Computers and Chemical Engineering 107 (2017) 49–60

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Hybrid ontology-learning materials engineering system for pharmaceutical products: Multi-label entity recognition and concept detection

Miguel Francisco M. Remolona[a,b], Matthew F. Conway[a], Sriram Balasubramanian[a], Linxi Fan[a], Ziyan Feng[a], Tianhao Gu[a], Hyungtae Kim[a], Prasad M. Nirantar[a], Sarah Panda[a], Nithin R. Ranabothu[a], Neha Rastogi[a], Venkat Venkatasubramanian[a,*]

[a] Complex Resilient Intelligent Systems Laboratory, Department of Chemical Engineering, Columbia University, New York, NY 10027, United States
[b] Chemical Engineering Department, College of Engineering, University of the Philippines, Diliman, Quezon City, Philippines

ARTICLE INFO

Article history:
Received 12 October 2016
Received in revised form 24 February 2017
Accepted 16 March 2017
Available online 21 March 2017

Keywords:
Natural language processing
Entity recognition
Ontology
Machine learning
Concept detection

ABSTRACT

The dawn of a new era in knowledge management due to information explosion is making old habits of modeling knowledge and decision-making inadequate. In the search for new modeling paradigms, we expect ontologies to play a big role. One of the critical challenges we face is the scarcity of semantically rich, properly populated, ontologies in most application domains in chemical and materials engineering. Developing such ontologies is a very challenging task requiring considerable investment in time, effort, and expert knowledge. One needs automation tools that can assist an ontology engineer to quickly develop and curate domain-specific ontologies. We consider our conceptual framework in this paper, a general approach for populating scientific ontologies, and its implementation as the prototype HOLMES, as an early attempt towards such an automated knowledge management environment. Our approach integrates a variety of machine learning and natural language processing methods to extract information from journal articles and store them semantically in an ontology. In this work, identification of key terms (such as chemicals, drugs, processes, anatomical entities, etc.) from abstracts, and the classification of these terms into 25 classes are presented. Two methods, a multi-class classifier (SVM) and a multi-label classifier (HOMER), were tested on an annotated data set for the pharmaceutical industry. The test was done using two different versions of the same data set, one using the BIO notation and the other not. The F1 scores for HOMER, were better in the BIO notation (63.6% vs 48.5%) while SVM performed better in the non-BIO version (54.1% vs 53.2%). However, the standard metrics did not consider the effect of the multiple answers that the multi-label classifier is allowed to obtain. As the results of our computational experiments show, while the performance of multi-label classifier is encouraging, much more remains to be done in order to develop a practically viable automated ontology-based knowledge management system.

1. Introduction

As the field of materials engineering continues to grow by leaps and bounds, high-throughput experiments, computational models, and conventional experimentation continue to generate large amounts of diverse data, posing challenges for scientists and engineers to store, manage, share, and use all this data effectively (Venkatasubramanian, 2009; Committee on Integrated Computational Materials Engineering et al., 2008). A proper framework for knowledge storage and retrieval, which exploits recent progress in machine learning (ML) and natural language processing (NLP) techniques, would enable scientists explore potential new materials more effectively. Such knowledge management would allow scientists to participate more actively in discovery informatics (Agresti, 2003). However, this requires an open, scalable, and flexible approach to knowledge modeling. These requirements can be met by the use of computer ontologies, which have been favored by the Semantic Web community (Taye, 2010). Ontological software has been developed with the scale and diversity of the World Wide Web in mind, so it is capable of handling the information

* Corresponding author.
E-mail address: venkat@columbia.edu (V. Venkatasubramanian).

http://dx.doi.org/10.1016/j.compchemeng.2017.03.012
0098-1354/

diversity in the materials engineering domain. Ontologies seek to model technical domain knowledge as a set of rules, a hierarchy of concepts, and the relations between them. These models can then be turned into first-order logic models, which are both computable, to check if they are logically consistent, and query-able in ways that allow for flexibly analysis of the information. These characteristics offer a foundation on which one can build innovative and intelligent systems for material design and discovery.

The use of ontologies to aid scientists has a large precedent in the biological and biomedical domain (Bard and Rhee, 2004); about a hundred and thirty ontologies for the biomedical domain are stored in the website of the Open Biological and Biomedical Ontologies (OBO) (Smith, et al., 2007). These ontologies have been used to great effect in understanding a variety of topics including metabolic and regulatory pathways (Guo et al., 2005), genes (Ashburner, et al., 2000), and computational systems biology (Sauro and Bergmann, 2008).

In material and manufacturing engineering domain, ontologies are a growing discipline. The most widely known is the OntoCAPE (Ontology for Computer-Aided Process Engineering) by Morbach, et al. (2007). There is also a manufacturing system engineering ontology developed by Lin and Harding (2007). In 2010, Ashino published his work on a materials ontology which contains information using data exchange with material databases. The contribution of Muñoz, et al., (2012, 2013) works on easing the decision-making in manufacturing industries using an ontology to relate corporate/management decisions to effects in manufacturing. Our work in this paper uses an extension of the Purdue Ontology for Pharmaceutical Engineering (POPE) (Hailemariam and Venkatasubramanian, 2010) as a basis for conceptualization of the HOLMES framework.

However, for ontologies to be useful in discovery informatics, they need to contain large amounts of accurate data. To be able to achieve this, there are two options: wait for ontologies to become the standard in information storage instead of data tables or databases, or, as in the HOLMES framework presented here, use advances in ML and NLP to extract information from scientific literature to populate the ontologies. In recent years, text-mining has been used to study the scientific literature (Carlson et al., 2010; Leaman and Gonzalez, 2008; Collins and Singer, 1999; Kim et al., 2003; Mausam et al., 2012; Agichtein and Gravano, 2000; Percha et al., 2012; Huang et al., 2004; Kudo and Matsumoto, 2001). In this contribution, we use text-mining and other machine learning algorithms to create an automated system to populate ontologies as an intermediate representation. This approach increases the scope of extraction a great deal. In addition, it is advantageous for two reasons: (i) ontologies provide constraints on how results from different algorithms agree with each other, and (ii) the output of the system is a structured ontology, over which users can run their own queries.

The biomedical industry is one of the first scientific domains to apply the information extraction techniques with applications such as Abner (Settles, 2004) and Banner (Leaman and Gonzalez, 2008). The domain has created a lot of machine learning training data that try to address the needs of the domain, such as the extensive GENIA corpus (Kim et al., 2003). However, since the domain is also large, there are a variety of these data sets (BioCreative, 2006). This has created the challenge of too much information, which is handled by segregating the information. But this segregation also gives rise to a diversity of tools required to completely process a single article.

The pharmaceutical industry is in the same dilemma as the biomedical, in terms of the amount of data and information processed. According to the 2015 PhRMA (Pharmaceuticals Research and Manufacturers of America) report (PhRMA, 2015), there were around 7000 total drugs in development all across the globe. Considering that the average length of the application for a new drug in the US is around 100,000 pages, the amount of information that the pharmaceutical companies, not to mention the FDA, is processing is enormous. Furthermore, for each drug approved, approximately 10,000 candidates are explored initially.

These large numbers suggest that the pharmaceutical industry is a "big data" industry that can benefit from recent progress in data science. Towards such a future, we propose in this paper a data science framework, and its implementation as a prototype, for knowledge management in the pharmaceutical domain. More specifically, the paper introduces three new ideas toward AI-assisted pharmaceutical product design: (1) a framework for the automatic population of a pharmaceutical product ontology, HOLMES; (2) a fully annotated Entity and Concept Databank for pharmaceutical product engineering; and (3) joint entity and concept recognition with a focus on using multi-label classification capture complex terms. To help in the readability and understanding of the text, a table of the more commonly used abbreviations in the text are shown in Table 7.

2. Framework for automatic population of ontologies

2.1. Ontologies

Ontologies were defined in 1993 by Gruber as an "explicit specification of a conceptualization" (Gruber, 1993). In the context of this paper, this specification refers to how information is organized and stored with as little loss as possible. It is important to note though that there is generally no single ontology for a given domain. Many instances of domain ontologies have been made with consideration to specific goals, a bottom up approach. This can be attributed to a lack of universally accepted top level or foundational ontology (Marquis, 2014), making it hard for a top-down construction of ontologies. While many attempts have been done to create this top-level ontology (Mascardi et al., 2006), none of these are considered as a standard.

One of the more prominent directions of research in ontologies is the *semantic web*. As information in the World Wide Web approaches 50 B Google-indexed web-pages (in 2015, (Kunder, 2016)), it is clear that semantic knowledge management at this level is obviously necessary. But in actuality, semantic web has not managed to infiltrate the internet. This can be attributed to several innate properties of the internet – the size, the uncertainty (Laskey et al., 2008), the inconsistency, to name a few.

One way to look at ontologies is as a set of graphical databases that are used to store information *semantically*. Nodes in this semantic graph are known as *individuals*. Edges represent which nodes are connected to other nodes. A collection of nodes defines an *ontological class*. A collection of edges represents *ontological properties*. One important aspect of the ontology is known as the *ontology reasoner*. The ontology reasoner is an inference engine that takes advantage of the description logic framework that ontologies are built upon. This allows first order logical reasoning within the basic structure of the ontology. Data filled ontologies can then derive logical conclusions similar to automated theorem proving.

