Advances in Cognitive Systems 5 (2017) 77-96 Submitted 6/2017; published 12/2017

Learning by Reading: Extending and Localizing Against a Model

Scott Friedman FRIEDMAN@SIFT.NET
Mark Burstein BURSTEIN@SIFT.NET
David McDonald DMCDONALD@SIFT.NET
Alex Plotnick APLOTNICK@SIFT.NET
Laurel Bobrow LBOBROW@SIFT.NET
Smart Information Flow Technologies, 319 N. 1st Avenue, Suite 400, Minneapolis, MN 55401 USA

Rusty Bobrow ROBERT.BOBROW@GMAIL.COM
Bobrow Computational Intelligence, LLC, 20 Draper Avenue, Arlington, MA 02474 USA

Brent Cochran BRENT.COCHRAN@TUFTS.EDU
School of Medicine, Tufts University, Boston, MA 02110 USA

James Pustejovsky JAMESP@BRANDEIS.EDU
Department of Computer Science, Brandeis University, Waltham, MA 02454 USA

Abstract

This paper describes R3 (*Reading, Reasoning, and Reporting*), our system for deep language understanding and extension of mechanistic models. The overall purpose of R3 is to read about biochemical signaling pathways from PubMed Central journal articles and integrate information into its model. Its initial background model of these biochemical pathways is derived from an imported Reactome model of biological pathways, events, complexes, and proteins. We describe some significant issues for semantic parsing in this domain and how R3 uses pre- and post-analysis reasoning to bridge the differences between the semantic information that can be derived from a text and the codified mechanistic information in the curated biomedical database. We also present extensions to relational structure mapping to detect corroboration between the semantic parse and the model and extend the model via analogical inference from the parse. We close with a description of empirical results with R3, including semantic parsing, model extension, grounding entity and event references, and modeling entity behavior using knowledge learned by reading.

1. Introduction

Machine reading does not end with a parse or even with a semantic interpretation of text. When we read to inform ourselves, we use our current model of the world to guide our interpretation of the text, and then we reconcile this interpretation with our model to determine consistency with our prior beliefs and perhaps to accept and incorporate the new information. Our interpretation might corroborate, extend, or conflict with our prior model and perhaps cause us to revise or extend it. We refer to this model-centric activity as *reading with a model*. This is the central goal of our ongoing work on the *Reading, Reasoning, and Reporting* (R3) cognitive system, as part of DARPA's Big

© 2017 Cognitive Systems Foundation. All rights reserved.

Mechanism program (Cohen, 2015). R3 reads articles in molecular biology to extend and revise its models of biological mechanisms, specifically those having to do with signaling pathways.

A central capability – and research challenge – for cognitive systems that read with a model is *localizing* (i.e., recognizing and retrieving) entities and events mentioned in the text when they appear in the prior model, in order to begin the process of reconciliation. Localization lets the system establish a mapping between the interpreted text and the model to enable bidirectional information flow between the model and the text interpretation process. That is, we seek to transfer information about mechanisms gleaned from the text into the model (*interpretation-to-model*), either to extend the model or to identify and annotate conflicts. If localization first establishes correspondence between parts of the model and the text, we can also improve the text interpretation process (*model-to-interpretation*) by making the reading system aware of details about known entities and processes (such as their types and relations to other mentioned entities) so that when they are mentioned only by reference, those references are not overly vague or ambiguous. If known entities and events in the text are not localized correctly within the model, then the interpretation is less successful since new related information is not properly integrated.

Building a system that reads and localizes to a pre-existing biological pathway model that has been developed and curated by domain experts involves many domain-general and domain-specific challenges, including:

- Texts frequently use the same word to reference different types in the model. For instance, "*RAS*" can refer to a protein, a gene, or a larger multi-protein complex, within a single article.
- Texts may describe things at different levels of abstraction than the model. For example, authors frequently talk about the *function* of entities, although the background model may only describe the entity interactions and molecular structures.
- One process or entity may be a component of many larger processes or entities in the model.

Some of these challenges are due to information mismatch between journal articles and domain models, where the model has no ontological categories or relations to represent the level of events in the text. For instance, articles frequently mention *post-translational modifications*, including *phosphorylation* (binding a phosphoryl group to a molecule). Another is the formation of complexes with multiples of the same molecule. *Dimerization* is the binding of two like molecules to form a *dimer*. These and other types of events are present in the forma model, but only implicitly: there are no categories for these events in the formal BioPAX model, so one must compare the reactants and products to infer these events. To rapidly recognize these events mentioned in the text, we must extend the model by describing them explicitly (Section 3.1).

Furthermore, biology articles often describe processes at a *functional* level, but at the time of this work, the background BioPAX Reactome domain model does not represent entity function. In the cell signalling domain, processes and entities are functionally described as being "switched" on or off, and processes or entities can "activate" or "inhibit" other processes or entities. Entities are "activated" when their structure or their bindings enable them to act as catalysts, inhibitors, or scaffolds in other reactions. Activation conditions vary across entities: proteins like MEK and ERK are activated when they are phosphorylated; others are *de*activated when they are phosphorylated;

others are activated when they are dimerized; and so forth. This functional language lets biologists compactly refer to entities with causal affordances without knowing their exact structure.

To capture the specific states of activation for the particular proteins in our model, we needed a source of information about the complexes involving those proteins when they were considered "active" or "inactive." We found our source embedded within the model itself: expert biologists wrote textual summaries about each reaction in the model, and associated these summaries with the reaction entities. For example:

- SOS1 is the guanine nucleotide exchange factor (GEF) for RAS. SOS1 *activates* RAS nucleotide exchange *from the inactive form* (bound to GDP) *to an active form* (bound to GTP).
- EGFR phosphorylates PLC-gamma1, thus *activating* it.
- *Activated MAPK proteins* negatively regulate MAP2K1:MAP2K2 heterodimers . . .

These examples of reaction summaries show how the language used for human consumption conveys the functional states of their primary participants, rather than their structural changes, which is described by the model. R3 learns functional knowledge – which it needs for localizing functional references in articles – by parsing these summaries and then analogically transferring knowledge into the associated events in its domain model. When subsequent texts describe entity function, R3 can identify the corresponding structures in its extended model.

R3 integrates deep semantic parsing, ontology mapping, interpretation-to-model structure mapping, and functional reasoning. Semantic parsing (Section 3.2) lets R3 extract precise descriptions and determine entity types from local lexical context. R3's ontology mapping (Section 3.3) lets it transfer its semantic interpretation into other ontologies to identify any and all corroborating events and entities. R3 extends structure mapping methods (Section 3.4) to support wide-scale event recognition and retrieval. Finally, R3's mechanism-level reasoning (Section 3.7) lets it reason about *functional* factors – such as what it means when an article describes an entity as *active* – despite lacking direct functional knowledge in its initial domain model.

Section 2 outlines the problem of extracting and recognizing biological events and interactions from text, focusing on challenges for natural language understanding. Section 3 describes the R3 approach to meeting these challenges. Section 5 describes empirical evidence of our claims that (1) R3 efficiently and robustly processes large bodies of text, (2) R3's learning by reading improves its accuracy of *subsequent* learning by reading, and (3) R3 can use information learned by reading to answer new types of queries that were not supported by its initial model. We close with a discussion of future work for R3.

2. Reading in the Biology Domain

Biomedical research articles are written to be read by other professional biologists who are presumed to have the requisite technical background. The brief mention of a well-known mechanism ("*RAS/RAF/MEK/ERK Pathway*") is sufficient to evoke all of the details of the mechanism in the mind of the reader. This lets them effortlessly fill in information gaps that cannot be supplied by standard discourse techniques ("*activated upon GTP loading and deactivated upon hydrolysis of*

GTP to GDP" – loaded onto or hydrolyzed from what?). We need knowledge sources that let our systems do this as well.

