Proceedings BioLINK SIG 2013

Roles for text mining in biomedical knowledge discovery and translational medicine

BioLINK: Linking Literature, Information and Knowledge for Biology
a Special Interest Group of the International Society for Computational Biology

ISMB/ECCB 2013
Berlin, Germany
July 20, 2013

Organizing Committee:
[Chair] Karin Verspoor, National Information and Communication Technology Australia (NICTA)
Christian Blaschke, Spanish Institute of Bioinformatics
Lynette Hirschman, MITRE Corporation
Hagit Shatkay, Dept. of Computer and Information Sciences, Center for Bioinformatics and Computational Biology, University of Delaware
Alfonso Valencia, Spanish National Cancer Research Centre

Program Committee:
Sophia Ananiadou, University of Manchester and NACTEM
Alan Aronson, US National Library of Medicine
Kevin Bretonnel Cohen, University of Colorado School of Medicine
Anna Divoli, Pingar Research
Larry Hunter, University of Colorado School of Medicine
Martin Krallinger, Spanish National Cancer Research Centre
Zhiyong Lu, US National Library of Medicine
Dietrich Rebholz-Schuhmann, University of Zurich and the European Bioinformatics Institute
Phoebe Roberts, Pfizer
Luis Rocha, Indiana University
Jun'ichi Tsujii, Microsoft Research Asia
John Wilbur, US National Library of Medicine
**BioLINK SIG: Roles for text mining in biomedical knowledge discovery and translational medicine**

https://sites.google.com/site/biolinksig2013/program

**Keynote:** Lars Juhl Jensen  
*The pragmatic text miner: From literature to electronic health records*

---

**Papers in this Volume**

**Long Papers**

[Pages 5-12] *Mining cis-Regulatory Transcription Networks from Literature.* Florian Leitner, Martin Krallinger, Sushil Tripathi, Martin Kuiper, Astrid Laegreid and Alfonso Valencia.


**Short Papers**

[Pages 28-31] *Neji: a tool for heterogeneous biomedical concept identification.* David Campos, Sérgio Matos and José Luis Oliveira.


**Poster Abstracts**

[Page 57] *A quantitative analysis of causal and associative events involving genes and proteins.* Phoebe Roberts.


[Page 62] *PubAnnotation - a storage system for sharing of literature annotation.* Jin-Dong Kim.


Long Papers

*Mining cis-Regulatory Transcription Networks from Literature.*
Florian Leitner, Martin Krallinger, Sushil Tripathi, Martin Kuiper, Astrid Laegreid and Alfonso Valencia.
Pages 5-12

*Occurrence of Gene Ontology, Protein Ontology, and NCBI Taxonomy Concepts in Text toward Automatic Gene Ontology Annotation.*
Michael Bada, Dmitry Sitnikov, Judith Blake and Lawrence Hunter.
Pages 13-19

*Comprehensive Benchmark of Gene Ontology Concept Recognition tools.*
Christoph Jacob, Philippe Thomas and Ulf Leser.
Pages 20-26
Mining cis-Regulatory Transcription Networks from Literature

Florian Leitner¹*, Martin Krallinger¹, Sushil Tripathi², Martin Kuiper³, Astrid Lægreid², and Alfonso Valencia¹*

¹Structural Computational Biology Group
Spanish National Cancer Research Center, Madrid
* {fleitner,avalencia}@cnio.es
²Department of Cancer Research and Molecular Medicine
Norwegian University of Science and Technology, Trondheim
³NTNU Semantic Systems Biology Group
Norwegian University of Science and Technology, Trondheim

While transcription regulation is a key biological process, to date no public reference repository for these interactions exists. The IMEx consortium unites several protein interaction DBs, but there is no similar public repository available for Transcription Regulation Events (TREs; DNA-binding of a transcription factor to a target gene’s promoter element that leads to the (up- or down-) regulation of the target). We now are building such a repository, based on TRE descriptions in the scientific literature, integrating text mining and manual curation. Annotating these interactions is a complex issue, as TREs are a subset of gene regulation events (GREs; direct or indirect regulator-target gene interactions that may involve a transcription event) that current text mining methods are extracting and go beyond the associations of transcription factors to DNA sequences derived by ChIP-seq. Detecting transcription factor DNA-binding events with their regulatory effect on target genes is a more complex issue than either text mining or ChIP-seq are currently solving. Additionally, most current text mining systems do not provide “normalized” (sequence DB-mapped) interactors, while curators spend a major part of their time resolving these identifiers.

We will present our ongoing work on an open-source, UIMA-based extraction system and the parallel effort to build a Gold Standard corpus of functional TREs consisting of directed, normalized transcription factor-target gene interactions. The key aspects of the system are: (1) its focus on transcription regulation events, (2) the integration of biologically relevant context, particularly experimental conditions and host tissue, (3) an organism-independent DB ID mapping of the participants, (4) a generic syntax expression language that enables syntactic pattern mining in UIMA, (5) an evaluation based on a hand-curated Gold Standard, (6) and the design of our curation standards for TREs to assemble this Gold Standard.

1. Introduction to Transcription Regulation Events

A significant proportion of a cell’s regulatory capabilities are directed towards its RNA expression landscape (Djebali et al., 2012), with cis-regulatory transcription events at the core of the intracellular gene regulatory network (Levine and Tjian, 2003). In eukaryotes, and particularly the metazoa (multicellular eukaryotes), transcription initiation requires the binding of transcription factors (TF) to particular sequence motifs at proximal regions several hundred base pairs upstream of the target gene’s (TG) transcription start site (TSS), and distal promoter regions that are further away (Lenhard et al., 2012). These events trigger the activation of RNA Polymerase II (RNA Pol II), which binds via the Pre-Initiation Complex (PIC) at the transcription start site (TSS) of protein coding and miRNA coding genes. RNA Pol II activation is mediated through a chain of protein interactions between a specific, DNA-binding TF and additional co-factors, and ultimately initiates mRNA synthesis for a particular target gene (TG) (Figure 1). In general, cellular signals are routed through these “network switches” that determine gene expression levels and thus cellular state. Given the relative sparsity (James et al., 2010) of these transcription regulation events (TREs) (Djebali et al., 2012) if compared to all possible genes a TF could pair

Figure 1: A transcription regulation event (TRE). The binding of a transcription factor (TF) to DNA in regulatory regions of the target gene (TG) leads to the regulation of RNA Pol II activity that binds via the PIC at the TSS, either directly or mediated via co-factors. A TF can be an activator or repressor (not shown) and bind distal or proximal. Downstream of the TSS, the first exon of the TG is shown.
with, a comprehensive map of the mammalian – and particularly, human – routing network is not (yet) available (Gerstein et al., 2012). However, given its central role in both normal and pathological processes, determining the transcription factor-target gene relations of gene regulatory networks is of general interest to a broad audience within the biological, medical and pharmacological sciences.

We will refer to the chain of events, from the TF binding to specific DNA regulatory regions at the TG to the particular outcome of up- or down-regulating the TG, as (direct, functional) TREs. This should be contrasted to (indirect, generic) gene regulation events (GREs), where any gene or protein can influence the expression of another (target) gene. In these cases, the actual chain of events leading to that outcome might involve several interactions via intra- and even inter-cellular signaling pathways (e.g. gene regulation in a given cell triggered by a ligand secreted from a different cell). In other words, while TREs are a subset of all GREs, the latter do not specifically describe the event of a TF directly activating or suppressing a TG, but may refer to more complex processes involving additional mediators, including intermediate transcription events. For example, the sentence “Furthermore, p50 binds and activates the CEBPA gene in myeloid cells.” allows an expert to infer a TRE because of the keyword binds and given the external knowledge that p50 is a TF. Our final goal is to extract the TF, the TG, the directionality, and cell type (p50, CEBPA, up-regulation, and myeloid cells, respectively) from this phrase. On the other hand, in “Thus, C/EBPα and p50 reciprocally regulate each other’s expression, establishing a positive feedback relationship.”, the two regulatory events (C/EBPα up-regulating p50 and p50 up-regulating C/EBPα) are clearly GREs, but the available context is insufficient to infer a TRE.1

Another example is the report of the activation of (the GLI transcription factor family member) GlI1, mediated via aPKC-ε/λ phosphorylation, that in turn gets regulated by protein smoothened (SMO), a hedgehog (HH) receptor. This chain of events results in the transcriptional activation of the aPKC-ε/λ gene (Prkci) in basal cell carcinomas (BCCs) (Atwood et al., 2013): In the paper, with sentences such as “Prkci is a HH target gene that forms a positive feedback loop with GLI and exists at increased levels in BCCs.”, the aPKC-ε/λ gene (Prkci) as well as GLI1 are presented several times as the “target genes” of hedgehog (HH). However, while the TRE is the transcriptional activation of Prkci (the TG) by GlI1 (the TF), HH and SMO are extra- and intracellular mediators of gene regulation events (GREs) with GlI1 and Prkci. Furthermore, it is probably impossible to deduce the TRE from the abstract alone, providing a cause to mine article bodies, too. Finally, the title “GLI activation by aPKC ε/λ regulates the growth of basal cell carcinomas” indicates a (reverse, protein-interaction-based) GRE, not the correct TRE.

2. Experimental TRE Detection

Over the last few years, high throughput methods have been used to trace gene regulatory networks, in particular chromatin immunoprecipitation sequencing (ChIP-seq) (Park, 2009) which detects the association of specific, genomic DNA sequences with an immunoprecipitated protein. However, determination of TREs still relies on a combination of several experimental approaches other than ChIP, many of which are mainly used in a small-scale, “low-throughput” manner, like electrophoretic mobility shift and reporter gene assays. Observations from these assays are warranted because ChIP-seq yields signals also in cases where the assayed TF binds TG regulatory regions indirectly, via another protein and thus does not ensure direct TF-DNA-binding. Furthermore, the actual transcription regulation event and its directionality (activation vs. repression) cannot be determined by ChIP-based methods (Valonev et al., 2008; Lickwar et al., 2012). To overcome such limitations, TF-sequence associations from ChIP-seq data are often combined with complementary gene expression data to predict the TGs (Sandmann et al., 2006; Huang et al., 2013). The fact that experimental data documenting functional TREs are mainly derived from small scale perturbation-type experiments performed over the last few decades implies that a wealth of TRE knowledge remains available only within the confines of a large body of published literature (A PubMed query for “Transcription Factors[MeSH Terms] AND Regulatory Sequences, Nucleic Acid[MeSH Terms]” on April 8, 2013, produced exactly 57,000 hits, and, for example, RegulonDB v8.0 (a yeast TRE DB) alone records 4667 papers.) Therefore, extracting these transcription events with text mining methods could provide a large number of direct, functional interactions that are not recorded anywhere else.

3. Methods for Extracting Gene Regulation Events

Despite a large number of database bio-curation efforts that are cataloging information relating to TFs and TREs2, a comprehensive collection based on purely manual approaches will likely remain elusive; Manual curation efforts do not scale with the exponential growth of available literature (Baumgartner et al., 2007), an effect that has been observed particularly for protein interaction data/curation (Ceol et al., 2008; Leitner et al., 2010a). Therefore, a possible option to bridge the widening gap between structured TRE-related repositories and the existing literature is text-mining facilitated extraction, similar to ongoing efforts for protein interactions (Krallinger et al., 2012). The earliest GRE extraction systems date back as far as 2004 (Saric et al., 2004; Valionyte et al., 2008).

---

1both sentences taken from PubMed abstract 21346255

2The NAR Database issue 2013 list 76 resources for TFs and their regulator sites at http://www.oxfordjournals.org/nar/database/subcat/1/4
Table 1: A classification of text mining systems for gene and/or transcription regulation events. Columns: Pattern-based: system uses (linguistic) patterns to detect relationships; Parser-based: uses dependency parsing to detect relationships; Machine Learning-based: uses statistical models to filter and/or rank (potential) interaction pairs; Gene Regulation Relations: explicitly extracts regulator-target gene relationships (*: limited; see text); Additional Relations: in addition, detects several (other) kinds of biologically relevant relationships; Target Gene: detects the target (gene) of the GRE; Gene Normalization: maps the detected regulator/TG to a database identifier; Model Organism Only: only extracts GREs for: S. cerevisiae (Saric, 2004 and R.-P., 2007), B. subtilis (Manine, 2009), A. thaliana (Krallinger, 2009), M. musculus (Roy, 2011).

Pan et al., 2004), and the former provided the basis of the STRING-IE system (Saric et al., 2006). (Hahn et al., 2009) compiled an excellent review and performance comparison of GRE extraction systems. By now, a number of text-mining systems have been published on the general topic of gene regulation event extraction; Table 1 shows several systems that have been created to detect such regulatory relationships, although not all systems shown can detect regulator-target relationships. It is important to notice that none of these systems has an explicit focus on functional TF-DNA-binding events (i.e., TREs as described in Figure 1).

With one exception, all relationship extraction systems will produce interactions and are evaluated against data that stem from descriptions of the indirect, more generic gene regulation events (e.g., pathway regulators or even extracellular peptides that trigger gene expression): Regarding TRE-based text mining systems, (Rodriguez-Penagos et al., 2007) did evaluate their text mining system against RegulonDB (yeast) data, which is comprised only of direct TF-TG interactions, although it should be noted that the regulome of yeast is less complex than that of metazoa. (Yang et al., 2008) extracted transcription factors contexts (associated GO and MeSH terms), but did not detect target genes, and the system does not normalize the TFs. (Aerts et al., 2008) applied a very successful gene normalization strategy, improving the mapping by BLASTing for sequence snippets found in the text in addition to mentions of gene symbols and names. Another noteworthy issue is that only one system (Krallinger et al., 2009) covers cell or tissue specificity for the mined interactions, which seems particularly important given its relevance in gene regulation. The work of (Wang et al., 2011) seems most related given the title, however, their evaluation was limited to only one TF family (HIF-1) in one organism (M. musculus), the patterns do not distinguish GREs from TREs, and no gene normalization is provided. Finally, (Klinger et al., 2011) and (Riedel et al., 2011) are cited as two example systems related to the BioNLP Shared Tasks (Kim et al., 2012). In this community challenge, within the “GENIA task 2”, gene expression events that are combined with (positive and negative) regulatory relations could be used to deduce regulator-target entity relationships. However, the BioNLP Shared Tasks does neither implement an explicit evaluation of the pairing between regulators and targets or the mapping of these entities to their databases (known as “Gene Normalization”, as for example required in the BioCreative protein interaction pair tasks (Krallinger et al., 2008; Leitner et al., 2010b)). In summary, only the systems of Rodriguez-Penagos and Krallinger come close to the objective of extracting di-
rect, functional TRE events triggered by TF-DNA binding that lead to regulation of target gene expression reported in the form of normalized (DB mapped) TF-TG relationships, but their approaches are limited to the yeast and A. thaliana model organisms, respectively.

4. The txtfnnl Pipeline

To quote Saric (2004):

It is often not known whether the regulation takes place at the level of gene transcription or translation or by an indirect mechanism. For this reason, and for simplicity, we decided against trying to extract how the regulation of expression takes place.

On the other hand, high-throughput ChIP-Seq is able to detect TF-DNA association events, but direct binding cannot be ascertained and it has difficulties identifying the functional transcription regulatory aspects (target gene, directionality) of a TRE. Given the complementary nature of the text mining and ChIP-seq approaches, we are creating a framework, the txtfnnl pipeline3 (Figure 2), to explicitly extract direct, cis-regulatory transcriptional regulation events using text mining that could be combined with ChIP-seq data to produce an optimal set of functional TRE relations. Our target is to extract normalized (i.e., sequence database-mapped) TREs from full-text, based on patterns identified by biologists as potentially describing cis-regulatory transcription events (Table 2, top, and next section).

For the “Natural Language Processing” step in Figure 2, the OpenNLP4 MaxEnt tagger (sentence segmentation), the GENIA Tagger (Tsuruoka et al., 2005) (part-of-speech tagging and chunking), and the BioLemmatizer (Liu et al., 2012) are used for linguistic pre-processing, and are wrapped as UIMA Annotators. To enable the “Pattern Detection” step, we have developed a library (libfsmg) that provides generic Java classes for compiling non-deterministic finite state machines using back-tracking for the identification of subgroup matches5. We then designed a sentence-level, (regular) syntax expression grammar that makes use of this library (see Table 2, bottom). The syntax patterns for this step are manually created from a “seed” collection of patterns (see Table 2, top), enriching them with linguistic properties using the displayed grammar (bottom). Terms in the generic patterns (top) shown in parenthesis are optional, but increase the specificity of a match, separating it farther from GREs and protein interactions events. In the case of site in the noun phrase head of a TF mention, the keyword may be replaced with “motif”, “element”, or “sequence”. For promoter, also “promoter region”, “enhancer”, and “silencer” may be applied. The TF/TG body of any noun phrase may be used as prepositional complement (“TG be direct target of TF” instead of “TG be direct TF target”). For the special preposition within, the TG promoter element does not need to be present: For all other prepositions, we detected ambiguity with protein interactions, but all others may be substituted with proper alternatives. Regulate/regulation can be replaced with any of its synonyms or coordinate terms, possibly indicating the directionality of the event (e.g., “activate” or “inhibition”). In other words, directionality is determined by the presence of selected verbs (or their nominalized forms). For example, the generic pattern “TF direct regulate TG” can be re-written as “direct repression of TG by TF”, indicating a down-regulation event.

To translate these patterns into syntax expressions, the LHS grammar rules (Table 2, bottom) in quotes or rectangle brackets represent terminals; <token> (describing a token) and <lemma> (describing the lemma of a token) can be substituted with any standard regular expressions. I.e., nested within the syntax expressions, regular expressions can be applied to match tokens and/or their lemmata. A Token can be quantified or generalized with a match-any expression (dot, “.”). Phrases (delimited by “[“ and “]”) can be declared optional (“?”). This is an example syntax expression that would implement “TF direct regulate TG” and detects an up-regulation event (“activate”) of a TF on a TG (subgroups, delimited by “(“ and “)“):

\[
\begin{align*}
[ \text{NP} ( . + ) ]. & \ast \text{RB}\_\text{direct} \ [ \text{VP} . \ast \text{activate} ] \ [ \text{NP} \ DT . * \ ? ( . + ) ]
\end{align*}
\]

This would match “Here, we show that intracellular Aβ42 directly binds and activates the p53 promoter, ...” (PMID 15548589) and produces the tokens “intracellular Aβ42” (TF) and “p53 promoter” (TG) as subgroup matches. The high specificity of these patterns also explains the need for mining article bodies, making it more likely to find instances of these restrictive patterns.

After the TRE pattern detection step, UIMA Annotators are provided for detecting lists of bio-entity terms (such as organism names or genes and protein symbols) in the two last steps, entity recognition and gene/protein mapping. For the first, the pipeline makes use of the Linnaeus Tagger (Gerner et al., 2010), a system specifically tailored for annotating term collections (“Gazetteers”) of bio-entities, and in particular, provides a dictionary for matching organism mentions. However, the underlying finite state automaton6 was not able to compile a state machine over the collection of approximately 23 million gene symbols collected for the txtfnnl pipeline. Therefore, instead, we implemented a concurrent PATRICIA trie7-based UIMA Annotator for handling very large collections (i.e., millions) of terms. Optionally, term matching in this trie-based Annotator can be limited to coin-

3http://github.com/fnl/txtfnnl
4http://opennlp.apache.org/
5http://github.com/fnl/libfsmg
6brics automaton, http://www.brics.dk/automaton
7http://code.google.com/p/concurrent-trees
Figure 2: The particular configuration of the UIMA-based txtfnnl text mining pipeline for transcription regulation event extraction. The pipeline consists of a series of UIMA annotators; the first extracts content from most filetypes (e.g., PubMed Central XML, Elsevier XML, MEDLINE abstracts, or plain HTML), wrapping a modified Apache Tika (http://tika.apache.org/). Several Natural Language Processing (NLP) Annotators apply sentence segmentation, part-of-speech tagging, phrase chunking, and lemmatization. The pattern annotator (based on the libfsmg library) detects sentences that match a particular TRE pattern (see Table 2). Finally, several Annotators tag named entities with concept identifiers (from ontologies or databases) of particular organisms, cell-lines and tissue types, and gene or protein identifiers (that are first collected using the gnamed tool). This produces (potential) TREs consisting of the TF, TG, the sentence(s) the TRE is mentioned in, as well as any cell lines or tissues mentioned. The pipeline itself is open source, available via GitHub, and can be installed using Apache Maven. Because the patterns and Gazetteers are fed into it as parameterized arguments, the pipeline can be applied to other tasks as well.

5. Resources for TRE Mining and Curation

In addition to the pipeline itself, several other resources are under construction: (1) a compilation of mammalian transcription factors (“TFCheckpoint”, (Chawla et al., 2013)), (2) an expansion to the PSI:MI ontology terms to cover experimental methods for TREs, (3) a collection of TRE-relevant patterns, (4) a Gold Standard of TRE-annotated articles, and (5) curation guidelines to create this Gold Standard.

The only pre-existing data that records functional TRE interactions on a per-article basis are the resources made available by transcription factor databases as, for example, TRED (Jiang et al., 2007) or ORegAnno (Griffith et al., 2008). However, these databases exclusively curate interactions for which relevant experimental evidence was generated in that article. In other words, if a direct, functional TRE is described only with cited evidence for the interaction, it will not be curated. Another important aspect to consider stems from the frequent use of supplementary or any other “external data” by human curators, such as following references to determine the correct sequence record to annotate. These issues imply that the use of commonly available bio-curation results as a Gold Standard for text mining systems will incur false positives for results that are not wrong per se, and will increase the false negatives for pairs that are impossible to extract (“external” material), leading to a possibly overly strict comparison and evaluation. While there do exist well known annotated corpora for GREs
Furthermore, we have created curation guidelines to establish a common curation standard for extracting these interactions. We are currently in the process of annotating these articles each with at least three curators (trained in molecular biology), and will then keep resolving agreement issues until a final consensus is reached that overlaps with the guidelines. The final Gold Standard should contain, for each TRE, the TF as UniProt accession and the TG as EntrezGene ID. In addition, two lists may be annotated: the relevant tissue/cell type(s) (as BRENDA Tissue Ontology IDs) and the experimental evidence code(s) that were used to trace the interaction (as PSI Molecular Interaction ontology IDs). However, the IMEx’ PSI MI ontology focusses on detection of protein interactions rather than transcription (regulation) events (Orchard et al., 2012). Therefore, we are collaborating with members of the IMEx consortium to explore ways of expanding the PSI MI ontology with relevant experimental evidence types for detecting TREs that are not (yet) part of this ontology.

To generate the syntax expressions for the text mining pipeline, we extracted and lemmatized clauses that contained both a TF and TG mention from 1932 TRED-annotated articles. With the use of the MyMiner bio-curation tool (Salgado et al., 2012), the 2756 clauses that appeared at least three times in the whole corpus were repeatedly classified by four independent expert curators as describing a direct TRE or not, until a consensus was reached for all of them, identifying 149 of the 2756 passages as describing direct TREs. The most generic patterns that cover these 149 cases are shown in Table 2 (top), while another 531 clauses, or roughly 3.5 times as many, were classified as describing GREs. All TRE clauses have some context-based reference in common that allowed the curator to infer a direct TRE; In the final patterns, this is ensured by either

1. The explicit mention of a direct transcription regulation event or TF binding.
2. The presence of the keyword promoter in the head of the noun phrase containing the TG.
3. The mention of a binding site in the head of the noun phrase containing the TF.
4. A reference to an experimental method that can be applied to detect a TRE.