Despite the obvious importance of this topic, there is not much work in chemical or pharmaceutical engineering. Some of the early work was by Venkatasubramanian et al. (2006), Sesen et al. (2010), Suresh et al. (2010a,b), and Hailemariam and Venkatasubramanian (2010). These papers discuss the development of a domain-specific ontology and its application for manufacturing, mathematical modeling, and regulatory compliance in the domain of pharmaceutical engineering. More recently, Kokossis describes a system for Port symbiosis (Lignos and Kokossis, 2014) using an ontology that contains pertinent information about Naval Ports and the surrounding areas. The system identifies the products of the ports and the

demands of the surroundings and tries to identify suggestions for optimal distribution of goods and resources within the area. Puigjaner and his co-workers have studied applications in integration of enterprise levels in ontology development (Muñoz et al., 2012, 2013). They have also developed ontologies for mathematical knowledge management to support decision-making (Muñoz et al., 2014).

The ontology referred to in this paper, an overview of which is shown in Fig. 2, is a modification of the Purdue Ontology for Pharmaceutical Engineering (Hailemariam and Venkatasubramanian, 2010). This new ontology, called Columbia Ontology for Pharmaceutical Engineering (COPE), contains 198 classes at the present and can be easily grown as needed. It also contains 202 relations including constraints and range constrictions. This level of detail and manageability makes it easier for the automated population of ontologies and statistical classification. Compared with OntoCAPE, which has 472 classes, 210 interclass relations and 1041 constraints (Marquardt et al., 2010), the complexity of COPE is considerably lower. By comparing the increase in COPE's complexity in contrast to OntoCAPE as the ontology is populated, a systematic analysis of the performance of the automatic population of COPE can be performed.

COPE is encoded in the Web Ontology Language, 2nd Edition (OWL 2). It contains relevant information on the following topics (i.e., sub-ontologies): materials; mathematical models; physical objects; unit processes and experiments; physical and chemical properties; physical, chemical and biological reactions; general scientific concepts; pure chemical substances; and values and dimensions. The following is a brief outline of the contents of the sub-ontologies.

2.1.1. Materials ontology

This ontology describes materials explicitly in its constituent substances and phase systems. This ontology contains information about homogeneous and heterogeneous mixture of substances and identifies a material by the composition and fraction of each component. It also specifies the role of a material in a process or reaction.

2.1.2. Mathematical models ontology

This ontology describes mathematical models. It handles the information for mathematical models, the variables, their relationships, and the concepts within. It identifies a model by specifying either the algorithm or expression, as well as the assumptions involved and the parameters. The concepts are identified by relating the model to the scientific concept ontology.

2.1.3. Physical objects ontology

This ontology describes physical objects and shapes. This ontology also contains spatial information including relative and absolute position, rotation, and other information of similar nature. This allows for more information on either molecular structure in the substance ontology as well as the material ontology.

2.1.4. Unit processes and experiments ontology

This ontology describes experimental and unit processes and the corresponding equipment involved in the processes. For each process, there is information on the parameters used in the process, the reactions that take place, the details on the material used, and the details on the material produced. It also contains information for the equipment like the specifications, processes where it is and can be used in and the operating parameters.

2.1.5. Physical and chemical properties ontology

This ontology identifies and classifies physical and chemical properties. It identifies a property with the classification and description. It also identifies the physical or chemical property that an ontological instance is associated with as well as its value.

2.1.6. Physical, chemical and biological reactions ontology

This ontology describes physical, chemical and biological reactions. It identifies reactions as processes that convert one material into other or those that change the phase system. It also specifies the reaction rate and has an option to identify the set of elementary reactions of a given reaction.

2.1.7. Scientific concepts ontology

This ontology describes scientific concepts as well as general information. It contains concepts about people, sources, scientific domains, and definitions. This ontology is also meant for storing the goals as well as the limitations of the papers that are being stored.

2.1.8. Pure chemical substances ontology

This is the ontology that describes molecular composition of pure chemical substances. It identifies a substance by conventional classification by structure, like inorganic or organic, basic organic or biological, etc. This ontology separates itself from the Materials ontology by considering that the substance ontology contains only those of a consistent molecular structure. It also contains information about classification by other features of substances, for example conductor, semiconductor, electrolyte, etc.

2.1.9. Values and dimensions ontology

This ontology describes value and the form in which value is presented. It also contains information about dimensions, and several data structures like arrays, and tables. It is also able to identify values as a distribution, a set, a pair, etc.

2.2. Knowledge acquisition for ontologies and HOLMES

The most common way of populating ontologies is through manually identifying individual entities and their connections to one another. This is similar to how a database is filled using a form that contains all the details to populate the database table. The problem with this is that it is a slow process and is impractical with the large amount of information that we want to store. An alternative approach is by matching the schema with that of a pre-existing database. There are, however, a number of problems with this. The most significant one being that the information transferred does not gain anything from using the ontologies. This defeats the purpose of using an ontology over a database. The additional information requires manual porting of the missing data.

The framework we propose aims to populate ontologies using text documentation. Several approaches have been pursued in this regard. One important framework, called OntoPOP, uses a combination of information extraction techniques and then following it up with a rule-based system to map information into ontologies (Amardeilh, 2006). Cimiano, on the other hand, presents a formal model of ontology and the idea of an "ontology learning layer cake" that separates different tasks, such as concept identification, relation extraction, and axiom generation (Cimiano, 2006). Different techniques for populating each layer and evaluating these techniques were also summarized. Pursuing another approach, Byrne started by using named entity recognition to identify key terms in her corpus (Byrne, 2008). Then she used relation extraction to identify relations between the key terms identified. The methodology she used is similar to ours, except for the difference of scale. The main difference is that her work is quite narrow in terms of the classes identified. It isn't specific enough to perform the more sophisticated reasoning and more specific queries that the semantic web and discovery informatics are looking for.

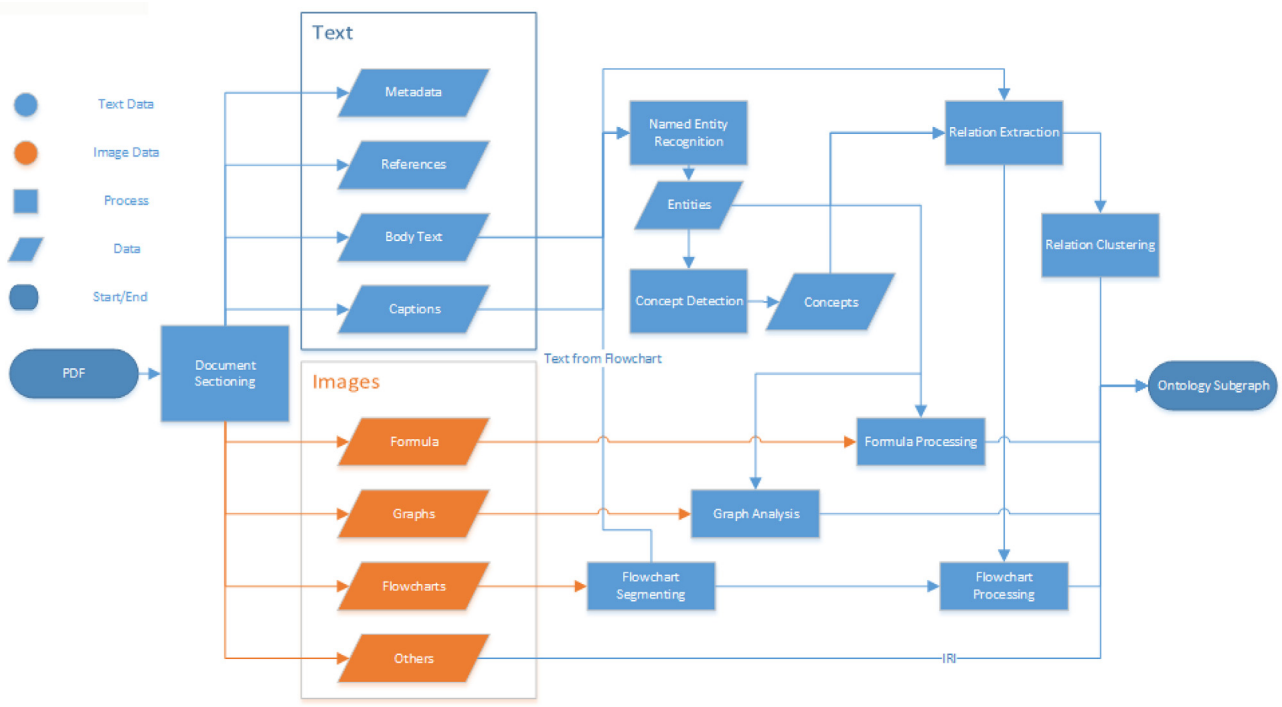

Fig. 1. Flowchart of HOLMES. The framework for the automatic population of ontologies.

