2013

A Hybrid Approach to Finding Relevant Social Media Content for Complex Domain Specific Information Needs

Delroy H. Cameron
Wright State University - Main Campus, cameron.20@wright.edu

Amit P. Sheth
Wright State University - Main Campus, amit@sc.edu

Nishita Jaykumar
Wright State University - Main Campus, jaykumar.2@wright.edu

Gaurish Anand
Wright State University - Main Campus, anand.7@wright.edu

Krishnaprasad Thirunarayan
Wright State University - Main Campus, t.k.prasad@wright.edu

See next page for additional authors

A Hybrid Approach to Finding Relevant Social Media Content for Complex Domain Specific Information Needs

Delroy Cameron, Amit P. Sheth, Nishita Jaykumar, Krishnaprasad Thirunarayan, Gaurish Anand, Gary A. Smith

[a] *Ohio Center of Excellence in Knowledge-enabled Computing (Kno.e.sis)*
Wright State University, Dayton OH 45435, USA

Abstract

While contemporary semantic search systems offer to improve classical keyword-based search, they are not always adequate for complex domain specific information needs. The domain of prescription drug abuse, for example, requires knowledge of both ontological concepts and "intelligible constructs" not typically modeled in ontologies. These intelligible constructs convey essential information that include notions of intensity, frequency, interval, dosage and sentiments, which could be important to the holistic needs of the information seeker. In this paper, we present a hybrid approach to domain specific information retrieval that integrates ontology-driven query interpretation with synonym-based query expansion and domain specific rules, to facilitate search in social media on prescription drug abuse. Our framework is based on a context-free grammar (CFG) that defines the query language of constructs interpretable by the search system. The grammar provides two levels of semantic interpretation: 1) a top-level CFG that facilitates retrieval of diverse textual patterns, which belong to broad templates and 2) a low-level CFG that enables interpretation of specific expressions belonging to such textual patterns. These low-level expressions occur as concepts from four different categories of data: 1) ontological concepts, 2) concepts in lexicons (such as emotions and sentiments), 3) concepts in lexicons with only partial ontology representation, called *lexico-ontology* concepts (such as side effects and routes of administration (ROA)), and 4) domain specific expressions (such as date, time, interval, frequency and dosage) derived solely through rules. Our approach is embodied in a novel Semantic Web platform called PREDOSE, which provides search support for complex domain specific information needs in prescription drug abuse epidemiology. When applied to a corpus of over 1 million drug abuse-related web forum posts, our search framework proved effective in retrieving relevant documents when compared with three existing search systems.

Keywords: Semantic Search, Domain Specific Information Retrieval, Complex Information Needs, Ontology, Background Knowledge, Context-Free Grammar

1. Introduction

The use of structured background knowledge (ontologies) to enhance search has gained considerable traction among contemporary information retrieval systems. Ontologies offer to improve search by capturing the meaning of real-world concepts and their associations. The formal representations modeled in ontologies have been used to positively impact many complex tasks, including interoperability, personalization and knowledge discovery.

While semantic search has gained credibility, compared to classical keyword-based and hyperlinked-based search, there is often a misalignment between the information needs of users and the knowledge model developed to meet such needs. Ontologies provide a means for interpreting some elements of complex information needs, but not all aspects of such needs [1]. The main issue is that ontologies often have limited scope, while users are unrestricted in the range of information they can seek on a given topic. A user information need can transcend data types and sources, exceeding what is formally modeled.

In spite of this, many semantic search applications [2, 3, 4, 5], semantic search engines (Hakia, Bing) and hybrid information retrieval approaches [1, 6, 7, 8] rely heavily on ontologies for query interpretation. While these approaches serve their intended purpose, they are generally unsuitable for domain specific applications, such as prescription drug abuse. General-purpose search engines such as Google and Yahoo that rely on keyword-based and hyperlinked-based models, may not perform well on domain specific data. This is because minimal (and often inadequate) support is provided for interpreting the additional elements that could be important to an information need, but not formally captured by the knowledge model.