Like other authors, biologists must keep their articles within length limits, so they use compaction techniques such as describing events using nominalized verbs and packing information into them as prenominal modifiers, such as "*EGFR and ERBB3 tyrosine phosphorylation*" and "*mitogen-induced signal transduction*." This changes the usual grammatical cues (such as one would use on newswire text) and requires knowledge-rich analysis techniques if parses are to be accurate.

A further property of biomedical text is that logically-related information is usually distributed across multiple sentences. In a typical example, the classification of the sites is given in the first sentence and their identity in the second, as in: "*We observed two conserved putative MAPK phosphorylation sites in ASPP1 and ASPP2. The ASPP1 sites are at residues 671 and 746, and the ASPP2 sites are at residues 698 and 827.*" In R3, we have enhanced our discourse history to combine information from both sentences into a single, logically complete, representation that specifies the binding sites on ASPP1 and ASPP2 where MAPK catalyzes phosphorylation.

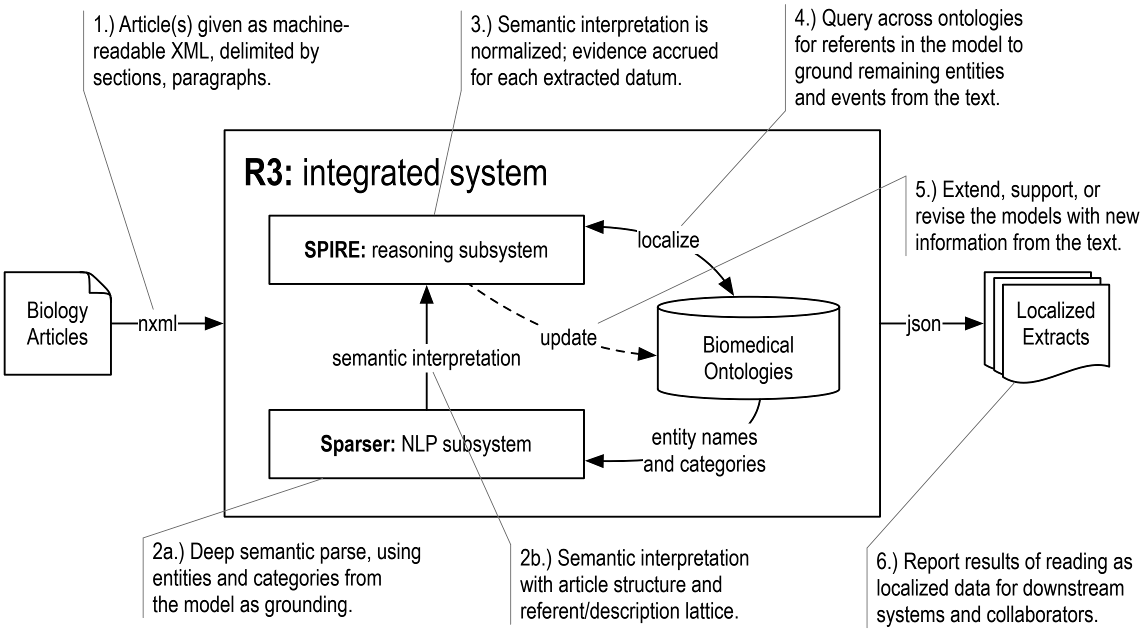
1.) Article(s) given as machine-readable XML, delimited by sections, paragraphs.
3.) Semantic interpretation is normalized; evidence accrued for each extracted datum.
4.) Query across ontologies for referents in the model to ground remaining entities and events from the text.
R3: integrated system
5.) Extend, support, or revise the models with new information from the text.
SPIRE: reasoning subsystem
localize
Biology Articles
nxml
update
Biomedical Ontologies
json
Localized Extracts
semantic interpretation
Sparser: NLP subsystem
entity names and categories
2a.) Deep semantic parse, using entities and categories from the model as grounding.
2b.) Semantic interpretation with article structure and referent/description lattice.
6.) Report results of reading as localized data for downstream systems and collaborators.

Figure 1. The R3 architecture, and the flow of information by which R3 reads articles, updates its mechanism models, and publishes extracted knowledge for human and machine collaborators.

3. Reading and Localizing Scientific Texts

Here we describe R3's architecture and information flow, using Figure 1 to guide our discussion. Before processing complete biology articles, the system bootstraps its domain model (Section 3.1) by inferring and representing events and segmenting its model to facilitate retrieval. It then parses the English textual labels and comments embedded in its biology model (Section 3.2), which were written by biologists. These textual labels and comments describe events (e.g., activation, deactivation) and properties (e.g., active and inactive states) that are not represented directly in the model. To learn by reading, R3 uses structure mapping to match the parsed semantics against its domain model (Section 3.4), uses analogical inference to transfer knowledge into the model (Section 3.6), and then propagates the information throughout the model, where relevant (Section 3.7). The system then reads biology articles using the same semantic interpretation and structure-mapping processes: it parses the article to extract the semantics of entities and events, and it then matches those semantics against events and entities represented in its model. This computes the locale of parsed entities and events within the model for the purpose of bidirectional knowledge transfer. We continue by describing the setup and operation of the domain model and the semantic parser, after which we discuss the post-parse reasoning mechanisms and operations on the domain model.

3.1 Bootstrapping the Domain Model and the Parser

R3 initializes its parser with domain vocabulary and grammar and uses inference rules to optimize and index its domain model. The system uses the UniProt knowledge base (UniProt Consortium, 2008) as a source of protein synonyms to enhance protein recognition during parsing. It maps each protein synonym to a unique identifier to enable cross-indexing in various biological ontologies.

R3 imports OWL domain models specified in Biological Pathway Exchange (BioPAX) (Demir et al., 2010). BioPAX specifies structural information about biochemical reactions (e.g., bindings, phosphorylations, and other interactions), complexes, proteins, catalysis, and reaction regulation. R3 uses domain-specific inference rules to extend the BioPAX domain model with additional structure to explicitly represent causal relations, transport events, post-translational modifications (e.g., phosphorylation), functional information (activation, deactivation), and some key structural molecular categories (e.g., homodimer, heterodimer) that are frequently referred to in text. Much of the enhanced content is *implicitly* described in BioPAX (e.g., a homodimer is identifiable as a complex with two instances of the same protein), but the system detects and explicitly represents this sort of additional structure to facilitate its search and localization during reading.

Finally, R3 uses a graph grammar to segment and index the enhanced BioPAX model into logical contexts. The grammar is equivalent to regular expressions over relational knowledge to describe how to traverse the model and segment it into indexable parts, such as by starting with `biochemical-reaction` entities and traversing `left` and `right` relations to the reactions' input and output molecules, respectively, and then descending recursively through sub-molecular structures via `compound` relations, etc. This ultimately indexes each entity and event in the model into its own logical context, so that the system can efficiently search the model to localize the information it reads, as we describe in Section 3.5.

3.2 Deep Semantic Parsing

The purpose of language analysis in R3 is to extract the semantic content of biomedical texts, which ultimately extend a domain model and provides a standard view of an article's content for downstream reasoners (e.g., Danos et al., 2009). Parsing normalizes syntactic and lexical variation to produce canonical, relational forms. References to entities and relations are also aligned with articles' document structure to facilitate search and context driven inferences. Ultimately, we seek to use the additional specificity of content localized from introductory remarks in papers to help the interpretation process, though this remains a goal for the future.