These 18 patterns represent the exhaustive list of cases that permitted the curators to infer a direct TRE assuming the regulator is a known TF for all inspected 2756 clauses. The main conclusions drawn from this exercise are that a transcription-relevant DNA element or experimental method term must be present to allow inference of cis-regulatory events by experts. The only valid alternative is a traceable author statement declaring a direct promoter binding event (“TF directly regulates TG”). Finally, the regulator always must be a known TF. These strict limitations on the patterns represent

<table>
<thead>
<tr>
<th>Tag</th>
<th>RegEx</th>
<th>Quantifier</th>
<th>Token</th>
<th>Chunk</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>→ Phrase S?</td>
<td>Capture S?</td>
<td>Token S?</td>
<td></td>
</tr>
<tr>
<td>Phrase</td>
<td>→ “[“ Chunk InPhrase “]” “?”</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capture</td>
<td>→ (“ S “)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>InPhrase</td>
<td>→ CapInPhr InPhrase?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CapInPhr</td>
<td>→ (“ InPhrase “)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chunk</td>
<td>→ “NP”</td>
<td>“VP”</td>
<td>“PP”</td>
<td>“ADVP”</td>
</tr>
<tr>
<td>Token</td>
<td>→ “.”</td>
<td>Quantifier?</td>
<td></td>
<td>RegEx Quantifier?</td>
</tr>
<tr>
<td>Quantifier</td>
<td>→ “*”</td>
<td>“?”</td>
<td>“+”</td>
<td></td>
</tr>
<tr>
<td>RegEx</td>
<td>→ “[&lt;token&gt; “.”]? Tag “.”]? &lt;lemma&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag</td>
<td>→ “JJ”</td>
<td>“NN”</td>
<td>“VBZ”</td>
<td>“...”</td>
</tr>
</tbody>
</table>

Table 2: The generic patterns judged by human experts to likely describe TREs (top). A syntax expression grammar (bottom) is then used to endow these generic patterns with linguistic context. The generic patterns (top) are ordered by the frequency of the first matching passage extracted from TRED full text articles and are being manually translated to the syntax expressions (bottom). See text for details.

- such as the GeneReg corpus (Buyko et al., 2010), or the LLL05 corpus (Nédellec, 2005), in addition to the already mentioned GENIA Shared Task corpora - these collections were not created specifically for (direct) transcription regulation events. For all these reasons, we are now creating a dedicated Gold Standard where, on one hand, cited interactions may be included if the description allows biologists to deduce a direct TRE (implicitly trusting authors are reporting true facts), while at the same time “external data” is never used to create annotations. A good Gold Standard should represent an as diverse as possible sample: We selected 70 articles since 1999 onwards from 33 different journals and describing TRE interactions for over 100 different transcription factors from all major TF families (as defined by (Vazquez et al., 2009)). Furthermore, we have created curation
the main difference of our approach to existing pattern-based systems. In another effort related to this project, (Chawla et al., 2013) have already created a repository of all known and putative mammalian (in particular, human, mouse, and rat) transcription factors and are now in the process of manually curating published experimental evidence that will allow classification of specific DNA-binding TFs. This work will be used to confine the extraction of the text mining pipeline by limiting the TF entity space that may be mapped for regulator mentions.

6. Prospects of this Work

The extraction system and resources presented explore and define standards for a public repository of transcription regulation events, similar to the existing MIMIx and IMEX standards for protein interactions (Orchard et al., 2007). The text mining aspects of the project provide a pipeline tweaked for high-precision to automatically extract TRES and bootstrap the low confidence repository, and a complementary high-recall version that will be used to assist human curation in the framework of a specific bio-curation tool developed for TREC extraction and annotation.

7. Acknowledgements

We would like to thank Karen Christie and Jane Lomax for introducing us to the transcription factor curation and ontology being developed at the JAX Laboratory and EBI. We are grateful to David Salgado for continuously updating MyMiner as to cater to our curation needs. FL wishes to acknowledge Miguel Vazquez particularly for the discussions on the algorithmic implementations of the pattern and entity matching annotators.

The work was supported by the grant BIO2007-66855 from the Spanish Ministry of Science and Innovation and by the Spanish National Bioinformatics Institute (INBISCIII).

References


R. Klinger, S. Riedel, and A. McCallum. Inter-Event Dependencies support Event Extraction from Biomedical Literature. In *Workshop on Mining Complex Entities from Network and Biomedical Data*, 2011.


Occurrence of Gene Ontology, Protein Ontology, and NCBI Taxonomy Concepts in Text toward Automatic Gene Ontology Annotation of Genes and Gene Products

Michael Bada¹, Dmitry Sitnikov², Judith A. Blake³, and Lawrence E. Hunter¹
¹University of Colorado Anschutz Medical Campus, Aurora, CO, USA
²Mouse Genome Informatics, The Jackson Laboratory, Bar Harbor, ME, USA

ABSTRACT
Annotations of genes and gene products in model-organism databases with Gene Ontology (GO) terms have become an important knowledge resource in biomedical research, which has spurred many efforts at automating this labor-intensive manual curatorial activity, including many text-mining approaches. In an effort to provide some guidance on these text-mining efforts, we have used a gold-standard manually annotated corpus to conduct an evaluation of the occurrence of three types of fundamental GO-annotation concepts in 34 journal articles that were the evidential bases of approximately 220 GO annotations largely created by the Mouse Genome Informatics (MGI) group.

In addition to an analysis of the occurrence of the GO concepts of the curated GO annotations associated with these articles in the corpus, we have analyzed the occurrence of NCBI Taxonomy (NCBITAXON) and Protein Ontology (PRO) concepts corresponding to the species-specific genes/gene products of these curated GO annotations. The GO, NCBITAXON, and PRO concepts corresponding to the curated GO annotations were analyzed both in the full-text versions of these articles as well as in only those sentences of the articles providing the strongest evidence for the GO annotations, as specified by an official MGI GO curator. Though this sample set may not necessarily be representative of all GO annotations, our analysis suggests that full-text articles mention substantial fractions of the GO concepts at least once; however, the mentions of these GO concepts constitute very low percentages of the mentions of all GO concepts in these articles. Nearly all PRO concepts corresponding to GO annotations are mentioned at least once in the full articles, and these PRO mentions constitute a substantial fraction of the mentions of all PRO concepts in these articles. Mus musculus is seldom mentioned, though mice (strictly corresponding to the genus Mus) are mentioned at least once in the full articles, and these Mus mentions also constitute a substantial fraction of the mentions of all NCBITAXON concepts in these articles. For all of the ontologies, counts of annotated concepts corresponding to the curated GO annotations in only the strongly evidential sentences are comparatively very low, amounting to several mentions or fewer per article. However, for most of the ontologies, concepts corresponding to the curated GO annotations appear overrepresented, though this must be viewed cautiously given that this is based on very low counts. Thus, it remains to be further examined whether this overrepresentation overrides the very low mention frequency and thus whether it would be beneficial for automatic GO-annotation systems to focus on these evidential sentences.

1 INTRODUCTION
Annotations of genes and gene products in model-organism databases with Gene Ontology (GO) terms have become an important knowledge resource in biomedical research (The Gene Ontology Consortium, 2000; Camon et al., 2004; Lee et al., 2005). This has spurred many efforts at automating this labor-intensive manual curatorial activity, including text-mining approaches (Camon et al., 2005; Winnenburg et al., 2008). In an effort to provide some guidance on these text-mining efforts, we have conducted an evaluation of the occurrence of three types of fundamental GO-annotation concepts in articles that were the evidential bases of GO annotations largely created by the Mouse Genome Informatics (MGI) group, who curate a wide range of data for the primary international database resource for the laboratory mouse (Drabkin and Blake, 2012).

For this effort, we have employed the Colorado Richly Annotated Full-Text (CRAFT) Corpus, a gold-standard corpus of journal articles whose full-text versions have been manually marked up with ~140,000 concept annotations, relying on nearly all of the concepts of eight prominent biomedical ontologies; it has also been manually marked up in a variety of other ways, including syntactic, coreferential, and discourse annotation (though these other types of annotation were not analyzed in this study) (Bada et al., 2012; Verspoor et al., 2012). In addition to an examination of the occurrence of the GO concepts of the curated GO annotations associated with these articles in the corpus, we have

¹To whom correspondence should be addressed.
analyzed the occurrence in the corpus of NCBI Taxonomy (NCBITAXON) (Sayers et al., 2009) and Protein Ontology (PRO) (Natale et al., 2011) concepts corresponding to the species-specific genes/gene products of these curated GO annotations. The GO, NCBITAXON, and PRO concepts corresponding to the curated GO annotations were analyzed both in the full-text versions of these articles as well as in only those sentences of the articles providing the strongest evidence for the GO annotations, as specified by an official MGI GO curator.

It is important to note that we are not examining the occurrence of these concepts merely in terms of their mentions as exact string matches to the concepts’ primary labels or synonyms in the text of these articles. Rather, every mention semantically equivalent to a concept in one of these ontologies (with rare exception) has been annotated with the corresponding concept according to the CRAFT concept-annotation guidelines (Bada et al., 2010), whether or not the textual mention matches the concept label or one of its synonyms. Thus, ours is an analysis of the potential for recognizing the species, genes/gene products, and biological functionalities of GO annotations by looking for these concepts in text rather than an examination of what is possible with current text-mining technology, which would very likely miss some of these gold-standard concept annotations and incorrectly annotate other spans of text.

2 METHODS

We omit from this paper discussion of the markup of the concept annotations of the CRAFT Corpus (including the NCBITAXON, PRO, and GO concept annotations, which are analyzed in this study), as it has been extensively described elsewhere (Bada et al., 2012).

The markup of the sentences of the articles of the CRAFT Corpus providing strong evidence upon which curators most relied for their GO annotations of genes/gene products associated with these articles was performed within Knowtator (Ogren, 2006), a tool for ontology-based annotation of text implemented as a tab plugin for Protégé-Frames (Gennari et al., 2003); this is the same tool that was used to perform all of the concept annotation of the CRAFT Corpus. For the annotation of these evidential sentences, a simple ontology was manually constructed, including one class representing GO annotations and another representing evidence annotations. For the former, properties were defined for the Entrez Gene ID of the annotated gene, the GO term ID and primary label used for the annotation, the GO evidence code of the annotation (which specifies the type of evidence supporting the annotation), qualifier(s) of the annotation (such as one indicating negation), and for the sentences supporting the given GO annotation. A new Protégé-Knowtator project was created based on this ontology, and the GO-annotation class was programmatically populated with instances of curated GO annotations, with the appropriate values of all properties (except for that for the evidential sentences) filled in. The annotation of the evidential sentences was performed by an official MGI GO curator (DS), who, for each curated GO annotation instance, created instances of evidence annotations by selecting appropriate sentences and added them as property values for the appropriate GO-annotation instances. This markup was periodically reviewed by the project lead (MB) to check that the evidential sentences were being consistently marked up.

Though the full CRAFT Corpus consists of 97 articles, only 67 of the articles have been included in the 1.0 public release. (The other 30 are being temporarily reserved for use in future text-mining competitions, after which these too will be released.) The articles of the CRAFT Corpus were partly selected based on their serving as evidential bases for curated annotations of genes/gene products with terms from the GO and/or the Mammalian Phenotype Ontology (MPO) (Smith and Eppig, 2010). Since in this study we are analyzing only the GO annotations associated with these articles, we narrowed down the 67 publicly released articles to the 36 articles associated with one or more curated GO annotations. One of these articles (PMID:14611657) is an outlier in that it is associated with 4,524 curated GO annotations; this very large number of annotations for this paper (which correspond to a large set of olfactory receptor genes identified through a screening of a mouse olfactory epithelium cDNA library) would have completely eclipsed all of the other annotations in this study, and so we excluded this paper and its annotations. Another paper (PMID:16870721) was associated with one curated GO annotation, but during the course of this project, it was discovered to be an erroneous annotation; it has since been removed by MGI from its database, and so this paper and its annotation were excluded from this study as well. Excluding these two papers results in 34 papers with 254 curated GO annotations. An additional 28 curated GO annotations were excluded from this study since no evidential sentences were selected from the corresponding articles for these annotations by the MGI GO curator. (The large majority of these annotations were based on sequence or structure similarity of the annotated genes/gene products to homologous sequences (GO evidence code ISS) that presumably were studied in the corresponding articles.) This resulted in 34 papers with 226 curated GO annotations.

This study includes an analysis of the occurrence of NCBITAXON and PRO concepts in these articles; however, the curated GO annotations were originally specified for genes/gene products by their Entrez Gene IDs. Therefore, we mapped these Entrez Gene IDs (which refer to species-specific genes) to their corresponding NCBITAXON and PRO concepts, designating species and species-nonspecific genes/gene products, respectively. Properties for PRO and NCBITAXON IDs were created in the Protégé-Knowtator...
project for the class of GO annotations, and the values for these IDs were manually entered for all of the GO-annotation instances. Thus, in the Protégé-Knowtator project of curated GO annotations, each GO annotation instance is formally associated with its corresponding GO, NCBITAXON, and PRO concepts as well as with the evidential sentences supporting the given annotation, all of which can be programmatically queried.

We subsequently noticed that for eight of the articles, there was a pair of curated GO annotations with identical NCBITAXON-PRO-GO triples; these are the result of the two annotations of a pair having only different GO evidence codes (i.e., based on different types of evidence) or of one of the two annotations of a pair having an additional qualifier specifying that the gene/gene product contributes to a functionality rather than possessing the functionality itself. For the full-text article analysis, since we are only analyzing the occurrence of NCBITAXON, PRO, and GO concepts in the full-text versions of the articles and not these other aspects, we removed one of each of these pairs so as not to double-count them within their respective articles, resulting in 218 curated GO annotations for the full-text analysis. For the analysis of the concepts only within the evidence annotations, four of the eight pairs of duplicate NCBITAXON-PRO-GO annotations have different evidence annotations and so the concept annotations to be analyzed are in different "documents", i.e., pieces of text; thus, for these four pairs, one annotation of each of the pairs was added back, resulting in 222 curated GO annotations for the evidence-annotation analysis.

Our analysis was implemented in a Java program. First, the Protégé-Knowtator project of curated GO annotations was queried via the Knowtator Java API for the NCBITAXON, PRO, and GO IDs and the start and end character positions of the span(s) of the evidence annotations for each GO annotation, and a mapping of the articles to their corresponding GO annotations was dynamically created. Then, for each category of concept annotation, the concept annotations for each article were retrieved. For the analysis of the occurrence of these concepts in the full-text versions of the articles, for each article each of its concept annotations was queried via the Protégé Java API to determine if this concept was an exact match to the concept used in each of the GO annotations associated with the given article, a subclass, a superclass, or none of these. For the analysis of the occurrence of these concepts in only the evidence annotations, for each article each of its concept annotations was first queried for its start and end character positions of its span(s), and these spans were compared to the spans of the evidence annotations of each of the GO annotations associated with the given article; only if the span(s) of the given concept annotation were found to be within the span(s) of any of the evidence annotations of a given associated GO annotation was the concept used in the concept annotation compared to the concept used in the GO annotation. These analyses were done once each for the NCBITAXON and PRO concepts and for each of the three branches of the GO, i.e., biological processes (BP), molecular functions (MF), and cellular components (CC).

3 RESULTS

Table 1 displays statistics for the percentages and fractions of the curated GO annotations for which there is at least one annotated mention of the associated NCBITAXON concept, both in the full-text versions of the articles and only in the evidential sentences identified for the annotations. As shown in the table, there are few articles explicitly mentioning species exactly corresponding to the curated GO annotations: The species of only 7.3% and 0.9% of the curated GO annotations is annotated at least once in the corresponding full-text articles and the evidential sentences, respectively.

![Table 1](https://example.com/table1.png)

<table>
<thead>
<tr>
<th>analysis</th>
<th>exact</th>
<th>superclasses</th>
<th>Mus</th>
</tr>
</thead>
<tbody>
<tr>
<td>full-text articles</td>
<td>7.3% (16/218)</td>
<td>100% (218/218)</td>
<td>99% (215/218)</td>
</tr>
<tr>
<td>evidence only</td>
<td>0.9% (2/222)</td>
<td>47% (102/222)</td>
<td>44% (96/222)</td>
</tr>
</tbody>
</table>

Table 1. Percentages and fractions of curated GO annotations for which there is at least one annotated mention of the associated NCBITAXON concept, both in the full-text articles or only in the evidential sentences. The second, third, and fourth columns respectively show statistics for exact NCBITAXON concept matches, for all superclasses of the exact species concept, and for only the genus Mus (the taxon of all mice).

In addition to an analysis of the occurrence of annotated mentions of concepts exactly corresponding to the curated GO annotations, we have included analysis of all superclasses (i.e., ancestors) of the directly corresponding concepts (e.g., for the species Mus musculus, a mention of the genus Mus, the subfamily Murinae, the family Muridae, the order Rodentia, etc.). We have included analysis of such superclass annotations throughout our study because even though these are not mentions of concepts exactly corresponding to the curated GO annotations, at least some of these may be coreferential mentions of the exact species or may be mentioned within assertions pertaining to the ancestor concepts that also hold true for the directly corresponding concepts; therefore, they are potentially useful as well. As Table 1 shows, there is at least one mention of an ancestor NCBITAXON concept for 100% and 47% of the curated GO annotations in the corresponding full-text articles and in the evidential sentences, respectively. (Though there are concepts more specific than species in the NCBI Taxonomy, e.g., subspecies, strains, there are no such annotated mentions of such subclasses of organisms associated with the GO annotations in these articles.)

The small fractions of articles with annotated mentions of species exactly corresponding to the curated GO annotations
is somewhat deceptive in that mentions of species are very often referred to as higher-level taxa; for example, the most common species of laboratory fruit fly, *Drosophila melanogaster*, is often referred to as “*Drosophila*” (indicating its genus, which contains more than 1,500 species), “fruit fly” (a common name that can refer to this genus), or even “fly” (a common name than can also refer to the order *Diptera*, the higher-level taxon of (true) flies, which contains an estimated 240,000 species). As MGI focuses on compiling data and knowledge for the laboratory mouse, a large majority of the GO annotations examined in this study pertain to the most common species of laboratory mouse, *Mus musculus*, which is analogously commonly referred to as “mouse”.

For all of the concept annotations of the CRAFT Corpus, we consistently sought to annotate mentions with the closest semantic match to the selected text, even in cases in which a more specific class is known from context; we have found that such a strategy avoids a great amount of labor in many cases and reduces error overall. Therefore, mentions of mice (*i.e.*, mice not explicitly mentioned as a specific species) are annotated with the NCBITAXON term *Mus*, the genus of mice, and not with *Mus musculus*, the species colloquially known as the house mouse, even in cases where it is known to be referring to the house mouse. As most mentions of mice in these articles (which are annotated with *Mus*) very likely refer to *Mus musculus*, we also specifically examined the occurrence of *Mus* annotations for GO annotations of *Mus musculus* genes/gene products in our study. Table 1 shows that *Mus* is mentioned at least once in the corresponding full-text articles and evidential sentences of 99% and 44% of the curated GO annotations, respectively.

As for occurrence of PRO concepts, we found that the corresponding PRO concepts of 98% (213/218) and 83% (180/222) of the curated GO annotations are mentioned at least once in the corresponding full-text articles and evidential sentences, respectively. The hierarchical structure of the Protein Ontology is relatively flat, with many protein concepts as children of the ontology’s basic protein concept, and the overwhelming majority of the PRO concept annotations were made using classes from this level. Thus, we did not examine the occurrence of superclasses and subclasses of the corresponding PRO concepts of the GO annotations.

As for the concepts from the three branches of the GO exactly matching the GO concepts used in the curated GO annotations, we found that CC concepts were most often mentioned at least once (with 59% and 55% of the GO CC annotations mentioned in the full-text articles and evidential sentences, respectively), followed by GO MF (with 39% and 27%, respectively) and GO BP (with 33% and 16%, respectively). There are also mentions of concepts more specific than the GO concepts used in the GO annotations; as these are subclasses of the latter, they deductively infer the former: Such subclasses are mentioned at least once in the full-text articles and evidential sentences for 7.4% and 0% of the curated GO BP annotations, respectively; for 45% and 12% of the GO MF annotations, respectively; and for 12% and 4% of the GO CC annotations, respectively. Finally, superclasses of the GO concepts used in GO annotations are mentioned at least once in the full-text articles and evidential sentences for 88% and 43% of the GO BP annotations, respectively; for 82% and 42% of the GO MF annotations, respectively; and for 69% and 45% of the GO CC annotations, respectively. These data for GO concepts are shown in Table 2.

Table 2. Percentages and fractions of curated GO annotations for which there is at least one annotated mention of the associated GO BP, MF, or CC concept, either in the full-text articles or only in the evidential sentences. The second, third, and fourth columns respectively show statistics for the exact GO concepts, for all subclasses of the exact GO concepts, and for all superclasses of the exact GO concepts.

<table>
<thead>
<tr>
<th>ontology</th>
<th>total annotations</th>
<th>average/median annotations per article</th>
<th>min/max annotations per article</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCBITAXON</td>
<td>3,566</td>
<td>105 / 95</td>
<td>19 / 378</td>
</tr>
<tr>
<td>PRO</td>
<td>8,437</td>
<td>248 / 238</td>
<td>61 / 625</td>
</tr>
<tr>
<td>GO BP</td>
<td>8,366</td>
<td>246 / 218</td>
<td>25 / 781</td>
</tr>
<tr>
<td>GO MF</td>
<td>2,106</td>
<td>62 / 50</td>
<td>4 / 235</td>
</tr>
<tr>
<td>GO CC</td>
<td>4,102</td>
<td>121 / 116</td>
<td>10 / 347</td>
</tr>
</tbody>
</table>

Table 3. Total counts of NCBITAXON, PRO, GO BP, GO MF, and GO CC annotations in the 34 journal articles analyzed in this study, with average, median, minimum, and maximum counts of annotations per article.

To provide context of the occurrence of the NCBITAXON, PRO, and GO concepts associated with the curated GO annotations relative to the occurrence of all NCBITAXON, PRO, and GO concepts in these articles, Tables 3 and 4 show counts for all NCBITAXON, PRO, and GO annotations in the journal articles analyzed in this study, along with counts of unique mentions of all of these concepts. In Table 3, it can be seen that the average/median annotation counts for all concepts of these ontologies per article range from 62/50 GO MF annotations per article to 248/238 PRO annotations per article; note, however, the very wide range of these counts in the minimum and maximum annotations per article for all of the ontologies. In Table 4, it can be seen that the average/median counts of unique concepts mentioned at least once per article range from 12/10 unique NCBITAXON concepts mentioned per
article to 44/44 unique GO BP concepts mentioned per article; as with the annotation counts, there is a very wide range of counts of unique concepts mentioned, as seen in the minimum and maximum counts for all of the ontologies. (Corresponding counts for all 67 articles of the 1.0 release of the corpus have been previously published (Bada et al., 2012.).)