We address this problem by developing and evaluating a hybrid approach to search that allows query specification and interpretation of diverse expressions, which are involved in various aspects of complex information needs. To illustrate our approach, consider a scenario in which an epidemiologist in the domain of prescription drug abuse is seeking insights into emerging patterns and trends in drug abuse using social media. For brevity, we present only one of many information needs explored in PREDOSE (`http://wiki.knoesis.org/index.php/PREDOSE`).

*Corresponding Author. Tel.: +1 937 775 5213; fax: +1 937 775 5133
Email address: `delroy@knoesis.org` (Delroy Cameron)

Information Need: *How are drug users engaging in the use of the semi-synthetic opioid Buprenorphine, through excessive daily dosage?*

Inherent in this information need is the following relevant background knowledge. Buprenorphine is an opioid antagonist used in the treatment of opioid addiction, including addiction to Heroin, OxyContin and Vicodin. Prescribed daily dosage varies by individual ranging from 4–32mg[1]. Buprenorphine is known to stabilize drug users from withdrawal symptoms, but can also induce an opiated effect. This treatment drug is therefore at risk for abuse. Epidemiologists are interested in understanding the dosage practices of Buprenorphine users, including amounts taken, frequency of use and side effect experienced, to better understand emerging patterns and trends of abuse.

A suitable user query provided by a domain expert may involve the following keywords: "buprenorphine dosage exceed 4mg daily." A robust search system may correctly interpret the keyword 'buprenorphine' as the standard DBpedia resource: http://dbpedia.org/resource/Buprenorphine. Then through non-trivial query expansion, the system may also associate the keywords 'bupe,' 'bupey,' 'suboxone,' 'subbies' and 'suboxone film,' with 'Buprenorphine,' as synonyms. Likewise, the search system may expand the keyword 'daily' with the synonyms 'day,' 'night,' 'morning' and 'afternoon,' using available (or manually created) lexicons that contain such mappings. However, the intricate challenge is interpreting the notion of excessive dosage, expressed as the phrase "dosage exceed 4mg."

In the development of Active Semantic Electronic Medical Records (ASEMR), Sheth et. al. [9] created rules expressed in RDQL [10] (precursor to SPARQL) to enable specification of additional constructs (including dosage) that compensate for deficiencies in the knowledge model. Similarly, in the Semantic Content Organization and Retrieval Engine (SCORE) [11, 12], Hammond et. al. implemented various rules derived using regular expressions to specify quantity-conveying metadata (such as 'currency,' 'percentage,' 'amount,' 'time' and 'dates') which were not present in the ontology. In the Knowledge and Information Management platform (KIM) [13], Popov et. al. modeled various lexical resources in the ontology such as currency, dates and abbreviations, which were subsequently used for document annotation. However, the information need presented here requires a more in-depth interpretation.

To appropriately interpret excessive dosage, the notion of dosage itself must first be specified using its constituent members: DOSAGE-OPERATOR (e.g., '>,' '<'), DOSAGE-AMOUNT (e.g., '4,' '10') and DOSAGE-UNIT (e.g., 'mg,' 'tablet'). In this way, the search term '>4mg' could be an abstraction of the search phrase "dosage exceed 4mg." Rules must then be used to interpret each constituent according to what is possible in the corpus. This is important because a DOSAGE-UNIT may have various lexical representations in text (e.g., mg, milligram, milli-gram). Likewise, the DOSAGE-OPERATOR can have multiple equivalent manifestations (such as '>', 'greater than,' 'more than' and 'above'). Similarly, DOSAGE-AMOUNT can be numeric or textual. According to these possible interpretations, '6mg,' 'ten milligrams,' 'about 8mgs,' 'a bit more than 30 milli-grams' etc, are all valid expressions for the query 'dosage exceed 4mg.' Given this representation, the matching documents for the entire query ("buprenorphine dosage exceed 4mg daily") filtering heuristics are then applied to text fragments in the corpus that match the interpretation of each query component. In this way, a hybrid approach to information retrieval would have been utilized, which leverages ontologies, lexicons and rules for query interpretation of domain specific data.

