ChemDataExtractor: A Toolkit for Automated Extraction of Chemical Information from the Scientific Literature

Matthew C. Swain and Jacqueline M. Cole*

Cavendish Laboratory, University of Cambridge, J. J. Thomson Avenue, Cambridge, CB3 0HE, UK

E-mail: jmc61@cam.ac.uk

Abstract

The emergence of "big data" initiatives has led to the need for tools that can automatically extract valuable chemical information from large volumes of unstructured data, such as the scientific literature. Since chemical information can be present in figures, tables, and textual paragraphs, successful information extraction often depends on the ability to interpret all of these domains simultaneously. We present a complete toolkit for the automated extraction of chemical entities and their associated properties, measurements, and relationships from scientific documents that can be used to populate structured chemical databases. Our system provides an extensible, chemistry-aware natural language processing pipeline for tokenization, part-of-speech tagging, named entity recognition and phrase parsing. Within this scope, we report improved performance for chemical named entity recognition through the use of unsupervised word clustering based on a massive corpus of chemistry articles. For phrase parsing and information extraction, we present the novel use of multiple rule-based grammars that are tailored for interpreting specific document domains such as textual paragraphs,

captions and tables. We also describe document-level processing to resolve data interdependencies, and show that this is particularly necessary for the auto-generation of chemical databases since captions and tables commonly contain chemical identifiers and references that are defined elsewhere in the text. The performance of the toolkit to correctly extract various types of data was evaluated, affording an F-score of 93.4%, 86.8% and 91.5% for extracting chemical identifiers, spectroscopic attributes, and chemical property attributes, respectively; set against the CHEMDNER chemical name extraction challenge, ChemDataExtractor yields a competitive F-score of 87.8%. All tools have been released under the MIT license and are available to download from `http://www.chemdataextractor.org`.

Introduction

Scientific results are typically communicated in the form of papers, patents and theses that contain unstructured and semi-structured data described by free flowing natural language that is not readily interpretable by machines. Yet, manual data abstraction by humans with expert knowledge is an expensive, labor-intensive and error-prone process. With the continued growth of new publications, it is becoming increasingly difficult to create and maintain up-to-date manually curated databases, and automated information extraction by machines is fast becoming a necessity.

The chemistry literature presents an attractive and tractable target for this automated extraction as it is typically comprised of formulaic, data-rich language that is well-suited for machine analysis with the potential for high recall and precision. The extracted chemical information can be used to create and populate databases of chemical structures, properties and observations, opening up new avenues for discovery through large-scale data mining studies that are of great value in diverse areas such as materials discovery, drug discovery, and intellectual property protection.

In recent years, efforts such as The Materials Genome Initiative[1] have led to an increased

focus on large-scale data-mining for materials discovery. Notable projects include the Harvard Clean Energy Project,[2] which focuses on materials for organic photovoltaics, and the Materials Project,[3] which focuses on battery materials. These existing projects are primarily confined to exploiting computational resources to predict chemical properties, an approach that would be well complemented by wider availability of machine-readable databases of experimental properties. Moreover, a generic method that can automatically generate a database for any type of material property would extend the reach of existing efforts to all areas of materials science, rather than pre-defining a focus on a specific area.

While there are many well-established text-mining tools in the biomedical domain,[4–7] chemistry and materials text-mining is less widespread and fewer tools have been developed. Reviews by Eltyeb and Salim,[8] Vazquez et al.[9] and Gurulingappa et al.[10] provide comprehensive overviews of the existing chemistry text-mining tools and methods. Most of these tools focus narrowly on extracting specific entity types from specific document domains, while there are relatively few methodologies that embrace a broader focus on the extraction of chemical information, including properties, experimental measurements and relationships between entities.

One such tool is ChemicalTagger,[11] which parses experimental synthesis sections of documents to determine chemical roles (e.g. reactant, solvent) and relationships with experimental actions (e.g. heated, stirred), through the use of an ANTLR grammar[12] for rule-based text parsing and OSCAR[13] for chemical named entity recognition. ChemicalTagger has been used in conjunction with the commercial tool LeadMine[14] for the extraction of melting points from patents,[15] and additionally, the ChemEx project[16] extended ChemicalTagger with additional biomedical entity recognizers and image recognition of 2D chemical structures using OSRA.[17]

Gurulingappa et al.[10] outline the various challenges to further progress, and, in particular, highlight the distribution of information across different components of documents, such as textual paragraphs, images, tables and captions, as one of the primary barriers to successful

extraction of chemical information.

In this paper, we present a comprehensive toolkit for the automated extraction of chemical information from scientific documents. The toolkit provides a complete natural language processing (NLP) pipeline that makes use of a wide range of state-of-the-art methods, including a chemistry-aware part-of-speech (POS) tagger, named entity recognizers that combine conditional random fields and dictionaries, rule-based grammars for phrase parsing, and word clustering to improve performance of machine learning methods through unsupervised training. In addition, the toolkit includes a table parser for extracting information from semi-structured tabulated data, and document-level post-processing algorithms to resolve data interdependencies between information extracted from different parts of a document.

By automating the extraction of chemical entities, properties, measurements and procedures, our toolkit enables vast chemical databases to be created and populated with minimal time, effort and expense.

Implementation

System overview

Our system provides an end-to-end text-mining pipeline that takes PDF, HTML and XML files as input and produces an output of machine-readable structured data that is suitable for depositing in a database. Figure 1 presents an overview of the system. Our approach to each stage of the process is described below.

Document processing

The first stage of the system is to process PDF, HTML and XML files to isolate the relevant document domains, extract the raw text, and merge potentially fragmented data from different sources to produce a complete document record. The end result is a consistent, simplified document structure that consists of a single linear stream of document title, ab-

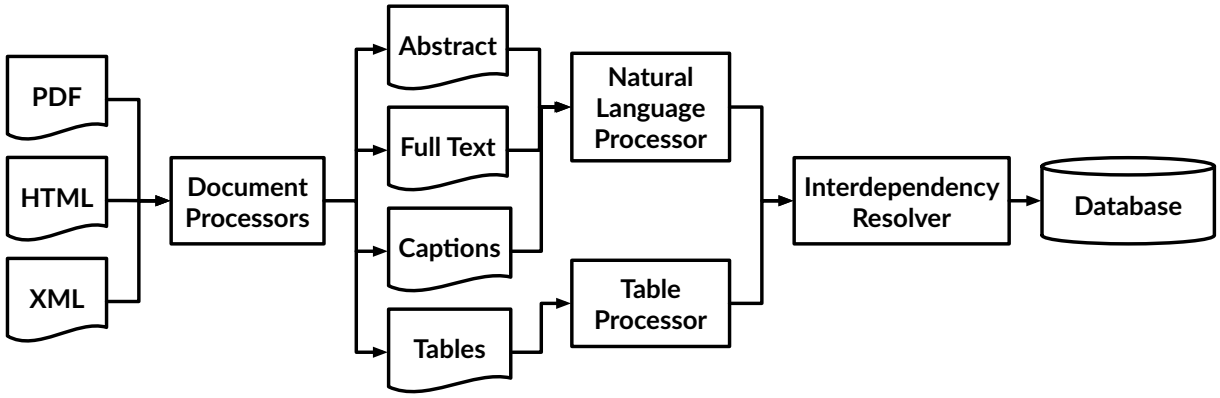

Figure 1: Overview of the complete information extraction system. Document Processors convert various input formats into a universal document model that consists of a single linear stream of elements such as paragraphs and tables that are each processed independently to extract information. This information is then merged to produce a single collection of chemical records for the overall document.

stract, heading, paragraph, figure, and table document elements. This allows subsequent components in the pipeline to process each document in exactly the same way, regardless of the original document format.

For text from HTML and XML sources, semantic markup of headings, paragraphs, captions and tables makes processing a trivial process. Once each text domain has been isolated, any further embedded markup (for example specifying bold and italic characters) is stripped to produce plain text for natural language processing. For tables, individual cells are treated as separate text domains, and stored in nested lists that represent the original table structure.

PDF documents present a greater challenge, as the format is not designed for the content to be easily interpreted by a machine. ChemDataExtractor provides layout analysis tools, built on top of the PDFMiner framework,[18] that use the positions of images and text characters to group text into headings, paragraphs and captions.

Natural language processing

The natural language processing pipeline extracts structured information from the English-language text in headings, paragraphs and captions. It is made up of five main stages:

tokenization, part-of-speech tagging, named entity recognition, phrase parsing, and information extraction. Figure 2 shows an overview of the pipeline, alongside an illustration of each stage applied to an example text passage.

Tokenization

The tokenization process converts text passages into a stream of tokens that are suitable for natural language processing. Text is first split into sentences, and then each sentence is further split into tokens that broadly correspond to individual words and punctuation.