HOLMES is the name of our framework for populating ontologies. It is an acronym for Hybrid Ontology-Learning Materials Engineering System. Our current focus is on scientific journals in the pharmaceutical engineering domain as they contain reliable peer-reviewed information. However, these are typically in the PDF format, which is not necessarily an easy format to extract data from. The PDF text can be extracted by software, but the extracted text is not organized any more than identifying how the words are aligned in the same line. There are also extractors that capture images, but the images are not flattened before extraction. This then leads to instances wherein the information from images are not extracted completely.

In contrast with other information extraction systems such as Ollie (Mausam et al., 2012), GATE (Cunningham et al., 2002) and RAPIDMINER (Hofman and Klinkenberg, 2013), the end goal of HOLMES is the automated allocation of the information from PDF's to the ontologies. HOLMES aims to extract as much information as possible including relations between terms, mathematical equations, flowcharts, with limited human assistance (except for data verification). The goal is not to limit the extraction of the information to specific topics in pharmaceutical engineering such as drug delivery, cancer treatment, reaction pathways, etc., but to be able to extract all of these along with their commonalities and differences. We do not, however, address the additional processing of this information for various applications.

Since our goal is to populate ontologies, the objectives of HOLMES are: (i) to identify the ontological entities or individuals (nodes), (ii) to identify the relationships among these individuals (edges), and (iii) then relate them to the ontological classes (group of nodes) and properties (group of edges). The overall hybrid architecture considers each task as a separate machine learning activity. These machine learning tasks are, in the order presented in this paper, Document Sectioning, Named Entity Recognition (NER), Concept Detection (CD), Relation Extraction (RE), Relation Clustering (RC), Formula Extraction, Graphical Image Processing, and Flowchart Processing. Fig. 1 shows the overall architecture of HOLMES. It starts by sectioning flattened images of a PDF document to text, images, and other sections. The section that contains the body text of the document is sent to Named Entity Recognition (NER), then Concept Detection (CD), then Relation Extraction (RE) and finally to Relation Clustering (RC) tasks. The mathematical and chemical formulas are sent into what we call the formula extraction task. The tables are stored as data tables, while making references to the text. Two dimensional graphs can be processed using image processing so that it can be converted to either relevant equations or table of values depending on the graph. The flowcharts are processed both as images and then text, as necessary. All images, regardless of classification, are cross referenced with the captions that are associated with it.

The image segmentation is done using the methodology proposed by Agrawal (Agrawal and Doermann, 2010). First, an area Voronoi diagram is generated. This is the closest white area that surrounds connected non-white pixels. The Voronoi edges are then generated and analyzed to determine which edges constitute a possible section divider. These edges are connected to each other to create polygons which are then converted to the closest rectangles. The rectangles represent a section, which are then classified using the methodology provided in the paper by Wang et al. (2006). It identifies 25 features that are relevant to identifying the 15 different section types that were mentioned in their paper. Sections identified as "main text" are used to filter the input into the Natural Language Processing (NLP) module, and sections identified as "mathematical formula" are passed to the formula extraction module. The other sections are sent to the relevant processing technologies. The first in the NLP module is the named entity recognition (NER). NER works by using statistical methods that need to be trained by pre-tagged data. Tagged data in this context is a series of sentences that have each word identified as part of an ontological class (a node), or as a non-node word. A simple NER can be built using Hidden Markov Models with only the words and their corresponding tags as your variables (Bikel et al., 1997). More complicated methods use feature vectors that uses contextual information in text to help determine these named entities (Nadeau and Sekine, 2007).

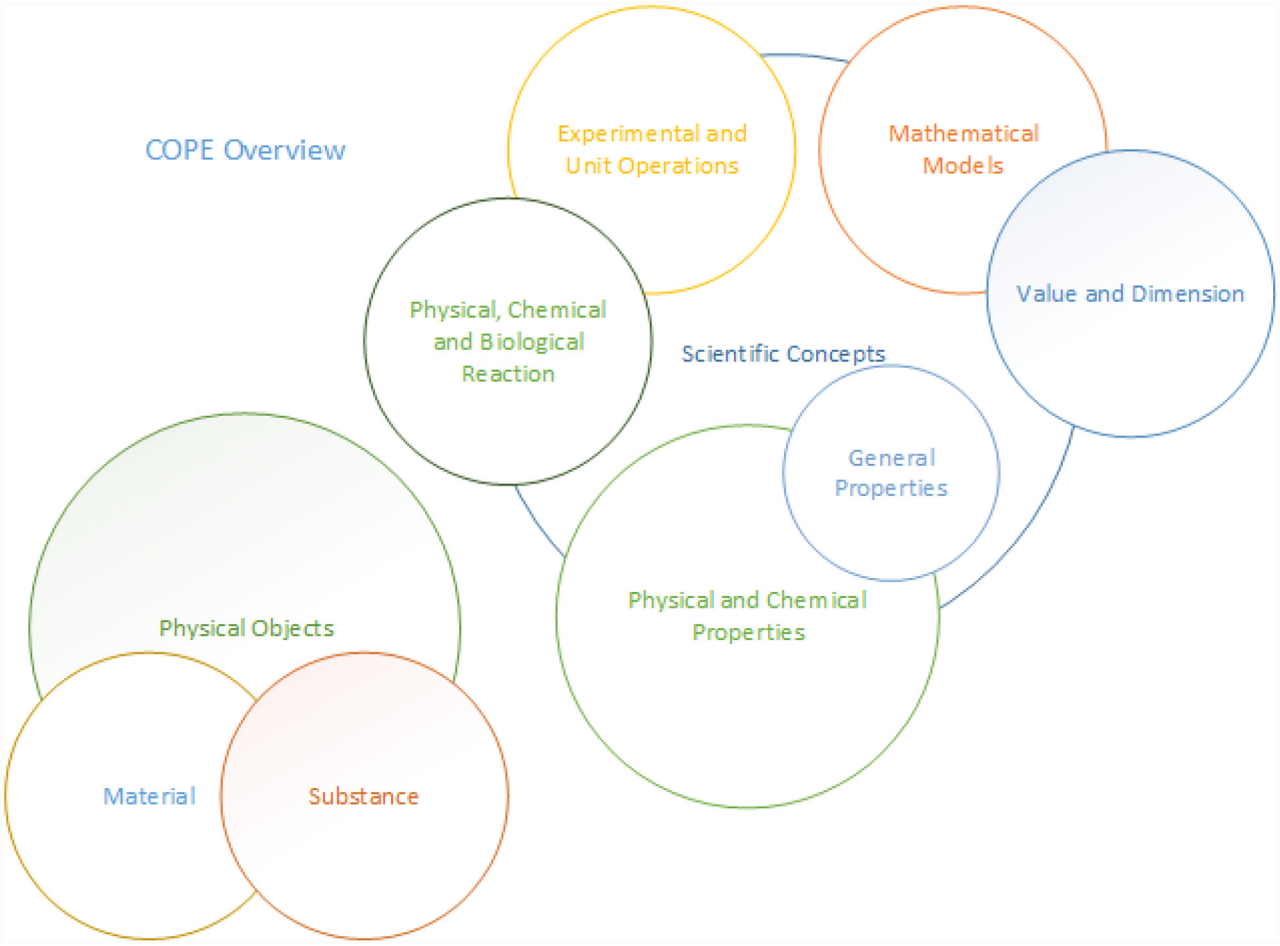

Fig. 2. Overview of the Columbia Ontology for Pharmaceutical Engineering (COPE).
The figure is depicted with overlapping regions representing either inclusion or partial inclusion in the ontology it overlaps. This shows the hierarchical relations between the main ontological classes.

The accuracies of these methods vary depending on the application. These accuracies are measured using what is known as an F-score, which is the average between how many words are tagged accurately (precision) and how many of those that have to be tagged are actually tagged (recall). Details on these metrics are discussed later. One of the early NER's, that identifies "Person" names, "Organization" names and "Others", has an accuracy between 89 and 96% (Bikel et al., 1997). Another NER by Collins, with partially unlabeled data, has an F1-score of between 81 and 91% (Collins and Singer, 1999). In the biomedical domain, a system called ABNER, reports accuracies of 70% (Settles, Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets, 2004). Another biomedical NER, called BANNER, reports accuracies of around 85% (Leaman and Gonzalez, 2008).

While the NER is focused on identifying entities or objects, a large part of the relevant information in the text are concepts. This makes it hard to simply combine different domain NER systems to create the relevant information identification. The information intersection between these different domain NER is in the concepts. Detecting these concepts is the role of concept detection (CD). Not much work has been done on generic concept detection; much of the reported work is in the biomedical domain. Scaria et al. (2013) report concept detection in the context of biological processes with an overall F1-score of about 55%. Expanding on this work to include also chemical processes, which no NLP method identifies, as well as physical, chemical and biological properties, would help in acquiring all the relevant information in a body of pharmaceutical text.