Concretely, our approach is based on a context-free grammar (CFG) that defines the query language of constructs interpretable by the search system. The grammar provides two levels of semantic interpretation: 1) a top-level CFG defines broad patterns that can be interpreted by the system and 2) a low-level CFG defines the specific interpretation of elements within user queries. The query language of the grammar is specified in a declarative information extraction specification called SystemT [14, 15], which is designed for information extraction from heterogeneous texts. This is advantageous because the rules developed using SystemT can ported to other texts in other domains. Some of these domains specifically: 1) biomaterials and materials science, 2) cannabis and synthetic cannabinoid research and 3) clinical texts on cardiology reports.

In an evaluation using a corpus of over 1 million web forum posts related to drug abuse, our hybrid search system retrieved a larger number of relevant documents when compared with three existing search systems. These systems are the: 1) semantic search engine Hakia, 2) crowd sourcing-based search engine DuckDuckGo and 3) popular search engine Google. Note that since these search engines are not specifically engineered to handle domain specific data, our results are not surprising. However, our experiments highlight the need for more effective approaches to domain specific search as noted in [16]. The specific contributions of this research are as follows:

- We develop a hybrid approach to domain specific information retrieval that interprets four categories of data. These are: 1) structured background knowledge in ontologies, 2) concepts in lexicons; 3) concepts in lexicons with partial ontology representation called *lexico-ontology* concepts (see Section 2.1.2) and 4) concepts defined using rules.
- We utilize a CFG to formally define the query language of strings interpretable by the system. The CFG provides two levels of semantic interpretation: 1) a top-level CFG for interpreting general textual patterns, and 2) a low-level CFG for interpreting specific expressions.
- We show that our approach is effective through an evaluation against three popular search systems.

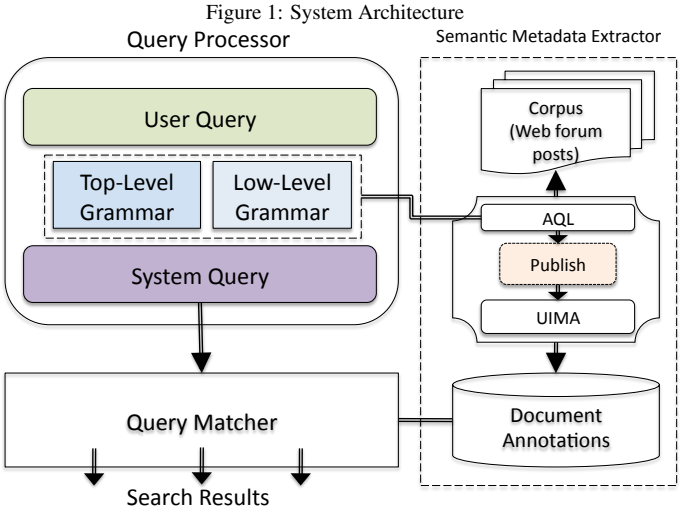
The rest of the paper is organized as follows. Section 2 describes the overall hybrid information retrieval framework, which includes modules for query interpretation in Section 2.1, semantic metadata extraction/document annotation in Section 2.2 and query matching in Section 2.3. Section 3 describes the evaluation and Section 4 covers related work.

[1]Note that the actual amounts used in examples throughout this manuscript are anecdotal only

2. Approach

Our hybrid information retrieval system (shown in Figure 1) consists of three components: 1) Query Processor, 2) Semantic Metadata Extractor and 3) Query Matcher. The *query processor* provides functionality for template-based query specification and domain specific query interpretation. The *semantic metadata extractor* identifies the offsets of text snippets that match the query interpretation in the corpus. The *query matcher* retrieves and filters the relevant documents for a given user query, based on query interpretation and document annotations in the corpus. Each component is discussed in detail in the following subsections.