<table>
<thead>
<tr>
<th>ontology</th>
<th>total unique concepts</th>
<th>average/median unique concepts per article</th>
<th>min/max unique concepts per article</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCBITAXON</td>
<td>113</td>
<td>12 / 10</td>
<td>3 / 49</td>
</tr>
<tr>
<td>PRO</td>
<td>523</td>
<td>19 / 20</td>
<td>5 / 40</td>
</tr>
<tr>
<td>GO BP</td>
<td>488</td>
<td>44 / 44</td>
<td>11 / 95</td>
</tr>
<tr>
<td>GO MF</td>
<td>187</td>
<td>13 / 12</td>
<td>1 / 26</td>
</tr>
<tr>
<td>GO CC</td>
<td>158</td>
<td>13 / 11</td>
<td>1 / 33</td>
</tr>
</tbody>
</table>

Table 4. Total counts of unique NCBITAXON, PRO, GO BP, GO MF, and GO CC concepts annotated in the 34 journal articles analyzed in this study, with average, median, minimum, and maximum counts of unique concepts annotated per article.

<table>
<thead>
<tr>
<th>analysis</th>
<th>exact</th>
<th>superclasses</th>
<th>Mus</th>
</tr>
</thead>
<tbody>
<tr>
<td>full-text articles</td>
<td>1.1 (1%)</td>
<td>66 (63%)</td>
<td>47 (45%)</td>
</tr>
<tr>
<td>evidence only</td>
<td>0.009 (0.6%)</td>
<td>1.7 (119%)</td>
<td>1.4 (97%)</td>
</tr>
</tbody>
</table>

Table 5. Counts of annotated NCBITAXON concept mentions associated with curated GO annotations, averaged over these GO annotations. These counts are also expressed as percentages relative to the average counts of all NCBITAXON annotations either throughout the entire articles (105, shown in Table 3) or only in the evidential sentences (1.4, data not shown). The second, third, and fourth columns respectively hold data for the exact concepts, for superclasses of the exact species concept, and for the genus Mus (the taxon of all mice).

Efforts at automatic GO annotation that attempt to find relevant GO concepts in text must not only be able to accurately identify concept mentions but also to choose the concept mentions that are relevant for GO annotation of genes/gene products from all of the identified concept mentions. For the PRO, there was found to be an average of 94 annotated mentions of the exact PRO concepts associated with the curated GO annotation in the full-text articles, amounting to 38% of the average total annotated PRO mentions per article of 248 (shown in Table 3); there was found to be an average of only 4.9 annotated mentions of the exactly matching PRO concepts in only the evidential sentences, but amounting to 92% of the average annotated mentions of all PRO concepts in the evidential sentences of 5.3 (data not shown). For the NCBITAXON, there were found to be averages of 1.1 mention of the exact species and 47 mentions of Mus, respectively amounting to 1% and 45% of the average total NCBITAXON annotations per article of 105 (shown in Table 3); the NCBITAXON data are shown in Table 5. Table 6 shows corresponding data for the GO concepts associated with curated GO annotations, all of which were found to have very low mention numbers averaged over the GO annotations.

<table>
<thead>
<tr>
<th>analysis</th>
<th>exact</th>
<th>superclasses</th>
<th>superclasses</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP full-text articles</td>
<td>5 (2%)</td>
<td>0.4 (0.1%)</td>
<td>25 (10%)</td>
</tr>
<tr>
<td>BP evidence only</td>
<td>0.2 (7%)</td>
<td>0.0 (0%)</td>
<td>2 (28%)</td>
</tr>
<tr>
<td>MF full-text articles</td>
<td>3 (5%)</td>
<td>5 (8%)</td>
<td>22 (35%)</td>
</tr>
<tr>
<td>MF evidence only</td>
<td>0.7 (19%)</td>
<td>0.7 (20%)</td>
<td>2 (59%)</td>
</tr>
<tr>
<td>CC full-text articles</td>
<td>8 (7%)</td>
<td>0.4 (0.3%)</td>
<td>10 (8%)</td>
</tr>
<tr>
<td>CC evidence only</td>
<td>2 (39%)</td>
<td>0.06 (1%)</td>
<td>1 (19%)</td>
</tr>
</tbody>
</table>

Table 6. Counts of annotated GO BP, MF, and CC concepts associated with curated GO annotations, averaged over these GO annotations. These counts are also expressed as percentages relative to the average counts of all GO BP, MF, or CC annotations either throughout the entire articles (246, 62, and 121, respectively, shown in Table 3) or only in the evidential sentences (5.4, 3.5, and 5.7 respectively, data not shown). The second, third, and fourth columns respectively hold data for the exact concepts, for subclasses of the exact concepts, and for superclasses of the exact concepts.

4 DISCUSSION

We have employed the 1.0 public version of the CRAFT Corpus to undertake an analysis of the occurrence in the corpus articles of annotations of NCBITAXON, PRO, and GO concepts corresponding to a set of official GO annotations largely curated by the Mouse Genome Informatics group. We have specifically relied on the concept annotations of the corpus, in which every mention of (nearly) every explicitly represented concept of eight prominent biomedical ontologies has been annotated with its corresponding ontological concept according to the CRAFT concept-annotations guidelines.

We have taken advantage of the fact that the articles of the CRAFT Corpus were selected partly based on their serving as evidential sources for curated annotations of genes/gene products with GO and/or MPO classes. With the complete concept annotation of these articles with the aforementioned ontologies, we are provided with an important knowledge resource for biomedical research, au-
omatic methods of GO annotation have become of interest, including a number of text-mining approaches. Since such approaches often look for mentions of relevant concepts in articles, it is of interest to determine the degree of potential of finding the species, gene/gene products, and aspects of biological functionality directly in the text.

In the full-text articles, we have found that the GO BP, MF, and CC concepts exactly matching the GO concepts of these curated GO annotations are mentioned at least once in substantial fractions (33%, 39%, and 59%, respectively) of the annotations’ corresponding articles; this coheres with empirical studies of occurrence of GO terms in biomedical text in which relatively low percentages of GO BP terms and higher percentages of GO CC terms are found due to their respective higher and lower complexities (e.g., McCray et al., 2002). Additionally, concepts that are more specific than (i.e., subclasses of) the GO concepts of the GO annotations are mentioned at least once in corresponding articles in substantially smaller fractions of the BP and CC annotations (7.4% and 12%, respectively) and in a slightly higher fraction of the MF annotations (45%) (though the difference in absolute numbers of annotations is very small).

In addition to the exactly matching concepts, occurrence of such subclasses is of interest because an annotation with a subclass deductively infers an annotation with each of the subclass’s superclasses—including the exactly matching class—according to the GO true-path rule (Camon et al., 2012). Concepts more general than (i.e., superclasses of) of the GO concepts used in the curated GO annotations are mentioned at least once in larger fractions of the annotations’ corresponding articles, which is intuitive since more general classes are more likely to be mentioned than specific ones. Even though these superclass annotations do not exactly correspond to the GO concepts of the curated GO annotations (nor do they deductively infer the exact GO concepts, as do the subclass annotations), they are potentially useful, since at least some of them may be coreferential mentions of the exact GO concepts or may be mentioned within assertions pertaining to the ancestor concepts that also hold true for the exact concepts.

NCBITAXON concepts representing the species of the curated GO annotations are very infrequently mentioned explicitly. However, mice (which in the corpus is annotated to the genus *Mus*) are very frequently mentioned in the articles corresponding to the GO annotations, and we expect such frequent mentions of higher-level common names for other species. These mentions of higher-level taxa are not guaranteed to refer to the most common laboratory species, of course; for example, there are species of mice other than *Mus musculus* that are used in laboratory experiments. However, such mentions could be leveraged as very reasonable abductive inferences. PRO concepts are also mentioned in very high fractions of the articles corresponding to the GO annotations, and the PRO concepts associated with the curated GO annotations constitute substantial fractions of the total mentions of all PRO concepts in the articles.

An interesting trend that can be seen for all of the ontologies concern the counts of annotated mentions of the concepts corresponding to the curated GO annotations, averaged over these GO annotations, in the full-text articles versus only in the evidential sentences. For all of the ontologies, the absolute counts of the associated ontological concepts in only the evidential sentences as expected are much lower than those in the full-text articles. At the same time, the annotated ontological concepts associated with the curated GO annotations seem to be overrepresented in the evidential sentences in that the ratios of counts of the ontological concepts associated with the GO annotations to the counts of all of the ontological concepts are higher within the evidential sentences than throughout the full-text articles. However, the counts within the evidential sentences are very low, this appearance of overrepresentation should be treated cautiously. Identification of these types of sentences seems a difficult task, as many of the evidential sentences annotated by the MGI curator describe low-level data (e.g., phenotypes of animals, biochemical assays), and the functionalities of the genes/gene products are often not straightforwardly mentioned but are inferable from the experimental results by domain experts. It remains to be seen whether a system could reliably identify such passages relevant for GO annotations, and if so, whether it might be beneficial to do so for the task of automatic GO annotation.

A caveat that should be stated is that not all of these annotated concept mentions will be relevant to the curated GO annotations associated with the articles. Therefore, we regard the statistics presented as upper bounds toward the inference of the GO annotations from the direct mention of these component concepts in text. A rigorous gold-standard investigation of which of these mentions are relevant to GO annotations extracted from the text would require an additional layer of annotation in which these concept annotations are appropriately relationally linked with each other. These annotated assertions, which would include links among mentions of species, genes/gene products, and aspects of biological functionality, could then be analogously analyzed to identify GO-annotation information. We do plan on performing such assertional annotation in the future. However, as was the case with the concept annotation of the CRAFT Corpus, this will almost certainly be a labor-intensive, multiyear effort. Furthermore, even annotation of direct assertions may not be sufficient, as GO-annotation information may require the use of coreferential information as well as other types of inference.
5 CONCLUSIONS

We have evaluated the potential for programmatic extraction of GO annotations of genes/gene products by examining the occurrence of direct mentions of NCBI-TAXON, PRO, and GO concepts corresponding to official curated GO annotations in a gold-standard manually annotated corpus of full-text articles partly selected as the basis for such GO annotations. GO concepts exactly matching the GO concepts of these GO annotations are mentioned in substantial fractions of the annotations’ corresponding articles at least once; however, the mentions of these GO concepts constitute very low percentages of the mentions of all GO concepts in these articles. Nearly all PRO concepts corresponding to GO annotations are mentioned at least once, and these PRO mentions constitute a substantial fraction of the mentions of all PRO concepts in these articles. This suggests that automatic textual recognition of PRO and NCBI-TAXON concepts relevant to GO annotations appear quite tractable relative to textual recognition of associated GO concepts. For all of the ontologies, counts of annotated concepts corresponding to the curated GO annotations in only the strongly evidential sentences are comparatively very low, amounting to several mentions or fewer per article. However, for most of the ontologies, concepts corresponding to the curated GO annotations appear overrepresented, though this must be viewed cautiously given that this is based on very low counts. Thus, it remains to be further examined whether this overrepresentation overrides the very low mention frequency and thus whether it would be beneficial for automatic GO-annotation systems to focus on these evidential sentences.

ACKNOWLEDGEMENTS

We thank William Baumgartner for programmatically importing GO-annotation information into the Protégé-Knowtator project of curated GO annotations. We also gratefully acknowledge support from NIH 5R01 LM008111, 2R01 LM009254, and 5T15 LM009451

REFERENCES


Comprehensive Benchmark of Gene Ontology Concept Recognition tools
Christoph Jacob, Philippe Thomas, Ulf Leser (thomas,leser}@informatik.hu-berlin.de)
Knowledge Management in Bioinformatics, Institute for Computer Science, Humboldt-Universität zu Berlin, Unter den Linden 6, 10099 Berlin, Germany

Abstract
The Gene Ontology has evolved as the de facto standard for describing gene function in the biomedical domain. Information about gene function can be often found in written articles. In this work we evaluate three tools capable of recognizing Gene Ontology concepts in text on an automatically generated gold standard of 88,573 articles. The analysis reveals differences in concept recognition for these tools. An ensemble based approach is implemented to exploit idiosyncrasies between different tools and substantially improves recognition quality.

Introduction
In the biomedical domain, the Gene Ontology (GO) has evolved as the de facto standard for providing a controlled and structured vocabulary of terms describing attributes of genes. The Gene Ontology is used for the process of gene function annotation. GO annotation involves two tasks: identifying genes and gene functions in free text and associating both. Gene function annotations are collected and stored by the Gene Ontology Annotation (GOA) project (Camon et al., 2004).

In the past years, the amount of available biomedical literature has been growing at an estimated double-exponential pace (Hunter and Cohen 2006), rendering manual curation of all relevant publications as too time consuming. Therefore, concept recognition tools like mgrep (Dai et al., 2008) and MetaMap (Aronson, 2001) have been proposed. Shah et al. (2009) compared mgrep and MetaMap in terms of runtime and evaluated recognition of terms from the biological process GO-branch on a set of 2,827 PubMed abstracts. Both tools performed equally well in terms of precision. Due to the lack of a gold standard, recall has not been evaluated.

The first evaluation of different GO recognition tools has been performed in the first BioCreative competition (Blaschke et al., 2005). Evaluation proved to be difficult, as curation of predictions was too time consuming. The recently published CRAFT corpus (Bada et al., 2012) provides annotations for nine different concept types for 67 full text articles.

However, it is currently unclear what performance can be expected when evaluating concept recognition tools in a real world scenario. Some terms might not be relevant for GO-curators as corpora are annotated with respect to specific guidelines. To overcome this, we provide a comprehensive benchmark of three different concept recognition tools in the context of GO terms. In difference to previous approaches, the tools are evaluated using a large automatically generated corpus, which we believe is a valuable complementary approach to evaluating on manually annotated corpora, since it better resembles the real annotation process. Furthermore, this corpus reflects the curation approach more closely, as only relevant GO terms are annotated (e.g. the GO term cell may appear literally in almost every PubMed article, but is rarely used for annotation purposes).
Materials and Methods

Gold Standard(s)
In this work, we derive a gold standard corpus by exploiting existing annotations from the GOA database. The GOA project collects and integrates annotations from various sources and thus provides the most comprehensive collection of GO-annotations. GOA entries consist of three elements. The gene of interest, the GO term, and the supporting PubMed article. In this work we simplify this data to distinct binary tuples consisting of GO term and PubMed-ID. This allows us to evaluate GO concept recognition independent of errors in gene name recognition and subsequent relationship extraction. This strategy allows us to generate a corpus which is about two orders of magnitude larger than those used in previous works.

MetaMap
MetaMap is a general purpose tool for concept recognition. It currently recognizes concepts contained in the Unified Medical Language System (UMLS) Metathesaurus. For this work, we used a local version of MetaMap 2010 using the UMLS2010AB dataset. We selected parameters which seemed to be appropriate for the task of GO term recognition in order to create a realistic setting environment. In particular, we used the strict matching model and activated word sense disambiguation.

mgrep
The other tool considered in our work is mgrep. The basic idea of mggrep is to search dictionary terms in a supplied text passage. It is left to the user to define a proper dictionary which incorporates synonyms and lexical variations for each concept. This means, that the quality of the results is primarily determined by the quality of the dictionary – mggrep only ensures that the input text is searched in an efficient manner by implementing a radix tree based string search algorithm.

The current workflow for generating a basic dictionary as suggested by the authors consists of the following steps: GO terms and synonyms are extracted from the respective OBO file. Subsequently, lexical variations of the extracted terms are built using NCBI’s lvg tool. This allows for the detection of small lexical variations.

t4rgot
We propose an additional method which we further refer to as “tool for recognition of GO terms” (t4rgot). In contrast to classical approaches to the problem of term searching, which are primarily dictionary based, the functionality of t4rgot is inspired by information retrieval techniques.

A schematic representation of t4rgot is shown in Figure 1. The tool consists of 2 components: the Indexer and the Recognizer. As a preprocessing step, the Indexer builds a bag of words (BOW) representation for each GO-term separately. Each bag contains all terms associated with one GO-term. The words in the bag are not stored in any particular order. Thus, positional information is given up for the purpose of efficiently dealing with variations in word order during the search. To alleviate this issue, bigrams are added to the BoWs and a score is assigned to each word/bigram in a bag. In our setting, we score bigrams 10 times as high as single words. Finally, the Indexer stores all BoWs in an index (2) which can later be processed and used by the Recognizer (3).

The Recognizer searches for GO terms in a provided input text (3). The search features the same preprocessing steps used by the Indexer. For each sentence, a set of GO-candidates is derived by matching the BOWs with words from the sentence. These candidates are then scored using the cosine similarity measure to determine which GO terms best describe the contents of the sentence. Depending on the use case, a variable number of highest scoring terms
for each sentence or article may be returned to the user. In addition, a threshold based on the determined score may be introduced to further filter the results (4).

Results
Gold Standard corpus
The generated gold standard corpus contains 248,847 distinct GO-ID, PubMed-ID tuples collected from 88,573 PubMed articles. For 3,683 articles the full text version was available in the PubMed-Central open access subset. The remaining 84,890 articles could only be retrieved as abstracts. Approximately 18,500 of all 33,311 GO terms (55.5%) can be found in current GOA-annotations and only 5,744 GO terms are contained in the available full texts. A ratio of tuples per article of 4.12 per full text article and 2.75 per abstract can be observed. These differences can be partially explained by two observations. First, for both sets a strong positive correlation between publication year and number of annotations is observed. Second, articles in PMC tend to be more recent (1992) than articles where no full-text is found (1979). Detailed information is shown in Table 1.

Table 1: Contents of the gold standard

<table>
<thead>
<tr>
<th></th>
<th>PubMed articles</th>
<th>GO term count</th>
<th>Tuple count</th>
<th>Tuples per article</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>90,700</td>
<td>18,523</td>
<td>252,578</td>
<td>2.78</td>
</tr>
<tr>
<td>Evaluation</td>
<td>88,573</td>
<td>18,500</td>
<td>248,847</td>
<td>2.81</td>
</tr>
<tr>
<td>Full text</td>
<td>3,683</td>
<td>5,744</td>
<td>15,159</td>
<td>4.12</td>
</tr>
<tr>
<td>Abstracts</td>
<td>84,890</td>
<td>18,105</td>
<td>233,688</td>
<td>2.75</td>
</tr>
</tbody>
</table>

Index size and runtime
We first investigated time and hard drive requirements to set up the three tools. The results are shown in Table 2. In case of MetaMap, determining the size of the index is not trivial. MetaMap uses a proprietary dataset derived from the UMLS 2010AB release and GO constitutes only a portion of this dataset. The entire UMLS dataset is about 3.6 gigabytes large. The dictionary used by mgrep is by far the largest compared to the other two tools. It occupies about 183 gigabytes of disk space which is remarkably high considering the fact that all of this data is derived from only 33,311 terms of the Gene Ontology. The smallest index of all three tools is used by t4rgot. Only about 20 megabytes of disk space is needed to store the BOW information. The main reason for the dictionary size for mgrep is, that lexical word variations and variations in word order are stored separately. In difference, the BOW approach of t4rgot saves every token exactly once and implicitly handles variations in word order.
Runtime was assessed on a 2.25 GHz machine with 60 gigabytes of main memory. The longest execution time can be observed for MetaMap with 2 days for all 84,890 articles on a single core. In contrast, execution time for mgrep and t4rgot is much lower for both abstracts and full texts. T4rgot is capable of recognizing GO terms in all abstracts from PubMed (21 million) within 18 days on a single core.

**Table 2: Index sizes and runtimes of t4rgot, mgrep and MetaMap**

<table>
<thead>
<tr>
<th></th>
<th>t4rgot</th>
<th>mgrep</th>
<th>MetaMap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index size</td>
<td>20.34 MB</td>
<td>187,874.84 MB</td>
<td>3,665.79 MB</td>
</tr>
<tr>
<td>Abstract</td>
<td>41 min</td>
<td>104 min</td>
<td>2,979 min</td>
</tr>
<tr>
<td>Full text</td>
<td>48 min</td>
<td>109 min</td>
<td>4,308 min</td>
</tr>
</tbody>
</table>

**Precision, recall and F measure**

Concept recognition results for all three tools on abstracts are shown in Table 3.

**Table 3: Comparison of results from t4rgot, mgrep and MetaMap on abstracts**

<table>
<thead>
<tr>
<th></th>
<th>t4rgot</th>
<th>mgrep</th>
<th>MetaMap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>843,959</td>
<td>843,469</td>
<td>663,612</td>
</tr>
<tr>
<td>True Pos.</td>
<td>36,456</td>
<td>40,655</td>
<td>37,634</td>
</tr>
<tr>
<td>False Pos.</td>
<td>807,503</td>
<td>802,814</td>
<td>625,978</td>
</tr>
<tr>
<td>False Neg.</td>
<td>197,232</td>
<td>193,033</td>
<td>193,402</td>
</tr>
<tr>
<td>Precision</td>
<td>4.32%</td>
<td>4.82%</td>
<td>5.67%</td>
</tr>
<tr>
<td>Recall</td>
<td>15.60%</td>
<td>17.40%</td>
<td>16.29%</td>
</tr>
<tr>
<td>F measure</td>
<td>6.77%</td>
<td>7.55%</td>
<td>8.41%</td>
</tr>
</tbody>
</table>

The highest recall on abstracts is achieved by mgrep. The tool correctly identifies more than 40,000 GO terms, leading to a recall of 17.40%. Of the 2.75 tuples which are on average contained in each abstract, mgrep is capable of correctly finding 0.48 tuples. MetaMap finds fewer true positives but also returns overall fewer terms, achieving the highest precision of 5.67% and subsequently the highest F-measure with 8.4%. T4rgot is not able to achieve the results of the other two tools for abstracts, as precision, recall and F measure fall about 2 percentage points (pp) short of the respective best result.

Table 4 shows the results for the three different tools when applied to full-text articles.

**Table 4: Comparison of results from t4rgot, mgrep and MetaMap on full texts**

<table>
<thead>
<tr>
<th></th>
<th>t4rgot</th>
<th>mgrep</th>
<th>MetaMap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>220,980</td>
<td>238,916</td>
<td>213,242</td>
</tr>
<tr>
<td>True Pos.</td>
<td>4,688</td>
<td>4,551</td>
<td>4,233</td>
</tr>
<tr>
<td>False Pos.</td>
<td>216,292</td>
<td>234,365</td>
<td>209,009</td>
</tr>
<tr>
<td>False Neg.</td>
<td>10,471</td>
<td>10,608</td>
<td>10,713</td>
</tr>
<tr>
<td>Precision</td>
<td>2.12%</td>
<td>1.90%</td>
<td>1.99%</td>
</tr>
<tr>
<td>Recall</td>
<td>30.93%</td>
<td>30.02%</td>
<td>28.32%</td>
</tr>
<tr>
<td>F measure</td>
<td>3.97%</td>
<td>3.58%</td>
<td>3.71%</td>
</tr>
</tbody>
</table>

Different observations can be made for full texts in comparison to abstracts. Here, t4rgot leads all tools in all three measures and achieves the highest rates of precision (2.12%), recall (30.93%) and F measure (3.97%). The tool extracts 4,688 correct GO terms. The lowest precision of 1.90% is achieved by mgrep. MetaMap returns the fewest amounts of tuples and true positives which results in the lowest recall of 28.32%.
It is also remarkable that the observed recall rates for full texts are generally twice as high as for abstracts. This is probably caused by the fact that abstracts only constitute a small portion of the entire article. During the process of manual annotation, curators typically use the entire full text to derive appropriate GO terms. It can be suspected that in many cases these GO terms could not be extracted from the abstract because the text passage containing the term was missing.