The next part in the NLP module is the relation extraction (RE). RE is where relations are identified between the named entities detected with the NER. As of this publication, there has been very few general research in this area, and the two most generic ones are mentioned here. The first of which is an algorithm, developed by Mausam, called Ollie (Mausam et al., 2012). Ollie looks for complex relations in sentences including relations within a complex sentence. The second is called Snowball (Agichtein and Gravano, 2000), which is more related to NER, except instead of features it uses the relations based on relative word locations as its training data. Domain specific relation extraction that has been studied so far depends to a high degree on the application of study. A natural language study in the pharmaceutical domain identified the relations between drugs and gene to identify the effects of drugs working together (Percha et al., 2012; Rindflesch et al., 2000). A similar study is used for protein-protein interactions (Huang et al., 2004).

Identifying which terms (both entities and concepts) have relations with each other is only the first step in dealing with relations. To be fully utilized in an ontology, these relations must also be classified into various classes, as with NER and CD. This classification into relation types is known as relation clustering (RC) or relation typing.

As RC is a relatively new problem area, there are no standard approaches yet. There are two promising methods which we consider in our framework. One is by Wang, et al., who extends other methods by putting constraints on entity classes (from NER) that

Table 1
(a) Example of a nested term. The labels are for the word or set of words directly above it. (e.g uridine 5′ triphosphate evoked –> process-concept, and uridine 5′ triphosphate –> substance) (b) BIO format for labels as applied to nested terms. The same term as in the example in labelled with the BIO format. c) Nested Term Relations. The relations due to nested terms.

a)

uridine	5′	triphosphate	evoked	store-operated	Ca2+	channel	entry
biological-process process-concept substance				process-concept	biological substance		

b)

uridine	5′	triphosphate	evoked	store-operated
B-biological-process B-process-concept B-substance	I-biological-process I-process-concept I-substance	I-biological-process I-process-concept I-substance	I-biological-process I-process-concept	I-biological-process B-process-concept
Ca2+ I-biological-process B-biological B-substance	channel I-biological-process I-biological	entry I-biological-process		

c)

Term		Parent
uridine 5′ triphosphate evoked	–>	uridine 5′ triphosphate evoked store-operated Ca2+ channel entry
store-operated	–>	uridine 5′ triphosphate evoked store-operated Ca2+ channel entry
Ca2+ channel	–>	uridine 5′ triphosphate evoked store-operated Ca2+ channel entry
uridine 5′ triphosphate	–>	uridine 5′ triphosphate evoked
Ca2+	–>	Ca2+ channel

are allowed in a relation type (Wang, et al., 2015). This work reports a normalized mutual information score that reaches 95% depending on the number of constraints applied. The other work (Berant, et al., 2014) focuses on biological processes, and classifies whether processes cause, enable or prevent another process.

Mathematical and Chemical Formulas are an essential part of the text that are not easy to process as text. This is from the nature of some parts of these formulae that are expressed as figures instead of some ASCII representation that a computer can easily parse. It is important that these formulae be extracted and placed into the ontology as it contains valuable information. Though these formulae are usually explained in text, the detail explained in text doesn't fully describe the equation as much as describing the constituent parts and focusing on only a few select relations that are important for the paper.

For mathematical formulae, mathematical characters have to be identified using an optical character recognition algorithm, as some journals publish formulae as images. The 2-D layout of the equation is extracted as a graph, the edges of which are given weights based on the likelihood that the classes of those glyphs would occur in that spatial relationship (i.e. it's likely for alphanumeric characters to be in the superscript position and unlikely for symbols, like+, to be in the subscript). The semantic meaning of the equation is found by searching for the maximum weight spanning tree of this graph via dynamic programming (Suzuki et al., 2003). The global syntax of the extracted formula is verified with a linear monadic tree grammar, as in the work by Fujiyoshi et al. (2010). Once the graph is verified, the variables in the equations are cross-referenced with the surrounding text to identify their meaning, and the formula is stored as annotated MathML.

For chemical formulae, there is an application called Optical Structure Recognition Application (OSRA) (Filippov, 2012). It analyzes a picture of the structure and then outputs a Simplified Molecular-Input Line-Entry System (SMILES) representation of the structure, which is a single string representation of the structure of a chemical formula (Weininger, 1987). It then allows for easier storage and comparison of chemical formula. A system for storing chemical formula and then aligning matching metabolic information with the use of ontologies has been reported (Kumar, 2014).

Most 2-D graphs can be considered as pixelated representations of equations. For two dimensional graphs, the conversion is relatively straightforward. This approach is commonly known as a plot or graph digitizer. The equation is then to be derived using curve fitting algorithms with high thresholds, r-squared score of 95 or higher, since the 2-D graphs are exact. If the threshold s not reached, then a data table is instead used. Flowcharts and similar images are either processes or some network structure. Rusiñol et al. (2014) used a series of graphic recognition techniques (to identify shapes), with optical character recognition to identify the nodes and the connectors in flowcharts. Improving this design and then integration of the results with the ontologies would be the next steps. In terms of integration, if it is a process, then its handling will be similar to that of processes defined in the text. For networks, the storage is in data structures within the value ontology.

HOLMES is an integration of all these components. From the text, NER and CD to identify the nodes and classifies them according to the ontological class they belong to. RE identifies which pair of nodes have an edge connecting them. Then RC identifies which ontological property the edge belongs to. The mathematical formula extraction identifies nodes by identifying constants and variables, while its innate structure identifies the relation these have to each other. Chemical formula is the same as mathematical formula as it identifies each element as a node and the structure identifies the relation. 2-D graphs require that the text is annotated to properly place them in the ontology. Flowcharts have to be reprocessed as text if it contains sentences. Then these flowchart sentences are connected with ontological edges as the flowchart shows.

2.3. Some challenges with this approach

One of the main challenges in developing machine learning algorithms for new domains, such as the one we are faced with, is the lack of labeled data with which models can be obtained through training. In certain well-researched domains, one doesn't have this problem. For instance, the most used natural language data bank, known as The Penn Treebank (Taylor et al., 2003), has about seven million annotated words. Genia, a domain specific dataset, has data

amounting to 89,862 terms for machine learning purposes (Kim et al., 2003). We don't have the luxury of such data banks in the domain of pharmaceutical engineering. This has to be one of the priorities going forward if we were to leverage the great progress in machine learning to improve drug discovery and manufacturing.

However, this problem of limited annotated data is partially mitigated with the use of active learning (Settles, 2009). Active learning is a set of techniques which, given a partially trained model and a set of "unlabeled data", can determine which unlabeled instances would be optimal for the new datum. This allows the system to prioritize the data so that annotators' time is spent wisely (Settles, 2001). Another workaround with limited data is the use of the ontology reasoner and the open world assumption in ontologies. This allows us to limit the NER and CD classes to just the classes that are directly descendants of the class "OWL:Thing", which is the ancestor of all ontological classes. The same concept applies to the ontological properties. This lack of training data is most apparent in machine learning algorithms that require separation between a large number of classes.

Scientific publications, especially in abstracts, are highly dense with respect to meaning per word. Terms can be complex wherein parts of the term can be classified as different from the term itself. For example, "Fourier Transform Infrared Spectroscopy" is classified as an "action-analytical process". In terms of grammar, it is known that the terms "Fourier Transform" and "Infrared" modify the word "Spectroscopy" for specificity in the type of spectroscopy that is being done. However, these terms alone existing somewhere else in the corpus will be classified differently from the way this specific term is classified. For the purpose of this paper, these complex terms are called nested terms. A more detailed example is mentioned in Table 1-a. This signifies the need for multiple labels in identifying and categorizing terms in multi-domain documents. However, as far as we could tell from the literature, there has been no study done in identifying the potential of multi-label classification in multi-domain documents or in entity recognition. We explore this opportunity by examining the potential of multi-label classifiers over the multi-class classifiers used in joint NER and CD given the corresponding data set.

Table 2
(a) Named Entity Recognition Expected Output. The NER output with the BIO formatting for the sentence "Antidepressant drugs, especially tricyclics have been widely used in the treatment of chronic pain, but not in acute pain." (b) Terms derived from NER results. Consecutive labeled words are used together to form terms. The terms can be classified to one of the labels it is given.

Token	Label
Antidepressant	B-process-concept
drugs	I-material
,	O
especially	O
tricyclics	I-biological
have	O
been	O
widely	B-process-concept
used	I-process-concept
in	O
the	O
treatment	B-biological-concept
of	O
chronic	B-biological
pain	I-biological-concept
,	O
but	O
not	B-value
in	O
acute	B-process-concept
pain	I-biological-concept
.	O

Key Terms	Possible Labels	
Antidepressant drugs	process-concept	material
tricyclics	biological	
widely used	process-concept	
treatment	biological-concept	
chronic pain	biological	biological-concept
not	value	
acute pain	process-concept	biological-concept

3. Methodology

Our computational experiments are designed to assess the accuracy of the current state of automated knowledge extraction systems and algorithms in the domain of pharmaceutical engineering. The initial tests are for evaluating the performance metrics of the joint named entity recognition and concept detection. Experiments are also conducted to resolve the possibility that a term is nested. An example result of the classification is shown in Table 2.

