed entries are used by our other client programs in the context of identifying biorelations and metabolic pathways from Medline.

SaRAD (http://www.hpl.hp.com/research/idl/projects/abbrev.html)

The Simple and Robust Abbreviation Dictionary (SaRAD) system (24) was created as a by product of a very different problem. The algorithms were initially designed for use in a gene-mining application (25) and were intended to extract abbreviation pairs for the purpose of disambiguation. Although the algorithms were very simple, it was discovered that they were quite robust and thus SaRAD was born. The SaRAD system consists of three components: a mechanism for finding definitions for abbreviations, the clustering of those definitions and the generation of information useful for refining PubMed searches. Only abbreviation/definition pairs that appear more than once are retained in the database.

Definition extraction is achieved in a similar fashion to the other systems. Specifically, a window of text is extracted preceding a parenthetical abbreviation. The algorithm then extracts 'paths' through the definition window that match the abbreviation. Each path is scored by four simple heuristics (e.g. for every abbreviation character that is at the start of a definition one is added to the score, for every extra word between the definition and the parentheses subtract one, etc.). The highest scoring path with a score over zero is considered the best match. The algorithm is easy to implement and is very fast in practice. Because scores can be calculated as each path is being built, and because of the large scale of MEDLINE unlikely definitions or complex windows can be removed quickly making the algorithm computationally attractive.

To make the results more useful SaRAD visually clusters related definitions. This is important for plural definitions (Estrogen Receptor/Receptors), nested abbreviations (E. Receptor) and other variants (Estradiol/Estrogen). While stemming addresses a number of these cases, it is not realistic given the complexity of biomedical language. Disambiguation is achieved first though the use of *n*-grams. Briefly, the system breaks apart each definition into *n*-character sequences

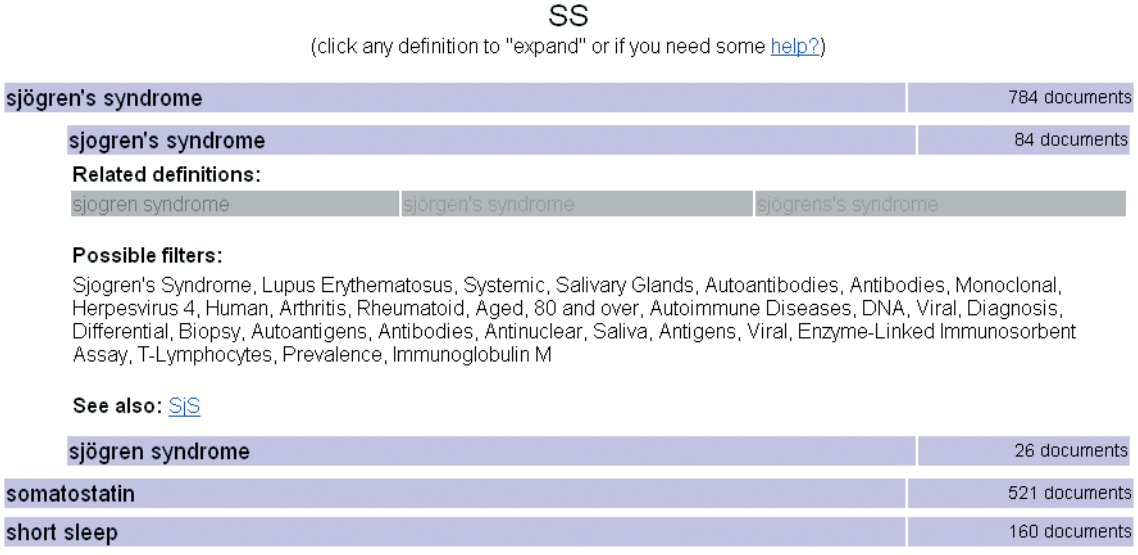

Figure 1. Screenshot of SaRAD. The user has searched for 'SS' and clicked to get details of the sub-definition 'sjorgen's syndrome'. The possible filters are MeSH terms useful for limiting search results.

(specifically tri-grams), represents those characters in vector-space (one dimension for each possible tri-gram), and performs a variant of hierarchical clustering. A secondary clustering uses the Medical Subject Headings (MeSH) annotations available in MEDLINE documents. Definitions extracted from documents with very similar MeSH headings are clustered.

Figure 1 is a screenshot of the SaRAD system where the user is looking at the details page for the abbreviation 'SS.' At the top of the page the interface displays the most popular definition in the cluster with all (MeSH clustered) variants listed below. Clicking on these definitions expands the display to reveal *n*-gram clustered results and any cross-references to other abbreviations with the same definition.

Users can narrow PubMed searches with MeSH terms extracted for clustering. For example, one could add the term 'Immunologic' to the query 'CDC' to get documents related to 'Complement Dependent Cytotoxicity' or appending 'Bile Acids and Salts' to find documents about 'Chenodexycholic Acid.' SaRAD contains a secondary interface (although non-public) that automatically clusters PubMed results based on these MeSH headings.

FUTURE DEVELOPMENT

Mapping biomedical abbreviations in an automated manner permits the continued refinement of recognition techniques, incorporation of and application to alternative domains of text, and flexibility in the data presented. While the overall false-positive rates in acronym-definition mapping are low, when processing large databases such as MEDLINE the primary challenge is that many such mapping events will occur and even a 1% false-positive rate can translate into tens of thousands of false-positive entries into the database. Nevertheless, we believe these databases and their algorithms will serve as foundations for the development of tools to analyze high-throughput biological data, and that currently the databases are useful resources for biologists.