R3 uses SPARSER for semantic parsing. SPARSER is a rule-based, type-driven semantic parser. Rules succeed only if the types of the constituents to be composed satisfy the type constraints (value restrictions) specified by the rule. SPARSER is also model driven. As described in McDonald (1996), writing a semantic grammar starts with a semantic model of the information to be analyzed along with a specification of all the ways each of the concepts can be realized in the language of the genre (e.g., biomedical research articles). A compiler takes the model and creates a semantic grammar from the realization specifications by drawing on a schematic standard English syntactic grammar. This ensures that every rule in the generated grammar has an interpretation, and thus everything SPARSER can model it can also parse.

SPARSER parses into a referential model, instead of solely into logical forms. The model uses a typed lambda calculus (McDonald, 2000) with top-level predicates from Pustejovsky's (1991) event model, and mid-level ontological concepts of location, time, people, measurement, and change. The lower-level domain-specific ontology represents concrete biological phenomena, structured according to how these phenomena are commonly described in journal articles.

Categories are frames in a conventional knowledge representation, with a specialization lattice that permits the inheritance of realization options, default values, variables (i.e., possible relations), and methods for type-specific reasoning. Individuals (i.e., instances of categories) represent the entities, events, and relationships that are identified when a text is read. SPARSER uniquely identifies each *individual* (i.e., category instance) in its parsing. Every individual with a unique set of property values is represented by a single object (Maida & Shapiro, 1982; McDonald, 2000) and managed by a description lattice that tracks the addition of properties (i.e., binding of role variables).

SPARSER's discourse component resolves pronominal and definite references using a structured history of entity and event mentions. This same facility organizes searches to expand partial descriptions of entities into full ones (frame completion) and in general to link individuals as they appear in different parts of an article. For example, *phosphorylation* events entail an agent protein or molecule, a substrate protein that is phosphorylated, and a site (i.e., residue) where the phosphate is added. A residue is identified by its amino acid and its location on a particular protein. If SPARSER reads about the sites of a phosphorylation and the requisite information is not supplied locally in the text, then we can assume that it is very likely to have been supplied elsewhere in the article, which motivates a search to identify it. Consider this text that compares what happens when a particular drug is or is not used:

> *In untreated cells, EGFR is phosphorylated at T669 by MEK/ERK, which inhibits activation of EGFR and ERBB3. In the presence of AZD6244, ERK is inhibited and T669 phosphorylation is blocked, increasing EGFR and ERBB3 tyrosine phosphorylation . . .*

There are two mentions of the phosphorylation of residue T669 in this text, one in each sentence. The mention in the second sentence (*T669 phosphorylation*) is marked by the sentence post-processor as being incomplete because it does not specify the agent or the substrate. This combination of event type and site is a unique individual stored in the description lattice. The discourse history records that this individual was also mentioned in the first sentence. This is enough to license SPARSER to trace up the structure on the first mention to identify the other properties it has, and to copy over any nonconflicting properties of the first to the second.[1]

3.3 Ontology Mapping

After semantic parsing with SPARSER, R3 localizes and learns from the interpreted events and entities. This requires representing these events and entities using the ontology of R3's domain models, which are presently described in BioPAX. The system must perform *ontology mapping* to re-represent SPARSER's output using the BioPAX ontology.

R3 performs ontology mapping using manually-created forward-chaining rules in its internal SPIRE reasoner (shown in Figure 1, center). It runs these rules exhaustively, binding each rule's left-hand side to relational knowledge in the SPARSER interpretation, and asserting the corresponding right-hand side in the BioPAX ontology. The article and the model may represent events at different granularity, so the SPIRE ontology mapping rules generate new symbols to represent the semantics at different levels of description. For example, if R3 reads, *X phosphorylates Y and Z*, it must create two *separate* phosphorylation events for Y and Z, with X as the `agent` role for both, in order to localize them independently: these may or may not correspond to the same event in the model.

3.4 Enhanced Structure Mapping

R3 uses SPIRE's *structure mapping* – constrained graph-matching over relational representations based on Gentner's (1983) theory of analogy and similarity – for two localization operations:

1. **Retrieval**: Given a *probe* description extracted from text, recognize and retrieve all potentially corresponding entities and events from the model.
2. **Transfer**: Given a semantic parse and a semantic description of an entity or event from the model, match the two and suggest the transfer of entities and relations into the model.

The core structure-mapping operation involves computing one or more *mappings* between two representations. Each mapping is a maximal common subgraph solution between the two representations, where each entity is a node, each relational assertion is a node, and each relation argument is a position-labeled edge. Following Forbus et al.'s (2017) computational model, each of SPIRE's common subgraphs describe *correspondences* (i.e., tuples describing isomorphic nodes across graphs), a *score* that rates the quality of the correspondences, and *inferences* describing complements of the common subgraphs (i.e., nonisomorphic relations and entities) that can be projected from one graph to the other. Structure-mapping inferences are not necessarily deductively sound, since they are based solely on structural similarity; however, in previous work, we have shown that these inferences can be practically used to revise beliefs and models (Burstein, 1988; Friedman et al., 2012). As we illustrate below, structure-mapping inferences are practical for extending the model while reading. Structure mapping reduces the space of legal mappings – thus making the problem

[1] The two eventualities differ in their existential status. The tense in the first sentence indicates that the phosphorylation occurs. The second states that it is blocked.

more tractable than traditional maximal common subgraph optimization problems – by adding two additional constraints:

- *Tiered identicality*: Category nodes can only correspond to other category nodes with identical categories and relation nodes can only correspond to relation nodes with identical predicates. Structure mapping allows symbol arguments (e.g., referring to entities or events) to correspond to nonidentical symbols.
- *Parallel connectivity*: If two relation or category nodes correspond, their arguments must correspond, in sequence. Applied globally, if two nodes correspond, then so must their reachable subgraphs.

These two constraints drastically decrease the solution space, so SPIRE's greedy maximal common subgraph algorithm is plausible and effective. Guaranteeing an optimal solution is out of scope for R3 due to tractability: the decision problem for maximal common subgraph is widely known to be NP-complete. As we demonstrate below, a greedy algorithm produces practical results for R3's model localization.

Computational models of structure mapping have been used widely to compute analogies across domains, identify structural similarities, and transfer knowledge. However, event recognition and localization require much tighter matching: R3 should not retrieve events that are *similar* to an event described in a scientific article and it should retrieve descriptions that could refer to the same events. We call this structure-mapping setting *recognition* rather than the more traditional setting of analogy. Although recognition differs from analogy in crucial ways that we mention below, structure mapping still offers important benefits for flexible localization from reading. Specifically, structure mapping supports *partial matches*: if the article mentions something not in the model (e.g., a new reactant within a known reaction), structure mapping will identify relevant candidates for model expansion.

Typically, texts will talk about specific proteins playing roles in various reactions or causal processes when, in fact and in the underlying model, these proteins are in various states of binding with other, unmentioned, molecules in complexes. Hence, it is critical that the structure matching be able to identify these proteins within these larger complex structures as either reactants, products or catalysts when relating the extracted text semantics to the model. Adapting structure mapping to this recognition task included three extensions:

Identifier intersection. Like any graph-matching optimization algorithm, if structure mapping can add a correspondence to its mapping, it will. This maximality bias yields higher-scoring mappings, but it can also produce erroneous results in an entity- or event-recognition setting. For instance, without additional constraints, the event *"SOS1 activates RAS"* will map nearly perfectly to the event *"MEK activates ERK"*; however, this is undesirable for coreference and recognition.

In its recognition setting, R3 computes *a priori* correspondence allowances, so that a parsed individual can only correspond to a model individual if their identifiers (e.g., list of name strings) intersect. This allows the parsed individual with namestrings {"SOS1," "SOS1_HUMAN," "SOS-1"} to correspond to the model entity with namestrings {"UniProt:Q07889 SOS1," "SOS1," "Son of sevenless protein homolog 1"} due to the "SOS1" intersection. This substantially increases recognition accuracy and reduces the search space for mappings.