Finally, the small variation in the number of true positives returned by all three tools raises the question whether all tools find approximately the same terms or if the tools find entirely different terms. This question is further investigated in the following subsection, by analyzing the true positives returned by all three tools within full text articles in more detail.

We first analyzed the distribution of distinctly found GO terms in comparison to their character length. The results are depicted in Figure 2. GO terms from the gold standard are displayed by the dark grey curve. Ideally, a tool should follow this curve. All three tools correctly identify a large portion of short terms consisting of fewer than 20 characters. The curves of MetaMap and mgrep look very similar and steadily decline for GO terms with more than 28 characters. In contrast, t4rgot seems to better correlate with the profile of the gold standard. Indeed, we observe a significant correlation between gold standard and t4rgot predictions (Kendall’s tau = 0.75; p-value < 0.01). For the other two tools we observe significant correlations of 0.55 in comparison to the gold standard.

T4rgot is apparently able to identify more of the longer GO terms compared to MetaMap and mgrep which is an indication that the approach of t4rgot works better for identifying longer GO terms. This suspicion is also confirmed by comparing the average lengths of all GO terms contained in all true positives for each tool. While MetaMap and mgrep extract terms with an average length of 18 characters, the terms found by t4rgot contain on average 24 characters which is much closer to the average length of a 29 characters in the gold standard. An analysis of missed GO terms revealed an average in length of 34 characters which supports the observation that longer GO terms are harder to identify.

Figure 2: Length distribution of distinct GO terms of the gold standard and distinct GO terms contained in true positives returned by each tool for full texts.
The average depth of distinct GO terms returned by t4got is 4.4 and 4.0 for mgrep and MetaMap. This difference is significant according to the Wilcoxon signed-rank test. In comparison, the average node depth in the gold standard is 5.4. For all three tools we observe that terms from the sub-ontology cellular component achieve with 4.9% the highest F1 score, followed by biological process with 2.9% and molecular function with 2.3%. Similar results have been reported for the BioCreative competition where cellular component terms had the highest fraction of correct predictions.

**Ensemble systems**

The previous analysis of true positive predictions shows that t4got produces different results in comparison to MetaMap and mgrep. This suggests that a hybrid system, consisting of two complementing systems, might lead to superior performance. To test this hypothesis we generated a hybrid system by combining t4got with mgrep. We evaluated two different combinations: first, to increase recall, we built the union between predictions of the two tools. Second, to increase precision, we built the intersection. Experiments are restricted to t4got and mgrep, as ensembles using MetaMap produce highly similar results (data not shown).

Results of the two different ensembles are shown in Table 5.

**Table 5: Results for the ensemble system of t4got and mgrep on abstracts and full texts**

<table>
<thead>
<tr>
<th></th>
<th>Union t4got &amp; mgrep</th>
<th>Intersection t4got &amp; mgrep</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>abstract</td>
<td>full text</td>
</tr>
<tr>
<td><strong>Count</strong></td>
<td>1,546,285</td>
<td>411,841</td>
</tr>
<tr>
<td><strong>True Pos.</strong></td>
<td>58,733</td>
<td>6,431</td>
</tr>
<tr>
<td><strong>False Pos.</strong></td>
<td>1,487,552</td>
<td>405,410</td>
</tr>
<tr>
<td><strong>False Neg.</strong></td>
<td>174,955</td>
<td>8,728</td>
</tr>
<tr>
<td><strong>Precision</strong></td>
<td>3.80%</td>
<td>1.56%</td>
</tr>
<tr>
<td><strong>Recall</strong></td>
<td>25.13%</td>
<td>42.42%</td>
</tr>
<tr>
<td><strong>F measure</strong></td>
<td>6.60%</td>
<td>3.01%</td>
</tr>
</tbody>
</table>

As expected, the union between t4got and mgrep achieves very high recall rates of 25.13% for abstracts and 42.42% for full texts. In comparison to the individual results, recall increases up to 12 pp. The ensemble system using the intersection between the two tools, achieves superior precision rates of 13.0% and 5.8% for abstracts and full texts respectively. While this system may not return a large amount of GO terms for each article, it may be more suitable for the purpose of manual annotation for two reasons. First, the system processes a full text article within two or three seconds and could thus be used in a real-time application environment. Second, curators usually use only one or two GO terms during the process of annotation. If the system were to return high precision results, the curator could quickly choose between 10 to 15 proposed GO terms for each article without having to go through a long list of possible candidates.

In conclusion, the proposed ensemble system may be of relevance for the current annotation efforts undertaken by the GOA project. While the individual systems each suffer from various problems, their combination is able to return results of higher quality. The results obtained from t4got present a valuable addition to the results of mgrep and can be used to construct a system with either high recall or high precision, depending on the use case. The intersection of the results of both tools leads to an increase in precision of 8.2 percentage points for abstracts and 3.72 percentage points for full texts compared to the individual systems. In contrast, the union of both results is able to increase recall by 7.73 percentage points for abstracts and 11.49 percentage points for full texts compared to the highest recall obtained by either t4got or mgrep.
Discussion

A drawback of the automatically generated gold standard is that not all GO terms have been annotated. In fact, only a very small fraction of the contained terms may have been used by curators. The reason is that curators follow specific guidelines when selecting terms relevant for curation. Therefore, unspecific top level terms such as cell are very rarely used as curators tend to annotate the most specific term. However, this is also one of the biggest advantages as this corpus reflects actual annotation behavior. We assume that all tools would achieve better results on a corpus manually annotated for all GO concepts. However, this corpus would not represent curator’s behavior, who are only interested in informative terms. We therefore believe that such a corpus is a realistic scenario to evaluate real world capabilities of GO concept recognition tools.

In summary, all tools achieve very low rates of precision. For abstracts, precision ranges between 4% and 6% while for full texts all tools achieve a precision of about 2%. Considering the results achieved by participants of the Bio-Creative contest (Blaschke et al., 2005) (20% to 80% precision) or results from previous comparisons of mgrep and MetaMap (Shah et al., 2009) (above 70% precision), the figures obtained here clearly contradict these results. However, evaluation in these cases was conducted manually, achieving almost complete coverage of all GO terms contained in the used corpora of text.

Conclusion

In this publication we present the first large scale evaluation of concept recognition tools. The collected large-scale gold standard reflects the curators’ needs better than a small manually annotated corpus. We also provide a detailed evaluation of three different tools and conclude that the combination of t4rgot and mgrep provide better results than a single system.

Acknowledgements

The authors would like to thank Fan Meng and Manhong Dai, the creators of mgrep from the Molecular and Neuroscience Institute at the University of Michigan for their support for the evaluation of mgrep and development of t4rgot. We also would like to thank the anonymous reviewers and the TAIR database curators for their helpful comments on the manuscript.

References


BioLINK SIG: Roles for text mining in biomedical knowledge discovery and translational medicine
https://sites.google.com/site/biolinksig2013program

Short Papers

Neji: a tool for heterogeneous biomedical concept identification.
David Campos, Sérgio Matos and José Luís Oliveira.
Pages 28-31

Natural language query processing for SPARQL generation: A prototype system for SNOMED-CT.
Jin-Dong Kim and K. Bretonnel Cohen.
Pages 32-38

Towards automatic large-scale curation of genomic variation: improving coverage based on supplementary material.
Antonio Jimeno Yepes and Karin Verspoor.
Pages 39-43

Towards an Integrated Compound to Compound Relatedness Measure.
Maria Kissa, Michael Schroeder and George Tsatsaronis.
Pages 44-47

Case studies in making sense of clinical text.
Pages 48-51

HistoNer: Histone modification extraction from text.
Philippe Thomas and Ulf Leser.
Pages 52-55
**ABSTRACT**

Motivation: Concept identification is an essential task in biomedical information extraction, presenting several complex and unsolved challenges. Current solutions are typically performed in an ad-hoc manner or optimized for specific biomedical concepts. Thus, the availability of general and modular solutions is scarce.

Results: This article presents Neji, an open source tool for biomedical concept recognition focused on four key characteristics: modularity, high-performance, speed, and usability. It integrates features for biomedical natural language processing, from sentence splitting to chunking and dependency parsing, and supports the most popular input and output formats. Concept recognition and normalization are provided through dictionary matching and machine learning, and the resulting annotations are stored in an innovative concept tree implementation. Neji was evaluated against the CRAFT corpus, achieving high performance F-measure results: species (95%), cell (92%), cellular component (83%), gene and protein (76%), chemical (65%), biological processes and molecular functions (63%). It also provides fast and multi-threaded data processing, annotating up to 1200 sentences/second when using dictionary-based concept identification. Considering the provided features, underlying characteristics and current state-of-the-art methods, Neji constitutes an important contribution to the biomedical community, streamlining the development of complex concept recognition solutions.

Availability and Implementation: Neji is implemented in Java and is available at [http://bioinformatics.ua.pt/neji](http://bioinformatics.ua.pt/neji). Contact: david.campos@ua.pt

1 INTRODUCTION

A growing amount of biomedical data is continuously being produced, resulting largely from the widespread application of high-throughput techniques, such as gene and protein analysis. This growth is accompanied by a corresponding increase of textual information, in the form of articles, books and technical reports. Managing these large amounts of information and knowledge is rapidly becoming a very difficult task, especially when dealing with unstructured information in natural language texts. This has naturally led to the application of text mining (TM) systems to aid in the creation and curation of knowledge bases. An initial and crucial step for this is Named Entity Recognition (NER), aimed at identifying chunks of text that refer to specific entities of interest. However, the identification of such mentions is hindered by the lack of naming standards and the specific characteristics of biomedical entity names (Zhou et al., 2004). Thanks to challenges such as BioCreative (Smith et al., 2008; Morgan et al., 2008; Lu et al., 2011) and JNLPBA (Kim et al., 2004), dozens of new solutions emerged for NER (e.g. Campos et al., 2013) and for normalization (Wermter et al., 2009). However, the resources provided by those challenges are often too specific and focused on the recognition of particular entity types (e.g., gene and protein), generating tailored solutions that provide high performance results on tested corpora. There are also solutions focused on providing annotation of heterogeneous biomedical concepts. For instance, Whatizit (Rebbolz-Schuhmann et al., 2008) and Cocol provide annotations of species, genes and proteins, and disorders, among others concepts. However, since they are provided as web-services, batch processing and application configurations are limited. MetaMap (Aronson, 2001) also provides annotation of heterogeneous concepts, using the Unified Medical Language System (UMLS) Metathesaurus (Bodenreider, 2004) and partial matching for extracting candidate strings with respective scores for concept names. Considering the current tools for the biomedical domain, we believe that there is a lack of solutions that allow batch processing of heterogeneous biomedical concepts, providing the complete set of recognized concepts and an easy to use integrated ecosystem. This document presents Neji, an open source tool optimized for heterogeneous biomedical concept recognition, supporting both machine learning and dictionary-based approaches, and combining the recognized concepts in a structured concept tree.

2 METHODS

2.1 Implementation

The core component of Neji is the processing pipeline, which allows users to submit various modules for execution following a FIFO (First In, First Out) strategy. A pipeline is a list of modules that are executed sequentially, considering specific goals and target chunks of text. Fig. 1 illustrates the idea of this modular and flexible architecture. Each module is implemented as a custom Deterministic Finite Automaton (DFA), with specific matching rules and actions. The hierarchical text processing features of Monq.jfa are used to support the pipeline infrastructure and module execution.

---

1To whom correspondence should be addressed.

2[http://monq.jfa.berlios.de](http://monq.jfa.berlios.de)
it is necessary variants of names can be generated and provided in the dictionary. Even so, solution, we decided to use case insensitive exact matching. Orthographic matching, and considering that we are dealing with a general biomedical a large amount of false positives may be generated using a modified version (Campos et al., 2013) of an existing tool, which provides a comprehensive set of features, serving as a good starting point to develop solutions for the biomedical domain. To complement G1imi, establishing a relation between the entity mentions and unique database identifiers, we developed a simple and general normalization algorithm based on prioritized dictionaries. Following this algorithm, if an identifier is found in the first dictionary, the match is complete and the algorithm finishes. If no match is found in the first dictionary, the second one is used to find a match, and so on. In the end, if no matches are found in the provided dictionaries, the annotation is discarded by default. This configuration works well if the first dictionary is a list of preferred names, and the remaining contain synonyms for each identifier. Moreover, it also provides flexibility to users, which only have to generate the various orthographic variants and prioritize them in the dictionaries. Regarding the matching approach, if a partial match of the annotation is found in the dictionary, it is accepted as a valid identifier for the complete chunk of text. For instance, if “BCRA1 gene” is recognized as an entity mention and only “BRCA1” is present in the dictionary, its identifier is associated with the annotation.

2.1.4 Data structure Neji provides a flexible and complete data structure to store the generated information, providing easy and fast access to obtained sentences, tokens, concepts and natural language processing output. Since nested and intersected annotations are common in the biomedical domain, it is important to integrate a data structure to support such characteristics in the best and most automated way as possible. In Neji, this is achieved through a tree of annotations, presenting various advantages over typical approaches (e.g., list of annotations), such as the automatic maintenance of structured concept annotations, and easy identification of ambiguity problems. Additionally, each concept may have multiple identifiers associated, where each identifier contains information regarding its source, unique identifier, semantic group and semantic subgroup.

2.1.5 Input formats Both XML and raw text are supported. XML format allows specifying the tags of interest. For instance, considering the Pubmed XML format, if only titles and abstracts have to be processed, only the content of the tags “ArticleTitle” and “AbstractText” are of interest. On the other hand, the raw format considers that all the input text is of interest to be processed.

2.1.6 Natural language processing The sentence splitting module uses the model included in the Lingpipe library, which was trained on biomedical corpora and presents high-performance results (Verspoor et al., 2012). Natural Language Processing (NLP) tasks are performed using GDep (Sagae, 2007), a dependency parser for the biomedical domain built on top of the GENIA tagger, which performs tokenization, lemmatization, part-of-speech (POS) tagging, chunking and NER. Since we are not interested in the named entities provided by GENIA tagger, we removed that feature and its dependencies. Moreover, we adapted the tokenizer behavior in order to make it more consistent, obviously respecting all task dependencies and resources (tokenization < POS < lemmatization < chunking < dependency parsing).

2.1.7 Dictionary matching Dictionary matching is offered using a modified version (Girner et al., 2010) of the dk.bricks.automaton library, a DFA implementation for exact and approximate matching. Since a large amount of false positives may be generated when using approximate matching, and considering that we are dealing with a general biomedical solution, we decided to use case insensitive exact matching. Orthographic variants of names can be generated and provided in the dictionary. Even so, it is necessary to pay special attention to terms that are common English words. Thus, a list of non-informative words for the biomedical domain (Kang et al., 2011) is ignored during the matching process. Similarly, tokens with less than three characters are also discarded.

2.1.8 Abbreviation resolution Neji also integrates abbreviation resolution, by adapting a simple but effective abbreviation definition recognizer (Schwartz and Hearst, 2003), which is based on a set of pattern-matching rules to identify abbreviations and their full forms. In this way, we are able to extract both short and long forms of each abbreviation in text. If one of the forms is already provided as a concept, the other one is added as a new concept with the identifiers of the existing one. Additionally, any further occurrences of that entity are also automatically annotated.

2.2 Usage Neji is provided as a simple but powerful Command Line Interface (CLI) tool, which provides a complete set of features: 1) Annotate using dictionaries and/or machine-learning models with respective normalization dictionaries; 2) Various input and output formats. When the XML input format is used, the XML tags should be indicated; 3) Parsing level customization. By default, Neji automatically finds the appropriate parsing level considering the ML model characteristics; 4) Number of threads customization; 5) Parallel processing. On top of the previously described features, Neji also supports multi-threading processing, automatically duplicating the required resources when necessary. This allows annotating multiple documents at the same time, significantly dropping processing times.
Wildcard input filter to properly indicate the files to process; and 6) Support for compressed and uncompressed files. Such features allow annotating a corpus using a simple bash command, such as:

```
./neji.sh -i input/ -f XML -o output/ -of XML -x AbstractText,ArticleTitle -d resources/dictionaries/ -m resources/models/ -c < 6
```

In this example, six threads are used to annotate the compressed XML documents in the input folder with the specified dictionaries and machine-learning models, providing the resulting XML documents to the output folder. Note that only the text inside the specified tags is annotated.

### 3 RESULTS

We evaluated Neji in terms of the quality of the provided concept annotations and the required processing time, given a specific configuration of dictionaries and ML models selected according to the corpus used in the evaluation.

#### 3.1 Corpus

Our analysis was centered on the CRAFT corpus (Bada et al., 2012), one of the largest publicly available gold standard corpora, focused on multiple biomedical concept types with heterogeneous characteristics. The initial release contains a set of 67 full-text articles (more than 21 thousand sentences) manually annotated with concepts from nine biomedical ontologies and terminological resources: Chemical Entities of Biological Interest (ChEBI); Cell Ontology; Entrez Gene; Gene Ontology (biological process, cellular component, and molecular function); NCBI Taxonomy; Protein Ontology and Sequence Ontology. Overall, it contains almost 100 thousand concept annotations.

#### 3.2 Resources

ML models and dictionaries were collected to recognize the biomedical concepts in the CRAFT corpus. Gene and protein names recognition was performed through a ML model trained on GENETAG using a complete and complex set of features, namely lemmas, POS, chunking, orthographic, local context (windows) and morphological features. LexEBI (Thompson et al., 2011), which contains a filtered version of BioThesaurus (Liu et al., 2006), was used to perform normalization. Two different dictionaries were created: the first with preferred names and the second with synonyms for each identifier. For each dictionary a set of orthographic and semantic variants was generated using the Lexical Variants Generation (LVG) tool (Bodenreider, 2004). Chemical recognition was based on a dictionary compiled from the ChEBI database of molecular entities (Degtyarenko et al., 2008). Regarding species, the dictionary provided by LINNAEUS (Gerner et al., 2010) was extended with NCBI Taxonomy entries assigned to taxonomical ranks above “species” and with synonyms from the UMLS Metathesaurus. Cell names were compiled from the “Cell” and “Cell Component” semantic types in the UMLS Metathesaurus. Finally, cellular components, biological processes and molecular functions were obtained from the corresponding sub-ontologies of the Gene Ontology (GO) (Ashburner et al., 2000), and expanded with synonyms from UMLS and with concepts from the semantic types “Physiologic Function”, “Organism Function”, “Organ or Tissue function”, “Cell function”, “Molecular function” and “Genetic function”. As a filtering step, we rejected names with one or two characters, names starting with a word from a strict list of stopwords (e.g. “the cell”), and also any single word name if that word was included in the list of most frequent words in MEDLINE. Some relevant terms that occur very frequently in MEDLINE, such as GO terms (e.g. “expression”, “transcription”) and species names (e.g. “human”, “Saccharomyces”), were removed from this list to allow identifying them in texts. In the end, our dictionaries contain almost 1 million concept identifiers with 7 million name variants.

#### 3.3 Concept annotation

Results previously presented using CRAFT are focused on text mentions, not evaluating the assigned identifiers. We follow the same approach, considering four matching strategies: exact (annotation is accepted if both left and right sides match); left (annotation is accepted if the left side matches); right (annotation is accepted if the right side matches); and overlap (annotation is accepted if there is any kind of match: exact, nested or intersected). Such matching strategies allow a better understanding of annotation quality, since a non-exact matching does not mean that the correct concept was not recognized. The common evaluation metrics of precision, recall and F-measure are used to analyze the results.

Considering the databases and ontologies used in the annotation of CRAFT, we defined six concept classes: species, cell, cellular component, chemical, gene and protein, and biological processes and molecular functions. Biological processes and molecular functions are grouped into a single class, since annotations are provided in a single file. Moreover, gene and protein annotations are evaluated against Entrez Gene. The comparison was performed against BANNER, the best performing system on (Von Laer et al., 2008), and Cocoa and Whatizit web services, with the provided classes precisely mapped to the corresponding CRAFT concept types.

Fig. 2 presents the results achieved by Neji, Whatizit, Cocoa and BANNER on the CRAFT corpus, considering the various matching strategies. Overall, Neji presents the best results, with significant improvements on various concept types, namely on concepts associated with GO (cellular component, biological process and molecular function), chemical and gene/protein. In more detail, we can see that Neji is the solution that presents overall best recall results without loss in precision. Neji obtained state-of-the-art results on the recognition of species and cell concepts, with overlap F-measure results of 94.7% and 91.5%, respectively. It achieved an F-measure of 83.2% on overlap matching in the recognition of cellular component names, which is significantly better than Cocoa and Whatizit. Regarding gene and protein recognition, Neji ML model with normalization presents better results than Cocoa, BANNER and Whatizit on left and overlap matching. Its performance drop on exact and right matching appears to be a consequence of the different annotation guidelines in CRAFT and GENETAG, which was used to train Neji’s ML model. Finally, the results achieved on chemical and biological processes and molecular functions are considerably better than Cocoa and Whatizit.
ACKNOWLEDGEMENTS

Funding: This research work was funded by FEDER through the COMPETE programme and by national funds through FCT - “Fundaçao Para a Ciencia e a Tecnologia” under the project number PTDC/EIA-CCO/100541/2008. S. Matos is funded by FCT under the Ciencia2007 programme.

REFERENCES


---


---

\[\text{Fig. 2. Comparison of precision, recall, and F-measure results on CRAFT corpus, considering exact (E), left (L), right (R) and overlap (O) matching.}\]
Abstract

SPARQL queries are a powerful method for querying the large and increasing number of linked open data repositories available through the Semantic Web. However, generating SPARQL queries can be difficult, even for experts. Interfaces that accept questions in natural language and convert them to SPARQL queries are one solution to this problem. We describe the Linked Open Data Question Answering (LODQA) system. LODQA is developed to generate SPARQL queries from natural language, with the goal of providing an easy-to-use interface to search linked open RDF data. The paper presents a prototype version of LODQA which works on SNOMED CT, discussing the design and implementation, together with the limitations of the current implementation and future directions for improvement.

1 Introduction

As data integration in the life sciences progresses, more and more relevant data from heterogeneous data sources are linked to each other, forming a web of data, also known as linked data. Meanwhile, information needs are becoming increasingly complex, which increases the need for a sophisticated query interface to allow complex and fine-grained search across a variety of data sources. As semantic web (SW) or linked open data (LOD) technology is actively applied to data integration, SPARQL\(^1\) is emerging as a standard query language. While it is powerful, SPARQL is not easy to learn or to write even for experts, due largely to the dynamic features of its vocabulary and semantics, as well as its complex syntax.

\(^1\)http://www.w3.org/TR/rdf-sparql-query/

A number of techniques to assist SPARQL query composition have been developed. They include SPARQL query editors and graphical query composition. We present another approach, based on natural language processing (NLP) technology. Natural language is one of the most natural ways of human communication, and its expressive power is high enough to represent complex queries. If queries expressed in natural language can be seamlessly translated into SPARQL queries, human users will be able to search linked RDF data without having to learn the complex SPARQL language.