Our system provides a sentence splitter that makes use of the Punkt algorithm by Kiss and Strunk,[19] which detects sentence boundaries through unsupervised learning of common abbreviations and sentence starters. This algorithm has been shown to be broadly applicable to many languages and text domains, and performs best when it has been trained on text from the target domain.[20] The unsupervised nature of this training process makes this method particularly well-suited to the chemistry domain, where there is a huge archive of literature available, and yet, very few collections have been manually annotated with features such as sentence boundaries. Our sentence splitter has been trained on the abstract, main text and captions of 3,592 chemistry articles published by The American Chemical Society (ACS), The Royal Society of Chemistry (RSC) and Springer. The sentence splitter identifies 702,132 individual sentences in the training articles, correctly distinguishing true sentence boundaries from full stops that occur in abbreviations, such as "et al.", "fig.", "ref." and "equiv." that are prevalent in the chemistry literature.

The word tokenizer has been designed to broadly match the Penn Treebank policy, with some modifications to better handle chemistry text. Tokens are split on all whitespace and most punctuation characters, with exceptions for brackets, colons, and other symbols in certain situations to preserve entities such as chemical names as a single token. Additionally, care is taken to consistently split units and mathematical symbols from numeric values, regardless of whether the source text contains a space between them.

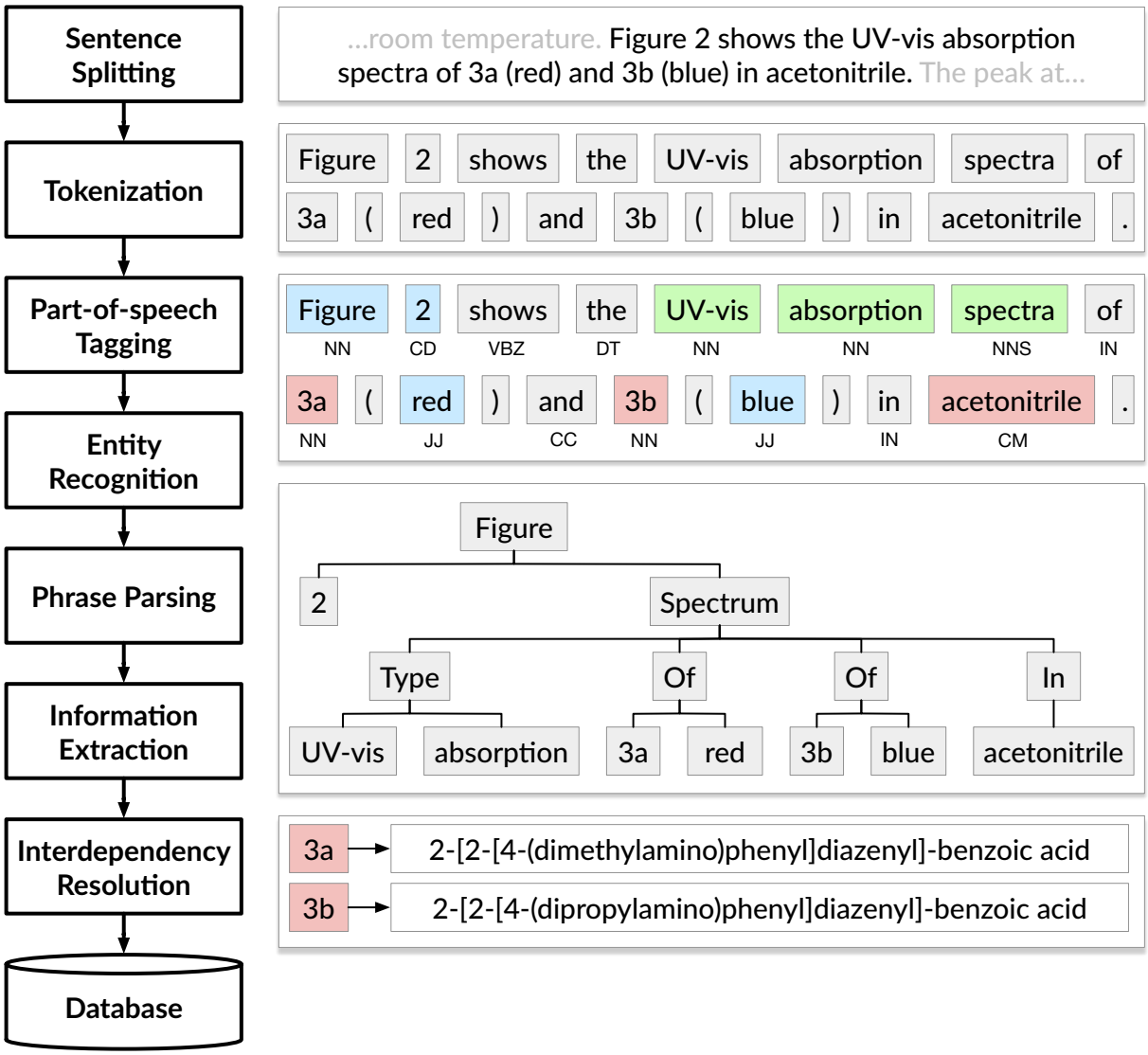

Figure 2: The natural language processing pipeline. Text is first split into sentences and then into individual tokens. The part-of-speech tagger and entity recognizer outputs are combined to assign a single tag to each token, which is then parsed using a rule-based grammar to produce a tree structure. This tree structure is interpreted to extract individual chemical records for this sentence, which are then combined with records from throughout the document to resolve data interdependencies and produce unified records for depositing in a database. Tags shown: NN = noun, CD = cardinal number, VBZ = verb (third person singular present), DT = determiner, NNS = noun plural, IN = preposition, JJ = adjective, CC = coordinating conjunction, CM = chemical mention.

Text normalization is an important step that removes commonly occurring inconsistencies that have a detrimental impact on the performance of machine learning and dictionary methods, and add unnecessary complexity to parsing rules. In contrast to other systems that normalize text prior to tokenization, our tokenizer is designed to operate on any unicode text input, and normalization is subsequently performed on the text content of each individual token. The advantage of this approach is that each token can retain a pointer to its exact start and end position within the source text, even if normalization then changes the length of tokens. Therefore, the original token text can always be recovered, and any information derived about a token can be easily annotated back onto the original document.

As part of the normalization, unicode characters with similar appearance that are often used interchangeably are standardized, all non-printing control characters are removed, and alternative chemical spellings are unified.

Word clustering

To achieve good performance, many machine learning techniques that are used in natural language processing must first be trained in a supervised fashion by providing a large collection of text from the target domain that has been manually annotated with the desired results. However, previous work has shown that the performance of these methods can be improved by adding unsupervised word representations as extra word features.[21] This is particularly useful in the chemistry domain, where the relative lack of annotated text collections for supervised training can be compensated for by using word cluster features derived from the extensive and widely available unannotated literature.

Our system makes use of features derived from Brown clustering,[22] a form of hierarchical clustering of words based on the contexts in which they occur. This has been shown to improve the performance of part-of-speech tagging and named entity recognition in a variety of domains.[21,23–26] Figure 3 shows how various components of our natural language processing pipeline incorporate both unsupervised and supervised learning.

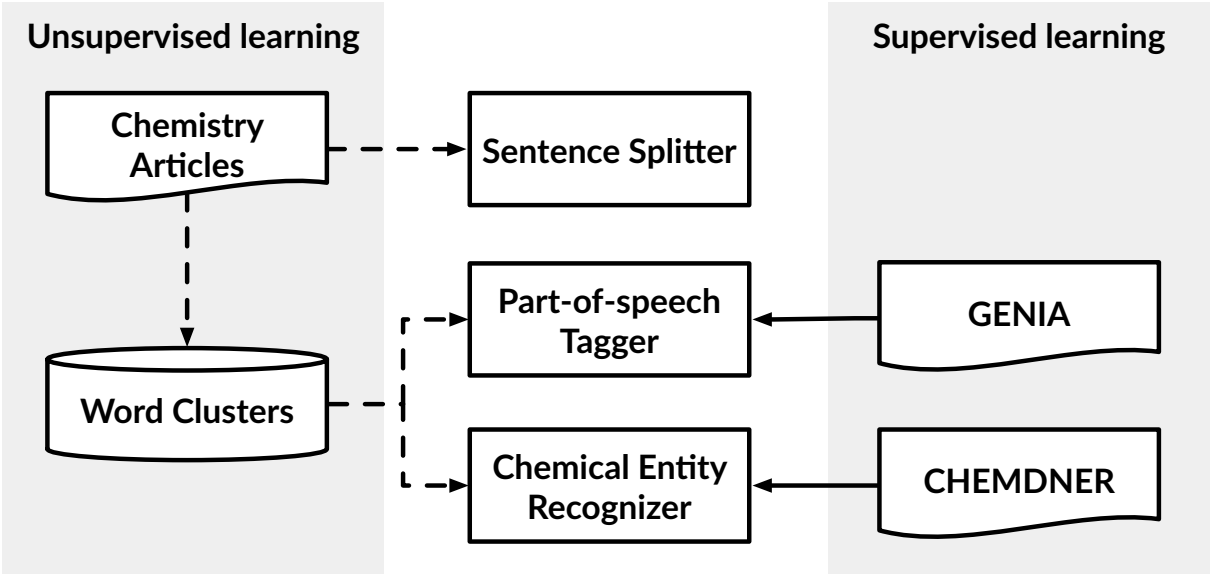

Figure 3: Supervised (solid lines) and unsupervised (dashed lines) training methods for machine learning-based NLP components. The sentence tokenizer relies entirely on unsupervised training using the raw text of chemistry articles, whereas the the part-of-speech tagger and chemical entity recognizer combine unsupervised features from word clusters with supervised training from labeled corpora, such as GENIA (2,000 MEDLINE abstracts with manually annotated part-of-speech tags) and CHEMDNER (10,000 PubMed abstracts with manually annotated chemical entity mentions).