Dependency constraints. Adding constraints on entities during mapping – such as only permitting two phosphorylation events to match if the phosphorylat*ed* entities also match – reduces erroneous mappings. The descriptions *"phosphorylated ERK"* and *"phosphorylated RAF"* describe the same property (i.e., phosphorylation modifications) but with nonintersecting object roles. The phosphorylation properties are therefore incompatible for recognition purposes. We use a domain-general mechanism for specifying and mapping with dependencies, but R3 uses domain-specific rules for asserting these dependencies, e.g., properties depend on their `object` role-filler. During the mapping process, when the `object` role-filler is selected for the mapping, the events are added to the search space.

Category and predicate subsumption. If R3 reads *"SOS1 activates RAS nucleotide exchange"*, it will assert `(activates-process txt-SOS1-ent txt-RAS-NE-ent)` to describe this relationship between the SOS1 referent `txt-SOS1-ent` and the nucleotide exchange referent `txt-RAS-NE-ent`.[2] However, in the corresponding model reaction, R3 has described this relationship with greater specificity, with the expression `(catalyzes-process-as-component mdl-SOS1-ent mdl-RAS-NE-ent)`, as SOS1 is a subcomponent of the catalyzing complex.

In R3's relational hierarchy, the `activates-process` relation from the text is a superordinate relation of the `catalyzes-process-as-component` relation in the model. SPIRE's structure-mapping algorithm supports nonidentical relation matches and nonidentical category matches albeit at a diminished score, based on the Jaccard index of their *superordinate locales*, which we define as the set of superordinate predicates or relations reachable in an upward walk of constant length k. The Jaccard index between locales is computed as $\frac{|(A \cap B)|}{|(A \cup B)|}$, so it is 0.0 (i.e., not allowed) for nonintersecting locales, 1.0 for identical locales (i.e., identical predicates or categories), and within the interval $(0, 1)$ for nonidentical predicates with intersecting locales. For R3, we use a locale distance of $k = 3$, including the relation or category itself and all relations or categories within two upward traversals. The k value is sensitive to the depth and specificity of the ontology.[3]

3.5 Retrieval and Localization

After mapping the extracted information – e.g., a description of an entity or process – into BioPAX, R3 localizes it by retrieving all matching entities and processes in the model. R3 uses a two-stage similarity-based retrieval algorithm, similar to MAC/FAC (Forbus et al., 1995): given a probe (i.e., the process or entity description) and a library (i.e., a set of entity and process descriptions from the model), the first stage is an efficient feature vector dot-product between the probe and each context to filter low-similarity descriptions, and the second stage is the structure-mapping recognition algorithm described above. The result is a similarity-ranked subset of the model library.

R3 uses *a priori* structure-mapping constraints to ensure that the explicitly-described entities and relations are in the mapping (e.g., for "MEK-directed phosphorylation of ERK," the MEK, ERK,

[2] We have renamed the symbols here for the sake of clarity.

[3] Other analogy systems (e.g., Falkenhainer, 1988) match nonidentical predicates and categories as a post-process. This differs from SPIRE's inclusion of nonidentical predicate matches in the initial search for correspondences.

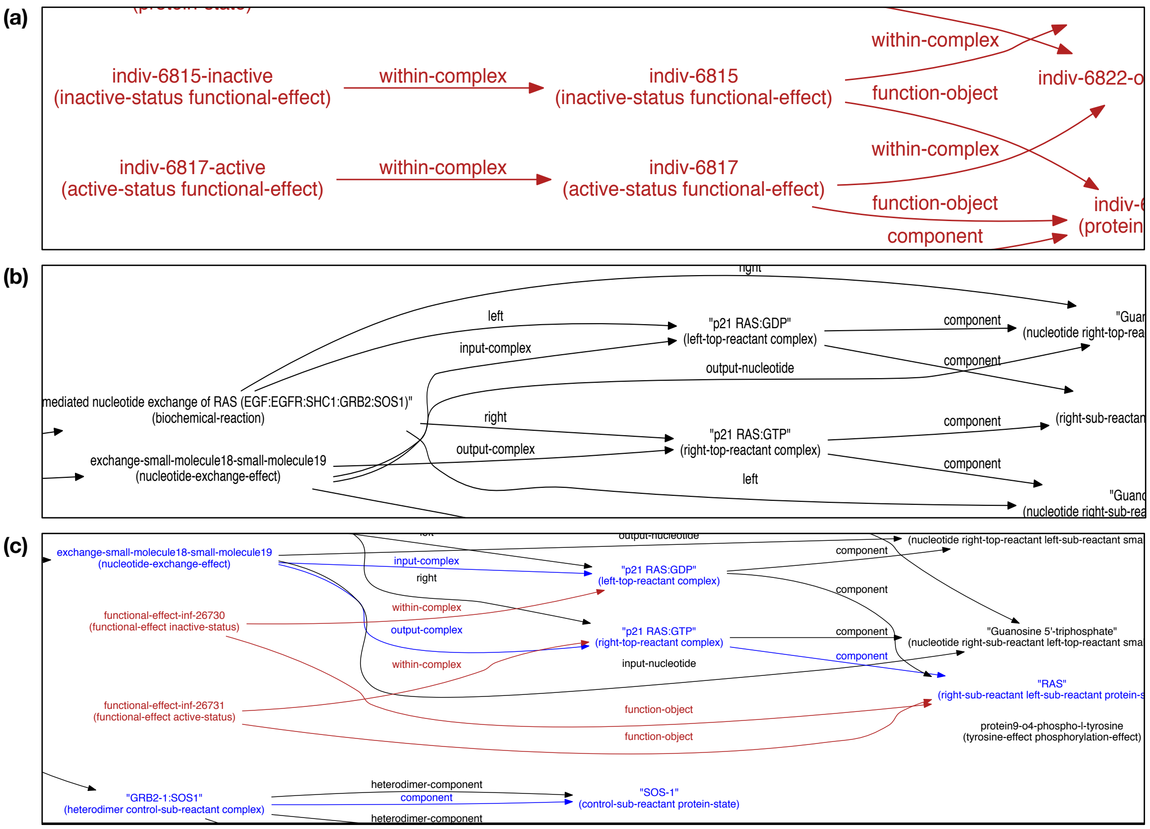
(a)
indiv-6815-inactive
(inactive-status functional-effect)
within-complex
indiv-6815
(inactive-status functional-effect)
within-complex
function-object
indiv-6822-o
indiv-6817-active
(active-status functional-effect)
within-complex
indiv-6817
(active-status functional-effect)
within-complex
function-object
component
indiv-6
(protein
(b)
right
left
input-complex
"p21 RAS:GDP"
(left-top-reactant complex)
component
"Guar
(nucleotide right-top-rea
component
output-nucleotide
mediated nucleotide exchange of RAS (EGF:EGFR:SHC1:GRB2:SOS1)"
(biochemical-reaction)
right
"p21 RAS:GTP"
(right-top-reactant complex)
component
(right-sub-reactan
output-complex
exchange-small-molecule18-small-molecule19
(nucleotide-exchange-effect)
component
left
"Guano
(nucleotide right-sub-rea
(c)
output-nucleotide
(nucleotide right-top-reactant left-sub-reactant sma
exchange-small-molecule18-small-molecule19
(nucleotide-exchange-effect)
input-complex
right
"p21 RAS:GDP"
(left-top-reactant complex)
component
component
functional-effect-inf-26730
(functional-effect inactive-status)
within-complex
output-complex
"p21 RAS:GTP"
(right-top-reactant complex)
component
"Guanosine 5'-triphosphate"
(nucleotide right-sub-reactant left-top-reactant sma
within-complex
input-nucleotide
component
"RAS"
(right-sub-reactant left-sub-reactant protein-s
functional-effect-inf-26731
(functional-effect active-status)
function-object
function-object
protein9-o4-phospho-l-tyrosine
(tyrosine-effect phosphorylation-effect)
"GRB2-1:SOS1"
(heterodimer control-sub-reactant complex)
heterodimer-component
component
heterodimer-component
"SOS-1"
(control-sub-reactant protein-state)