There are two main classifying algorithms that are used for entity recognition. The first is a multi-class classifier using the support vector machine (SVM) classifier (Kudo and Matsumoto, 2001) in the WEKA Machine Learning Library (Hall et al., 2009). The second is a multi-label classifier called HOMER (Tsoumakas et al., 2008). This algorithm is a type of problem transformation multi-label classifier and is one of the higher scoring algorithms based on tests comparing multi-label classifiers done by Madjarov et al. (2012). The HOMER algorithm is available in the MULAN library extension of WEKA.

Multi-class classification takes into account the problem of assigning a single label from a set of classes to a set of unlabeled words. Multi-label classification tries to identify multiple labels from the set of classes to a set of unlabeled words. In terms of sets, for a set of classes, Y, the difference between multi-class and multi-label is that for every word in the dataset, $x \in X$, a multi-class classifier will assign a maximum of one class, $y \in Y$, while a multi-label classifier will assign a set of classes, L, where $L \subset Y$.

The ER uses a total of 16 feature sets that are commonly used in entity recognition. These are: four Bigrams (forward and backward; tokens and number normalized), four Trigrams (forward and backward; tokens and number normalized), Token, Parts of Speech (token, forward and backward Bigrams), Word Shape, Prefix and Suffix (of one, two and three letters). A more detailed discussion of these features can be found in the work of Nadeau and Sekine (2007).

The classifying algorithms are applied to Data Set 2, which is identified in the data section below, in two different formats, using a 4-fold cross validation. The first format is a standard format for WEKA, wherein we separate the data into words, and each word retains the label of the terms that it is included in. For the multi-class classifier, the word retains only the label of the longest term it is a part of. For the multi-label classifier, it retains all of the labels of all the terms that it is a part of. For the purpose of this paper, this format set is called a non-BIO format.

The second format follows on the BIO formatting (Ramshaw and Marcus, 1995). In this format, the word that constitutes the start of a term is indicated with a "B-" attached to the label of that word. The rest of the words in the term is identified with an "I-" attached to the label. The same distinction as with the first format can be identified in this format comparing multi-class and multi-label classifiers. An example of this is shown in Table 1-b.

3.1. Data

We gathered the relevant information from four standard sources in pharmaceutical science and engineering: *The AAPS*

Table 3
Data Composition Statistics. The annotated data relevant statistics in terms of the entity recognition.

Item	Count
Tokens	24855
Sentences	417
Classes	24
Terms	7948
Feature Sets	16
Total Features	131096
Nested Relations	2154

Journal, *Molecular Pharmacology*, *The Journal of Pharmacological Scientists* and *The Journal of Pharmaceutical Innovation*. All abstracts from the web of science up to the 1st quarter of 2014 were retrieved. The total number of abstracts collected, removing duplicated and null entries, is 12,040. Of these, 1730 abstracts were selected based on diversity of topics to be included in the initial data set for testing of the Natural Language Processing methods. The data set used was annotated using the UAM Corpus Tools (O'Donnell, 2008). The labelled data for entity recognition consists of a total of 24,855 words/tokens in 417 sentences. These words are then grouped into a total of 7948 terms that each have their own labels. The total number of possible labels, *Y*, is 24 including the super-classes. A summary of this data composition is shown in Table 3. Some of these labeled texts are nested terms as was mentioned earlier. These nested terms are important since they contain information that can be used to specify mechanisms, behavior or other supplementary information about the entity. It also allows differentiation of these terms from other entities of the similar token composition. This presented a problem of how to extract these nested terms, which we have addressed.

3.2. Annotation schema

For text processing, the annotations require tagging both the significant terms as well as the relations in sentences. There is need for a system which allows a systematic identification followed by an encoding of these terms and relations. This system should allow for groups of words, terms, to be tagged as well as words themselves. In addition, the words are chunked into terms using the Beginning – Inside – Outside (BIO) chunk tagging. This system of tagging was introduced by Ramshaw and Marcus (1995). The Beginning signifies the start of a term. Inside refers to the other parts, or the inside, of the term. And finally, Outside refers to words that are not included in a term. This allows us to identify two or more consecutive terms separately. The classification scheme is a basic identification of actions, objects and concepts followed by a secondary layer for further classification into corresponding subtypes. The complete tree is shown in Fig. 3. This classification scheme was chosen to make the classes as different from one another as possible while still maintaining an ontology schema. While it can be argued that there is no categorical classification that would completely and exclusively include all entities (Marquis, 2014), this classification is meant only to differentiate between the major domains within pharmaceutical engineering.

3.3. Metrics

We follow the standard metrics used in machine learning and natural language classification, namely, precision, recall and F1-score. For each class, *y*, the results can be separated into four categories. True positives, P_T, are those results where the classifier was correct in identifying members of *y*. True negatives are the results where the classifier was correct in identifying those that are not members of *y*. False positives, type I error (E_I), are those results that were incorrectly classified by the classifier as a member of *y*. False negatives, type II error (E_{II}), are those results where the classifier failed to identify them as members of *y*.

3.3.1. Precision

Precision quantifies how accurately the machine learning algorithm identifies the term class. It is the number of true positives compared to the sum of false positives and true positives.

$$Precision = \frac{P_T}{P_T + E_I}$$

3.3.2. Recall

Recall evaluates the amount of correctly tagged classes that is recovered compared to what can actually be recovered. This is the ratio between the true positives over the sum of false negatives and true positives.

$$Recall = \frac{P_T}{P_T + E_{II}}$$

3.3.3. F1-Score

F-score is a metric that combines recall and precision using a weighting factor that determines the priority of one over the other.

$$F_x = \left(1 + x^2\right) \frac{Precision \times Recall}{x^2 \times Precision + Recall}$$

The F1-score has $x = 1$, giving equal bearing to both precision and recall.

$$F_1 = 2\frac{Precision \times Recall}{Precision + Recall}$$

3.3.4. Micro metrics

In multi-class classification, the aforementioned scores are taken as a weighted average over all labels. The weights considered in the average is in terms of the denominator used in the metric in consideration. For example, the precision metric reported is the average over all labels weighted by the sum of number of true positives and sum of the number of type I errors. This is equivalent to the micro metrics used in multi-label classification. This is the metrics used in the majority of the comparison.

3.3.4.1. Micro-Precision. Micro-precision is the example-label averaged precision. Using the same variables as the formula for calculating the precision, the micro-precision is:

$$Precision_{micro} = \frac{\sum_{y \in Y} P_{Ty}}{\sum_{y \in Y} P_{Ty} + \sum_{y \in Y} E_{Iy}}$$

3.3.4.2. Micro-Recall. Micro-recall, similar to the micro-precision metric, is the example-label averaged recall. In the same way precision was relabeled by putting a summation over all labels, the formula for micro-recall is as follows:

$$Recall_{micro} = \frac{\sum_{y \in Y} P_{Ty}}{\sum_{y \in Y} P_{Ty} + \sum_{y \in Y} E_{IIy}}$$

3.3.5. Other multi-label classification metrics

3.3.5.1. Macro-precision and macro-recall. Macro-precision is the label averaged precision. It takes the precision of each label and takes a non-weighted average of the values. In the same vein,

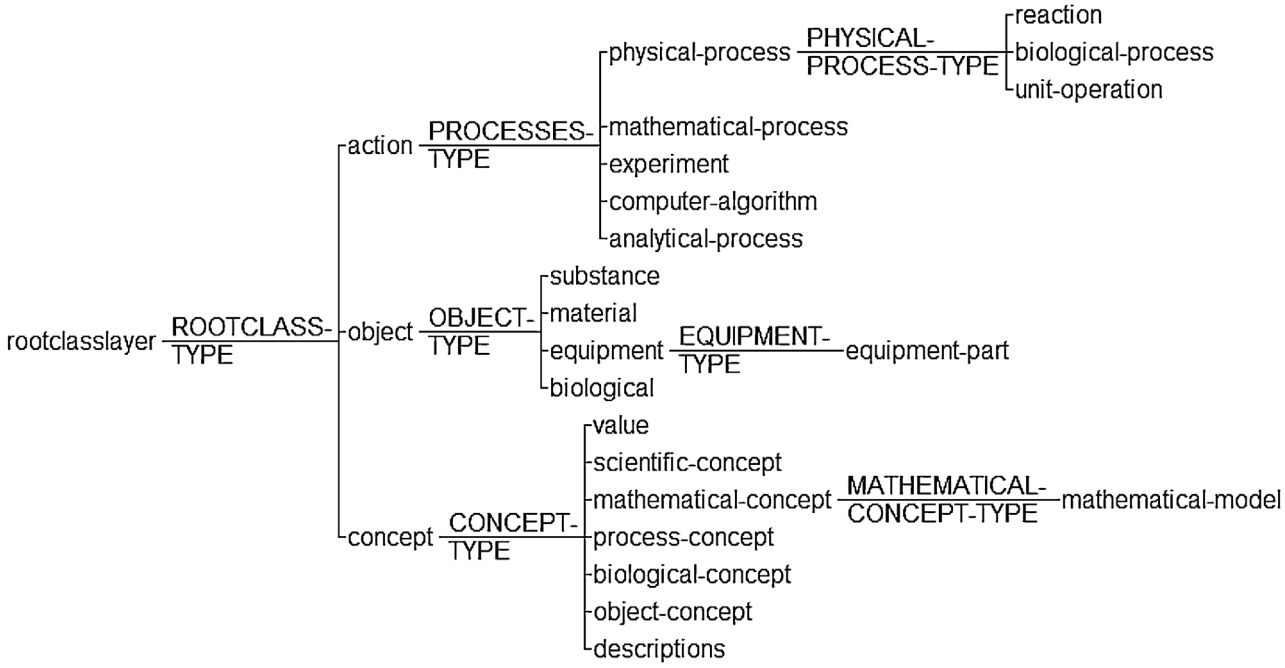

Fig. 3. Classification Schemas. The schema used for initial annotations as the primary division is based more on how the words are more commonly used in sentence.

macro-recall is the label averaged recall. The formula for these are as follows:

$$Precision_{macro} = \frac{1}{|Y|}\sum_{y \in Y}\frac{P_{Ty}}{P_{Ty} + E_{Iy}}$$

$$Recall_{macro} = \frac{1}{|Y|}\sum_{y \in Y}\frac{P_{Ty}}{P_{Ty} + E_{IIy}}$$

where $|Y|$ is the total number of labels in set Y.