Figure 1: System Architecture



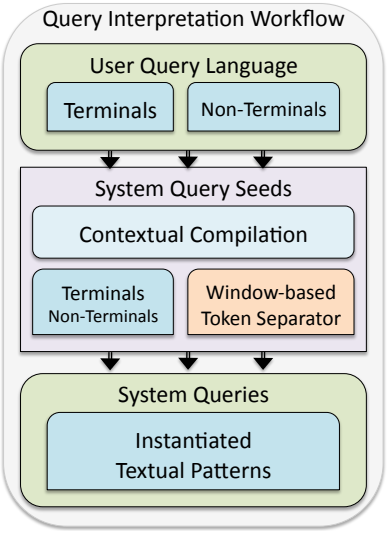
2.1. Query Processor

The process of searching for information from text commonly involves certain interactions between a user and a system. Users typically possess a conceptualization of their information need that can be framed using a mental model, as noted by Tran et al. [2]. The search system must provide an environment for users to adequately express their information need in terms of language primitives or a *user query language* that can be understood by the system (Figure 2, top left). The system must then provide a specification for translating the user query into a *system query* (Figure 2, left center), based on the interpretation of the user query. In our application, the query processor provides a *user query interface* for users to specify their queries. It then performs the translation from user query into system query based on the underlying specification in the grammar. To account for domain specific constructs in the knowledge model, user queries are specified using templates, instead of free-form queries. These templates abstract data from the aforementioned (four) categories of data elements, which are important to the domain. The CFG is presented in the next section.

2.1.1. Context-Free Grammar

The context-free grammar used in our hybrid information retrieval system is formally defined in this section, along with anecdotal examples to illustrate how it is used in practice. Definitions 1-3 cover the top-level grammar, while Definitions 4-7 cover the low-level grammar.

Figure 2: Workflow for translation of user queries into system queries



Definition 1. *The context-free grammar G for the query language U of the hybrid information retrieval system H is a quadruple (N, T, P, S), where N is a finite set of non-terminals, T is a finite set of terminals (or alphabet), P is a finite set of rules (or productions) and S is a Start Symbol.*

The set of nonterminals N is partitioned into two sets, N^S and N^P, where N^S denotes the set of nonterminals found directly in the right-hand sides (RHS) of the productions associated with the start symbol S, and $N^P = N - N^S$, where the symbol '–' is the set-difference operator. The set N^S contains 11 nonterminals, including ⟨DOSAGE⟩, ⟨FREQUENCY⟩ and route of administration ⟨ROA⟩ (see Table 2, Section 2.1.2 and Appendix A), which abstracts broad template classes of data relevant to the domain. The user query language of H is formally defined as follows:

Definition 2. *The user query language U of the hybrid information retrieval system H is the set of sentential forms over $(N^S \cup T)^*$ derivable from G = (N, T, P, S). That is, a user query q, may consist of terminals and nonterminals that appear only in the start symbol productions.*

For example, the production: ⟨TEMPLATE PATTERN⟩ → ⟨ENTITY⟩ ⟨PRONOUN⟩ ⟨DOSAGE⟩ ⟨INTENSITY⟩ is a broad template pattern that abstracts the information need given in Section 1 in G, where ⟨TEMPLATE PATTERN⟩ is the start symbol, ⟨ENTITY⟩, ⟨PRONOUN⟩, ⟨DOSAGE⟩ and ⟨INTENSITY⟩ are nonterminals, or template classes, in N^S. The more specific user query: ⟨q⟩ ::= ⟨Buprenorphine⟩ ⟨PERSONAL PRONOUN⟩ ">4mg" ⟨BY DAY | BY HOUR⟩ is a valid user query for the given information need, derived from this production, where ⟨Buprenorphine⟩ is a member of the template class ⟨ENTITY⟩, ⟨PERSONAL PRONOUN⟩ is a member of