Figure 2. (a) Subset of BioPAX semantic parse; (b) subset of existing event in model; (c) subset of extended event in model: model complement (black), isomorphism (blue), and parse complement (red). After parsing text, R3 maps the semantic parse into the model ontology (a) and uses the retrieved portion of the model (b) to compute a mapping (c). R3 uses corroborating, isomorphic structure (blue) as a scaffold to transfer the complement semantic structure from the parse (red) directly into the model to extend it.

and phosphorylation event are all required); otherwise, the mapping operation terminates with a score of zero. The system thereby identifies and ranks portions of the domain model according to their structural similarity to the extracted knowledge. As we show in Section 5, this localization approach recognizes entities and processes with high precision and recall; however, it does not account for context and causal locality, which we revisit when we discuss future work in Section 6.

3.6 Updating the Model

After R3 interprets text (Section 3.2), maps it into BioPAX (Section 3.3), and identifies relevant portion(s) of the model (Section 3.5), it updates the model with the interpretation. The update operation is based on structure-mapping inferences (outlined in Section 3.4). This transfers relational structure from the semantic parse into the model, using the maximal common subgraph of the two as a scaffold.

An example structure-mapping inference operation is displayed graphically in Figure 2, which illustrates the relationship between the semantic interpretation, the model, their isomorphic subgraph, and the inferred (i.e., transferred) subgraph. The semantic representation from the text (Figure 2a) and the corresponding portion of the BioPAX model (Figure 2b) are the inputs to structure mapping, which computes the maximal common subgraph (shown in blue in Figure 2c, using the symbol names from the model). The complement (i.e., nonisomorphic portion) of the semantic interpretation provides structure-mapping inferences (shown in red in Figure 2c) that R3 transfers into the model. Contentwise, the interpretation in Figure 2 corresponds to the text:

> SOS1 activates RAS nucleotide exchange from the inactive form (bound to GDP) to an active form (bound to GTP).

The interpretation includes four statements about a nucleotide exchange and an RAS protein,

```
(isa indiv-6822 nucleotide-exchange)
(input-complex indiv-6822 indiv-6822-in-complex)
(output-complex indiv-6822 indiv-6822-out-complex)
(name indiv-6690 "RAS")
```

as well as others. Structure mapping computes that the nucleotide exchange event parsed from the text is isomorphic to a known nucleotide exchange event in the model. Also, the protein, input, and output complexes of this event in the text are isomorphic to the respective protein, input, and output complexes of the event in the model. However, the semantic interpretation also contains novel (i.e., nonisomorphic) information that the RAS is *inactive* in the input complex of the nucleotide exchange and is *active* in the output complex:

```
(isa indiv-6815 inactive-status)
(function-object indiv-6815 indiv-6690)
(within-complex indiv-6815 indiv-6822-in-complex)
(isa indiv-6917 active-status)
(function-object indiv-6817 indiv-6690)
(within-complex indiv-6817 indiv-6822-out-complex)
```

The isomorphic structure (shown in blue in Figure 2c) provides a scaffold to transfer this novel information (shown in red in Figure 2c) into the model, importing new categories and relations describing existing entities and events in the model, and generating new symbols for novel events and entities.

For our evaluation described in Section 5, the system only transfers inferences that describe protein function and behavior, such as active and inactive forms of proteins, and processes that activate and deactivate proteins. In the case of Figure 2, R3 learned that p21 RAS is active when bound to GTP and inactive when it is bound to GDP.

3.7 Propagating Learned Knowledge

After transferring the inferences into its model as described in the previous section, R3 propagates information throughout the model and makes secondary inferences. At present, the system only propagates information about protein function and activation, but we are expanding the scope of propagation in ongoing work. For instance, when the system learns that a certain protein structure is active or inactive, it revises all relevant reactions and super-complexes in the model. It then detects *changes* in active status in each relevant reaction, and labels status-changing reactions as activation or deactivation processes. Updating the model with one fact can thereby cause broad ripples of secondary inferences throughout the events and entities in the model.

Updates to the model change the relational structure of the entities and reactions that R3 uses for localization, as described in Section 3.5. For example, after updating its model to describe p21 RAS:GTP as *active*, the system will retrieve p21 RAS:GTP when localizing a mention of "active RAS," and it will localize any reaction that produces p21 RAS:GTP (such as RAS nucleotide exchanges) when it localizes mentions of "RAS activation." In Section 5, we show how this process of model extension drastically increases R3's precision when localizing subsequent mentions from text. In this fashion, learning by reading improves the system's *subsequent* learning by reading.

4. Related Work

Biology has been a focus of NLP research for decades as part of the broader research area of bioinformatics. Until recently, the NLP field has concentrated its efforts and challenges on the problem of recognizing proteins and other bio-entities (Funk et al., 2014), with some early efforts at extracting larger-scale, protein-protein relations from PubMed abstracts or full articles (Hunter et al., 2008; Pustejovsky et al., 2002). Given the community focus on challenge problems and annotated data (`bioNLP.org`), curated article collections have also been developed, notably the CRAFT corpus (Cohen et al., 2017).

But with the advent of DARPA's Big Mechanism program (Cohen, 2015), there has been a marked growth in interest in information extraction from full Biomedical articles at large scale (tens to hundreds of thousands of articles). The Reach system (Valenzuela-Escárcega et al., 2015, 2017) is widely used. Like SPARSER, Reach is primarily rule driven, using a cascade of simple, easily understood rules (Valenzuela-Escárcega et al., 2016). DRUM (Allen et al., 2015) is another rule-driven system that reliably extracts entities and events from biomedical texts. It is closer to SPARSER in that it is a modification of the domain-independent, grammar based TRIPS parsing technology (Allen et al., 2008). There are also several efforts to apply machine learning techniques to the problem of recognizing events and event-event relationships in professional biomedical articles, notably at Microsoft (Parikh et al., 2015; Poon & Vanderwende, 2010) and ISI (Garg et al., 2016).

While some of these other large-scale reading efforts have comparable skill in event and entity extraction (e.g., Reach and DRUM), only R3 goes further and localizes the extracted information within a domain model. And while there are other systems using structure mapping for machine reading – such as Learning Reader (Forbus et al., 2007), which uses structure mapping for offline rumination – R3 is the only system, as far as we are aware, that uses structure mapping for online localization (i.e., retrieval of model components) and transfer to extend the model.

5. Empirical Evaluation

Here we describe in turn three experiments with R3 to support the claims that (1) its parser extracts scientific knowledge from text reliably and efficiently (Section 5.1), (2) its reading operations extract information that improves subsequent reading (Section 5.3), and (3) extracting information lets the system answer qualitatively different types of queries than with only its initial domain model (Section 5.4).

5.1 Evaluating Breadth and Efficiency of Information Extraction

We evaluated R3's semantic parser and its ability to extract information, filter irrelevant information (i.e., entities or events not in the domain model), and merge duplicate information against 17,209 biology articles from PubMed Central. This supports our claim that the system can efficiently and robustly perform deep semantic analysis.

We configured R3 to extract information about post-translational modifications such as phosphorylation and ubiquitination reactions, as well as positive and negative regulations of processes, and increases or decreases in molecule concentrations. Other information – including binding events, indirect causal relations, translocation events, transcription events, and more – was also parsed and analyzed with respect to the domain model. The extractions were output as JSON structures that contained the event and descriptions of all arguments (e.g., proteins, genes, cellular regions) to the event. Additionally, the system used epistemic filtering to ignore historical, hypothetical, or negated statements, in order to focus on positive information.