ACKNOWLEDGEMENTS

We would like to thank the National Library of Medicine for providing us with electronically available copies of MEDLINE for analysis. This work was funded by NSF-EPSCoR EPS-0132534 (J.D.W.).

REFERENCES

1. Raychaudhuri,S., Chang,J.T., Imam,F. and Altman,R.B. (2003) The computational analysis of scientific literature to define and recognize gene expression clusters. *Nucleic Acids Res.*, **31**, 4553–4560.
2. Wren,J.D. and Garner,H.R. (2004) Shared relationship analysis: ranking set cohesion and commonalities within a literature-derived relationship network. *Bioinformatics*, **20**, 191–198.
3. Hirschman,L., Park,J.C., Tsujii,J., Wong,L. and Wu,C.H. (2002) Accomplishments and challenges in literature data mining for biology. *Bioinformatics*, **18**, 1553–1561.
4. Mack,R. and Hehenberger,M. (2002) Text-based knowledge discovery: search and mining of life-sciences documents. *Drug Discov. Today*, **7**, S89–S98.
5. Salwinski,L., Miller,C.S., Smith,A.J., Pettit,F.K., Bowie,J.U. and Eisenberg,D. (2004) The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.*, **32**, D449–D451.
6. Marcotte,E.M., Xenarios,I. and Eisenberg,D. (2001) Mining literature for protein–protein interactions. *Bioinformatics*, **17**, 359–363.
7. Jenssen,T.K., Laegreid,A., Komorowski,J. and Hovig,E. (2001) A literature network of human genes for high-throughput analysis of gene expression. *Nature Genet.*, **28**, 21–28.
8. Wren,J.D., Bekeredjian,R., Stewart,J.A., Shohet,R.V. and Garner,H.R. (2004) Knowledge discovery by automated identification and ranking of implicit relationships. *Bioinformatics*, **20**, 389–398.
9. Federiuk,C.S. (1999) The effect of abbreviations on MEDLINE searching. *Acad. Emerg. Med.*, **6**, 292–296.
10. Weeber,M., Schijvenaars,B.J., Van Mulligen,E.M., Mons,B., Jelier,R., Van Der Eijk,C.C. and Kors,J.A. (2003) Ambiguity of human gene symbols in LocusLink and MEDLINE: creating an inventory and a disambiguation test collection. *Proc. AMIA Symp.*, 704–708.
11. Chen,L., Liu,H. and Friedman,C. (2004) Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics*, in press.
12. Wain,H.M., Lush,M.J., Ducluzeau,F., Khodiyar,V.K. and Povey,S. (2004) Genew: the Human Gene Nomenclature Database, 2004 updates. *Nucleic Acids Res.*, **32**, D255–D257.
13. Park,Y. and Byrd,R. (2001) Hybrid text mining for finding abbreviations and their definitions. In *Conference on Empirical Methods in Natural Language Processing (EMNLP 2001)*, Pittsburgh, PA, June 3–4, 2001.
14. Larkey,L., Ogilvie,P., Price,A. and Tamilio,B. (2000) Acrophile: An Automated Acronym Extractor and Server. In *Proceedings of the 5th ACM Conference on Digital Libraries*, San Antonio, TX, June 2–7, 2000, pp. 205–214.
15. Taghva,K. and Gilbreth,J. (1995) Recognizing acronyms and their definitions. Information Science Research Institute (ISRI), University of Nevada at Las Vegas.
16. Yeates,S. (1999) Automatic extraction of acronyms from text. In *Proceedings of the Third New Zealand Computer Science Research Students' Conference (NZCSRSC'99)*, University of Waikato, Hamilton, New Zealand, 6–9 April, pp. 117–124.
17. Yoshida,M., Fukuda,K. and Takagi,T. (2000) PNAD-CSS: a workbench for constructing a protein name abbreviation dictionary. *Bioinformatics*, **16**, 169–175.
18. Yu,H. (2001) Knowledge-based disambiguation of abbreviations. In *Proceedings of the AMIA Annual Symposium (AMIA 2001)*, Washington, DC, November 3–7, 2001.
19. Schwartz,A.S. and Hearst,M.A. (2003) A simple algorithm for identifying abbreviation definitions in biomedical text. *Pac. Symp. Biocomput.*, 451–462.
20. Yu,H., Hripcsak,G. and Friedman,C. (2002) Mapping abbreviations to full forms in biomedical articles. *J. Am. Med. Inform. Assoc.*, **9**, 262–272.
21. Wren,J.D. and Garner,H.R. (2002) Heuristics for identification of acronym-definition patterns within text: towards an automated construction of comprehensive acronym-definition dictionaries. *Methods Inf. Med.*, **41**, 426–434.
22. Chang,J.T., Schutze,H. and Altman,R.B. (2002) Creating an online dictionary of abbreviations from MEDLINE. *J. Am. Med. Inform. Assoc.*, **9**, 612–620.
23. Pustejovsky,J., Castano,J., Cochran,B., Kotecki,M. and Morrell,M. (2001) Automatic extraction of acronym-meaning pairs from MEDLINE databases. *Medinfo*, **10**, 371–375.
24. Adar,E. (2004) SaRAD: a Simple and Robust Abbreviation Dictionary. *Bioinformatics*, **20**, 527–533.
25. Adamic,L.A., Wilkinson,D., Huberman,B.A. and Adar,E. (2002) A literature based method for identifying gene-disease connections. In Markstein,V.a.M.P. (ed.), *Proceedings of the First IEEE Computer Society Bioinformatics Conference (CSB2002)*, Stanford University, Palo Alto, CA, August 14–16, 2002. IEEE Press, NY, pp. 109–117.