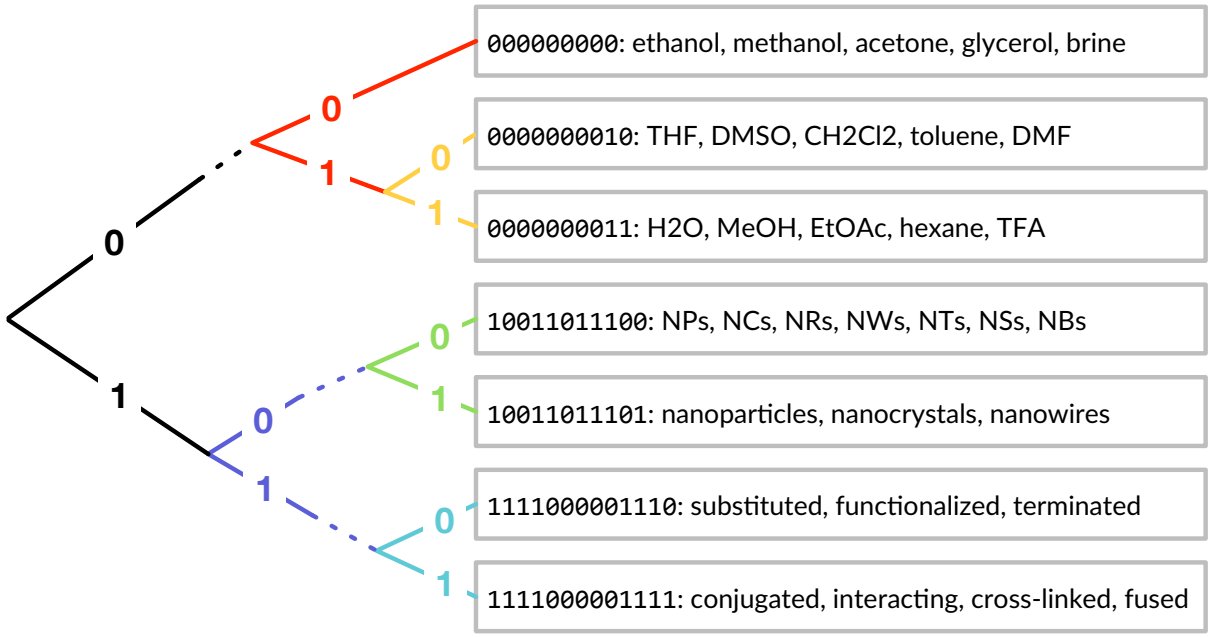

Figure 4: The most common words in seven example word clusters. The binary path defines the hierarchical position for each cluster. Larger cluster supersets can be obtained by considering different length prefixes of the binary path.

Clustering was performed on the full text and captions of 3,592 chemistry articles published by the ACS, the RSC and Springer. Once tokenized, this collection consists of about 20 million words in about 700,000 sentences. Clustering was performed using the Liang C++ implementation[27] to produce 1,500 clusters containing 372,799 unique words. Figure 4 shows the highest frequency words in seven example clusters. As Brown clusters are hierarchical, different length prefixes of the binary path correspond to cluster supersets, which can also be used as features in machine learning methods.

Part-of-speech-tagging

Part-of-speech (POS) tagging involves assigning a tag to each token that describes its syntactic function, for example as a noun, verb or adjective. Entity recognition, phrase parsing and information extraction tools routinely use the POS tags on tokens as part of their input, and thus their success is often highly dependent on the accuracy of the tagging process.

The vast majority of publicly available natural language processing tools provide POS taggers that have been trained on newspaper articles, and therefore do not necessarily perform well on chemistry literature. Tsuruoka et al.[4] found that by training a POS tagger on a combined corpus of newspaper articles (WSJ corpus[28]) and MEDLINE abstracts (GENIA corpus[29]), performance in the biomedical domain was greatly improved. In the absence of any equivalent POS-annotated corpus that covers the wider chemistry domain, the POS tagger in our system makes use of the same newspaper and biomedical training corpora, but also supplements these with unsupervised word cluster features derived from chemistry articles. This improves performance across a wider range of subject areas and document domains (such as captions) that are not well-covered by the training corpora.

Table 1: Features used in the POS tagger. A context window is used, such that some features for the token at index i are derived from the token text (w) of surrounding tokens.

Feature	Context	Description
Word	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	Normalized lowercase token text
Bigrams	$w_{i-2}w_{i-1}$, $w_{i-1}w_i$, w_iw_{i+1}, $w_{i+1}w_{i+2}$	Combinations of consecutive tokens
Word shape	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	Simplified token representation
Brown clusters	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	4, 6, 10, and 20 bit binary path prefixes
Length	w_i	Number of characters in token
Prefixes	w_i	1-5 character prefixes
Suffixes	w_i	1-5 character suffixes
Hyphenated	w_i	Contains a hyphen character
Alphabetical	w_i	Contains only alphabetical characters
Case	w_i	Is upper, lower, or title cased
Number	w_i	Is a number in digit or word form
Punctuation	w_i	Contains only punctuation characters

The POS tagger uses a linear-chain conditional random field (CRF) model, trained using the Orthant-Wise Limited-memory Quasi-Newton (OWL-QN) method as implemented by the CRFsuite framework.[30] The features for each token are shown in Table 1. The word shape feature is derived by replacing every number with ‘d’, every greek letter with ‘g’, and every latin letter with ‘X’ or ‘x’ for uppercase and lowercase respectively.

Chemical named entity recognition

In order to extract information such as relations and properties, the named entities involved must first be recognized. The task of recognizing chemical entity mentions in text is an area that has recently received significant attention. The best performing approaches typically involve a hybrid approach that combines dictionary and rule-based methods with machine learning methods. The OSCAR4 recognizer,[13] which uses a maximum-entropy Markov model (MEMM), and ChemSpot,[31] which uses a CRF model, are two of the most well-established systems. More recently, the CHEMDNER community challenge has promoted the development of a number of new systems,[32] and provided the CHEMDNER corpus of 10,000 PubMed abstracts with 84,355 manually annotated chemical entity mentions.[33]

Due to the wide variety of methods with differing strengths, our approach is to provide a modular architecture for named entity recognition that allows the results from multiple methodologies to be combined using heuristic techniques. We primarily use a CRF-based recognizer for chemical names, in combination with a dictionary-based recognizer that provides improved performance for trivial and trade names, and a regular expression-based recognizer that excels for database identifiers and chemical formulae.

The dictionary-based recognizer uses a word list compiled from the Jochem chemical dictionary[34] with an automatic filtering process based on the method described by Lowe and Sayle[14] that excludes entries that lead to false positives. For efficient storage and fast string matching, the dictionary is stored as a directed acyclic word graph (DAWG), which uses a graph-like representation to eliminate redundancy between similar names.

The CRF-based recognizer uses a linear-chain CRF model, trained using the OWL-QN method as implemented by the CRFsuite framework.[30] The features that are generated for each token are listed in Table 2. An "IOB" labelling scheme was utilized, where each token is labelled as the beginning of a chemical name (B), in a chemical name (I), or outside a chemical name (O). Training was performed using the training subset of the CHEMDNER corpus and the word clusters derived from chemistry articles.

Table 2: Features used in CRF chemical named entity recognizer. A context window is used, such that some features for the token at index i are derived from the token text (w) of surrounding tokens.

Feature	Context	Description
Word	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	Normalized lowercase token text
POS tags	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	Part-of-speech tag
Word shape	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	Simplified token representation
Brown clusters	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}	4, 6, 10, and 20 bit binary path prefixes
Length	w_i	Number of characters in token
Counts	w_i	Digit, upper and lower case letter counts
Prefixes	w_i	1-5 character prefixes
Suffixes	w_i	1-5 character suffixes
Hyphenated	w_i	Contains a hyphen character
Alphabetical	w_i	Contains only alphabetical characters
Case	w_i	Is upper, lower, or title cased
Number	w_i	Is a number in digit or word form
Punctuation	w_i	Contains only punctuation characters
URL	w_i	Looks like a URL

Phrase parsing

In general, parsing natural language is a challenging problem, due to ambiguities that mean a single sentence can sometimes be parsed in multiple different ways to produce different meanings. In practice, the formulaic and precise nature of the chemistry literature means that this occurs less often, and parsing to a level that is adequate for information extraction is much more tractable than in other domains.