R3 read all of the 17,209 articles in 67 minutes on a MacBook Pro 2017, running five simultaneous parallel Common Lisp (SBCL) processes, resulting in an average speed of 0.23 seconds to process an article on the laptop. The first four threads each processed 3,500 articles in 67 minutes, or slightly less than 1.15 seconds per article, per process. In total, the system extracted 812,266 semantic descriptions of the targeted events, across all sections of all papers. This includes events that were duplicate mentions in a single article, since multiple sentences often refer to the same event. Since we do not have a human expert's gold standard to judge precision and recall for R3 on these 17,209 documents, this experiment does not provide evidence of the system's extraction accuracy or ability to reason with the parsed output. That is the purpose of two additional experiments.

5.2 Example: Localization in a Pathway

We now describe and plot an example of R3's localization in a small subset of the entire domain model to clarify the process, the representations, and the result. Figure 3 shows the events and top-level reactants in the "RAF/MAP Kinase Cascade" subset of the system's background model. This conforms to a single "pathway" but clearly illustrates multiple causal chains and feedback cycles.

Consider a sentence given to R3: *"RAS activation ultimately promotes the phosphorylation of downstream effectors MAPK3 and MAPK1."* The system parses this sentence into lexical semantics and then maps its parse into an extended BioPAX ontology. From the BioPAX description, R3 identifies three candidates for localization, and automatically plots them in Figure 3:

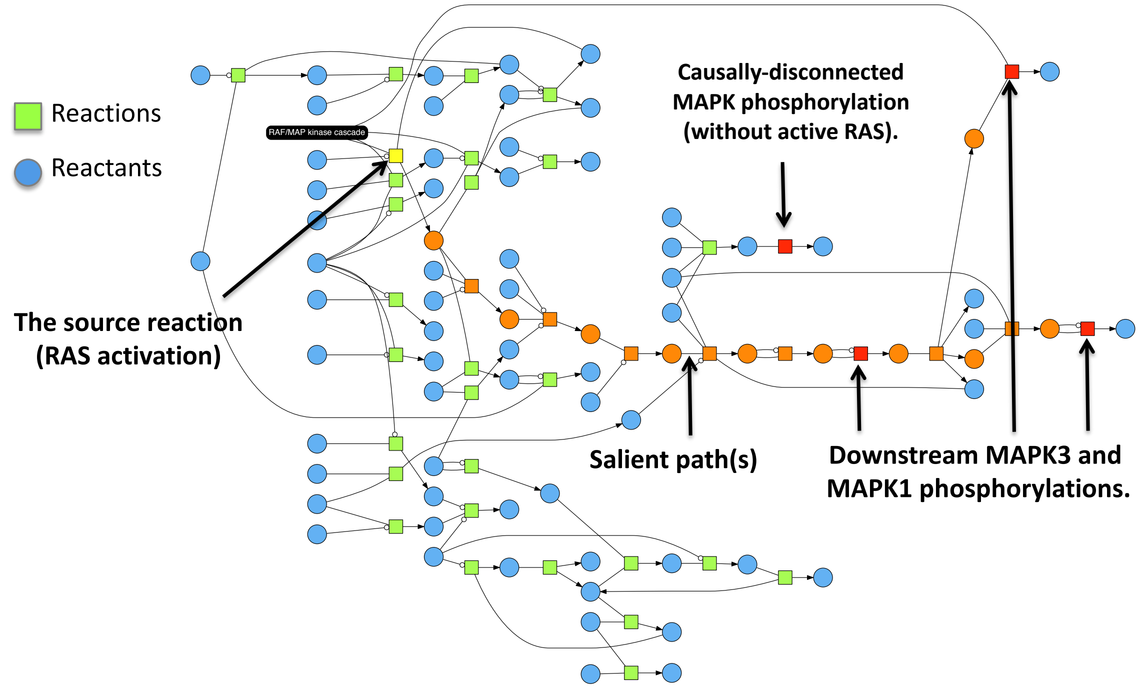

Figure 3. Localizing: "RAS activation ultimately promotes the phosphorylation of downstream effectors MAPK3 and MAPK1". Within the RAF/MAP Kinase Cascade pathway, R3 localizes two mentions of events in the same sentence: (1) a source event (yellow square, upper left in figure) and (2) three consequent events (red squares, middle right in figure) described as "downstream" in the text. R3 uses the causal path (orange) to exclude a disconnected event.

1. Reaction where RAS is activated (yellow square).
2. Reactions that phosphorylate MAPK1 (red squares).
3. Reactions that phosphorylate MAPK3 (red squares).

Note that the sentence relates RAS activation and MAPK1/3 phosphorylation with *"promotes"*. The system uses this causal relation to plot a *causal path* through its model from the promoting event to the promoted events, rendering this in orange in Figure 3.

Also note that one MAPK phosphorylation (i.e., red square) is not reachable from the RAS activation in the model. This phosphorylation of MAPK is RAS-independent; the sentence is referring to the other two phosphorylations (i.e., the red squares connected via the orange path). This exemplifies localization of multiple events – and a causal path connecting them – in a single sentence. We believe these operations are fundamental for a machine to identify corroboration, extension, or conflict of complex scientific texts with complex domain models.

5.3 Evaluating Localization

Next, we demonstrate that R3 can learn functional knowledge by reading, and that this knowledge improves the system's subsequent ability to localize extracted knowledge as it reads. For this experiment, we used the entire "Signaling by EGFR" subset of the open-source, peer-curated Reactome pathway database.[4] Reactome pathway models describe reactions, reactants (complexes, proteins, and other molecules), catalysis and regulation relations, and protein modifications (e.g., phosphorylation, ubiquitination). The "Signaling by EGFR" Reactome subset contains 128 biochemical reactions and 911 molecules (proteins, complexes, small molecules, and other physical objects).

R3 parsed summaries (i.e., multi-sentence descriptions) and display-names (i.e., labels) of reactions that refer to molecules as "active," "inactive," "stimulated," or "activated," or alternatively that refer to "activation" or "activating" a protein. These mentions of protein activity describe functional knowledge, which the BioPAX model does not represent natively. Since the summaries and display-names are related *directly* to a corresponding reaction in the model, there is no need for the system's localization step; it simply maps the parser's semantic interpretation directly against the reaction in the model, and updates the model as described in Section 3.6 and Section 3.7. R3 thus reads textual passages embedded within its model to extend the model itself. We refer to this automatic process as *bootstrapping* the system with functional knowledge.

To qualify the effect of R3's bootstrapping on its ability to localize extracted information, we ran the system's localization operations on an article and compared the f-measure before and after. In this article, R3 extracted six mentions of biochemical processes that had correspondences in the domain model: three activations of ERK, one activation of MEK, one MEK-ERK association, and one MEK-directed phosphorylation of ERK. Some entities and events in the paper were *not* in the model, so these did not count against the system's recall.

Before its functional knowledge from bootstrapping, R3 could not localize these *activation* mentions – since the system had never encountered this type of event – but it properly localized the rest, so it scored a precision of 1.0 and a recall of 0.33. After bootstrapping by reading about activation and deactivation, the system scored an average recall of 1.0, and its average precision dropped slightly to 0.94 (the system incorrectly retrieved a dimerization event when localizing MAPK activation). Similarly, for localizing eight distinct molecules mentioned by the article, two of which were described as "active," R3 scored an average precision and recall of 1.0 and 0.75 without bootstrapping, and a precision and recall of 1.0 and 1.0 with bootstrapping. This provides preliminary evidence that the system's process of bootstrapping through learning by reading improves its ability to localize extracted knowledge during subsequent reading.