Development of natural language interfaces (NLI) involves optimization over many nondeterministic processes. We are developing a natural language query processing system, which we call LODQA (Linked Open Data Question Answering), as a long-term and open project. We have developed a prototype version of LODQA, based on which further research and development for SPARQL NLI will be conducted. The prototype system is designed with emphases on modularity and flexibility, so that contributions from different parties can easily be incorporated.

2 Related Work

The evaluation conducted in (Kaufmann and Bernstein, 2007) contained a usability test with 48 end-users of SW technologies, including four types of query language interfaces. They concluded that NLI were the most acceptable, being significantly preferred to menu-guided and graphical query language interfaces.

Since the potential of NLI as a natural and easy-to-use interface to information systems has been recognized for a long time, there have been several attempts to develop NLI. AquaLog (Lopez et al., 2007) used shallow parsing and WordNet for converting (controlled) natural language queries to SPARQL. ORAKEL (Cimiano et al., 2007) used...
a LTAG-inspired lexicalized grammar which they called Logical Description Grammars (LDGs) for natural language parsing. The grammars were tightly coupled with selectional restrictions specified with respect to an ontology, which made porting of the grammar to other domains a bit costly. Ran and Lencevicius (2007) developed an NLI for mobile devices for applications like personal information management. QuestIO (Tablan et al., 2008) used a part-of-speech tagger and a morphological analyzer for linguistic analysis, and an ontological gazetteer lookup to produce SPARQL or SeRQL from natural language queries. AutoSPARQL (Lehmann and Bühmann, 2011) implemented a dialog interface based on a series of short natural language expressions. BioQA2 implemented a NLI for question answering for the genomics domain, based on the data from the TREC 2005 Genomics track.

3 LODQA design and implementation

The design of LODQA shares several features with previous work. To the extent that TREC Genomics track queries were the model queries of NLI, BioQA is similar to LODQA. However, the target knowledge source of BioQA was natural language documents and the approach was a kind of paraphrase searching. This differs from LODQA, which converts natural language queries to SPARQL, targeting LOD of RDF statements.

Like several previous systems, LODQA performs linguistic analysis and ontology lookup. For linguistic analysis, LODQA adopts Enju(Miyao and Tsujii, 2008), one of the state-of-the-art English parsers. Enju is based on the HPSG grammar formalism, and, as one of its special features, it produces predicate-argument relations of the words in a sentence. We used a version of Enju that is trained on English-language questions (Hara et al., 2011). For ontology lookup, LODQA uses OntoFinder3, which searches ontologies in BioPortal for ontology terms.

Similarly to the work by Ran and Lencevicius (2007), LODQA produces SPARQL queries in two steps with different specificity: it first produces a pseudo query based only on linguistic analysis, then a final SPARQL query by incorporating the vocabulary of the target SPARQL endpoint, separating the language-dependent and schema-dependent processes. By this separation, we aim to improve the modularity of the system, to allow divide-and-conquer-style development, and to minimize the cost of adapting the system to different endpoints. While Ran and Lencevicius (2007) extensively explored the role of domain ontologies, minimizing discussions of linguistic parsing, LODQA, intended to be a long-term project, currently focuses on a more general approach with a full-featured state-of-the-art parser, Enju, putting more emphasis on sensitivity than on specificity of search.

Figure 1 (all figures are currently at the end of the paper) shows the results of LODQA for the query What devices are used to treat heart failure?. Note that the interface of the prototype LODQA is designed for the purpose of assisting in the development and debugging of the system, and therefore it shows all the intermediate results step-by-step. The query shown in the Pseudo SPARQL section in the figure is the result of linguistic analysis, and the query in the SPARQL section is the result of incorporating the vocabulary of the target endpoint into the pseudo query. The following sections explain the two processes in detail.

3.1 Pseudo SPARQL Production

The production of pseudo SPARQL queries is a form of language-dependent processing. LODQA uses the Enju English parser, which is based on the HPSG grammar formalism. An advantage of Enju is that it produces predicate-argument relations of the words in a natural language sentence, finding deep subjects and objects of predicates.

In Figure 1, the Semantic analysis section shows the predicate-argument relations in the given query, as analyzed by Enju. To produce the pseudo query from the parsing result, LODQA implements the following sub-processes:

1. Base noun chunking to find base noun chunks.
2. Targeting to find the target of the query.
3. Encoding to produce a pseudo SPARQL query.
   (a) Instantiation to instantiate the entities.
   (b) Relation to relate the instances.

Base noun chunks (BNCs) are minimal noun phrases without rightward modifiers such as relative clauses or prepositional phrases and without leftward modifiers such as articles, e.g. a or the. In the example, devices and heart failure are found as BNCs, based on part-of-speech patterns. The
BNCs become the building blocks of the pseudo SPARQL query. The targeting step finds the targets of queries among the BNCs. In the example, devices is determined as the target of the query (colored in red), as it is modified by the wh-word, what. The encoding step is to find how the target is restricted in its relation with other noun chunks as expressed in the query, and to encode the restriction in a form of SPARQL. For this step, first the entities represented by the BNCs are instantiated:

\[
\text{?t1 rdf:type [devices]}
\text{?t2 rdf:type [heart failure]}
\]

Then, the instances are related as expressed in the query. LODQA determines the relations from the predicate-argument graph: when there is a path between any two noun chunks without another noun chunk intervening, it is assumed that there is a relation between the two and that the relation is represented by the words on the path. Note that for simplicity we only consider the shortest path. To find shortest paths between any two noun chunks, Dijkstra’s algorithm(Dijkstra, 1959) is used. In the example, used → to → treat is the shortest path connecting devices and failure, which is represented by [used to treat] in the pseudo query:

\[
\text{?t1 [used to treat] ?t2}
\]

As the pseudo query expresses the structure of the final SPARQL query, leaving terms yet to be resolved to actual URIs or values, we call it the “skeleton” of the SPARQL query.

As Enju finds deep subjects and objects of predicates, we can obtain an abstraction of the natural language questions. As a result, we can get a similar pseudo query from a natural language question which has seemingly very different structure but has similar semantics, as exemplified in Figure 2.

Note that the step for pseudo SPARQL generation does not require any information about the target SPARQL endpoint, which is an important feature that we intended for a modularized development and for collaboration with groups of different expertise.

### 3.2 Final SPARQL Production

Given a pseudo query, we need to resolve the terms and relations that are represented as a sequence of words in the pseudo query. The final SPARQL queries are actual queries to be executed on the target SPARQL endpoints, and thus have to be written using the vocabulary (URIs) of the target endpoints. To get the URIs of the terms in the subject or object position, e.g. devices and heard failure, LODQA uses the REST API of OntoFinder. OntoFinder takes as input a list of terms and returns their corresponding URIs. If the REST call arrives with a specification of the ontologies to be searched, only the URIs in those ontologies are returned. While OntoFinder can search all of the ontologies in BioPortal, the prototype version of LODQA is implemented using only the Systematized Nomenclature of Medicine Clinical Terms (SNOMED CT)\(^4\) as the target SPARQL endpoint.

The result of vocabulary lookup using OntoFinder for the terms in the example query is shown in the Term mapping section in Figure 1. In the example, one URI is found for the term represented by [devices], and two for [heart failure]. For the latter case, we may treat the two as alternatives and construct the final query using both to maximize sensitivity. Alternatively, one could perform disambiguation to choose only one and maximize specificity. For the prototype implementation, we simply choose the one that is most highly ranked by OntoFinder\(^5\).

For the relations, the ultimate goal is to find the right predicates corresponding to the linguistic cues, e.g., used, to, and treat in the brackets, probably considering contextual information and casting it as a relation classification task. For the prototype system, however, we make the maximum generalization, modeling every relation as the “any” relation that is represented by a free variable in the example. This choice of implementation maximizes the sensitivity of LOD search.

The approach of making the maximum generalization may seem likely to lead to random relations. However, in fact, it is quite controlled, because the types of the two entities are constrained. For example, when one entity is constrained to be a device and another to be a heart failure, there is a limited number of relations that actually exist between them. So, even without explicit specification of the relation, the maximum generalization approach can work reasonably well. Nevertheless, it is true that there is a possibility of unintended relations being allowed in the search, and we will need to implement a filtering to improve the specificity. This remains as future work. We may be able to cast it as a disambiguation task.

---

\(^4\) The virtual Id of SNOMED CT in BioPortal is 1353.

\(^5\) The ranking is based on string matching distances.
By incorporating the vocabulary of SNOMED CT and specifying the target graph to be SNOMED CT, the final SPARQL query works on the SNOMED CT SPARQL endpoint. The Results section shows the result of executing the example query on SNOMED CT, which gives four entries from the endpoint.

4 Discussion and conclusions

LODQA is a long-term project with the goal of developing an NLI for LOD related to the life sciences. In its early stage, we developed a prototype system with SNOMED CT as the target SPARQL endpoint. It is the first working version of the system, marking a significant milestone of the project. The focus of the design and implementation was put on making a modularized system and adopting a reasonable solution for each module, so that subsequent research and development for NLI can be conducted in a flexible and efficient way based on the initial system. The methods adopted to implement the prototype system are as follows:

- Enju for parsing.
- Pattern matching for base noun chunking.
- Pattern matching for targeting.
- Dijkstra’s algorithm for shortest path finding.
- OntoFinder for ontology lookup.
- Choice of a default relation for predicate determination.

The prototype system is available online at [removed for anonymization purposes]. While we believe the adopted methods are reasonably chosen, the performance of the system has yet to be thoroughly evaluated in both qualitative and quantitative ways, which itself is in fact a highly challenging task. Our attempt at subjective evaluation based on many trials suggests that the current system performs reasonably well for simple queries without special linguistic constructs, e.g. coordination or negation, although Enju often fails with imperative sentences, as exemplified in Figure 3.

For a more objective evaluation, we are planning to collect natural language questions in various ways, which may include:

- implementation of query collection in LODQA itself,
- exploration of the space of extant ontologies and knowledge bases,
- automatic generation of queries, and
- crowd-sourcing for the evaluation of the results.

One of our goals is to more fully explore the range of linguistic variability in natural language questions. To this end, we have implemented a language generation application that produces 372 varieties of queries that share the same semantic content but have different surface linguistic forms. For example, for a query in which the user is looking for representations of the term cell, we generate questions that include the following:

- What ontology has the term cell?
- Which ontology has the term cell?
- What ontologies have the term cell?
- Which ontologies have the term cell?
- What vocabularies have the term cell?
- Which vocabularies have the term cell?
- What terminologies have the term cell?
- Which terminologies have the term cell?

To ensure robustness in the face of ways that native and non-native speakers might enter the same query, we generate various punctuational variants, e.g.

- What ontology has the term cell?
- What ontology has the term cell.
- What ontology has the term cell

We generate variability in singular versus plural number and verb agreement, e.g.

- What ontology has the term cell?
- What ontologies have the term cell?

We generate variants in the specific verbs in the query:

- Which vocabularies contain the term cell?
- Which vocabularies have the term cell?

We also generate questions that probe a variety of types of relationships between concepts, including presence in an ontology, is-a (hyponymy), synonymy, preferred name, inverse, and the presence of specific relations:

- What ontology has the term cell? (Presence in an ontology)
- What terms have the parent cell? (is-a)
• What term has the synonym nuclear? (Synonymy)
• Which concept is the preferred name for nuclear? (Preferred name)
• What concept is the inverse of proliferation? (Inverse)
• Which concepts are regulated? (Presence of a specific relation)

We also explore the space of pragmatics, in the sense of the variety of linguistic forms that can be used to ask questions in English, such as imperatives, passives, and focus shifts in addition to conventional question forms:
• Find me the devices to treat heart failure. (Imperative)
• What devices are used to treat heart failure? (Passive)
• Heart failure can be treated by what devices? (Passive, focus shift)

In addition to the sorts of meta-level questions about ontologies and their concepts and terms, and medical questions targeting SNOMED CT, that we have illustrated above, we also have test suites prepared that address instance-level questions, in particular from UNIPROT, such as:
• What are the ligands of human TP53?
• What are the isoforms of human TP53?

We have also constructed natural language queries that correspond to the sample SPARQL queries at the web page beta.sparql.uniprot.org.

As future work, improvement of each module of the system will be conducted, including but not limited to the following:
• Adding devices to treat coordination and negation.
• Improving the performance of Enju, especially for imperative sentences.
• Testing alternative solutions for ontology lookup, e.g., MetaMap (Aronson and Lang, 2010) or BioPortal Annotator (Jonquet et al., 2009).

The development of different modules may contribute to the performance of LODQA individually. However, we have also found that sometimes the contribution is complementary. For example, with the query shown in Figure 3, Enju fails to produce the correct analysis. However, the loose implementation of the following steps, e.g., the use of OntoFinder, which is an implementation of a string similarity algorithm, and the use of the default general relation, does not make a difference in the final SPARQL query. The improvement of individual modules therefore has to be evaluated not only individually but also in the aggregate.

References


Figure 1: Screenshot of LODQA results for *What devices are used to treat heart failure?*
Figure 2: LODQA results for *Heart failure can be treated by what devices?*

Figure 3: LODQA results for *Find me the devices to treat heart failure.*
Towards automatic large-scale curation of genomic variation: improving coverage based on supplementary material

Antonio Jimeno Yepes  
National ICT Australia  
Victoria Research Laboratory  
Melbourne, Australia  
antonio.jimeno@gmail.com  

Karin Verspoor  
National ICT Australia  
Victoria Research Laboratory  
Melbourne, Australia  
karin.verspoor@nicta.com.au  

Abstract

There are ongoing large-scale efforts to catalog genomic variation related to disease in structured databases. Much of the relevant information is available only from unstructured sources, including the scientific literature. The ability of text mining tools to recover the mutations cataloged in the COSMIC and InSiGHT databases based on the article text has been demonstrated to be far less than what would be expected based on the excellent performance on intrinsic evaluation of mutation extraction tools. We explore the impact of processing tables and supplementary material associated to relevant literature, and find that the coverage of variants improves dramatically, from 2% to over 50%. This result highlights the importance of processing all of the data associated with a publication.

1 Introduction

A major thrust of modern biological research is the understanding of how genomic variation relates to disease. There are large-scale efforts to catalog the results of this research in structured databases, including the Online Mendelian Inheritance in Man (OMIM) database and the Human Gene Mutation Database (HGMD). Much of this information is available only from unstructured sources, including the scientific literature.

There have been several systems developed to target extraction of mutations and other genetic variation from the literature (Baker and Witte, 2006; Caporaso et al., 2007; Krallinger et al., 2009; Doughty et al., 2011; Naderi and Witte, 2012), *inter alia*. The performance of these tools has been claimed to achieve high precision and recall on *intrinsic* evaluation.

In previous work, we performed an *extrinsic* evaluation of a mutation extraction tool with respect to the task of curation of the literature for the purpose of populating a database of genetic variation information (Jimeno Yepes and Verspoor, 2013). We found that the ability of the text mining tool to recover the mutations catalogued in the databases is far less than what would be expected based on the excellent performance on intrinsic evaluation.

Here, we extend our mutation extraction analysis to include not only the mutations extracted from the abstract and full text of the articles, but also those in supplementary material. Our results show that the coverage obtained by using supplementary material reaches over 50% of the gene-mutation pairs for the analyzed articles, while the coverage with full text alone is approximately 2%. This indicates that the supplementary material is critically important for complete processing of the information available from publications.

2 Methods

We identified two mutation databases that have explicit, curated links to the source literature for individual genetic variants. We accessed that literature, collecting full text where possible, and applied a tool to identify genetic variants in the text.

2.1 The mutation databases

2.1.1 COSMIC database

COSMIC (Bamford et al., 2004) contains comprehensive, curated, information on somatic mutations in human cancer. We used version v62 available from COSMIC’s FTP site1,

---

1ftp://ftp.sanger.ac.uk/pub/CGP/cosmic/
including mutation information curated from 9,950 unique PubMed citations (referenced via PubMed identifiers, or PMIDs). The database associates 7,868 publications to individual mutations in specific genes. The remaining articles contain non-coding mutations and COSMIC does not record specific mutations. Genes are referenced by name and by HGNIC (HUGO Gene Nomenclature Committee) (Povey et al., 2001) identifier.

2.1.2 InSiGHT database
The International Society for Gastrointestinal Hereditary Tumours (InSiGHT) maintains a database of genetic variants for both Lynch Syndrome and Familial Adenomatous Polyposis. The database has curated mutations for four genes: MLH1, MSH2, MSH6 and PMS2. We accessed the database on 02/Jan/2013. The data includes variants with curated associations linked to 809 PubMed citations. Amino acids in protein variants were normalized to single letter abbreviation form.

2.2 Article collection
We collected the PMIDs available from each of the databases and searched PubMed Central (PMC-OA) subset for available articles. We were able to obtain PMC XML files for 13 articles in InSiGHT and 563 in COSMIC. We aim to extract the abstract and article text from these XML files. However, of the 13 InSiGHT PMC XML files, 4 files contained only the abstract with a link to the full text in PDF format. In the COSMIC collection, this issue occurred in 76 of the 563 PMC XML files. We downloaded the PDF articles for these articles to obtain the full text content; the PDF version of the articles contains the article text, references and images that are not included as supplementary material. These PDF versions were downloaded from the European PMC2, which offers a straightforward link to them. They were converted into plain text using Apache Tika 1.3; no specific problems were noted. The conversion maintains the column formatting; for a two column layout, the second column is appended at the end of the first.

2.3 Mutation identification in text
We selected the EMU tool (Doughty et al., 2011) to perform mutation extraction from text. Compared to other existing mutation annotation tools, EMU is able to identify a broader range of mutations, including DNA insertions and deletions, dbSNP (Sherry et al., 2001) identifiers, and point mutations. In addition, it links the mutations to the proteins and genes that appear in text and, optionally, performs sequence verification against existing databases to increase the precision of the annotations.

We post-processed the output of EMU to be comparable to the information in the COSMIC and InSiGHT databases, i.e., normalizing the mutation mentions to the HGVS format (Den Dunnen et al., 2000). The dbSNP API4 is queried to recover all available candidates for DNA and protein mutations for a given dbSNP identifier.

Protein missense mutations, mutations in the DNA that result in an amino acid change, identified by EMU are normalized to amino acid (wild type), position, amino acid (mutated). Single letter amino acid abbreviations are used. Thus, a mutation identified by EMU with wild type amino acid Ala, position 140 and mutated amino acid Thr is converted into A140T.

We normalize DNA mutations identified by EMU to the format “c.[position][wild type nucleotide]>[mutated nucleotide]”. In the case of insertion and deletions, given position ranges, hyphens are replaced by the underscore character (e.g. c.597-598delGA to c.597_598delGA).

We identified some mentions in which the position of the DNA or protein mutation as the exon/intron number or codon position. The exon and intron positions were converted to the three candidate nucleotide positions. Exon and intron mentions were removed since no precise position could be derived.

2.4 Gene normalization
EMU identifies gene mentions based on string matching of a dictionary of gene names from the Human Genome Organization (HUGO) and from NCBI’s gene database. From this

---

4 http://europepmc.org/  
3 http://tika.apache.org/1.3

dictionary, gene names identical to codon names were removed and the P53 gene name, absent in both gene dictionaries, was added. InSiGHT curated genes are easy to map since only 4 genes are included. The COSMIC database contains the gene name and in most cases a HGNC identifier. We normalized the gene mentions identified by EMU to the NCBI Gene database, and then mapped them to the corresponding HGNC identifier.

3 Article processing

Previous work (JimenoYepes and Verspoor, 2013) hypothesized the presence of mutations in tables and supplementary materials as one explanation for the low recall of mutations in full text. Here, we access the tables and supplementary materials to investigate the impact of processing those elements.

3.1 Table processing

It has been previously shown that genetic mutation information can appear in tables (Wong et al., 2009). We therefore extracted the tables and table captions associated to the XML articles in our data sets and processed them with EMU.

From the COSMIC database we found 394 articles with tables. After processing the articles with EMU, 197 articles were identified as having mutations in the tables. From the InSiGHT database there are only 8 articles with tables, of which 4 contain mutations. In these articles, no mutations were found in the abstract or full text content at all.

3.2 Supplementary material

Authors often include supplementary material with their publication that contains information supporting the claims in the paper. Supplementary material appears in a variety of file formats as shown in table 1. The InSiGHT set includes only one supplementary material file, while COSMIC has a larger set linked to the papers (505 files associated to 138 articles).

In order to process the supplementary material files with EMU, their content was converted to text using Apache Tika. It handles a large number of file types while preserving the original layout of the document, compared to other possible solutions like Open Office SDK.

Table 1: Count of supplementary file types

<table>
<thead>
<tr>
<th>Set</th>
<th>COSMIC</th>
<th>InSiGHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS Word</td>
<td>176</td>
<td>87</td>
</tr>
<tr>
<td>MS Excel</td>
<td>111</td>
<td>57</td>
</tr>
<tr>
<td>PDF</td>
<td>82</td>
<td>70</td>
</tr>
<tr>
<td>MS Powerpoint</td>
<td>34</td>
<td>17</td>
</tr>
<tr>
<td>CSV</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Images</td>
<td>101</td>
<td>36</td>
</tr>
<tr>
<td>Total</td>
<td>505</td>
<td>138</td>
</tr>
</tbody>
</table>

Manual inspection shows that MS Word documents and MS Excel files are converted to text without any problem for our coverage analysis purposes, even though further processing might be required to properly identify different sections in these documents. MS Powerpoint documents seem to contain many images, which are not processed. On the other hand, no mutations seem to be reported in them. A small number of PDF files could not be converted properly since the PDF contained scanned images, but no mutations were reported in these files.

In this work, we did not attempt to process images in the supplementary material. Manual inspection of a random selection of images show that no mutation information can be found in them.

4 Results

We assessed coverage of mutation extraction by evaluating matching of each {PMID, gene, mutation} triple extracted by EMU to the curated mutations present in the databases. From the set of 13 articles for the InSiGHT database, we find 252 mutation triples. For COSMIC, 33,814 mutation triples were identified for the 563 articles.

Table 2 shows the mutations matched (M) and the recall (R) per database and mutation source. Recall measures how many of the database mutations are also identified through the text processing. As we reported in previous work, MEDLINE® abstracts and full text articles provide very limited mutation coverage (only 3% of COSMIC mutations and ~8% of InSiGHT mutations in the complete collection we evaluated; the numbers for the PMC-OA subset considered here are comparable).
Table 2: Variant extraction results. Art=articles in set, M=Mutations matched, R=Recall (%)

<table>
<thead>
<tr>
<th>Set</th>
<th>InSiGHT</th>
<th>COSMIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Art M R (%)</td>
<td>Art M R (%)</td>
<td></td>
</tr>
<tr>
<td>Abstracts</td>
<td>13 1 0.40 563</td>
<td>140 0.41</td>
</tr>
<tr>
<td>XML Full Text (FT)</td>
<td>9 20 7.94 487</td>
<td>694 2.05</td>
</tr>
<tr>
<td>PDF Full Text (PDFFT)</td>
<td>4 7 2.78 76</td>
<td>23 0.07</td>
</tr>
<tr>
<td>Tables</td>
<td>8 18 7.14 394</td>
<td>466 1.38</td>
</tr>
<tr>
<td>FT+PDFFT+Tables</td>
<td>13 44 17.46 563</td>
<td>929 2.75</td>
</tr>
<tr>
<td>Supp. Mat.</td>
<td>1 88 34.92 138</td>
<td>17015 50.59</td>
</tr>
<tr>
<td>All</td>
<td>13 115 45.63 563</td>
<td>17896 52.92</td>
</tr>
</tbody>
</table>

The recall shows scant improvement using the few additional PDF full text files that we have added here. Relaxing the gene requirement from the evaluation triple, to allow for the possibility of gene normalization errors, does not provide a significant recall boost.