The ChemicalTagger project pioneered the use of a rule-based grammar for parsing experimental synthesis sections of chemistry texts. Their strategy was to attempt to build one universal grammar to parse all possible inputs, but this was pushing at the practical limits of a single rule-based grammar, even within their relatively narrow target domain.

Our alternative strategy is to make use of multiple, more specialized grammars that are tailored to extracting more specific types of chemical information. Similarly to ChemicalTagger, our system produces input for the parser in the form of a merged list of tags from the part-of-speech tagger and chemical entity recognizer. Each grammar consists of a series of nested rules that describe how sequences of tagged tokens can be translated into a tree

a) Chemical identifier grammar

```
name = T('B-CM') + ZeroOrMore(T('I-CM'))
label = R('[1-9][a-z]?') | R('[IVX]+')
cem: name + W('(') + label + W(')')
```

b) Parse tree for " ... 1,2-dihydroxybenzene (3a) ... "

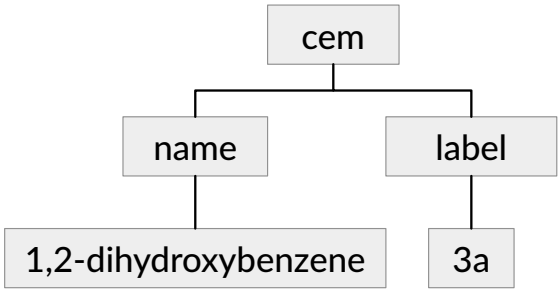

Figure 5: An example rule-based parsing grammar that recognizes chemical names with an associated alphanumeric label.

model that represents the syntactic structure of each sentence. Grammars are defined in simple Python code, and unlike other tools, they do not need to be compiled before use.

Figure 5a shows a simplified grammar, which recognizes definitions of alphanumeric compound labels in terms of a full chemical name. The rules are primarily constructed using three core elements: `T`, which matches a token based on its POS or entity tag, `W`, which matches the exact text of a token, and `R`, which matches text patterns using regular expressions. The `+` operator is used to define a required sequence of tokens, while the `|` operator is used where just one of multiple alternatives is required. Additional elements such as `Optional`, `ZeroOrMore` and `Not` allow more complex rules to be constructed.

In the example grammar shown in Figure 5a, the first rule, `name`, matches a token with the tag `B-CM`, followed by zero or more tokens with the tag `I-CM`, corresponding to the output tags of the chemical named entity recognizer. The second rule, `label`, defines two regular expression patterns, one for alphanumeric labels and one for Roman numerals, either of which

may be matched. The final rule, `cem`, is defined in terms of the first two rules. It requires a `name`, followed by a `label` enclosed within brackets. Figure 5b shows an illustration of the tree data structure that results from applying this grammar to an example sentence.

Table parsing

Tables are a highly attractive target for information extraction due to both their high data density and also their semi-structured nature that facilitates interpretation in comparison to completely unstructured natural language. Despite the lack of strict table format standardization, many tables in the chemistry literature follow broad conventions that make accurate interpretation possible through rule-based methods.

An overview of the table parsing system is shown in Figure 6. At present, ChemDataExtractor primarily targets tables in which each row corresponds to a single chemical entity, and each column describes property values for that entity. By treating each individual table cell as a short, highly formulaic sentence, information can be extracted using a specialized version of the natural language processing pipeline. This consists of a more fine-grained tokenizer and a series of rule-based parsing grammars, each tailored specifically for extracting a certain property type.

Column headings are parsed first, to determine the type of data in the cells below, as well as any relevant units. Column headings can also contain contextual data themselves, such as temperatures, concentrations or solvent names, which are applied to the property values in every cell below that heading. Interpreting the property values in each cell can be as simple as reading a single numeric value, but multiple bracketed and comma-separated values within a single cell are commonplace. In these cases, interpretation of any corresponding bracketed or comma-separated structure in the column heading is often necessary to successfully parse the values and assign the correct units to the correct values.

Figure 6 shows an example of a simple table that contains a UV-vis absorption peak wavelength and extinction value for a single compound. After each cell has been separately

tokenized and tagged, the heading cells are parsed to determine the column types and units. In this case, the first column contains chemical entity identifiers, and the second column contains combined UV-vis wavelength and extinction values. Each subsequent row is then processed individually to produce a chemical record, taking into account the column classification from each heading to choose the appropriate parsing grammar for the cells below.

Data interdependency resolution

In many cases, the information extracted from a single sentence, caption or table can be meaningless or even misleading without the context provided by the rest of the document. The final stage of our system involves post-processing to resolve these data interdependencies and combine the data from individual document domains into a single complete structured record for each unique chemical entity that is mentioned in the document.

Chemical identifier disambiguation

Initially, each heading, paragraph, caption and table is processed completely independently to produce structured chemical records that are defined in terms of whatever chemical name, abbreviation or identifier is used locally in that context. The records from throughout the document are then combined into a single list, and records that refer to the same chemical entity are merged into a single record.

Our system detects definitions of chemical abbreviations and labels using a method based on the algorithm by Hearst and Schwartz.[35] This algorithm is applied to all sentences in the document to produce a list of mappings between abbreviations and their corresponding full unabbreviated names. These mappings are then used to merge data that is defined in terms of different identifiers into single records for each unique chemical entity.

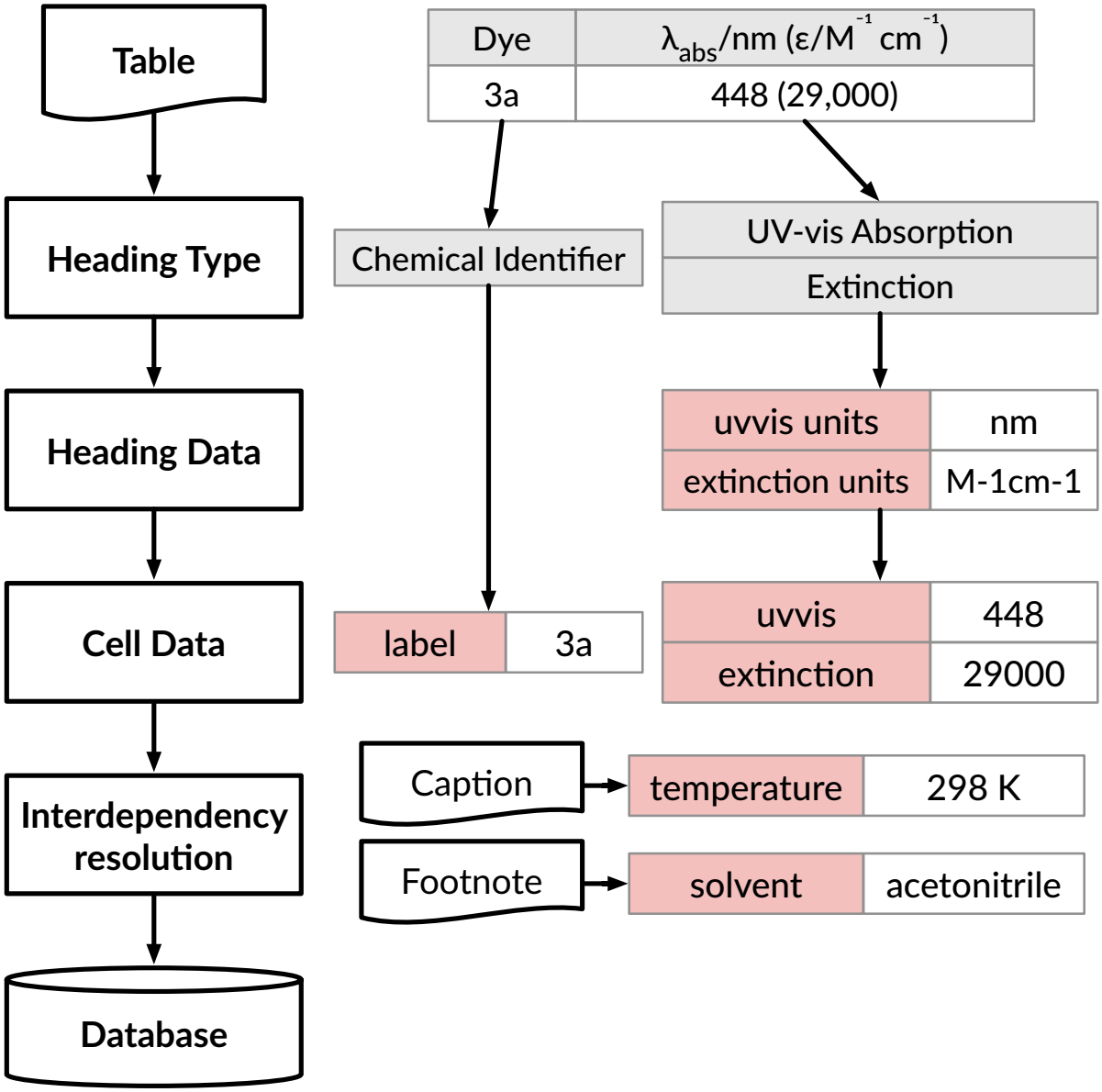

Figure 6: Overview of the main stages in the table parsing system (left) applied to a simplified example table (right). First, table headings are parsed to classify the type of each column, and then all the cells in each row are parsed to produce a compound record. The data interdependency resolution process incorporates information from the table caption and elsewhere in the document to produce the final record.