5.4 Demonstrating Learned Knowledge

In addition to using its learned knowledge to improve model localization, R3 can display the functional knowledge that it learned by reading. Figure 4 shows a graph generated by the system to describe the function – including activation, deactivation, and event behavior – of the MAPK protein. Before reading, R3 had no concept of "active" or "inactive" RAS/RAF/MAP2K/MAPK, since this functional knowledge is not modeled in the structural BioPAX domain model.

The system parsed English summaries and display-names (i.e., descriptive labels) written by biologists about 17 reactions that refer to molecules as "active," "inactive," "stimulated," or "activated," or that describe events such as "activation" of a protein or how a protein "activates" another.

[4] The BioPAX OWL files are downloadable via the pathway browser http://www.reactome.org/PathwayBrowser/.

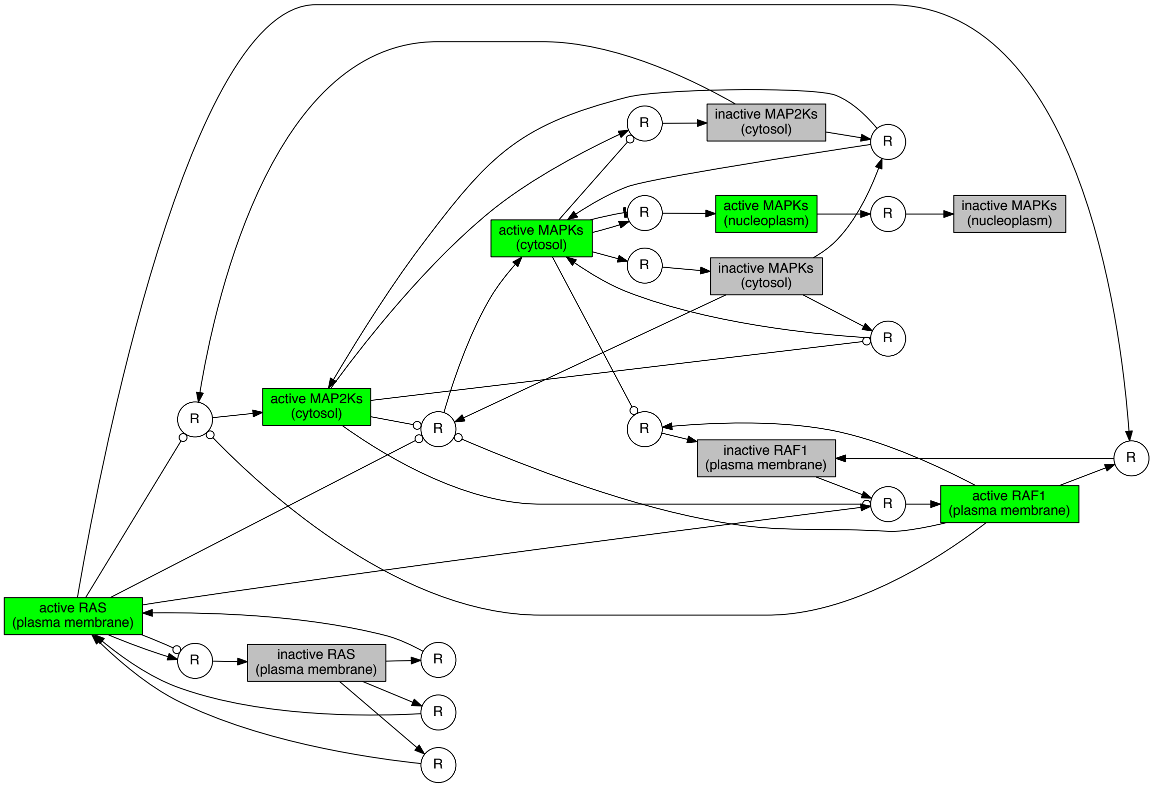

Figure 4. R3-generated graph that explains the activity (i.e., function) and interrelated event structures of RAS, RAF1, MAP2K (MEK), and MAPK (ERK), including activations, deactivations, and translocations of the entities.

Reading the text and analogically transferring the content into its domain model, as described in Section 3, extends R3's model with additional knowledge about the active forms and the *function* of the described proteins.

After reading, the system is able to describe the functional event structure of MAPK – relative to activating components RAS, RAF, and MAP2K, and including its translocation to the nucleus – which is an accurate representation of MAPK function in EGFR signaling. This is shown in Figure 4, where "R" nodes are reactions in the model, triangular arrowheads indicate input and output reactants of each reaction, circular arrowheads indicate that the molecule catalyzes the reaction, and tee arrowheads indicate that the molecule negatively inhibits of the reaction in a regulatory role. Each active and inactive form of the proteins plotted in Figure 4 correspond to structural configurations of of the individual proteins that R3 learned via reading to be active or inactive. The structural properties are not shown in Figure 4, but the system's model of active RAS is GTP-bound, its inactive RAS is GDP-bound, and its active RAF and MAP2K and MAPK are phosphorylated at specific sites. This automatically-generated event structure concisely summarizes the well-known RAS-RAF-MAP2K-MAPK activation cascade.

5.5 Summary of Claims and Results

Our first experiment (Section 5.1) demonstrated the efficiency and robustness of R3's textual parsing components, parsing 17,209 articles in just over an hour, or 0.23 seconds per article, extracting 812,266 semantic descriptions of mentioned entities and events. This is state of the art efficiency, which supports our claim that the system can efficiently extract information; however, its localization operations (described in Section 3.5) are substantially slower than the parsing subsystem, and its duration is directly proportional to the size of the model. R3 uses name-based indexing to quickly filter down possible entity matches, but the graph-matching operations (described in Section 3.4) still contribute to the duration.

The second experiment (Section 5.3) characterized the precision and recall of R3's localization before and after the system reads comments written by biologists. Since the system accrued information about new types of events and states (activation, deactivation, active state, inactive state), its recall of localizing events from an article nearly tripled when it first read biologists' comments in the model. Although R3's recall increased nearly threefold, its precision dropped 6% after learning by reading due to an error during model propagation (see Section 3.7). These errors could contribute to subsequent errors in learning by reading, but the future error rate and the possibility of human correction are areas of future work. Precision and recall both considered, the system's nearly threefold recall improvement supports the claim that its learning by reading improves its accuracy of subsequent learning by reading.

The third experiment (Section 5.4) characterized R3's ability to construct new types of explanations after it read about previously-unseen types of events. Specifically, the system plotted proteins' related active and inactive states in a signaling pathway. This provides clear evidence that R3 can use information it learns by reading to answer new types of queries that the initial model could not.

6. Plans for Future Work

We have shown that R3 reads scientific texts written by experts, localizes extracted information within its complex domain model, and improves its model by reading to facilitate future reading. Our initial results are encouraging, but important future work remains.

The system presently uses semantic similarity of a single semantic parse to localize events in the model, but a single mention, e.g., "*SOS activates RAS*," may match a dozen specific reactions. Identifying the referenced subset of reactions requires using context of the surrounding text. We are implementing a *causal relevance* measure for ranking localization candidates by model proximity to previous high-confidence localization targets. This assumes that biology articles describe causally-related events and entities rather than unrelated events and entities, which holds true in our experience.

We anticipate using our R3 approach on much larger domain models. This may present tractability challenges during localization, since the system must match events and entities from its semantic parse to find all referred-to events and entities in the model. Denser indexing of the model, or using causal relevance (mentioned above) as a hard search constraint may maintain tractability within massive models, but this needs empirical validation. Also, R3 presently only *extends* the model with new information, but learning by reading also involves detecting and reconciling conflicts. Important near-term future work on our system will enable it to automatically identify possible conflicts in these extensions and then pose possible resolutions to these conflicts.