Our data shows that tables contribute another ~1% of the mutations. Combining the information from both the full text, including the articles available as PDF, and the tables (FT+PDFFT+Tables) shows that these sources are complementary. Finally, supplementary material has the largest coverage, exceeding by far any other mutation source considered (35% recall for InSiGHT, and 50% for COSMIC). Combining all the sources results in close to 50% recall in the case of the InSiGHT database, and over 52% recall in the case of the COSMIC database. In the supplementary material, most of the mutations are found either in MS Word documents or MS Excel files.

5 Discussion

The results confirm the hypothesis that most of the mutations being curated in the considered databases are not in the main article text but appear in the supplementary material. Article tables also contribute to the extracted mutations but with more limited coverage.

Even with the boost in coverage provided by supplementary material, the coverage is still only around 50%. The main reason is that mutations are represented in tables and supplementary material differently to how they are expressed in unstructured text. These representations do not correspond directly to the patterns that tools such as EMU use to recognize mutations and they are therefore missed.

In particular, elements of the mutation, such as a specific base change and the location of that change, can appear in different columns of a table or external data source. Figure 1 of (Wong et al., 2009) exemplifies this sort of representation. In addition, information can be distributed across discontiguous spans of text, such as across several sentences, document sections, or even multiple supplementary files. The current tools do not consider this.

Our work has implications for the curation of mutation databases. Biocuration workflows rely mainly on using textual data (Hirschman et al., 2012; Verspoor et al., 2013) but our results indicate that all the material linked to the articles is required to fully cover all the mutations. Our results could explain the limited coverage of not only mutations but, as well, residue extraction methods that rely on text data (Nagel et al., 2009) and possibly contribute to existing tools like LEAP-FS (Verspoor et al., 2012).

The current study is based on articles available from PMC-OA, which is a reduced set compared to the articles available from PubMed. Applicability of our work is limited to the access of licensed content and diversity of formats available from different journals.

6 Conclusions and Future Work

We have presented an analysis of text mining for genetic variant extraction, extending previous work by considering supplementary material. The achieved recall of approximately 50% of curated mutations dramatically exceeds previously reported results. We plan to build on these results by developing targeted methods for mutation extraction in tables and supplementary material, possibly also includ-
ing mutation extraction from images. The results we have provided here indicate that such methods are critical for achieving effective information extraction of genetic variant data from the literature.

Our work has focused on recall of mutations. We would also like to evaluate more carefully the precision of the mutation extraction. In particular, we hope to refine the current tools to address more specific requirements of database curators (e.g., a given database may be limited to either germ line or somatic mutations, or the database may be restricted to genetic variants related to a specific disease) or to provide further information about the mutations like the reference sequence considered in the reported mutations. We therefore plan to continue this work, linking extracted mutations to more contextual details.

7 Acknowledgements

We thank the InSiGHT database curator, John-Paul Plazzer of the Royal Melbourne Hospital, for sharing the InSiGHT data and helping us to interpret the database fields. We also thank the COSMIC team for helpful details about their database.

Funding: National ICT Australia (NICTA) is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References


N. Naderi and R. Witte. 2012. Automated extraction and semantic analysis of mutation impacts from the biomedical literature. BMC Genomics, 13(Suppl 4):S20–.


Towards an Integrated Compound to Compound Relatedness Measure

Maria Kissá, Michael Schroeder and George Tsatsaronis

Biotechnology Center (BIOTEC) of the Technische Universität Dresden, Tatzberg 47-51, 01307 Dresden Germany.

ABSTRACT
The wealth of the publicly available data repositories related to chemical compounds and substances allows current research methodologies to integrate pieces of information across different resources. Typical compound-to-compound relatedness measures are based on structural commonalities between the compounds, sequential information for their targets or/toxicological liabilities. In this paper, we take a step further towards compound-to-compound relatedness and integrate a significantly larger number of compound characteristics, including chemical properties, indications, related sequences, pathways, genes, toxicity, and other pharmacological information. The main novelty of the suggested methodology is the systematic data integration and the combination of several similarity measures, including a string kernel, Jaccard and Tanimoto. We evaluate the performance of the proposed relatedness measure through a manually curated benchmark dataset, also introduced in this work. Our results suggest that the proposed method generates meaningful associations among the tested compounds; examples of these associations are presented and discussed analytically.

Keywords: compound relatedness, integration of compound resources, substance characteristics, evaluation

1 INTRODUCTION
Over the past decade there has been a growing interest in research methodologies that aim at producing computationally associations between compounds. The range of applications varies and, among others, includes drug-target prediction (Perlman et al., 2011), biological activity elucidation (Klebe et al., 1994), computational drug design (Zhou et al., 2010), identification of drug resistance factors (Gamo et al., 2007), prediction of drugs’ adverse events (Bender et al., 2007) and indications (Gottlieb et al., 2011), as well as drug repositioning (Li and Lu, 2012).

With regards to the methodological approaches followed, in the majority of the cases, the following aspects of the chemical compounds are considered: (i) adverse events, (ii) chemical features, (iii) sequence data, and, (iv) combinations of the aforementioned. More specifically, regarding the use of adverse event profiles, it has been shown to lead to the identification of shared targets between drugs (Campillos et al., 2008). For the same purpose, drug actions and phenotypic information are also considered (Mizutani et al., 2012). In addition, similarities between compounds have been inferred based on chemical features (Tabei et al., 2012) or sequential information of drug targets (Yamanishi et al., 2008). In parallel, the role of text mining in identifying related compounds for drug repurposing purposes, has also been found extremely important (Roberts and Hayes, 2008).

In this paper we adopt a wider perspective of the compound characteristics that may be utilized in a compound-to-compound relatedness measure. More precisely, we integrate the information from numerous popular databases pertaining to compound data, and employ an extended set of similarity measures in order to compute compound relatedness in a wide range of considered dimensions. In total, 30 distinct compound characteristics are taken into consideration for the quantification of the relatedness between two given compounds. Notably, apart from the high data diversity, a significant bottleneck arising towards the systematic evaluation of the suggested relatedness measure was the lack, to the best of our knowledge, of a high quality benchmark dataset, which could be used to assess the reliability of the associations generated via the proposed methodology. Hence, an additional contribution of this work is the creation of such a manually curated benchmark dataset for the quality assessment of the suggested approach. In the following, we give an overview of the used resources (Section 2.1), a description of the created benchmark dataset (Section 2.2) and the considered compound characteristics (Section 3.1), as well as the employed similarity measures (Section 3.2). Finally, we discuss the potentiality of our method based on a case study of 10 known chemical compounds and their produced associations (Section 4).

2 MATERIALS
2.1 Data sources
The publicly available repositories comprising compound related data vary. Herein we attempt to exploit as much of this information as possible, by integrating data from five popular repositories, namely DrugBank (Knox et al., 2011), SIDER (Kuhn et al., 2010), CTD (Davis et al., 2012; Wiegert et al., 2009), PharmGKB (Whirl-Carrillo et al., 2012), and STITCH (Kuhn et al., 2011).

DrugBank is a popular drug repository comprising approximately 6,700 drug entries regarding approved, experimental and nutraceutical drugs and their targets. The set of drugs included in this database serves as the drug set to which our methodology is applied. Each drug record (drugcard) consists of structural, chemical and protein information regarding a drug and its targets. The respective fields constitute some of the features used by our method. Additionally, for each compound in DrugBank we retrieved the

* to whom correspondence should be addressed: maria.kissa@biotec.tu-dresden.de

1 In the remaining of the paper, the terms relatedness and similarity might be used interchangeably to describe the notion of the similarity for a given pair of compounds.
potential against compounds such as tin detail, type that exists in the produced benchmark is shown in Table 1. In Pregabalin norphine set comprises measure to a case study subset of the aforementioned data set. This bringing the opposite effect (i.e., one drug treats a condition which function synergically in a therapeutic application are being consider to a new disease and the other drug is the recommended medication for the treatment of this disease. Additionally, drugs that function synergically in a therapeutic application are being consider. Moreover, pairs of derivative drugs as well as pairs of drugs bringing the opposite effect (i.e., one drug treats a condition which is reported to be the adverse event of the second drug) are also taken into account.

For the purposes of our evaluation we applied our relatedness measure to a case study subset of the aforementioned data set. This set comprises 10 known compounds, namely: Methadone, Buprenorphine, Vancomycin, Thalidomide, Lenalidomide, Ropinirole, Pregabalin, Gabapentin, Sitagliptin, and Metformin. Their relation type that exists in the produced benchmark is shown in Table 1. In detail, Pregabalin was designed as a potent successor to Gabapentin for the treatment of neuropathic pain, among other conditions. Buprenorphine, an analgesic compound, was found to have a new application in the opioid replacement therapy, so far controlled with compounds such as Methadone. When Thalidomide’s therapeutic potential against multiple myeloma was discovered, Lenalidomide was proposed as an effective derivative without the teratogenic liabilities of its analog. Towards the therapy of diabetes mellitus, Sitagliptin and Metformin are co-administered. Ropinirole was found able to substantially replace Methadone in the therapy of restless legs syndrome, while Thalidomide’s new applications included treatment of conditions such as prurigo nodularis and chronic bullous disease of childhood; conditions for which Vancomycin has been also administered.

<table>
<thead>
<tr>
<th>Compound I</th>
<th>Compound II</th>
<th>Condition</th>
<th>Relation Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gabapentin</td>
<td>Pregabalin</td>
<td>neuropathic pain</td>
<td>derivatives</td>
</tr>
<tr>
<td>Buprenorphine</td>
<td>Methadone</td>
<td>opioid replacement therapy</td>
<td>repositioning</td>
</tr>
<tr>
<td>Lenalidomide</td>
<td>Thalidomide</td>
<td>multiple myeloma</td>
<td>derivatives</td>
</tr>
<tr>
<td>Metformin</td>
<td>Sitagliptin</td>
<td>diabetes mellitus</td>
<td>synergic</td>
</tr>
<tr>
<td>Ropinirole</td>
<td>Methadone</td>
<td>restless legs syndrome</td>
<td>repositioning</td>
</tr>
<tr>
<td>Thalidomide</td>
<td>Vancomycin</td>
<td>prurigo nodularis</td>
<td>repositioning</td>
</tr>
<tr>
<td>Thalidomide</td>
<td>Vancomycin</td>
<td>chronic bullous disease of childhood</td>
<td>repositioning</td>
</tr>
</tbody>
</table>

Table 1. The considered compound pairs and their relations, which are used in our case study.

corresponding adverse events information available in the SIDER database. SIDER currently counts roughly 1,000 drugs and 4,200 side effects.

Taking into account that a drug is not only characterised by its adverse effects but also from the diseases that it has been found to be associated with, we chose to include information from the Comparative Toxicogenomics Database (CTD). CTD provides detailed reports about the diseases that a drug is either found to be correlated to, have a therapeutic application, or a potential etiological role in its development. In addition, we include the respective pathway information provided by the PharmaGKB database, for the drugs that it was already made available. Finally, we employed STITCH to collect chemical-to-chemical interactions and chemical-to-protein relations. The overall size of the raw data that we collected adds up to 16.3 GB. Each of the data sources was stored in an independent database, and as a final step, an integrated database was produced in-house using as keys the DrugBank and PubChem (Bolton et al., 2008) ids.

2.2 Benchmark Dataset

For the evaluation of the suggested measure, a gold standard set comprising pairs of drugs with similar properties is a prerequisite. Unfortunately, such a gold standard dataset does not exist. To overcome this issue, we manually built and curated such a dataset from the literature pertaining to drug repositioning. The dataset consists of roughly 150 drug pairs belonging to four distinct categories: (1) derivative compound, (2) known repositioned compounds, (3) compound characteristics which are taken into account for the computation of the compound relatedness values. These are shown in Table 2, and they are grouped according to the similarity measure that is used for each of the characteristics. The similarity measures used are explained analytically in the next section.

As the table shows, we have collected for each compound a total of 30 characteristics, which may be split into five categories: (1) fields for which we have a set of string values for each compound, where the values are stemming from a fixed and predefined vocabulary, e.g., the interacting drugs for each compound, or the pathways which the compound affects; for a pair of compounds, their relatedness in these fields may be measured using the Jaccard similarity coefficient, (2) fields for which we have numeric values, e.g., logP or the polarizability of the compound; in these fields the relatedness may be measured after normalizing the numeric values for each field in the range [0, 1], and then, given a pair of such values, the distance may be computed, through which the similarity may be inferred by subtracting from the unit, (3) fields for which we have free textual descriptions; in these fields the relatedness may be measured using a string kernel that considers n-grams, (4) fields for which we have the FASTA sequences of related target, carrier, transporter or enzyme proteins, as well as the respective DNA sequences; the relatedness in these fields may be computed using BLAST (Altschul et al., 1990), but since for each compound we have a set of such sequences, we employ a measure called THESUS (Varlamis et al., 2004) to generalize the similarity to the sets of a given pair of compounds, (5) the 3D structural representation of the compounds, in which case we compute the Tanimoto similarity for a given pair of compounds in 3D.

3 METHODS

3.1 Compound Characteristics

Following the integration of the resources described in Section 2.1, we present here the compound characteristics which are taken into account for the computation of the compound relatedness values. These are shown in Table 2, and they are grouped according to the similarity measure that is used for each of the characteristics. The similarity measures used are explained analytically in the next section.

As the table shows, we have collected for each compound a total of 30 characteristics, which may be split into five categories: (1) fields for which we have a set of string values for each compound, where the values are stemming from a fixed and predefined vocabulary, e.g., the interacting drugs for each compound, or the pathways which the compound affects; for a pair of compounds, their relatedness in these fields may be measured using the Jaccard similarity coefficient, (2) fields for which we have numeric values, e.g., logP or the polarizability of the compound; in these fields the relatedness may be measured after normalizing the numeric values for each field in the range [0, 1], and then, given a pair of such values, the distance may be computed, through which the similarity may be inferred by subtracting from the unit, (3) fields for which we have free textual descriptions; in these fields the relatedness may be measured using a string kernel that considers n-grams, (4) fields for which we have the FASTA sequences of related target, carrier, transporter or enzyme proteins, as well as the respective DNA sequences; the relatedness in these fields may be computed using BLAST (Altschul et al., 1990), but since for each compound we have a set of such sequences, we employ a measure called THESUS (Varlamis et al., 2004) to generalize the similarity to the sets of a given pair of compounds, (5) the 3D structural representation of the compounds, in which case we compute the Tanimoto similarity for a given pair of compounds in 3D.

3.2 Similarity Measures

In the following, we give the details of the five different employed similarity measures, and we discuss how we combine them to measure the overall relatedness for a given pair of compounds considering all of the 30 compound characteristics. For the description of the measures we will use the notation shown in Table 3.

<table>
<thead>
<tr>
<th>Similarity</th>
<th>Characteristic</th>
<th>Similarity</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>String Kernel</td>
<td>Drug Description</td>
<td>Tanimoto</td>
<td>3D structure of the compound</td>
</tr>
<tr>
<td></td>
<td>Mechanism of Action</td>
<td>BLAST</td>
<td>DNA sequences</td>
</tr>
<tr>
<td></td>
<td>Pharmacology</td>
<td>String Kernel</td>
<td>DNA sequences</td>
</tr>
<tr>
<td></td>
<td>Protein Binding</td>
<td>Jaccard</td>
<td>DNA sequences</td>
</tr>
<tr>
<td></td>
<td>Toxicity</td>
<td>LogP</td>
<td>DNA sequences</td>
</tr>
<tr>
<td></td>
<td>Metabolism</td>
<td>LogS</td>
<td>DNA sequences</td>
</tr>
</tbody>
</table>

Table 2. The considered compound characteristics grouped according to the employed similarity measure.

BioLINK SIG 2013 p. 45
Table 3. Summary of notation used for the description of the similarity measures.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_i$</td>
<td>A field (characteristic) of a compound among the 30 of Table 2</td>
</tr>
<tr>
<td>$F_1$</td>
<td>The set of the Jaccard similarity compound fields</td>
</tr>
<tr>
<td>$F_2$</td>
<td>The set of the string kernel similarity compound fields</td>
</tr>
<tr>
<td>$F_3$</td>
<td>The set of the numeric similarity compound fields</td>
</tr>
<tr>
<td>$C_{1}, C_{2}$</td>
<td>The set of the BLAST similarity compound fields</td>
</tr>
<tr>
<td>$C_{1}, C_{2}$</td>
<td>Two input compounds</td>
</tr>
<tr>
<td>$F_{C_{1}}, F_{C_{2}}$</td>
<td>All the fields of $C_{1}$ and $C_{2}$ respectively for which values are not null</td>
</tr>
<tr>
<td>$f_{(i,c_{1})}, f_{(i,c_{2})}$</td>
<td>The set of values of field $f_i$ for $C_{1}$ and $C_{2}$ respectively</td>
</tr>
<tr>
<td>$</td>
<td>f_{(i,c_{1})}</td>
</tr>
<tr>
<td>$C$</td>
<td>The set of all compounds</td>
</tr>
</tbody>
</table>

3.2.1 Jaccard Similarity Coefficient: Given the fields $f_i \in F_1$, we can compute the similarity in each such field between $C_1$ and $C_2$, as shown in the following equation. This similarity is always in the range $[0, 1]$.

$$J(f_{(i,c_{1})}, f_{(i,c_{2})}) = \frac{|f_{(i,c_{1})} \cap f_{(i,c_{2})}|}{|f_{(i,c_{1})} \cup f_{(i,c_{2})}|}$$ (1)

3.2.2 String Kernel: Given the fields $f_i \in F_2$, we can compute the similarity in each such field between $C_1$ and $C_2$, using the string kernel suggested by Lodhi et al. (2002). The used kernel receives as input two texts and computes their inner product in the feature space generated by all character subsequences of maximum length $n$ (n-grams). We use $n = 5$, and the implementation provided by Mallet2. The resulting similarity between two input texts is always in the range $[0, 1]$.

3.2.3 Numeric Distance as Similarity: Given the fields $f_i \in F_3$, we can compute the similarity in each such field between $C_1$ and $C_2$, by normalizing first the values $f_{(i,c_{1})}$ and $f_{(i,c_{2})}$ in the range $[0, 1]$, as shown in Equation 2 for $C_1$, but the same can be applied for $C_2$, and then, computing the similarity as shown in Equation 3. The resulting similarity between two numeric values is always in the range $[0, 1]$, because their distance is also in the same range.

$$f_{(i,c_{1})} = \frac{f_{(i,c_{1})} - \text{MINC}_{C_{1}}(f_{(i,c_{1})})}{\text{MAXC}_{C_{1}}(f_{(i,c_{1})}) - \text{MINC}_{C_{1}}(f_{(i,c_{1})})}$$ (2)

$$\text{NumSim}(f_{(i,c_{1})}, f_{(i,c_{2})}) = 1 - \frac{|f_{(i,c_{1})} - f_{(i,c_{2})}|}{|f_{(i,c_{1})}| - |f_{(i,c_{2})}|}$$ (3)

where $|f_{(i,c_{1})} - f_{(i,c_{2})}|$ symbolizes the absolute value of the Euclidean distance between $f_{(i,c_{1})}$ and $f_{(i,c_{2})}$.

3.2.4 Tanimoto: Given the one and only 3D structural field for $C_1$ and $C_2$, we compute the similarity between them using the Tanimoto similarity measure applied on the 3D fingerprints of the compounds. For the purposes of our implementation, we are using the Marvin suite AP1. The resulting Tanimoto similarity between the two compounds is always in the range $[0, 1]$, as the measure is equivalent to the Jaccard similarity index.

3.2.5 THESUS on BLAST: Given the fields $f_i \in F_4$, we can compute the similarity in each such field between $C_1$ and $C_2$, by computing first the NCBI’s BLAST sequence alignment3 for every pair of sequences belonging to $f_{(i,c_{1})}$ and $f_{(i,c_{2})}$. Then, we compute the overall similarity between the two sets $f_{(i,c_{1})}$ and $f_{(i,c_{2})}$ by applying the THESUS set similarity measure (Varlamis et al., 2004). In summary, THESUS finds for each sequence in $f_{(i,c_{1})}$: the maximum BLAST similarity with any sequence in $f_{(i,c_{2})}$. Once it computes all the $|f_{(i,c_{1})}|$ maximum similarities, it does the same reversely, for each sequence in $f_{(i,c_{2})}$. Finally, it averages the sum of all the computed maximum similarities (in total there are $|f_{(i,c_{1})}| + |f_{(i,c_{2})}$) such. Since the BLAST similarities are in the range $[0, 1]$, in this case THESUS also returns an overall similarity in the same range.

3.2.6 Overall Relatedness: Finally, we compute the overall relatedness between $C_1$ and $C_2$ by averaging the individual similarity scores on all common fields between $F_{C_{1}}$ and $F_{C_{2}}$.

4 RESULTS

The results of our experimental evaluation using the 10 drugs dataset described in Table 1 follows. First, we computed all the pairwise relatedness scores between all 10 drugs, i.e., a total of 45 pair relatedness scores. Next, for each drug we produced the ranking of the remaining 9 based on the relatedness scores, and measured for each one the R-precision, where $R$ is the number of related drugs that we should identify for the examined drug based on our dataset. Overall, for 7 out of the 10 drugs, R-precision was found 100%, for 2 out of the 10 it was found 50% and for only one was found 0%. This shows that the suggested methodology produced very meaningful rankings in the majority of the cases.

In addition, we explored whether there is any relation between any of the drugs and its top-ranked related drugs which we might have missed in our dataset. The results of this manual curation were very encouraging. In the majority of the cases, though the top-ranked was in our dataset, we identified a missing relation between the second ranked drug as well. We analyze this process for 4 out of the 10 drugs in the following, with a visualization shown in Figure 1. In the case of Lenalidomide, the top-ranked (Thalidomide) is its analog, which we knew from our dataset. Lenalidomide was designed as an analog to avoid the teratogenic effects of Thalidomide. The second drug is the anti-epileptic compound Pregabalin. This drug is reported to treat myeloma neuropathy, a symptom of myeloma to which Lenalidomide is used as a primary treatment, a relation which we did not know it existed. Regarding Pregabalin, its known predecessor Gabapentin received the highest similarity score. Lenalidomide follows, as described previously, which shows consistency between the results. Moving to Gabapentin, Pregabalin ranks first, and Methylamin follows. Interestingly, it is reported that co-administration of these compounds may result in reversible hearing loss, therefore it is highly probable that they share some contradictory functionalities. Lastly, Vancomycin has been found similar to Thalidomide and Sitagliptin. With regards to Thalidomide, the reason is obvious, since that was the compound repositioned for the condition prurigo nodularis, to which Vancomycin constitutes also a therapeutic agent. As far as Sitagliptin is concerned, this compound along with Vancomycin is used in the treatment of renal failure, a relation we also did not know it existed.