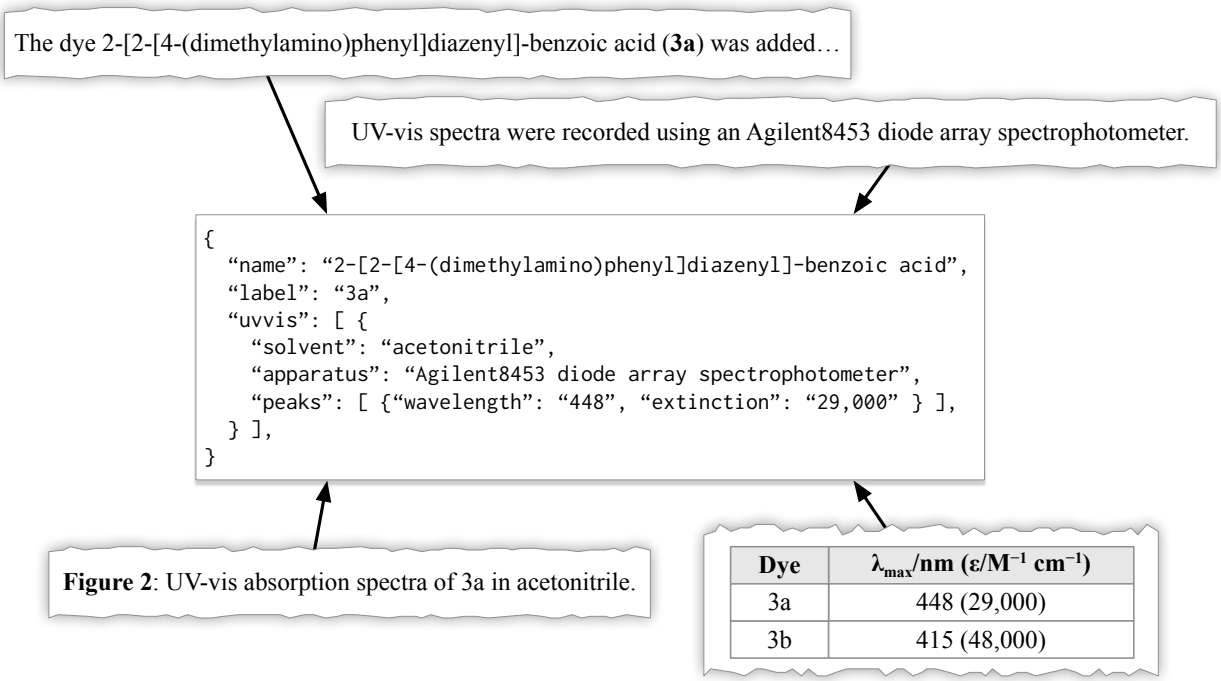

Figure 7: An example of how information from sentences, captions and tables is combined to produce a structured data record.

Global contextual information

In some cases, a sentence contains spectroscopic attributes or property information but is lacking any specific chemical identification information. Often, for example in experimental sections of a research article, the chemical identification information may be available in the preceding sentence or heading, and if so, this is used. Otherwise, these sentences typically contain contextual information (for example, a temperature, solvent or apparatus) that is applicable to all properties or spectra of a certain type. In these cases, the information is merged into all other records for all spectra or properties of that type and the record itself is removed.

Final data model

Figure 8 presents the schema of the final data model. The extracted data are primarily based around chemical entity records that contain all the names, abbreviations and labels that were

used in the document to refer to a given chemical entity. Each chemical entity record can have multiple associated spectra and properties, and each spectrum can also optionally contain information about individual peaks. ChemDataExtractor comes with built-in parsers and extractors for the specific property and spectrum types shown in Figure 8; however, the extensible and modular design of ChemDataExtractor means that it is straightforward for users to build additional parsers and extractors for other property and spectrum types.

The final chemical records may be saved directly to a document-oriented NoSQL database, or to a relational database through the use of an object-relational mapper. Alternatively, they may be exported to a variety of file formats including Microsoft Excel, SDF, CSV or JSON.

Evaluation

Evaluation metrics

For all aspects of the system, performance has been evaluated in terms of precision (the percentage of retrieved results that are correct), recall (the percentage of correct results that are retrieved) and F-score (harmonic mean of precision and recall). These are defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

$$\text{F-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

where TP is a true positive that the system correctly identified, FP is a false positive that the system incorrectly identified, and FN is a false negative that the system failed to

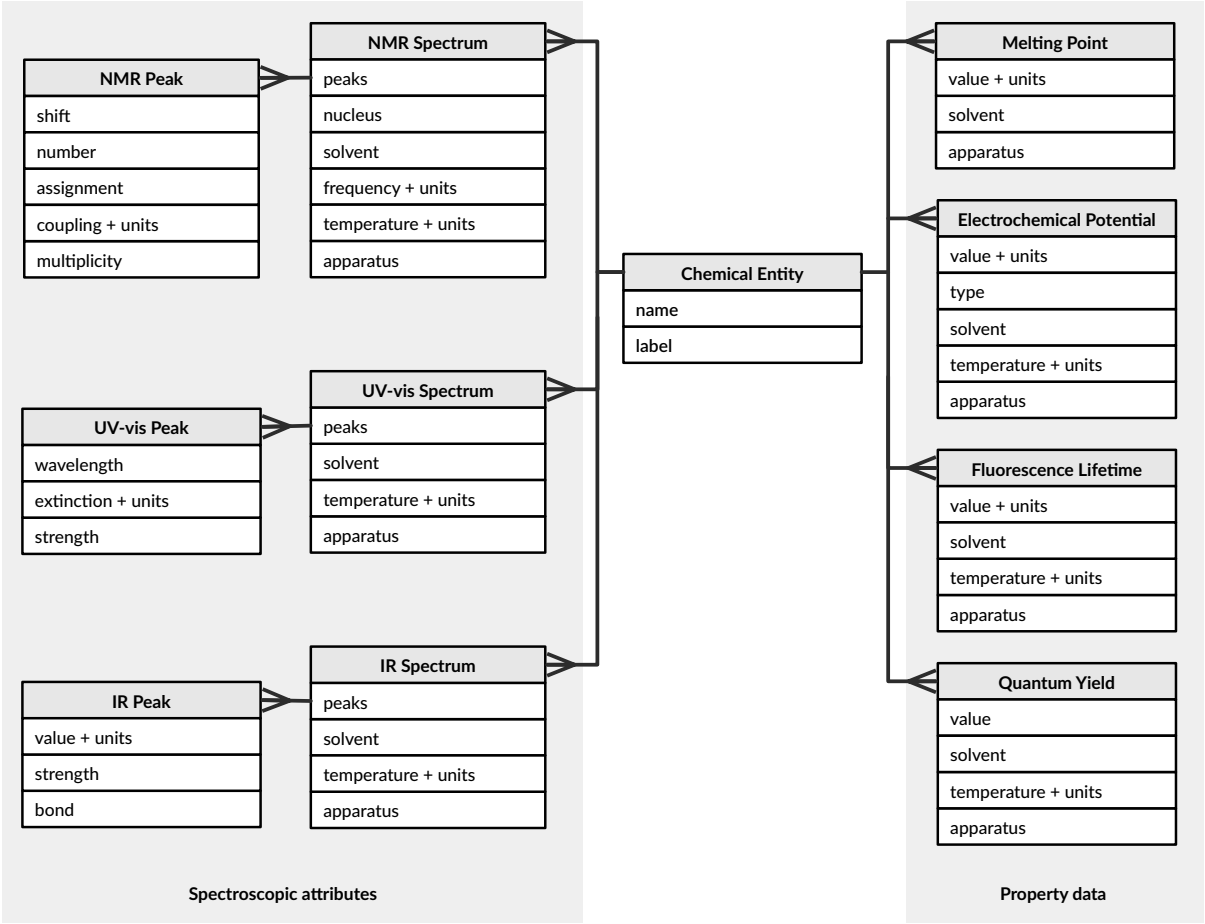

Figure 8: The data model for extracted chemical entities and their associated experimental properties and spectroscopic attributes, as currently provided by ChemDataExtractor. Users of the toolkit may extend this data model by defining their own custom parsers.

recognize.

Evaluation of information extraction from academic journals

Overall performance was evaluated by applying ChemDataExtractor to a test collection of 50 open-access chemistry articles that were selected from journals published by the ACS, the RSC and Springer. The precision, recall and F-score for the extraction of various different kinds of information were calculated by comparing ChemDataExtractor's output with a gold standard output[36] that was manually compiled especially for this evaluation. Strict guidelines were developed to ensure manual extraction was performed consistently.