Finally, we must characterize R3's pattern of errors over time as it accumulates information. Other learning by reading systems, such as NELL (Carlson et al., 2010), receive negative human feedback over time to reduce error. While NELL still receives negative human feedback after hundreds of iterations, the amount of human correction decreases over time (Mitchell et al., 2015). We believe this human feedback evaluation is valuable for R3, but the concepts of *correct* and *incorrect* may be too coarse for dynamic scientific domains where conflicting expert models must be held under consideration simultaneously.

7. Conclusion

Our ongoing work on the R3 system addresses the problem of learning by reading with a model-centric approach, since reading scientific texts requires substantial background knowledge, and vast third-party domain models can serve this purpose. This paper focused especially on localizing the text within the model via constrained graph-matching, and then using analogical inference to marshal information to and from the model locale. The system reads with a model in this fashion, and it includes relevant advances in semantic parsing and structure mapping to accurately extract information and localize it within a large third-party domain model.

We described evaluations of R3's parsing, its model localization, and its explanation of protein function using the information that it learned by reading in conjunction with its domain model. These evaluations empirically support our claims that (1) R3 efficiently extracts model-relevant information, (2) the system's learning by reading improves its accuracy of subsequent learning by reading, and (3) it can use information learned by reading to answer new types of queries that were unanswerable in its initial domain model. We are encouraged by the success of our approach to date, but we also see areas that are ripe for improvement and we look forward to tackling these and other challenges.

Acknowledgements

This work was supported by Contract W911NF-14-C-0109 with the Defense Advanced Research Projects Agency and the U. S. Army Research Office. Approved for public release, distribution unlimited. The views expressed are those of the authors and do not reflect the official policy or position of the U. S. Department of Defense or the U. S. Government.

References

Allen, J., de Beaumont, W., Galescu, L., & Teng, C. M. (2015). Complex event extraction using DRUM. *Proceedings of the Fourteenth Workshop on Biomedical Natural Language Processing* (pp. 1–11). Beijing, China: ACL.

Allen, J. F., Swift, M., & De Beaumont, W. (2008). Deep semantic analysis of text. *Proceedings of the 2008 Conference on Semantics in Text Processing* (pp. 343–354). Venice, Italy: ACL.

Burstein, M. H. (1988). Combining analogies in mental models. In D. H. Helman (Ed.), *Analogical reasoning,* 179–203. Dordrecht, NL: Springer.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr., E. R., & Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence* (pp. 1306–1313). Atlanta, GA: AAAI Press.

Cohen, K. B., Verspoor, K., Fort, K., Funk, C., Bada, M., Palmer, M., & Hunter, L. E. (2017). The Colorado richly annotated full text (CRAFT) corpus: Multi-model annotation in the biomedical domain. In N. Ide & J. Pustejovsky (Eds.), *Handbook of linguistic annotation,* 1379–1394. Dordrecht, NL: Springer.

Cohen, P. R. (2015). DARPA's Big Mechanism program. *Physical Biology*, *12*, 045008.

Danos, V., Feret, J., Fontana, W., Harmer, R., & Krivine, J. (2009). Rule-based modelling and model perturbation. In C. Priami, R. J. Back, & I. Petre (Eds.), *Transactions on computational systems biology XI,* 116–137. Berlin: Springer.

Demir, E., et al. (2010). The BioPAX community standard for pathway data sharing. *Nature Biotechnology*, *28*, 935–942.

Falkenhainer, B. (1988). *Learning from physical analogies: A study in analogy and the explanation process* (Technical Report UIUCDCS-R-88-1479). Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL.

Forbus, K. D., Ferguson, R. W., Lovett, A., & Gentner, D. (2017). Extending SME to handle large-scale cognitive modeling. *Cognitive Science*, *41*, 1152–1201.

Forbus, K. D., Gentner, D., & Law, K. (1995). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, *19*, 141–205.

Forbus, K. D., Riesbeck, C., Birnbaum, L., Livingston, K., Sharma, A., & Ureel, L. (2007). Integrating natural language, knowledge representation and reasoning, and analogical processing to learn by reading. *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence* (pp. 1542–1547). Vancouver, BC: AAAI Press.

Friedman, S. E., Barbella, D. M., & Forbus, K. D. (2012). Revising domain knowledge with cross-domain analogy. *Advances in Cognitive Systems*, *2*, 13–24.

Funk, C., Baumgartner, W., Garcia, B., Roeder, C., Bada, M., Cohen, K. B., Hunter, L. E., & Verspoor, K. (2014). Large-scale biomedical concept recognition: An evaluation of current automatic annotators and their parameters. *BMC Bioinformatics*, *15*, 59.

Garg, S., Galstyan, A., Hermjakob, U., & Marcu, D. (2016). Extracting biomolecular interactions using semantic parsing of biomedical text. *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (pp. 2718–2726). Phoenix, AZ: AAAI Press.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, *7*, 155–170.

Hunter, L., Lu, Z., Firby, J., Baumgartner, W. A., Johnson, H. L., Ogren, P. V., & Cohen, K. B. (2008). OpenDMAP: An open source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-type-specific gene expression. *BMC Bioinformatics*, *9*, 78.

Maida, A., & Shapiro, S. (1982). Intensional concepts in propositional semantic networks. *Cognitive Science*, *6*, 291–330.

McDonald, D. (1996). The interplay of syntactic and semantic node labels in parsing. In H. Bunt & M. Tomita (Eds.), *Recent advances in parsing technology,* 295–323. Dordrecht, NL: Springer.

McDonald, D. D. (2000). Issues in the representation of real texts: The design of KRISP. In L. M. Iwanska & S. C. Shapiro (Eds.), *Natural language processing and knowledge representation: Language for knowledge and knowledge for language,* 77–110. Cambridge, MA: The MIT Press.

Mitchell, T. M., et al. (2015). Never-ending learning. *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (pp. 2302–2310). Austin, TX: AAAI Press.

Parikh, A. P., Poon, H., & Toutanova, K. (2015). Grounded semantic parsing for complex knowledge extraction. *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 756–766). Denver, CO: ACL.

Poon, H., & Vanderwende, L. (2010). Joint inference for knowledge extraction from biomedical literature. *Proceedings of the 2010 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 813–821). Los Angeles: ACL.

Pustejovsky, J. (1991). The syntax of event structure. *Cognition*, *1*, 47–81.

Pustejovsky, J., Cochran, B., & Castaño, J. (2002). MEDSTRACT: Natural language tools for mining the biobibliome. *Proceedings of the Second International Conference on Human Language Technology Research* (pp. 413–415). San Diego, CA: Morgan Kaufmann Publishers.

UniProt Consortium (2008). The universal protein resource (UniProt). *Nucleic Acids Research*, *36*, D190–D195.

Valenzuela-Escárcega, M. A., Hahn-Powell, G., Hicks, T., & Surdeanu, M. (2015). A domain-independent rule-based framework for event extraction. *Proceedings of the Fifty-Third Annual Meeting of the Association for Computational Linguistics and the Seventh International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing: Software Demonstrations* (pp. 127–132). Beijing, China: ACL.

Valenzuela-Escárcega, M. A., Hahn-Powell, G., & Surdeanu, M. (2016). Odin's runes: A rule language for information extraction. *Proceedings of the Tenth International Conference on Language Resources and Evaluation* (pp. 322–329). Portoroz, Slovenia: European Language Resources Association.

Valenzuela-Escárcega, M. A., et al. (2017). Large-scale automated reading with Reach discovers new cancer driving mechanisms. *Proceedings of the Sixth BioCreative Challenge Evaluation Workshop* (pp. 200–202). Bethesda, MD: CNIO.