From the aforementioned discussion we can conclude that the suggested measure seems to produce meaningful associations between the compounds. We also observed that the same pipeline might also be used towards a (semi-)automated approach for the production of a benchmark set in order to evaluate drug or compound similarity measures. However, there are still many parameters to be considered in order to apply the suggested measure in a larger scale and produce rankings for thousands of related compounds.
of examined compounds, and conduct an analysis per relation type.

We are planning to explore several alternatives to the used similarity measures. To do this, we created a benchmark dataset of compound pairs for which there exists a strong reported relation, i.e., derivative compounds, known similarly but with opposite effects. Our evaluation in a set of 30 considered compound characteristics. Naturally, this needs to change specifically to the application for which the relatedness measure is applied, e.g., by learning the respective weights of the characteristics for this application.

5 CONCLUSIONS AND FUTURE WORK

In this paper we presented a novel relatedness measure for chemical compounds. The novelty of the measure lies in the utilization of 30 different compound characteristics stemming from the integration of several popular databases, namely DrugBank, SIDER, CTD, PharmaGKB, and STITCH. The measure uses five different similarity measures to compute the overall compound-to-compound relatedness, depending on the nature of the values of each of the 30 characteristics considered. For the purposes of our evaluation, we created a benchmark dataset of compound pairs for which there exists a strong reported relation, i.e., derivative compounds, known repositioning examples, synergistic drugs, or compounds that act similarly but with opposite effects. Our evaluation in a set of 10 compounds showed that the produced ranked associations of the suggested measure are meaningful and consistent. As a future work, we are planning to explore several alternatives to the used similarity measures, such as semantic smoothing kernels for the text characteristics, and to expand our evaluation in a significantly larger number of examined compounds, and conduct an analysis per relation type of the benchmark compound pairs.

REFERENCES


Case studies in making sense of clinical text


Stanford University, California USA

ABSTRACT

It is widely anticipated that the full value of data in electronic health records will come from mining clinical notes. In practice, in order to perform data-mining studies, we need to process and analyze more than 10 million notes even for our relatively small population of patients at Stanford. Using well-known NLP tools, this processing would either take years or require the kind of horsepower that makes logistics, expertise, security, and access a different kind of barrier to overcome. Using a clinical text processing system inspired by the NCBO Annotator and Resource Index, we process over 3 million notes per hour per CPU, generate a patient–feature matrix of nearly a trillion cells, and use this data to answer biomedical questions in four application areas ranging from pharmacovigilance to comparative effectiveness.

1 INTRODUCTION

Domingos sums up the argument for the unreasonable effectiveness of data by the catch phrase, “A dumb algorithm with lots and lots of data beats a clever one with modest amounts of it.”¹ It is widely anticipated that healthcare data will dwarf that of the financial sector, and we have yet to see the kind of analytics that we should in the healthcare space. We need to keep pace with data that has grown exponentially, and, most of all, we need the text-processing tools to be simpler for people to actually use in practice.

Natural language processing (NLP) tools for clinical text have evolved to the point where they are becoming “accurate enough to be applied to real-world medical problems.”² Yet, barriers remain, and among them, usability, access, and scalability often make the pain of applying NLP tools in real clinical settings outweigh the perceived benefits.

The National Center for Biomedical Ontology developed a Web service to recognize terms from biomedical ontologies in user submitted text.³ The service was applied to dozens of publicly available data sources, and resulted in the creation of the Resource Index, which won the 2010 Semantic Web Challenge Open Track.⁴ By configuring the system with a selection of ontologies, including several from the Unified Medical Language System (UMLS) as well as the Human Disease Ontology, we show that Annotator can be used for concept recognition on clinical text.

We recently evaluated this system to estimate its overall performance using the 2008 i2b2 Obesity Challenge NLP dataset textual judgments.⁵ This corpus provides annotations for 16 commonly comorbid conditions in patients with obesity based on textual mentions in their notes. The textual judgments do not make inferences about lab values or symptoms that a doctor might use to infer a diagnosis. In addition, our team manually annotated the notes in this corpus with 9 additional conditions of interest for evaluation purposes.

We report the results of our evaluation in this paper, and illustrate several different use cases for the resulting data-structure, which we call the “patient–feature” matrix, that our text-processing pipeline produces. We will highlight several results from our studies: on profiling of drug safety and off-label drug use using EHRs, on uncovering a natural experiment in which patients with congestive heart failure (CHF) were prescribed a drug despite its black box warning, and on examination of the hypothesis that juvenile arthritis patients with allergies may be more prone to develop chronic uveitis as a complication.

2 PROCESSING CLINICAL TEXT

There are a few features of our annotation pipeline that are relevant to our goal of analyzing large volumes of clinical notes. The pipeline uses publicly available ontologies to generate a lexicon for annotation; keeps unambiguous terms that are predominantly noun phrases representing one of drugs, diseases, devices or procedures semantic types; uses the cleaned-up lexicon for term recognition in the clinical notes to tag or annotate the text; excludes negated terms or terms that apply to family and past medical history; normalizes all terms using the ontology hierarchies; and finally uses the timestamps of the note to produce a de-identified, temporally ordered patient–feature matrix.

In contrast with traditional NLP approaches, there is no sentence parsing, no noun-phrase identification, no coreference resolution, and no temporal inferences. The essence is to create a lexicon of clinical terms from the selected BioPortal ontologies, and use it for dictionary style string matching in clinical notes. The radix tree based implementation can handle large dictionaries very rapidly at ~1 kilobyte of text per millisecond (our dictionary contains 3.2 million terms). Finally, the implementation keeps track of the position of each term as it appears in the text. Thus, we can post-

¹ To whom correspondence should be addressed.
process the annotations using regular expressions to determine context, such as negation or family history. The entire pipeline has been documented in detail previously.\(^6\)

### 3 I2b2 Corpus Evaluation

Besides the 16 conditions in the 2008 i2b2 obesity corpus, we added 9 more: myocardial infarction, venous thrombosis, acute renal failure, cardiac valve fibrosis, QT prolongation, pancytopenia, urinary bladder cancer, rhabdomyolysis, and progressive multifocal leukoencephalopathy (PML). In total there were 1,237 notes in the corpus that our team manually read through and annotated.

Table 1 Evaluation on the 2008 i2b2 NLP corpus, textual judgments task. The first 16 conditions are part of the original corpus; our team annotated the bottom 9 manually. The evaluation measures the performance of judgments made by the annotator as compared to those of human curators.

<table>
<thead>
<tr>
<th>CONDITION</th>
<th>SENS</th>
<th>SPEC</th>
<th>FL</th>
</tr>
</thead>
<tbody>
<tr>
<td>asthma</td>
<td>100%</td>
<td>99%</td>
<td>96%</td>
</tr>
<tr>
<td>atherosclerotic cardiovascular disease</td>
<td>77%</td>
<td>86%</td>
<td>82%</td>
</tr>
<tr>
<td>congestive heart failure</td>
<td>72%</td>
<td>95%</td>
<td>81%</td>
</tr>
<tr>
<td>depression</td>
<td>97%</td>
<td>96%</td>
<td>87%</td>
</tr>
<tr>
<td>diabetes mellitus</td>
<td>90%</td>
<td>96%</td>
<td>94%</td>
</tr>
<tr>
<td>gallstones/cholecystectomy</td>
<td>16%</td>
<td>100%</td>
<td>27%</td>
</tr>
<tr>
<td>gastroesophageal reflux disease</td>
<td>95%</td>
<td>99%</td>
<td>94%</td>
</tr>
<tr>
<td>gout</td>
<td>94%</td>
<td>99%</td>
<td>94%</td>
</tr>
<tr>
<td>hypercholesterolemia</td>
<td>53%</td>
<td>99%</td>
<td>69%</td>
</tr>
<tr>
<td>hypertension</td>
<td>78%</td>
<td>93%</td>
<td>86%</td>
</tr>
<tr>
<td>hypertriglyceridemia</td>
<td>46%</td>
<td>100%</td>
<td>63%</td>
</tr>
<tr>
<td>obstructive sleep apnea</td>
<td>37%</td>
<td>100%</td>
<td>54%</td>
</tr>
<tr>
<td>osteoarthritis</td>
<td>69%</td>
<td>93%</td>
<td>67%</td>
</tr>
<tr>
<td>peripheral vascular disease</td>
<td>64%</td>
<td>98%</td>
<td>72%</td>
</tr>
<tr>
<td>venous insufficiency</td>
<td>74%</td>
<td>99%</td>
<td>64%</td>
</tr>
<tr>
<td>obesity</td>
<td>61%</td>
<td>100%</td>
<td>76%</td>
</tr>
<tr>
<td>myocardial infarction</td>
<td>63%</td>
<td>94%</td>
<td>70%</td>
</tr>
<tr>
<td>venous thrombosis</td>
<td>95%</td>
<td>91%</td>
<td>63%</td>
</tr>
<tr>
<td>acute renal failure</td>
<td>89%</td>
<td>90%</td>
<td>59%</td>
</tr>
<tr>
<td>cardiac valve fibrosis</td>
<td>78%</td>
<td>82%</td>
<td>11%</td>
</tr>
<tr>
<td>QT prolongation</td>
<td>76%</td>
<td>98%</td>
<td>46%</td>
</tr>
<tr>
<td>aplastic anemia / pancytopenia</td>
<td>100%</td>
<td>100%</td>
<td>92%</td>
</tr>
<tr>
<td>bladder cancer</td>
<td>67%</td>
<td>100%</td>
<td>71%</td>
</tr>
<tr>
<td>rhabdomyolysis</td>
<td>75%</td>
<td>100%</td>
<td>86%</td>
</tr>
<tr>
<td>progressive multifocal leukoencephalopathy</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td><strong>GRAND TOTAL:</strong></td>
<td>0.74</td>
<td>0.96</td>
<td>0.71</td>
</tr>
</tbody>
</table>

Following the guidelines in \(^3\), we counted each present textual judgment as condition positive. Absent, questionable or unmentioned were counted as condition negative for evaluating specificity. Absent judgments include negated conditions, so the evaluation also tests our negation and family history detection ability. In other words, mention of “ROMI” or “ruled out for MI” should be counted as Absent for myocardial infarction. There is a range of sensitivity and specificity across the conditions. No patients had PML in the corpus and our annotator found none as well. Nevertheless, we do not report the 100% specificity since there are no positive cases. The overall results reached 74% sensitivity and 96% specificity (Table 1), which would have been considered among the top performers in the 2008 challenge.

### 4 Case Studies

The annotator produces a patient–feature matrix in which every row is a patient, and every column is a drug, disease, device, or procedure term (Figure 1). The third dimension is time. Temporality is not extracted from the text itself, but rather attributed by the timestamp of each note. Thus, in most analyses, we focus only on first mentions of concepts, knowing that in many cases notes could say, for example, “patient had an acute infarct 3 years ago.” Terms can be mapped, normalized, and aggregated to higher-level concepts using BioPortal terminology information. Drugs can be further normalized to active ingredients using RxNoRM, and aggregated using ATC hierarchies. We analyzed this matrix to answer a variety of questions and highlight four of them below.

#### 4.1 Pharmacovigilance: PPIs vs. H\(_2\)-blockers

We demonstrate the feasibility of using large amounts of free-text notes for pharmacovigilance, which could advance drug safety surveillance by using this yet untapped data source. In terms of performance, such analysis for safety surveillance achieves an area under the receiver operator characteristic curve (AUC) of 0.80 for the single-drug scenario and could provide early warning for 6 out of 9 drug recalls in the past decade\(^6\), on par with performance using FDA adverse event reporting data (AUC of 0.71—0.83)\(^7\).
clinical text, along with other features encoding domain knowledge (e.g., molecular targets in DrugBank). Second, we determine which used-to-treat relationships are already known or approved usages. Using a gold standard from the Medi-Span® Drug Indications Database, our SVM classifier achieved a precision of 0.933, specificity of 0.987, recall of 0.734 and F1 score of 0.821 on a hold out test set. In feature ablation experiments, our study found that the textual features add a complementary dimension for such discovery—boosting sensitivity or recall by 10–30%.

4.3 Natural experiments: cilostazol and PAD

For peripheral artery disease (PAD), the best treatment available is cilostazol, but it is contra-indicated for use in patients with congestive heart failure (CHF). The perceived risks could be undeserved. Because milrinone put CHF patients at undue risk for adverse cardiovascular events in a prospective trial, and cilostazol happens to also be a type III phosphodiesterase inhibitor, it got a black box warning.
4.4 Hypothesis testing: uveitis and JIA

Juvenile idiopathic arthritis (JIA) is the most common rheumatology disease in children, commonly complicated by chronic uveitis, which can lead to blindness. We can analyze to the patient–feature matrix to confirm four previously known predictors for uveitis, in terms of odds ratio and 95% confidence intervals: oligoarticular-onset disease, ANA positive status, and the presence of psoriasis are positive predictors, while rheumatoid factor (RF) positive is a known negative predictor (Figure 5).

Based on the intuition of a pediatric rheumatologist and communications with family members of JIA patients, we hypothesized that allergic conditions may be associated with uveitis in JIA patients. We analyzed combinations of allergic conditions with prescriptions of allergy medications in the patient–feature matrix to evaluate the association, yielding OR 2.54 [95% CI 1.22-5.4], which supports the hypothesis.

5 DISCUSSION

Our approach is simple in comparison with advanced natural language processing systems that may have better accuracy in identifying nuanced attributes of disease conditions. Yet, by sacrificing some note-level accuracy we gain the ability to analyze large amounts of clinical data that is buried in textual notes. The system can run on a simple computer like a laptop and process 3 million notes per hour. The tradeoff in speed versus volume enables answering population-level questions “accurately enough” as envisioned by Chapman. With the increasing availability of better and faster NLP tools, we expect results to only improve in the future. We demonstrated four different use cases ranging in applications of pharmacovigilance to comparative effectiveness and hypothesis testing. In our view, this is the tip of the iceberg and we expect increasing use of electronic health records, specifically clinical notes, for informatics research.

ACKNOWLEDGEMENTS

We acknowledge support from the National Center for Biomedical Ontology (U54-HG004028). We thank contributors to the i2b2 evaluation: Tanya Podchiyska, Emily Doughty, Amogh Vasekar, Sandy Huang, and Rave Harpaz.

REFERENCES

HistoNer: Histone modification extraction from text

Philippe Thomas Ulf Leser
Humboldt-Universität zu Berlin
Knowledge Management in Bioinformatics
Unter den Linden 6
Berlin, 10099, Germany
{thomas,leser}@informatik.hu-berlin.de

Abstract

Systematic recognition of histone modifications in text is an important task to cope with the fast increase of biomedical literature. The high variability of phrases to express histone modifications renders keyword based search as insufficient for information retrieval. We present HistoNer, a rule based system for the recognition of histone modifications from text. Patterns are collected semi-automatically and manually corrected. With 305 distinct patterns the system achieves an $F_1$ measure of 93.6 % on an unseen test set of 1,000 annotated documents.

HistoNer is licensed under GNU General Public License Version 3 and available at http://code.google.com/p/histoner/. The repository contains corpora, evaluation scripts, and intermediate files generated during pattern development.

1 Introduction

In eukariotic cells, DNA is densely packed around proteins. These proteins are referred to as histones. Winding DNA around histones greatly reduces the amount of space required for the DNA. The status of histones primarily determines the availability of DNA to binding proteins like transcription factors. Thus, histone modifications are known to directly affect transcription and regulation of other genes. Some of these modifications are involved in disease progression and are an a promising target for novel drug therapies (Kelly et al., 2010).

The vast majority of novel research findings is initially presented in scientific literature. Over the years, the amount of accumulated text has grown enormously and has reached a point where finding specific information becomes troublesome. Although the Brno nomenclature for the description of histone modification has been developed (Turner, 2005), a high variety of natural language expressions describing histone modifications has been observed (Kolářik et al., 2009). The high number of possible phrases describing histone modifications hinders systematic retrieval of relevant articles describing such modifications.

1.1 Related work

Extraction of histone modification terms has been previously tackled by Kolářík et al. (2009). They use conditional random fields to detect histone modification mentions. Recognized modification mentions are subsequently term normalized to a self-developed ontology, which is inspired by the Brno nomenclature. Their system achieves an $F_1$ measure of 81 % on a test corpus of 1,000 documents.

Rule based systems often achieve good precision but lack a high recall due to the high variability of free text. More importantly, the development of hand crafted patterns is a time intensive and laborious task. Several approaches have been proposed to rapidly engineer such patterns with little or without any human intervention. For instance, Hakenberg et al. (2008) automatically derive phrase–motifs describing protein–protein interactions from scientific abstracts using a knowledge base with known protein–protein interactions. This workflow is usually referred to as distant supervision (Mintz et al., 2009). Rinaldi et al. (2010) also
follow the same rationale to generate potential patterns describing protein–protein interactions. But in difference to Hakenberg et al. (2008) the automatically generated patterns are manually evaluated and removed if too unspecific.

In this publication we generate patterns in a semi-automatic fashion by following the approach from Caporaso et al. (2007) which was proposed for the generation of mutation patterns. The approach uses background knowledge about mutations to generate potential patterns. Potential patterns are subsequently refined by the authors.

2 Methods
In this section we describe the corpus generation strategy, followed by a brief discussion of the pattern generation process and the evaluation strategy.

2.1 Histone modification definition
We define a histone modification following the specifications from the Brno nomenclature, which characterizes a histone modification as having the following four arguments:

1. Histone name (H1, H2a, H2b, H3, or H4)
2. Modification type (e.g. phosphorylation, dimethylation, acetylation, ...)
3. Modified amino acid (e.g. lysine, arginine, ...)
4. The amino acid position where the modifications occurs on the histone polypeptide (e.g. 7, 23, ...)

Hence, histone modification recognition can be regarded as quaternary relationship extraction. In this work we regard all four arguments as mandatory.

2.2 Corpus
For development and evaluation of our tool we use the 1,187 abstracts originally annotated by Kolářík et al. (2009). The corpus originally contained under-specified annotations, like “acylated histones”. We re-annotated the corpora and retained only histone modification terms mentioning all four modification arguments. The corpus has been split into training and test corpus consisting of 187 and 1,000 documents respectively by Kolářík et al. (2009). In this work we used the same corpus splits. The smaller training corpus contains 603 and the testing corpus 224 histone modification mentions. Differences in ratios between mentions of histone modifications to number of articles are due to the corpus selection strategy. The 187 training documents are manually selected by Kolářík et al. (2009) to cover a large variety of different histone modifications. The evaluation corpus has been randomly sampled from 24,635 articles annotated with the MeSH term “epigenetics”.

2.3 Generation of patterns
Patterns are generated in a semi-automatic fashion by the following strategy originally proposed by Caporaso et al. (2007). Citations from MEDLINE and fulltext articles from PMC open access have been separated into sentences by using a segmentation model trained for biomedical publications (Buyko et al., 2006). These sentences have been searched for mentions of amino acids, modification terms, and numbers. For recognizing amino-acids we generated a list of different amino-acid terms, where we used long-forms, three letter abbreviations, and one letter abbreviations (e.g. Lysine, Lys, K, ...). For numbers we used a regular expression matching all number mentions between 1 to 999. Longer numbers are ignored as all histone proteins are shorter than 999 amino-acids. The Brno nomenclature currently lists seven different types of histone modifications (e.g. acetylation, methylation, ribosylation, ...). For all these modification terms we generated regular expressions matching verb, noun and adjective forms, like acetylation, acetylates, acetylating. We further build possible word inflections and active/passive word forms.

Mentions of amino-acids, numbers, and modifications are searched in the unannotated sentences. Detected mentions are replaced by a generic symbol. For instance, amino-acids are replaced by <aa>, modification terms by <mod>, and numbers by <number>. Patterns are derived from sentences containing all four required arguments. In other words, from sentences which contain at least two numbers, one amino-acid, one modification mention. To build these surface patterns we selected the shortest span between all relevant mentions and the words between them. Potential patterns are sorted
by their occurrence in MEDLINE and PMC. Patterns occurring at least twice are manually evaluated. This pattern generation strategy is also exemplified in Figure 1.

<table>
<thead>
<tr>
<th>Input Sentence</th>
<th>&quot;The major function of MYST is acetylation of H4 at the K16 residue.&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Term recognition</td>
<td>&quot;The major function of protein MYST is &lt;mod&gt; of H&lt;number&gt; at the &lt;aa&gt;&lt;number&gt; residue.&quot;</td>
</tr>
<tr>
<td>Potential Pattern</td>
<td>&lt;mod&gt; of H&lt;number&gt; at the &lt;aa&gt;&lt;number&gt;</td>
</tr>
<tr>
<td>Annotation</td>
<td>&lt;mod&gt; of H&lt;number&gt; at the &lt;aa&gt;&lt;number&gt;</td>
</tr>
</tbody>
</table>

Figure 1: Example of the different steps for pattern generation. First, relevant terms are replaced by the respective class (i.e. <aa>, <number>, <mod>). Second, surface patterns are generated. Third, patterns are evaluated and manually refined.

This procedure results in a set of patterns which can later be used to find histone modifications. During the search phase generic symbols are replaced by regular expressions. For instance the symbol aa is replaced by all possible amino acids “(lysine|lys|K|...)”. Recall, that for pattern generation the system did not use any information contained in the training corpus. All patterns are learned from the sentences provided in MEDLINE and PMC.

2.3.1 Pattern refinement

Manually annotated patterns are automatically refined by the following steps:

1. Mentions of conjunctions (and/or) are replaced by the regular expression “and|or”
2. Prepositions are replaced by the regular expression “to|of|at|on|in”
3. For the collocation “histone H”, two additional patterns are induced, containing only “histone” and “H”
4. For patterns containing only “histone” or “H” an additional pattern with “histone H” is produced

3 Results

3.1 Pattern generation

We retrieve potentially important articles by using the prefix query “histon*” on a set of ~20 million articles. This query leads to a set of 52,113 documents with 3,656,587 sentences. 81,526 sentences contained all four required elements and are transformed into potential patterns. Patterns occurring at least twice over all MEDLINE are subsequently annotated by the authors. This leads to 268 manually annotated pattern, which are refined into 305 different patterns by the steps described in Section 2.3.1.

3.2 Evaluation

Both sets of patterns (original and refined) are used to find histone modification mentions on the two corpora. Results are shown in Table 1. Using the linguistically refined patterns improves recall by approximately 5 percentage points on both corpora. On training and test corpus we observe similar results in terms of precision, recall and $F_1$ measure.

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Training</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
</tr>
<tr>
<td>Original</td>
<td>98.8</td>
<td>81.5</td>
</tr>
<tr>
<td>Refined</td>
<td>98.9</td>
<td>87.2</td>
</tr>
</tbody>
</table>

Table 1: Performance of HistoNer on the training and testing corpus. Original refers to the unrefined pattern, whereas refined refers to the modified pattern.