Chemical entities, spectra and properties were extracted from the abstract, main text, tables and figure captions. For this evaluation, extraction of melting point, oxidation and reduction potentials, photoluminescence lifetime, and quantum yield properties was considered, as well as NMR, UV-vis, and IR spectroscopic attributes. The relevant chemical entities were restricted to those with associated spectra or properties, or with an assigned alphanumeric label. All information that is defined solely within a scheme or figure image or a separate supplementary information document was considered outside the scope of our system and was therefore excluded from the evaluation.

Table 3 presents the overall precision, recall, and F-score values for the extraction of chemical identifiers, spectroscopic attributes, and chemical properties. These evaluation metrics consider the extraction of an entire data record as an individual unit, as shown by the schema diagram in Figure 8. A record is considered a false negative if any part of the record is missing, a false positive if any part of the record is incorrect, and a true positive only if it is exactly correct.

The following sections present a more detailed evaluation of the individual components that make up each of the record types in Table 3.

Table 3: The precision, recall and F-score measures for the extraction of various kinds of information.

	Precision	Recall	F-Score
Chemical identifier records	94.1%	92.7%	93.4%
Spectrum records	88.3%	85.4%	86.8%
Chemical property records	93.5%	89.6%	91.5%

Chemical Identifiers

Table 4 presents an evaluation of ChemDataExtractor's ability to extract the names and alphanumeric labels of chemical entities in a document. Any identifier that is present in public chemical databases or is resolvable using IUPAC naming rules is considered a name, while all other identifiers that are typically only applicable within the context of the containing document are considered labels.

Chemical name extraction is primarily dependent on the performance of the underlying chemical named entity recognizers, while extraction of labels depends on their proximity to a recognized chemical name or their presence within a table.

An F-score of 93.4% was obtained when considering chemical records as a whole, reflecting ChemDataExtractor's ability to identify which names and alphanumeric labels correspond to the same chemical entity. Accurately matching alphanumeric labels to the relevant chemical name is a vital prerequisite to successfully extracting any spectra and properties that are defined solely in terms of a label.

Table 4: The precision, recall and F-score measures for the extraction of chemical identifiers.

	Precision	Recall	F-Score
Chemical names	97.4%	96.3%	96.8%
Alphanumeric labels	99.3%	97.3%	98.3%
Full chemical identifier records	**94.1%**	**92.7%**	**93.4%**

Spectroscopic attributes

An evaluation of ChemDataExtractor's ability to extract various spectroscopic attributes is shown in Table 5. As well as assessing the extraction of each individual spectrum attribute,

the extraction of overall spectrum records is presented.

Table 5: The precision, recall and F-score measures for the extraction of spectroscopic attributes.

	Precision	Recall	F-Score
Spectrum type	99.9%	96.7%	98.4%
Chemical subject	93.4%	90.3%	91.8%
Peak values	98.6%	95.4%	96.9%
Solvent	99.5%	96.7%	98.1%
Temperature	100%	87.5%	93.3%
Apparatus	96.9%	91.0%	93.8%
Full spectrum records	**88.3%**	**85.4%**	**86.8%**

Precision is consistently high across all attributes, and some even have no false positives at all within the test set. This is due to the rule-based nature of the parsers in ChemDataExtractor, which are well-suited to the formulaic language and structure of scientific articles that leave little room for mis-interpretation.

Contextual spectroscopic attributes such as temperature and apparatus present the greatest challenge in terms of resolving interdependencies between information that has been extracted from different document domains. For example, a spectrum may have peaks listed in a table, temperature mentioned in a figure caption, and apparatus mentioned in the main text, leading to the slightly lower recall values of 87.5% and 91.0% for temperature and apparatus respectively. The ability to accurately match together these disparate pieces of information is a unique strength of ChemDataExtractor, yet also presents the greatest opportunity for further improvement.

Chemical properties

Table 6 presents the precision, recall and F-score for the extraction of different aspects of chemical property information, as well as the overall property record.

Errors in property extraction typically occur where properties are reported in sentences within the main text, rather than in a table. In these cases, while the value and units are normally extracted without issue, complex sentence structures often result in a failure

Table 6: The precision, recall and F-score measures for the extraction of chemical properties.

	Precision	Recall	F-Score
Property value	100%	95.9%	97.9%
Property units	100%	94.8%	97.4%
Chemical subject	93.5%	89.6%	91.5%
Solvent	100%	94.4%	97.1%
Temperature	100%	88.9%	94.1%
Apparatus	100%	87.5%	93.3%
Full property records	**93.5%**	**89.6%**	**91.5%**

to assign a chemical subject. Errors in extraction from tables also occasionally arise from complex table structures; for example where some table cells are merged across multiple rows or columns, as this can introduce ambiguity around the exact scope of the cell contents.

Evaluation of information extraction from patents

The performance of ChemDataExtractor was also tested against patent documents. To this end, a case study on melting points was used for the evaluation since it offers a good comparison to the recent work of Tetko et al.,[15] who have generated a dataset of 241,958 melting points, which were mined from US patents using LeadMine[14] in combination with a customized version of ChemicalTagger.[11] A comparative evaluation was performed by applying ChemDataExtractor to a representative subset of the patents used in their study and comparing the extracted melting point records from the two generated datasets. The subset comprised a sample of 2,000 patents, which were drawn from a random selection of US patent grants that were published in the years 2005-2014 and were present in the melting point dataset published by Tetko et al..

In total, ChemDataExtractor obtained 18,180 unique melting points while Tetko et al. obtained 13,198 from this sample. There is an overlap of 8,978 melting points that match exactly between the two datasets, giving a shared total of 22,400 unique melting points. Therefore, in addition to the 8,978 (40%) that are common to both datasets, there are 9,202 (41%) that occur only in the ChemDataExtractor dataset and 4,220 (19%) that occur only

in the Tetko dataset. These non-matching subsets include a further 202 compounds that are present in both datasets but with differing melting point values. Of the overall set of 9,180 common compounds, 97.8% have identical melting points, while the remaining melting point pairings differ by a root mean squared deviation of 63 °C.

When interpreting these values, it is important to note that Tetko et al. removed duplicate values across their entire dataset, prior to the evaluation sample being taken, while the ChemDataExtractor duplicate values were only removed by considering the evaluation sample itself. The published Tetko dataset only references a single patent for each melting point, even if it was extracted from multiple ones, so many of the melting point values that appear to be missing from the LeadMine-extracted evaluation sample from 2,000 patents were in fact intentionally discarded because they were considered to be a duplicate of a value from another patent outside the sample. 2,557 compounds for which melting points appear to have been only extracted by ChemDataExtractor do in fact occur in the overall Tetko dataset as extracted from a different patent. If the benefit of the doubt is given and all of these cases are assumed to be duplicate removals, 11,535 (51%) of values are common to both tools, 6,645 (30%) occur only in the ChemDataExtractor dataset, and 4,220 (19%) occur only in the Tetko dataset. Tetko et al. also discarded an unspecified number of melting points that failed with descriptor calculation programs, which may further account for differences between the two datasets.

Despite these differences, Table 7 shows that there is good agreement between the datasets in terms of this classification according to the average melting point temperature, average molecular weight, and average number of non-hydrogen atoms in the compounds involved. Moreover, Figure 9 shows broad agreement in the distribution of melting point temperatures for the different datasets, including being able to reproduce the peaks at 250 °C and 300 °C.