3.3.5.2. Coverage. Coverage is the average depth on which you have to search the rankings of the labels per word until all the correct labels are recognized. Coverage is calculated as follows:

$$Coverage = \frac{1}{|X|}\sum_{x \in X}(\max_{y \in L} rank(x, y) - 1)$$

where $rank(x, y)$ is the rank of the label y for the word x according to the calculated results and L is the correct set of results.

3.3.5.3. Average precision. This metric is different from the previous precision metrics because it also adds importance into the ranking of the label. It takes into account the fraction of incorrect labels that are higher ranked than a correct label.

$$Precision_{Ave} = \frac{1}{|X|}\sum_{x \in X}\frac{1}{|L|}\sum_{y \in L}\frac{\sum_{y' \in Y} f(x, y', y)}{rank(x, y)}$$

$$f\left(x, y', y\right) = \begin{cases} rank(x, y) \geq rank(x, y') & 1 \\ otherwise & 0 \end{cases}$$

4. Results and discussion

In our investigation, we have used multi-class classification as a baseline to evaluate the effectiveness of multi-label classification in scientific text documents. Multi-label is used as a means to both reduce confusion in nested terms as well as to identify possible relations within these nested terms. Take for example, as shown in Table 1-b., the term "uridine 5′ triphosphate evoked store-operated Ca2+ channel entry." This is a nested term wherein the term can be broken down into different parts. These parts can then be related to one another in various ways, as in Table 1-c.

The results shown in Table 5 suggest an increase in performance of the multi-label classifiers (i.e., HOMER – 0.636) over the multi-class classifiers (i.e., SVM – 0.485) in the BIO format. The improvement on the recall (0.585 vs 0.462) is generally expected since the multi-label is capable of allowing more than one label per term, increasing the chance that a term contains a correct label. The improvement in precision (0.698 vs 0.512) is somewhat surprising as there are more opportunities for additional unwanted labels being included in the classification, thereby potentially affecting the performance negatively. For the non-BIO format data, SVM performed better in recall (0.555 vs 0.458). This is counter to the intuition that recall should be higher for a multi-label classifier. The higher precision of HOMER is along the lines of the BIO case (0.635 vs 0.527). Table 6 provides the performance for individual labels. Certain labels such as the biological-process (0.229 vs 0.225), equipment (0.303 vs 0.229), object-concept (0.428 vs 0.416) and unit-operation (0.333 vs 0.212) are a bit lower in recall performance in comparison.

The increase in performance for precision doesn't translate as cleanly to the individual classes when looking at the results for the BIO format. In this case, each label is considered separately for the "B-" and "I-" parts. Recall from the methodology section, that the "B-" stands for the beginning of a term and is appended to the first word of that term, while "I-" is appended to every other word in the term. Due to this appended modifier, the scores for a particular class is divided into two parts. Only 17 of the 48 labels scores in multi-label are above the scores of the corresponding label in multi-class, the most significant being the B-biological (0.727 vs 0.464) and the "B-" and "I-" for the process-concept (0.388 vs 0.365 for "B-" and 0.652 vs 0.510 for "I-") and value (0.748 vs 0.548 for "B-" and 0.898 vs 0.846 for "I-") labels. The remaining 12 labels that scored higher are B-analytical-process (1.00 vs 0.571), I-analytical-process

Table 4
Label Distribution on Words in Terms of the BIO classification system. Addition of the B-Counts and I-Counts gives the distribution for the non-BIO system. Addition of the B-Counts identifies the total number of terms.

Index	Label	B-Counts	I-Counts
1	action	12	18
2	analytical-process	56	82
3	biological	1159	1191
4	biological-concept	469	242
5	biological-process	122	110
6	computer-algorithm	36	46
7	concept	88	110
8	equipment	51	94
9	equipment-part	14	17
10	experiment	189	155
11	material	338	257
12	mathematical-concept	444	133
13	mathematical-model	29	52
14	mathematical-process	30	24
15	object	20	22
16	object-concept	927	725
17	physical-process	369	265
18	process-concept	1712	1016
19	reaction	21	10
20	scientific-concept	145	77
21	substance	747	444
22	unit-operation	20	12
23	value	589	352

Table 5
a) Recall and Precision Scores. Basic results reporting the recall and precision scores of the multiclass (SVM) and the multilabel (HOMER) classifiers. b) Coverage and Average Precision Scores. Comparing the coverage and average precision scores of the multilabel classifiers.

a)

	F1	Recall	Precision
BIO			
SVM	0.485	0.462	0.512
HOMER	0.636	0.585	0.698
Non-BIO			
SVM	0.541	0.555	0.527
HOMER	0.532	0.458	0.635

b)

HOMER	Coverage	Average Precision
BIO	9.55	0.210
Non-BIO	3.86	0.265
Average Depth	1.15	

(0.686 vs 0.650), B-concept (0.508 vs 0.333), I-concept (0.436 vs 0.353), I-equipment-part (1.0 vs 0.0), I-experiment (0.835 vs 0.734), B-mathematical-concept (0.704 vs 0.658), B-mathematical-model (1.00 vs 0.846), I-mathematical-model (0.917 vs 0.857), B-scientific-concept (0.525 vs 0.200), B-substance (0.751 vs 0.725), and B-unit-operation (0.875 vs 0.800). The reason the micro-precision is higher is due to the higher counts of the actual data points that constitute these labels, as is shown in Table 4 (e.g. amount of data for "B-computer-algorithm" is 36, while the "B-biological" is 1159).

The performance in recall also did not translate directly to individual classes (in Table 6). There are only 11 out of the 48 label scores in the multi-label, that outscores their counterparts for multiclass classification. Of these eleven, only the labels for B-mathematical-concept (with recall scores of 0.353 vs 0.311) and B-object-concept (0.312 vs 0.270) were of a significantly high count in terms of actual data points (444 and 927 respectively). The other nine labels are B-concept (0.070 vs 0.036), I-concept (0.131 vs 0.065), B-equipment-part (0.188 vs 0.00), I-equipment-part (0.246 vs 0.00), I-mathematical-model (0.417 vs 0.360), B-reaction (0.332

Table 6
Per label precision and recall scores. Scores of the SVM and HOMER algorithms associated with each particular label. BIO scores are segregated into the B-labels and the I-labels.

		Non-BIO				BIO									
						SVM				HOMER					
		SVM		HOMER		B		I		B		I			
		Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall		
1	action	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
2	analytical-process	0.6379	0.2741	0.6686	0.2385	0.5714	0.1455	0.6500	0.3250	1.0000	0.1211	0.6861	0.2811		
3	biological	0.5257	0.7666	0.5560	0.7555	0.4641	0.6379	0.5205	0.6964	0.7272	0.3486	0.5117	0.5903		
4	biological-concept	0.7363	0.3426	0.7060	0.3123	0.6404	0.2511	0.7692	0.3415	0.6055	0.2149	0.5796	0.1979		
5	biological-process	0.7619	0.2286	0.8183	0.2252	0.6875	0.2075	0.6667	0.2308	0.5696	0.1199	0.6625	0.1908		
6	computer-algorithm	0.8571	0.2432	0.8768	0.2761	0.8000	0.1333	0.8182	0.4091	0.7500	0.1156	0.8125	0.3373		
7	concept	0.4783	0.0748	0.5521	0.0513	0.3333	0.0364	0.3529	0.0652	0.5083	0.0704	0.4365	0.1306		
8	equipment	0.9362	0.3034	0.8068	0.2289	0.7778	0.1373	0.9000	0.2872	0.5000	0.0278	0.7304	0.1403		
9	equipment-part	0.0000	0.0000	0.9167	0.3214	0.0000	0.0000	0.0000	0.0000	0.5000	0.1875	1.0000	0.2458		
10	experiment	0.8022	0.4307	0.8335	0.3608	0.7590	0.3405	0.7342	0.3766	0.6753	0.2835	0.8348	0.3315		
11	material	0.8167	0.4066	0.8098	0.4320	0.7623	0.3875	0.7281	0.3430	0.6124	0.3449	0.6871	0.1874		
12	mathematical-concept	0.6797	0.2963	0.7257	0.3494	0.6577	0.3106	0.7000	0.1810	0.7040	0.3532	0.2738	0.0660		
13	mathematical-model	0.8611	0.4079	0.8939	0.3715	0.8462	0.4231	0.8571	0.3600	1.0000	0.3875	0.9167	0.4174		
14	mathematical-process	0.5000	0.0400	0.0000	0.0000	0.5000	0.0714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
15	object	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
16	object-concept	0.5581	0.4276	0.5462	0.4163	0.4411	0.2699	0.6067	0.4512	0.3657	0.3120	0.6004	0.2646		
17	physical-process	0.5878	0.3031	0.6232	0.2536	0.4894	0.1673	0.6610	0.3348	0.3302	0.1218	0.5963	0.1815		
18	process-concept	0.4598	0.6891	0.6346	0.4702	0.3649	0.6319	0.5099	0.5832	0.3881	0.5513	0.6524	0.2763		
19	reaction	0.5000	0.0500	0.7500	0.2402	1.0000	0.0833	0.6667	0.2500	0.7500	0.3321	0.0000	0.0000		
20	scientific-concept	0.3846	0.0559	0.6730	0.0852	0.2000	0.0381	0.0000	0.0000	0.5250	0.0902	0.0000	0.0000		
21	substance	0.7101	0.4673	0.7972	0.3991	0.7254	0.4602	0.7316	0.3432	0.7511	0.3946	0.6505	0.2065		
22	unit-operation	0.8000	0.3333	0.7917	0.2118	0.8000	0.2857	0.8333	0.5000	0.8750	0.3690	0.5833	0.5417		
23	value	0.8092	0.6687	0.8758	0.5961	0.5484	0.5025	0.8462	0.6275	0.7484	0.4917	0.8984	0.6076		