4 Discussion

For named entity recognition, the approach of Kolářík et al. (2009) achieves an $F_1$ of 81 % on the test corpus. Due to the slightly different scope of the two tools (HistoNer extracts only histone modifications normalizable to Brno), the results serve only as indicator for the high quality of HistoNer and can not be directly compared. An advantage of our pattern based strategy is that term normalization to the Brno nomenclature is implicitly performed by the usage of regular expressions.

Finally, we applied HistoNer on a local repository of more than 21 million PubMed citations, where our system detects 97,563 histone modifications. An overview of the five most frequently used patterns is shown in Table 2.
Table 2: Overview of the five most frequently matching patterns. Terms in brackets are replaced by the corresponding regular expression as described in Methods. To simplify the regular expression we introduced the symbol “\p{“ matching an arbitrary punctuation mark or whitespace.

5 Conclusion

HistoNer is a stand alone tool capable of recognizing histone modification mentions in text. Detected mentions are normalized to the Brno nomenclature. For recognition it uses a set of 305 patterns, which have been automatically generated and subsequently manually corrected. The automatic refinement strategy is capable of improving recall by about 5 percentage points with unchanged precision. HistoNer achieves a remarkable performance of roughly 93 % F1 on two unseen data sets.

Recognized histone modifications are integrated in our web service GeneView1 (Thomas et al., 2012). The tool, including the set of regular expressions, evaluation scripts, intermediate files generated during pattern engineering, and documentation are freely available at http://code.google.com/p/histoner/.

We have shown that the bootstrapping strategy introduced by Caporaso et al. (2007) can be extended to another NER task, namely histone modification recognition.

References


1http://bc3.informatik.hu-berlin.de/
Poster Abstracts

A quantitative analysis of causal and associative events involving genes and proteins.
Phoebe Roberts
Page 57

Automatic Generation of BEL Statements from Text-mined Biological Events.
Haibin Liu, William Baumgartner Jr., Natalie Catlett, Andrea Matthews, Phoebe Roberts, Christophe Roeder, Aaron Hooser, Daniel Ziemek, Lawrence Hunter and Kevin Bretonnel Cohen
Page 58

Biotea.
Alexander Garcia, Casey McLaughlin and Leyla Jael García Castro
Page 59

Explaining genome-wide association study results using concept profile analysis and the Kyoto Encyclopedia of Genes and Genomes pathway database.
Kristina M. Hettne, Harish Dharuri, Reinout van Schouwen, Peter A.C. 'T Hoen, Barend Mons and Marco Roos
Page 60

In the Pursuit of Open Science.
Alexander Garcia, Leyla Jael García Castro and Olga Giraldo
Page 61

PubAnnotation - a storage system for sharing of literature annotation.
Jin-Dong Kim
Page 62

PDFJailbreak - a communal architecture for making biomedical PDFs semantic.
Alexander Garcia, Peter Murray-Rust, Gully APC Burns, Robert Stevens, Dominika Tkaczyk, Casey McLaughlin, Amaury Belin, Angelo Di Iorio, Leyla García, Célya Gruson-Daniel, Ross Mounce, Andrea Giovanni Nuzzolese, Silvio Peroni Jeremy Spinks, Boris Villazon-Terrazas, Oscar Corcho, Olga Giraldo, Mike Wabiszewski
Page 63

Text mining for characterizing cells and tissues.
Mariana Neves, Alexander Damaschun, Nancy Mah, Fritz Lekschas, Stefanie Seltmann, Harald Stachelscheid, Jean-Fred Fontaine, Andreas Kurtz and Ulf Leser
Page 64
A quantitative analysis of causal and associative events involving genes and proteins

Phoebe M. Roberts
Computational Sciences Center of Emphasis
Pfizer Inc., Cambridge MA

Biomedical research relies on indirect readouts to test hypotheses and reach conclusions from a set of experimental results. For instance, to unequivocally demonstrate that a gene region from a disease association study contains a disease-causing mutation requires the following: identifying the segregating mutation, demonstrating that it affects protein levels or function, and showing how the altered protein levels or function disrupt biological processes and lead to disease (Musunuru, Strong et al. 2010). These high-impact causal events are the culmination of sets of experiments that collectively refute alternative hypotheses and uncover the mechanistic underpinnings of disease. In the absence of support for causality, many findings are reported as associative when the data show a correlation between a gene product and disease (e.g. a biomarker for disease) or a genetic association.

To better understand how causal and associative gene and protein events are represented in biomedical abstracts, a set of causal and associative verbs was assembled by using known representative relationships to retrieve verbal connectors (referred to in (Rodriguez-Esteban, Roberts et al. 2009). For example, vascular endothelial growth factor (VEGF) is a well-known angiogenesis-inducing protein, and inhibiting signaling through VEGF has led to several marketed therapies for the treatment of cancer and age-related macular degeneration (Ferrara 2009). A straightforward “VEGF [verbal relationship] angiogenesis” query retrieves verb phrases that reflect causal, inductive relationships, such as “induces”, “stimulates”, and “is critical for”. Causal inhibitory and associative verb phrases were collected using similar well-studied relationships, and they were consistent with other biomedical verb classification studies (Rebholz-Schuhmann, Jimeno-Yepes et al. 2010).

With a gene dictionary and set of relevant verbs in hand, 334 noun phrases that followed the verb phrases were collected and classified into categories that were determined by an initial review of the results: biological processes, state changes of genes and gene products, phenotypes, diseases, and an “other” category for noun phrases that did not fit elsewhere. The four categories together were sufficient to classify 95% of the results. The difference in distribution of term classes between causal vs. associative verbs was striking: biological processes and state changes of genes accounted for 72% of the causal events, vs. 15% of the associative events. In contrast, diseases and phenotypes were found in 22% of the causal relationships, vs. 81% of the associative relationships.

This work has identified distinct patterns in causal vs. associative relationships involving genes and gene products. Prevalent terms from causal gene relationships have the potential to provide intermediary connections in gene-disease relationships.

BIBLIOGRAPHY
The Biological Expression Language (BEL; http://www.openbel.org/) designed by Selventa™ aims to represent scientific findings in the life sciences with a focus on capturing causal relationships. Knowledge in BEL is represented as BEL statements that form the basis for several successful approaches to interpreting transcriptional and genetics data in light of prior causal knowledge. However, the manual curation of BEL statements remains a bottleneck of the subsequent reasoning process. Text mining has made significant progress in recent years in extracting semantic events involving genes and proteins, such as binding and regulatory events, from the biomedical literature. Since nested event structures correspond to the causal relationship chains in BEL, automatic generation of BEL statements from text-mined biological events is feasible.

This work reports our initial attempt to bridge biological events with causal BEL statements. A knowledge-driven, surjective mapping schema is proposed to transform events of nine GENIA event types into BEL statements. A probability-based confidence score is assigned to each statement. The Turku event extraction system is used to extract biological events from Medline abstracts. Protein mentions are subsequently normalized to concepts in the Protein Ontology. When applied to the EVEX database consisting of events extracted by the Turku system over the 2009 distribution of Medline, our conversion results in a large-scale set of 126,880 BEL statements in 90,620 sentences from 63,434 Medline abstracts. In addition, evaluation of our protein normalization system against the CRAFT corpus shows a 0.81 precision and a 0.47 recall when using an overlapping-span matching criterion.
Biotea
Alexander Garcia1, 2, Leyla Jael Garcia Castro2 and Casey McLaughlin1
1Institute for Digital Information and Scientific Communication, College of Communication and Information, Florida State University, Tallahassee, Florida, 32306-2651, USA.
2Temporal Knowledge Bases Group, Department of Computer Languages and Systems, Universitat Jaumé I, Castello de la Plana, Valencia, 12071, Spain.

ABSTRACT

Motivation: Scientific information is usually locked up in discrete documents that are not always interconnected or machine-readable. The connectivity issue provided by RDF technology has not yet been widely used to support the generation of self-describing, machine-readable documents. In this paper we present our approach to machine-processable documents. We have semantically modeled and enriched the full-text open-access subset of PubMed Central. Our model delivers a highly interconnected and semantic dataset.

Introduction

In spite of technological advances, scientific publications remain poorly connected to each other as well as to external resources. Furthermore, most of the information remains locked up in discrete documents without machine-processable content. Such interconnectedness and structuring would facilitate interoperability across documents as well as between publications and online resources. Scholarly data and documents are of most value when they are interconnected rather than independent.

Methods

We use BIBO [1], DCMI Terms [2], and the Provenance Ontology (PROV-O) [3] to model the bibliographic metadata. BIBO provides classes and properties to represent citations and bibliographic references. BIBO can be used to model documents and citations in RDF or to classify documents within a hierarchy. Dublin Core (DC) [4] offers a domain-independent vocabulary to represent metadata; such vocabulary aims to facilitate cross-resource exploration. In order to identify biological terms, we use two entity recognition tools: Whatzit [5] and the NCBO Annotator [6]. Both tools are based on exact string matching and pre-defined dictionaries. By doing so, relevant biological identifiers such as UniProt accessions and ChEBI and GO identifiers are added. We are working with more than 20 biomedical ontologies.

The workflow that we followed to generate the RDF files for PubMed Central (PMC) articles is illustrated in Fig. 1. The main input for our process is the XML offered by PMC for open-access articles. We are also using available vocabularies to represent the metadata as well as the content in RDF; such vocabularies have been mapped to Java classes by using the RDFReactor. The article itself is modeled as bibo:Document; whenever it is possible, a more precise class is also added, e.g., bibo:AcademicArticle for research articles. Publisher metadata is modeled using BIBO, including publisher name, the International Standard Serial Number (ISSN), volume, issue, and starting and ending pages. Authors are modeled as a bibo:authorList, where each member is a foaf:Person. Abstract and sections are modeled as a doco:Section with a cnt:chars containing the actual text with formatting omitted. Well-known identifiers such as PubMed ids and DOIs are included in the output. In this way it is possible to track the original source of the article; the same principle is also applied to the references. In order to identify biological terms within the RDFixed article, it is processed with Whatzit and the NCBO Annotator. Those terms are modeled as semantic annotations, i.e., annotations associated to ontological concepts such as proteins, components, drugs, diseases, and medical terms. Whenever it is possible, we also link to entities in Bio2RDF [7] and identifiers.org as well as to relevant web pages.

Fig. 1. Biotea workflow

Results

We have semantically processed the full-text, open-access subset of PubMed Central. Our RDF model and resulting dataset make extensive use of existing ontologies and semantic enrichment services. We expose our model, services, prototype, and datasets at http://biotea.idiginfo.org/. The semantic processing of biomedical literature presented in this paper embeds documents within the Web of Data and facilitates the execution of concept-based queries against the entire digital library. Our approach delivers a flexible and adaptable set of tools for metadata enrichment and semantic processing of biomedical documents. Our model delivers a semantically rich and highly interconnected dataset with self-describing content so that software can make effective use of it.

References

Explaining genome-wide association study results using concept profile analysis and the Kyoto Encyclopedia of Genes and Genomes pathway database

Kristina M. Hettne¹, Harish Dharuri¹, Reinout van Schouwen¹, Peter A.C. 't Hoen¹,², Barend Mons¹,², Marco Roos¹,²

¹ Department of Human Genetics, Leiden University Medical Centre, Leiden, The Netherlands
² Netherlands BioInformatics Centre, The Netherlands

Genome-wide association studies (GWAS) with metabolomic phenotypes yield several statistically significant single nucleotide polymorphism (SNP)-metabolite associations (e.g. [1]). The information needed to arrive at an understanding of the mechanistic basis of the association requires integration of disparate structured (such as pathway databases) and unstructured (such as scientific literature) data sources. For example, the Kyoto Encyclopedia of Genes and Genomes (KEGG) RESTful Web services [2] can be used for pathway annotation, and the by us developed concept profile analysis Web services [3] can be used as a source of text-mining based annotation. The concept profile analysis technology uses the vector space model to relate two concepts (such as SNPs and biological process from the Gene Ontology) to each other and measure the strength of the relationship [4].

We evaluated the utility of KEGG pathways and concept profiles in facilitating the biological interpretation of statistically significant SNP-metabolite pairs using the Illig et al. GWAS dataset [1]. Workflows utilizing the KEGG Web services and concept profile analysis Web services were created in the Taverna workbench 2.4 [5] and made available on the workflow collaborative platform myExperiment [6,7]. Our workflow based on the KEGG pathway database was able to map 10 out of the 15 top hits in the Illig et al. study, to genes that participated in pathways relevant to the associated metabolite. The text-mining workflow was also able to reproduce 10 of the 15 manually curated SNP functions, and gave suggested annotations for the remaining five that can serve as material for further investigation and wet lab validation. This gives us a sensitivity measure of 67% (10/15). This high sensitivity is a validation of the method that seeks to utilize background knowledge present in pathway databases and literature, to make sense of SNP-metabolite pairs from genome-wide association studies of intermediate phenotypes.

In the Pursuit of Open Science

Alexander García¹, Leyla García¹, Olga Giraldo², Olga X Giraldo²
1 Institute for Digital Information and Scientific Communication, Florida State University, Tallahassee, Florida, USA. 2 Knowledge Bases Group, Universitat Jaumé I, Castelló de la Plana, Valencia, Spain.

In the Pursuit of Openness

"Life, Liberty, and the pursuit of Happiness" is a paramount phrase in the United States Declaration of Independence. The phrase illustrates the various "unalienable rights" which the Declaration says all human beings have been given by their Creator and for the protection of which they institute governments. Although "we the research community" enjoy various "unalienable rights" inherent to our work, the right of controlling our creations is not one of them. Moreover, in a similar way to that of the early colonies inheriting and adapting an onerous or tyrannical form of government, "we the researchers" are seeking more fairness and justice for the outcomes of our work—the content we generate is as diverse as research itself.

Citations, commonly referred to as citation data, illustrate an interesting situation arising from old practices in new times. Although authors carefully gather references supporting statements, once the publication is available as a PDF, citation data is significantly locked up. Furthermore, basic bibliographic metadata pertaining to the PDF is not always available. Meaningful entities such as figures, tables, captions, domain specific names and author information are often lost for practical purposes. Extracting information from PDFs as well as embedding content from PDFs into the Web of Data is challenging, though possible, but is an unnecessary barrier. How should authors link specific parts in documents to datasets? Once content is machine processable, how do we best support social participation around particular parts of the document? Science is becoming less dependent on one final document and a single central narrative; it is heading towards an aggregation of context and problem dependent self-describing research objects. Such a fluid array of research objects should support shareability, reproducibility, discoverability and reusability. Although interoperability is central to open science, "we the people" are struggling in the face of change.

Scientific content is varied, takes shapes known as scientific papers, database entries, presentations, code, etc. Although diverse, scientific content is usually related to a specific research question, hypothesis, or problem statement. By the same token, these are usually brought together in research projects. Research objects are therefore difficult to define and constrain.

Here, we present our approach to research objects as semantic aggregators. Central to our approach is the interoperability across structured ROs. We argue that ROs should be self-describing artifacts living in a fluid grid of relationships, a hyperontology that provides dynamic structuring mechanisms. Within our approach, documents, as self-describing entities, are not central to the communication of research; there is, however, a narrative structure that should be automatically built depending on the aggregation and structuring of ROs. Such aggregation is context and problem dependent; documents within this scenario are to be born semantic. Also fundamental to our approach is the idea of documents and research related files being semantic from their conception.

We illustrate our approach with two scenarios. The recreation of the research reported by Gutierrez et al in "Identification of a Rice stripe necrosis virus resistance locus and yield component QTLs using Oryza sativa × O. glaberrima introgression lines". Within this scenario, we present ROs, ancillary research-related files, laboratory notebooks, LIMS records, laboratory protocols, results, code, etc. We semantically structure all entities within a hyperontology; we then present a simple dynamic context dependent aggregation over such fluid array.

ACKNOWLEDGMENTS

Special thanks to Robert Stevens for his comments on the manuscript.

References

PubAnnotation is a storage system for sharing of literature annotation. It maintains texts from PubMed and PubMed Central (the open access subset) in a canonicalized form. Annotations, which are even produced without any connection to PubAnnotation, can be uploaded to the storage. What PubAnnotation does during the upload is to align the annotations, e.g., the character offsets, to the canonicalized texts, so that users do not need to worry about frequent small variations, e.g., extra spacing or different encoding of Greek letters, in the texts, which often causes a problem when compiling annotations from different sources. As a result, all the annotations uploaded to PubAnnotation become directly comparable to each other even if they have come from different sources or projects.

The annotations stored in PubAnnotation also can be downloaded to the local storage of a user. At that time, the user can specify a specific version of the texts of his/her own, to which the annotations to be downloaded will be aligned. In the way, the annotations become portable to variants of the base texts. It can be illustrated as follows:

\[ \text{text}_{v_1} \leftrightarrow \text{annotations} \rightarrow \text{text}_{v_c} \]

All the annotations stored in PubAnnotation are aligned to the canonicalized version of the base text (indicated by \((v_c)\)). A user may have the same texts but with some variations (indicated by \((v_1)\)). Annotations made to either version of the text become portable to the different versions of text through PubAnnotation.

We experimented the functionality of PubAnnotation with three open corpora with annotations: Genia, AlMed, and Genetag. The base texts of those corpora were collected and preprocessed by the corpus developers using different pipelines. However, we found that all the annotations in the corpora could be successfully aligned to the canonicalized texts, which were taken from PubMed, by simply uploading them to PubAnnotation.

We expect the aligning technology implemented in PubAnnotation to substantially reduce the cost of the community to seek interoperability of literature annotation.
**PDFJailbreak - a communal architecture for making biomedical PDFs semantic**

Alexander Garcia, Peter Murray-Rust, Gully APC Burns, Robert Stevens, Dominika Tkaczyk, Casey McLaughlin, Amaury Belin, Angelo Di Iorio, Leyla Garcia, Célya Gruson-Daniel, Ross Mounce, Andrea Giovanni Nuzzolese, Silvio Peroni, Jeremy Spinks, Boris Villazon-Terrazas, Oscar Corcho, Olga Giraldo, Mike Wabiszewski

1Institute for Digital Information and Scientific Communication, Florida State University, Tallahassee, Florida, USA. 2Department of Chemistry, Lensfield Road, Cambridge, CB2 1EW, UK. 3Information Sciences Institute, Marina del Rey CA 90292. 4The University of Manchester, Oxford Road, Manchester, UK, M13 9PL. 5Interdisciplinary Centre for Mathematical and Computational Modeling, University of Warsaw, Poland. 6Human-Computer Interaction, LIRIS, University of Lyon - France. 7Temporal Knowledge Bases Group, Universitat Jaume I, Castello de la Plana, Valencia, Spain. 8http://hackyouyourphd.wordpress.com/. 9Dept of Biology & Biochemistry, University of Bath, Claverton Down, Bath BA2 7AY. 10Department of Computer Science and Engineering, University of Bologna, Bologna, Italy. 11STLab-ISTC, National Research Council, Rome, Italy. 12ISOCC, Intelligent Software Components, Campo de las Naciones, 28042 Madrid. 13Ontology Engineering Group. Facultad de Informática, Universidad Politécnica de Madrid, Spain.

**Jail breaking the PDF**

Most biomedical literature is only available in PDF form and this contains a wealth of unused data when liberated. PDFJailbreak is a communal project to create a formal flexible infrastructure to extract semantic information from such documents (papers, grants, theses, reports, guidelines etc.). PDFJailbreak covers the workflow from reading raw PDFs (both interactively and high throughput processing) to domain-specific annotations, argumentation, and data extraction.


The functionality of these systems include: (1) extraction and normalization of PDF primitives (characters, paths, and images), (2) reconstructions of blocks (whitespace-separated chunks), (3) zoning and general annotation of blocks, (4) extraction of data from tables and figures, (5) extraction of citations, (6) scholarly-specific annotation of components (e.g., citation typing, bibliographic metadata, indexing of materials-and-methods), (7) linking to DBPedia (http://dbpedia.org/) and other scientific semantic services, and (8) domain-specific analysis (e.g., phylogenetic trees, chemistry, sequences).

In many of these there are alternative approaches, often complementary. For example, in blocking we use: machine learning, iterative parameter optimization, heuristics and crowdsourcing. In contrast, traditional XML offerings (e.g., JATS – http://jats.nlm.nih.gov/) only provides a subset of this and only about 10% of the literature is openly available as JATS. PDFJailbreak makes the information in most modern PDF documents fully accessible, including the detailed interpretation of tables and figures.

The architecture is designed for open community development and favours APIs that can be used at various stages in the workflow. Among the initial uses are citation typing, extraction of phylogenetic trees from diagrams, and bio/chemical structures and reactions.

**ACKNOWLEDGMENTS**

Special thanks to Pablo Mendes, Alexander Constantin, Steve Pettifer and Greg Riccardi.

**REFERENCES**


Text mining for characterizing cells and tissues

Mariana Neves\textsuperscript{1,2}, Alexander Damaschun\textsuperscript{2}, Nancy Mah\textsuperscript{3}, Fritz Lekschas\textsuperscript{2}, Stefanie Seltmann\textsuperscript{2}, Harald Stachelscheid\textsuperscript{2}, Jean-Fred Fontaine\textsuperscript{3}, Andreas Kurtz\textsuperscript{2}, and Ulf Leser\textsuperscript{1}

\textsuperscript{1}Humboldt-Universität zu Berlin, WBI, Berlin, Germany
\textsuperscript{2}Berlin Brandenburg Center for Regenerative Therapies, Charité, Berlin, Germany
\textsuperscript{3}Max Delbrück Center for Molecular Medicine, Berlin, Germany

Regenerative medicine is an important field for translational medical research provided its potential for repair, restoration and replacement of tissues \cite{1}. One of the requirement of regenerative approaches is the well characterization of therapeutic cell populations based on reliable measurement and analysis techniques. The biological scientific literature contains a huge number of citations on the expression of genes/proteins in a variety of cell lines, cell types or tissues, for instance (PMID 20085633): \textit{Pros is never expressed in glial cells}. In this work, we show that text mining can help characterizing cell and tissues and we describe the results we have achieved so far in the scope of the CellFinder database.

The CellFinder database\textsuperscript{1} is a repository of cell research which aims to integrate data derived from many sources, such as literature curation and microarray data. As part of the text mining development, a set of 20 full text documents on human embryonic stem cell \cite{2} and on kidney research have been manually annotated with almost 3,000 gene expression events on cells and tissues. Later, they have been utilized for evaluation and training of supervised learning methods as part of a text mining pipeline \cite{3}, which include document triage, named-entity recognition for a variety of types and event extraction. A first evaluation of the pipeline resulted on the curation of more than 1,800 gene expression events.

CellFinder currently includes more than 4,500 facts on the expression of particular genes in particular cells, derived from more than 800 full text publications. This literature-derived data have been automatically normalized to a variety of ontologies (CL, CLO, FMA, EHDAA2, UBERON) and to the EntrezGene database to allow integration into the database, after previous manual validation of the identifiers. Currently, validated data corresponds to more than 150 and 800 distinct anatomical terms and genes/proteins, respectively.

\textbf{Funding:} This work was supported by the DFG [LE 1428/3-1 and KU 851/3-1].

\textbf{References}


\textsuperscript{1}http://www.cellfinder.org/