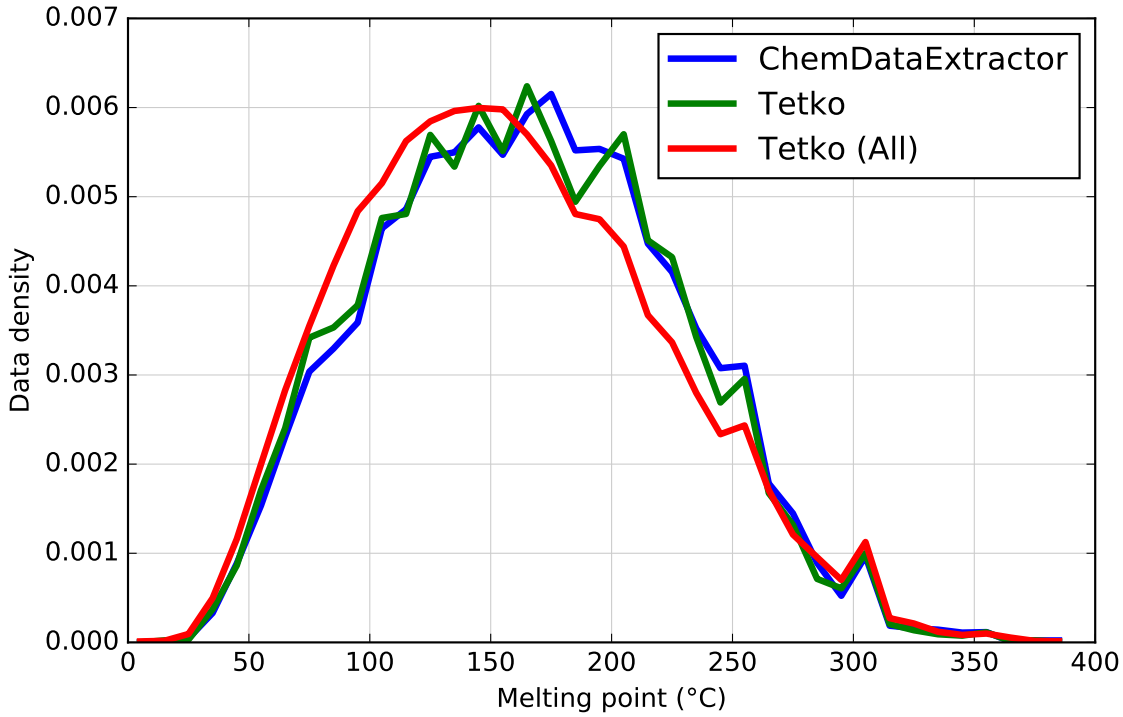

Figure 9: Data distributions of melting point temperature values extracted by ChemDataExtractor (blue), and Tetko (green) from the evaluation sample of 2000 patents. The distribution for the entire Tetko dataset of 241,958 melting points is also shown in red. Note that duplicate removal was performed on the entire Tetko dataset, prior to the evaluation sample being taken.

Table 7: The total number of melting point values (count), with the corresponding average temperature (T), average molecular weight (MW) and average number of non-hydrogen atoms (NA) of the compounds involved in the data records extracted by ChemDataExtractor and Tetko et al. for the evaluation sample of 2000 patents. Values for the entire Tetko dataset are also shown for comparison.

Dataset	Count	Average T (°C)	Average MW	Average NA
ChemDataExtractor	18,180	166.3	385.5	26.9
Tetko	13,198	164.1	385.1	27.0
Tetko (All)	241,958	159	357	25.0

Evaluation of natural language processing components

The performance of each individual component in the natural language processing pipeline can place an effective upper bound on the ability to accurately extract information. Imperfect performance in earlier stages carries through the pipeline and can degrade the performance of each subsequent stage. For example, incorrect POS tags can have a direct negative impact on both the recognition of chemical entities and the parsing of a sentence. Likewise, missing or incorrectly recognized chemical entity names can invalidate the extraction of all associated spectroscopic data and properties.

Therefore, it is important to quantify the performance of each component to identify the greatest barriers to improved performance of the overall system.

Chemical entity mentions

To facilitate comparison with other text mining tools, recognition of individual chemical mentions was evaluated. This evaluation was performed using the CHEMDNER corpus.[33] This consists of 3,000 abstracts from across the chemistry domain that have been manually annotated with 25,351 chemical entity mentions. Results were calculated using the `bc-evaluate` tool provided by the CHEMDNER organizers.

Table 8 shows the precision, recall and F-score for the individual CRF, dictionary and regular expression components of the chemical entity recognition system, as well as for the overall combined system. The overall combined F-score of 87.8% exceeds the scores

achieved by all the tools that officially entered the CHEMDNER chemical names extraction challenge.[32] The two best entries were tmChem by Leaman et al.,[37] which achieved an F-score of 87.39%, and a system by Lu et al.,[38] which achieved an F-score of 87.11%. Lu et al. have since published an alternative version of their system that achieved an F-score of 88.06%, which outperforms ChemDataExtractor on this statistic by 0.3%.

The CRF recognizer with word cluster features is the best performing individual component, with an F-score of 84.9%. While the dictionary recognizer has similar precision to the CRF, it has inferior recall, primarily caused by poor recognition of systematic names and chemical formulae. This weakness is typical of dictionary-based methods, where each chemical name must be present in the dictionary for it to be successfully recognized.

The regular expression recognizer is only designed to recognize a limited set of chemical identifier patterns, such as database registry numbers and chemical formulae, and therefore has poor recall of just 11.0% when applied on its own. However, these types of chemical identifiers can pose the greatest difficulty for the CRF and dictionary methods, and therefore the regular expression component still makes a worthwhile contribution to the overall combined system.

Table 8: Precision, recall and F-score of conditional random field (CRF), dictionary, and regular expression chemical named entity recognizers when used separately and in combination.

System	Precision	Recall	F-score
CRF	90.5%	80.0%	84.9%
Dictionary	88.6%	70.2%	78.3%
Regular expression	89.4%	11.0%	19.6%
Combined system	**89.1%**	**86.6%**	**87.8%**

POS tagging

POS tagging performance was evaluated through the use of two different corpora that have been manually annotated with POS tags: The Wall Street Journal (WSJ) corpus,[28] which consists of 1 million words from 1989 Wall Street Journal news articles, and the GENIA

corpus,[29] which consists of 2,000 MEDLINE abstracts that cover the biomedical domain. The standard WSJ splitting convention was used, with sections 0–18 for training, 19–21 for development, and 22–24 for evaluation. The first 90% of the GENIA corpus was used for training, and the remaining 10% for evaluation, matching the split used by Tsuruoka et al.[4] in developing a biomedical POS tagger.

Table 9 presents the POS tagging accuracy of different training configurations on the WSJ and GENIA evaluation corpora. Supervised training was performed using the WSJ and GENIA training corpora individually, and also both combined. The effect of adding unsupervised features from word clusters was also evaluated on each of these three configurations.

Table 9: POS tagging accuracy of different training systems evaluated on the WSJ and GENIA evaluation corpora.

Training system	WSJ	GENIA
WSJ	97.19%	83.50%
WSJ+clusters	97.23%	84.15%
GENIA	78.88%	98.53%
GENIA+clusters	81.19%	98.62%
WSJ+GENIA	97.02%	98.26%
WSJ+GENIA+clusters	97.08%	98.34%

Taggers trained individually on either the WSJ or the GENIA corpus achieved the best performance when evaluated on that same corpus, but afforded the poorest performance when evaluated on the opposite corpus. The tagger trained on the WSJ training corpus achieved an accuracy of 97.23% on the WSJ evaluation corpus, which falls to 84.15% on the GENIA evaluation corpus. Likewise, the tagger trained on the GENIA training corpus achieved an accuracy of 98.62% on the GENIA evaluation corpus, which falls to 81.19% on the WSJ evaluation corpus.

The tagger trained on the combined WSJ and GENIA corpora achieves good accuracy on both evaluation corpora, with 97.08% on the WSJ evaluation corpus and 98.34% on the GENIA evaluation corpus. This matches the observations of Tsuruoka et al., indicating that

using the combined newspaper and biomedical training sets extends coverage over both and has little negative impact compared to the specialized training for a specific domain.

The addition of word cluster features has a positive effect in all cases, but this is most significant in the cases where there is a mismatch between the training corpus and the evaluation corpus. The accuracy of the WSJ-trained tagger on the GENIA corpus rises from 83.50% to 84.15%, and the accuracy of the GENIA-trained tagger on the WSJ corpus rises from 78.88% to 81.19%. This suggests that unsupervised word cluster features are capable of broadening the coverage of a POS tagger outside the domain of its supervised training, and therefore should lead to improved performance across the wider chemistry domain for the tagger trained on the combined WSJ and GENIA corpus.

Conclusions

ChemDataExtractor is able to automatically extract chemical information from scientific documents, facilitating the creation of massive chemical databases with minimal time and effort. The system consists of a modular document processing pipeline with extensible components for natural language processing that achieve state-of-the-art performance for POS tagging and chemical named entity recognition.

In contrast to most existing text-mining systems that focus on extracting entities from individual sentences, ChemDataExtractor provides a table processor for extraction of tabulated experimental properties and document-level processing algorithms to resolve data interdependencies and produce unified chemical records that incorporate information from multiple document domains. Evaluations demonstrate good performance in the extraction of chemical entities and their associated experimental properties and spectroscopic data.

The generic and extensible design of the system means it can applied to the extraction of any chemical properties, measurements and relationships with minimal additional effort. This leads to the ultimate goal of quickly auto-generating chemical structure and property

databases for materials science and other fields. Future work will focus on extending the system to further property and spectrum types, and improving the performance of individual natural language processing components.

Software and Data Availability

ChemDataExtractor is released under the MIT license and is available to download from `http://chemdataextractor.org`. An interactive online demo is available at `http://chemdataextractor.org/demo` and a user guide, code examples and full API documentation are available at `http://chemdataextractor.org/docs`. Datasets produced by ChemDataExtractor as part of the evaluation are available at `http://chemdataextractor.org/evaluation`, including melting points extracted from 2,000 patents and full data records extracted from 50 journal articles.

Acknowledgement

M.C.S. is grateful to the EPSRC for a DTA PhD studentship (Grant No. EP/J500380/1). J.M.C. is indebted to the 1851 Royal Commission for the 2014 Design Fellowship.

Notes and References

(1) National Science and Technology Council,; Office of Science and Technology Policy, *Materials Genome Initiative for Global Competitiveness*; 2011.