Table 7
Table of acronyms. Table containing the more commonly used acronyms in the paper.

Acronym	Meaning
OntoCAPE	Ontology for Computer-Aided Process Engineering
POPE	Purdue Ontology for Pharmaceutical Engineering
COPE	Columbia Ontology for Pharmaceutical Engineering
PhRMA	Pharmaceuticals Research and Manufacturers of America
FDA	Food and Drug Administration
NER	Named Entity Recognition
CD	Concept Detection
RE	Relation Extraction
RC	Relation Clustering
NLP	Natural Language Processing
BIO	Beginning-Inside-Outside

vs 0.083), B-scientific-concept (0.090 vs 0.038), B-unit-operation (0.369 vs 0.286), and I-unit operation (0.542 vs 0.500). There are several other metrics that support this misalignment of the components that are considered in multi-label classifiers. Two metrics of note are the average precision and coverage. These factors identify the amount of results that have to be filtered through to get to the relevant results. Examining the average precision (in Table 5-b) of HOMER, the low values suggest that there are a lot of labels that are higher ranked in the classification that are not relevant. Therefore, this requires considerable post-processing to identify the relevant labels.

The coverage tells a similar story (in Table 5-b). Comparing the coverage of the Non-BIO format to the average number of labels per word, there is only an excess of 3.71 labels each (3.86 minus the 0.15 excess on the average number of labels; the formula for coverage already considers the first label in the −1 term in calculations). The fact that there is a label of "O" for non-labelled words are already considered by the classifier. This means that there are about four out of every five labels that need to be removed from consideration. This is especially true for the results in the BIO format which has an excess of 9.4 labels (9.55–0.15 as before).

This then suggests that even though the multi-label classifier performed better in terms of the standard metrics of precision, recall and F1-scores, the results are skewed. This can be attributed to the number of data points that the multi-label classifier has in contrast to the multi-class classifier. The total number of labels is 7968, but the total number of base labels, i.e., the label of the most complete term in the nested term, is only 5284. It has to be noted, though, that the relations between the labels in consecutive terms are not considered in multi-label classifiers. Improving these scores on these metrics can be done using a number of different ways. One way is the consideration of rules as a post-processing step to reduce the number of irrelevant terms (e.g. ensuring that the first word in the term has a "B-" label for the BIO format). Another is the incorporation of correlations between the labels of consecutive labels in the classifier's model learning algorithm (i.e., the equivalent of using Hidden Markov Models or HMM's over SVM's for multi-label classification).

It is recommended that the two methods work hand in hand in identifying the nested terms for named entity recognition. The SVM classifier or other multiclass classifiers perform well in identifying the primary entity as is proven by previous researches in the biomedical domain and is partially shown in the experiments done. Removing this base class from the options in multi-class classifiers then reduces the complexity of filtering through the results.

5. Conclusion

As we enter the new era of data and knowledge explosion, old ways of modeling knowledge – i.e., storing, searching, and managing knowledge – and using it for decision-making are woefully suboptimal. We need a new paradigm for knowledge modeling and management. In this regard, ontologies are expected to play a big part in the future of process systems engineering. However, one of the limiting factors today is that properly populated ontologies are scarce in most application domains. Properly populated ontologies are those that contain large amounts of concepts with enough connections among them to mimic the underlying semantics. However, developing such ontologies is a very challenging task requiring considerable investment in time, effort, and expert knowledge. One needs automation tools that can assist an ontology engineer to quickly develop and curate domain-specific ontologies. This paper is an early attempt towards such a future. We consider our conceptual framework, a general approach for populating scientific ontologies, and its implementation as the prototype HOLMES, as the beginnings of a long intellectual journey that we expect to take at least a decade, if not more, before we have the descendants of HOLMES that can be used readily in practical applications.

The hybrid architecture of HOLMES integrates several machine learning and natural language processing tasks, such as, Document Sectioning, Named Entity Recognition (NER), Concept Detection (CD), Relation Extraction (RE), Relation Clustering, Formula Extraction, Graphical Image Processing and Flowchart Processing. The technologies for a majority of these components are available, though they are not yet coherent. Ours is the first attempt to have a coherent system for pharmaceutical engineering. Our investigation demonstrates the feasibility of such a conceptual framework. As the results of our computational experiments show, while the performance of multi-label classifiers is encouraging, much more remains to be done in order to develop a practically viable automated ontology-based knowledge management system. The authors are currently working on different aspects of this framework. For example, image extraction for both the identification of document parts and for data analysis is being done. The intermediate work of correlating these parts into coherent wholes is also being carried out. We are also working on deepening concept detection as well as filtering through the multi-label results.

References

Agichtein, E., Gravano, L., 2000. Snowball: extracting relations from large plain-text collections. In: Fifth ACM Conference on Digital Libraries, New York, NY, pp. 85–94.

Agrawal, M., Doermann, D., 2010. Context-aware and content-based dynamic Voronoi page segmentation. In: 9th IAPR International Workshop on Document Analysis Systems, New York, NY: ACM, pp. 73–80.

Agresti, W.W., 2003. Discovery informatics. In: Communications of the ACM., pp. 25–28.

Amardeilh, F., 2006. OntoPop or how to annotate documents and populate ontologies from text. European Semantic Web Conference.

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Laurie, 2000. Gene Ontology: tool for the unification of biology. Nat. Genet., 25–29.

Ashino, T., 2010. Materials ontology: an infrastructure for exchanging materials information and knowledge. Data Sci. J., 54–61.

Bard, J.B., Rhee, S.Y., 2004. Ontologies in biology: design, applications and future challenges. Nat. Rev. Genet., 213–222.

Berant, J., Srikumar, V., Chen, P.-C., Huang, B., Manning, C.D., Linden, A.V., Clark, P., 2014. Modeling biological processes for reading comprehension. In: Conference on Empirical Methods in Natural Language Processing, Qatar.

Bikel, D.M., Miller, S., Schwartz, R., Weischedel, R., 1997. Nymble: a high-performance learning name-finder. In: Applied Natural Language Processing., pp. 194–201, Stroudsburg, PA.

BioCreative, 2006. BioNLP Corporac Retrieved from Biocreative. http://biocreative.sourceforge.net/bio_corpora_links.html.

Byrne, K., 2008. Populating the Semantic Web-Combining Text and Relational Databases as RDF Graphs (Doctoral dissertation) Retrieved from http://homepages.inf.ed.ac.uk/kbyrne3/docs/thesisfinal.pdf.

Carlson, A., Betteridge, J., Wang, R.C., Hruschka, E.R., Mitchell, T.M., 2010. Coupled semi-supervised learning for information extraction. In: Third ACM International Conference on Web Search and Data Mining, New York : ACM, pp. 101–110.

Cimiano, P., 2006. Ontology Learning and Population from Text. Springer, Germany.

Collins, M., Singer, Y., 1999. Unsupervised Models for Named Entity Classification. Association for Computational Linguistics, pp. 100–110.