(2) Olivares-Amaya, R.; Amador-Bedolla, C.; Hachmann, J.; Atahan-Evrenk, S.; Sanchez-Carrera, R. S.; Vogt, L.; Aspuru-Guzik, A. Accelerated Computational Discovery of High-performance Materials for Organic Photovoltaics by Means of Cheminformatics. *Energy Environ Sci* **2011**, *4*, 4849–4861.

(3) Jain, A.; Ong, S. P.; Hautier, G.; Chen, W.; Richards, W. D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; Persson, K. A. Commentary: The Materials Project: A Materials Genome Approach to Accelerating Materials Innovation. *APL Mater.* **2013**, *1*, 011002.

(4) Tsuruoka, Y.; Tateishi, Y.; Kim, J.-D.; Ohta, T.; McNaught, J.; Ananiadou, S.; Tsujii, J. In *Advances in Informatics*; Bozanis, P., Houstis, E. N., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2005; pp 382–392.

(5) Fundel, K.; Küffner, R.; Zimmer, R. Relex—Relation Extraction Using Dependency Parse Trees. *Bioinformatics* **2007**, *23*, 365–371.

(6) Zweigenbaum, P.; Demner-Fushman, D.; Yu, H.; Cohen, K. B. Frontiers of Biomedical Text Mining: Current Progress. *Brief. Bioinform.* **2007**, *8*, 358–375.

(7) Simpson, M. S.; Demner-Fushman, D. *Mining Text Data*; Springer US: Boston, MA, 2012; pp 465–517.

(8) Eltyeb, S.; Salim, N. Chemical Named Entities Recognition: A Review on Approaches and Applications. *J. Cheminf.* **2014**, *6*, 1–12.

(9) Vazquez, M.; Krallinger, M.; Leitner, F.; Valencia, A. Text Mining for Drugs and Chemical Compounds: Methods, Tools and Applications. *Mol. Inf.* **2011**, *30*, 506–519.

(10) Gurulingappa, H.; Mudi, A.; Toldo, L.; Hofmann-Apitius, M.; Bhate, J. Challenges in Mining the Literature for Chemical Information. *RSC Adv.* **2013**, *3*, 16194–16211.

(11) Hawizy, L.; Jessop, D.; Adams, N.; Murray-Rust, P. ChemicalTagger: A Tool for Semantic Text-mining in Chemistry. *J. Cheminf.* **2011**, *3*, 1–13.

(12) Parr, T. J.; Quong, R. W. ANTLR: A Predicated-LL(k) Parser Generator. *Software: Practice and Experience* **1995**, *25*, 789–810.

(13) Jessop, D. M.; Adams, S. E.; Willighagen, E. L.; Hawizy, L.; Murray-Rust, P. OSCAR4: A Flexible Architecture for Chemical Text-mining. *J. Cheminf.* **2011**, *3*, 1–12.

(14) Lowe, D. M.; Sayle, R. A. LeadMine: A Grammar and Dictionary Driven Approach to Entity Recognition. *J. Cheminf.* **2015**, *7*, S5.

(15) Tetko, I. V.; Lowe, D. M.; Williams, A. J. The Development of Models to Predict Melting and Pyrolysis Point Data Associated with Several Hundred Thousand Compounds Mined from PATENTS. *J. Cheminf.* **2016**, *8*, 1–18.

(16) Tharatipyakul, A.; Numnark, S.; Wichadakul, D.; Ingsriswang, S. ChemEx: Information Extraction System for Chemical Data Curation. *BMC Bioinformatics* **2012**, *13*, S9.

(17) Filippov, I. V.; Nicklaus, M. C. Optical Structure Recognition Software To Recover Chemical Information: OSRA, An Open Source Solution. *J. Chem. Inf. Model* **2009**, *49*, 740–743.

(18) Shinyama, Y. PDFMiner. `https://euske.github.io/pdfminer/`, [Online; accessed 4-August-2016].

(19) Kiss, T.; Strunk, J. Unsupervised Multilingual Sentence Boundary Detection. *Comput. Linguist.* **2006**, *32*, 485–525.

(20) Read, J.; Dridan, R.; Oepen, S.; Solberg, L. J. Sentence Boundary Detection: A Long Solved Problem? Proceedings of COLING 2012: Posters. 2012; pp 985–994.

(21) Turian, J.; Ratinov, L.; Bengio, Y. Word representations: A Simple and General Method for Semi-supervised Learning. **2010**, 384–394.

(22) Brown, P. F.; deSouza, P. V.; Mercer, R. L.; Pietra, V. J. D.; Lai, J. C. Class-based N-gram Models of Natural Language. *Comput. Linguist.* **1992**, *18*, 467–479.

(23) Miller, S.; Guinness, J.; Zamanian, A. Name Tagging with Word Clusters and Discriminative Training. HLT/NAACL. 2004; pp 337–342.

(24) Ganchev, K.; Crammer, K.; Pereira, F.; Mann, G.; Bellare, K.; McCallum, A.; Carroll, S.; Jin, Y.; White, P. Penn/Umass/CHOP Biocreative II Systems. 2007; pp 119–124.

(25) Täckström, O.; McDonald, R.; Uszkoreit, J. Cross-lingual Word Clusters for Direct Transfer of Linguistic Structure. HLT/NAACL. 2012; pp 477–487.

(26) Owoputi, O.; O'Connor, B.; Dyer, C.; Gimpel, K.; Schneider, N.; Smith, N. A. Improved Part-of-Speech Tagging for Online Conversational Text with Word Clusters. HLT/NAACL. 2013; pp 380–390.

(27) Liang, P. Semi-Supervised Learning for Natural Language. M.Sc. thesis, Massachusetts Institute of Technology, 2005.

(28) Bies, A.; Mott, J.; Warner, C. *English News Text Treebank: Penn Treebank Revised LDC2015T13*; Linguistic Data Consortium: Philadelphia, 2015.

(29) Tateishi, Y.; Tsujii, J. Part-of-Speech Annotation of Biology Research Abstracts. LREC. 2004.

(30) Okazaki, N. CRFsuite: A Fast Implementation of Conditional Random Fields (CRFs). 2007; `http://www.chokkan.org/software/crfsuite/`.

(31) Rocktaschel, T.; Weidlich, M.; Leser, U. ChemSpot: A Hybrid System for Chemical Named Entity Recognition. *Bioinformatics* **2012**, *28*, 1633–1640.

(32) Krallinger, M.; Leitner, F.; Rabal, O.; Vazquez, M.; Oyarzabal, J.; Valencia, A. CHEMDNER: The Drugs and Chemical Names Extraction Challenge. *J. Cheminf.* **2015**, *7*, S1.

(33) Krallinger, M.; Rabal, O.; Leitner, F.; Vazquez, M.; Salgado, D.; Lu, Z.; Leaman, R.; Lu, Y.; Ji, D.; Lowe, D. M.; Sayle, R. A.; Batista-Navarro, R. T.; Rak, R.; Huber, T.; Rocktäschel, T.; Matos, S.; Campos, D.; Tang, B.; Xu, H.; Munkhdalai, T.; Ryu, K. H.; Ramanan, S. V.; Nathan, S.; Žitnik, S.; Bajec, M.; Weber, L.; Irmer, M.; Akhondi, S. A.; Kors, J. A.; Xu, S.; An, X.; Sikdar, U. K.; Ekbal, A.; Yoshioka, M.; Dieb, T. M.; Choi, M.; Verspoor, K.; Khabsa, M.; Giles, C. L.; Liu, H.; Ravikumar, K. E.; Lamurias, A.; Couto, F. M.; Dai, H.-J.; Tsai, R. T.; Ata, C.; Can, T.; Usié, A.; Alves, R.; Segura-Bedmar, I.; Martínez, P.; Oyarzabal, J.; Valencia, A. The CHEMDNER Corpus of Chemicals and Drugs and Its Annotation Principles. *J. Cheminf.* **2015**, *7*, S2.

(34) Hettne, K. M.; Stierum, R. H.; Schuemie, M. J.; Hendriksen, P. J. M.; Schijvenaars, B. J. A.; Mulligen, E. M. v.; Kleinjans, J.; Kors, J. A. A Dictionary to Identify Small Molecules and Drugs in Free Text. *Bioinformatics* **2009**, *25*, 2983–2991.

(35) Schwartz, A. S.; Hearst, M. A. A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text. PSB 2003. 2003.

(36) The manually-extracted gold standard output is available from `http://chemdataextractor.org/evaluation` along with the full text of the 50 source articles.

(37) Leaman, R.; Wei, C.-H.; Lu, Z. tmChem: A High Performance Approach for Chemical Named Entity Recognition and Normalization. *J. Cheminf.* **2015**, *7*, S3.

(38) Lu, Y.; Ji, D.; Yao, X.; Wei, X.; Liang, X. CHEMDNER System with Mixed Conditional Random Fields and Multi-Scale Word Clustering. *J. Cheminf.* **2015**, *7*, S4.

Graphical TOC Entry