Committee on Integrated Computational Materials Engineering, National Materials Advisory Board, Division on Engineering and Physical Sciences, National Research Council, 2008. Integrated Computational Materials Engineering. National Academies Press.
Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., 2002. GATE: an Architecture for Development of Robust HLT Applications. In: Paper presented at the Proceedings of the 40th Anniversary Meeting of the Association of Computational Linguistics, Philadelphia.
Filippov, I. (2012, September 12). OSRA: Optical Structure Recognition Application. (National Cancer Institute) Retrieved February 2013, from National Cancer Institute.
Fujiyoshi, A., Suzuki, M., Uchida, S., 2010. Grammatical verification for mathematical formula recognition based on context-free tree grammar. Math. Comput. Sci., 279–298.
Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowl. Acquis. 5 (2), 199–220.
Guo, X., Shriver, C.D., Hu, H., Liebman, M.N., 2005. Analysis of metabolic and regulatory pathways through gene ontology-derived semantic similarity measures. AMIA Annual Symposium Proceedings, 972.
Hailemariam, L., Venkatasubramanian, V., 2010. Purdue ontology for pharmaceutical engineering: part I. Conceptual framework. J. Pharm. Innovation 5 (3), 88–99.
Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H., 2009. The WEKA data mining software: an update. SIGKDD 11 (1).
Hofman, M., Klinkenberg, R., 2013. RapidMiner: Data Mining Use Cases and Business Analytics Applications. Chapman and Hall/CRC.
Huang, M., Zhu, X., Hao, Y., Payan, D.G., Qu, K., Li, M., 2004. Discovering patterns to extract protein-protein interactions from full texts. Bioinformatics, 3604–3612.
Kim, J., Ohta, T., Tateisi, Y., Tsujii, J., 2003. GENIA corpus – a semantically annotated corpus for bio-textmining. Bioinformatics, 180–182.
Kudo, T., Matsumoto, Y., 2001. Chunking with support vector machines. In: The Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies, Pittsburgh, pp. 1–8.
Kumar, A., 2014. Rapid ontology alignment in large metabolic information databases. In: 14th AIChE Annual Meeting, Atlanta.
Kunder, M.d. (2016, February 29). WorldWideWebSize. Retrieved from www.worldwidewebsize.com.
Laskey, K.J., Laskey, K.B., Costa, P.C., Kokar, M.M., Martin, T., Lukasiewicz, T. (Eds.), 2008, Retrieved from World Wide Web Consortium: http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/.
Leaman, R., Gonzalez, G., 2008. Banner: an executable survey of advances in biomedical named entity recognition. Pacific Symposium on Biocomputing, 652–663.
Lignos, G., Kokossis, A.G., 2014. Semantically enabled technology for port symbiosis. In: 14th AIChE Annual Meeting, Atlanta.
Lin, H.K., Harding, J.A., 2007. A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration. Comput. Ind., 428–437.
Madjarov, G., Kocev, D., Gjorgjevikj, D., Džeroski, S., 2012. An extensive experimental comparison of methods for multi-label learning. Pattern Recognit., 3084–3104.
Marquardt, W., Morbach, J., Wiesner, A., Yang, A., 2010. Overview on OntoCAPE OntoCAPE – A Re-usable Ontology for Chemical Process Engineering, 35–56.
Marquis, J.-P., 2014. Category theory. The Stanford Encyclopedia of Philosophy, Retrieved June 2015, from http://plato.stanford.edu/archives/win2014/entries/category-theory/.
Mascardi, V., Cordi, V., Rosso, P., 2006. A Comparison of Upper Ontologies.
Mausam, Schmitz, M., Bart, R., Etzioni, O., 2012. Open language learning for information extraction. In: Empirical Methods in Natural Language Processing and Computational Natural Language Learning., pp. 523–534, Stroudsburg, PA.
Morbach, J., Yang, A., Marquardt, W., 2007. OntoCAPE – a large-scale ontology for chemical process engineering. Eng. Appl. Artif. Intell., 147–161.
Muñoz, E., Capón-García, E., Espuña, A., Puigjaner, L., 2012. Ontological framework for enterprise-wide integrated decision-making at operational level. Comput. Chem. Eng., 217–234.
Muñoz, E., Capón-García, E., Laínez, J.M., Espuña, A., Puigjaner, L., 2013. Integration of enterprise levels based on an ontological framework. Chem. Eng. Res. Des., 1542–1556.
Muñoz, E., Capón-García, E., Laínez-Aguirre, J.M., Espuña, A., Puigjaner, L., 2014. Using mathematical knowledge management to support integrated decision-making in the enterprise. Comput. Chem. Eng., 139–150.
Nadeau, D., Sekine, S., 2007. A survey of named entity recognition and classification. Linguist. Investig., 3–26.
O'Donnell, M., 2008. Demonstration of the UAM CorpusTool for text and image annotation. Association for Computational Linguistics, pp. 13–16.
Percha, B., Garten, Y., Altman, R., 2012. Discovery and explanation of drug-drug interactions via text mining. Pacific Symposium of Biocomputing, 410–421.
PhRMA, 2015. 2015 PhRMA Profile, Retrieved from PhRMA: http://www.phrma.org/sites/default/files/pdf/2015_phrma_profile.pdf.
Ramshaw, L.A., Marcus, M.P., 1995. Text Chunking using Transformation-Based Learning. Association for Computational Linguistics, pp. 82–94.
Rindflesch, T.C., Tanabe, L., Weinstein, J.N., Hunter, L., 2000. EDGAR: extraction of drugs, genes and relations from the biomedical literature. Pac. Symp. Biocomput., 517–528.
Rusiñol, M., Heras, L.-P.d., Terrades, O.R., 2014. Flowchart recognition for non-Textual information retrieval in patent search. Inf. Retr., 545–562.
Sauro, H.M., Bergmann, F.T., 2008. Standards and ontologies in computational systems biology. Essays Biochem., 211–222.
Scaria, A.T., Berant, J., Wang, M., Manning, C.D., Lewis, J., Harding, B., Clark, P., 2013. Learning biological processes with global constraints. 10th Conference on Empirical Methods in Natural Language Processing.
Sesen, M.B., Suresh, P., Banares-Alcantara, R., Venkatasubramanian, V., 2010. An ontological framework for automated regulatory compliance in pharmaceutical manufacturing. Comput. Chem. Eng. 34 (7), 1155–1169, http://dx.doi.org/10.1016/j.compchemeng.2009.09.004.
Settles, B., 2001. Closing the loop: fast, interactive semi-Supervised annotation with queries on features and instances. In: Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Edinburgh, Scotland, UK, pp. 1467–1478.
Settles, B., 2004. Biomedical named entity recognition using conditional random fields and rich feature sets. Proceedings of the COLING 2004 International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications.
Settles, B., 2009. Active Learning Literature Survey. University of Wisconsin, Madison.
Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Lewis, S., 2007. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. In: Ashburner, M., Mungall, C., Lewis, S., Ruttenberg, A., Scheuermann, R.H., Smith, B., Haendel, M. (Eds.), Nature Biotechnology. , pp. 1251–1255, June 20, Retrieved from The Open Biological and Biomedical Ontologies: www.obofoundry.org.
Suresh, P., Hsu, S.-H., Akkisetty, P., Reklaitis, G.V., Venkatasubramanian, V., 2010a. OntoMODEL: Ontological Mathematical Modeling Knowledge Management in Pharmaceutical Product Development, 1: Conceptual Framework. Ind. Eng. Chem. Res. 49, 7758–7767.
Suresh, P., Hsu, S.-H., Reklaitis, G.V., Venkatasubramanian, V., 2010b. OntoMODEL: Ontological Mathematical Modeling Knowledge Management in Pharmaceutical Product Development, 2: Applications. Ind. Eng. Chem. Res. 49, 7768–7781.
Suzuki, M., Tamari, F., Fukuda, R., Uchida, S., Kanahori, T., 2003. INFTY – An integrated OCR system for mathematical documents. In: ACM Symposium on Document Engineering, Grenoble.
Taye, M.M., 2010. Understanding semantic web and ontologies. J. Comput. 2 (June (6)), 182–192.
Taylor, A., Marcus, M., Santorini, B., 2003. The penn treebank: an overview. In: Abeillé, A. (Ed.), Treebanks. , pp. 5–22.
Tsoumakas, G., Katakis, I., Vlahavas, I., 2008. Effective and efficient multilabel classification in domains with large number of labels. Proceedings of the ECML/PKDD Workshop on Mining Multidimensional Data, 30–44.
Venkatasubramanian, V., 2009. DROWNING IN DATA: informatics and modeling challenges in a data-rich networked world. AIChE J., 2–8.
Venkatasubramanian, V., Zhao, C., Joglekar, G., Jain, A., Hailemariam, L., Suresh, P., . . . Reklaitis, G.V., 2006. Ontological informatics infrastructure for pharmaceutical product development and manufacturing. Computers & Chemical Engineering 30 (10–12), 1482–1496, http://dx.doi.org/10.1016/j.compchemeng.2006.05.036.
Wang, Y., Phillips, I.T., Haralick, R.M., 2006. Document zone content classification and its performance evaluation. Pattern Recognit., 57–73.
Wang, C., Song, Y., Roth, D., Wang, C., Han, J., Ji, H., Zhang, M., 2015. Constrained information-theoretic tripartite graph clustering to identify semantically similar relations. IJCAI'15 Proceedings of the 24th International Conference on Artificial Intelligence.
Weininger, D., 1987. SMILES, a chemical language and information system. Am. Chem. Soc., 31–36.