Table 1: Derivation of system query strings using the CFG

1	⟨TEMPLATE PATTERN⟩	::= ⟨ENTITY⟩		⟨PRONOUN⟩		⟨DOSAGE⟩		⟨INTENSITY⟩	
2	User Query	::= ⟨Buprenorphine⟩		⟨PERSONAL PRONOUN⟩		"›4mg"		⟨BY DAY | BY HOUR⟩	
3	System Query Seed	::= ⟨Subs⟩	⟨RANGE⟩	I	⟨RANGE⟩	"32mg"	⟨RANGE⟩	"a day"	
4	System Query	::= Subs I was taking 32mg a day							
5	⟨TEMPLATE PATTERN⟩	::= ⟨ENTITY⟩		⟨PRONOUN⟩		⟨DOSAGE⟩		⟨INTENSITY⟩	
6	User Query	::= ⟨vicodin⟩		⟨PERSONAL PRONOUN⟩		"›28mg"		⟨BY DAY | BY HOUR⟩	
7	System Query Seed	::= ⟨vicodin⟩	⟨RANGE⟩	I	⟨RANGE⟩	"32mg"	⟨RANGE⟩	"every day"	
8	System Query	::= vicodin habit and I was taking 28mg of buprenorphine every day							

the template class ⟨PRONOUN⟩, ⟨BY DAY⟩ and ⟨BY HOUR⟩ are nonterminals, derived from the template class ⟨INTERVAL⟩. The expression ">4mg" is a terminal, which requires special interpretation (discussed later in this section). User queries may therefore consist of permutations of terminals and nonterminals, or sentential forms in G. Lines 2 and 6 in Table 1, show two valid user queries derived from the given production. Lines 4 and 8 show two specific system queries (which are matching text snippets in the corpus) derived from the user queries in lines 2 and 6.

The translation of user queries into system queries is a two-step process. *System query seeds* must first be generated from user queries (Figure 2, center) and then transformed into system queries by instantiating a window-based token separator. For instance, the system query seed: "Subs ⟨RANGE⟩ I ⟨RANGE⟩ 32mg ⟨RANGE⟩ a day," is generated from the user query: ⟨q⟩ → ⟨Buprenorphine⟩ ⟨PERSONAL PRONOUN⟩ ">4mg" ⟨BY DAY | BY HOUR⟩, where "Subs" is a synonym for "Buprenorphine," "I" is a personal pronoun, "32mg" is greater than "4mg" and "a day" is an expression for "by day". Upon instantiating the three successive ⟨RANGE⟩ values that capture window size to 0, 2 and 0 respectively, the specific system query: "Subs I was taking 32mg a day," can be obtained. The sequence of tokens that occupy the range are shown in underline. Given this example, the interpretation of a user query is therefore defined as follows:

Definition 3. *The interpretation of a user query $q \in U \subseteq (N^S \cup T)^*$ is a set of all terminal strings derivable from q in the grammar Q = (N, T, P, q), where the Start Symbol is replaced by q, a single sentential form obtained over $(N^S \cup T)$.*

The previous definitions cover the top-level grammar. However, the precise translation of a user query into system query seeds requires additional preliminaries, especially to interpret compound expressions such as ">4mg." The DOSAGE-OPERATOR, DOSAGE-AMOUNT and DOSAGE-UNIT have instantiations that require appropriate expansion. For example, "much more than 4mg", "five mg," "60 milligrams" and "a hundred milligrams" are valid interpretations for the query fragment ">4mg." To capture this, we introduce the notion of a *contextual compilation* (Figure 2, center) to formalize the translation of any terminal to its semantic equivalent according to its interpretation in G. Let *cc(">4mg," ⟨DOSAGE⟩)* denote the contextual compilation of the expression ">4mg," which belongs to the class ⟨DOSAGE⟩. Formally:

Definition 4. *A contextual compilation cc of a terminal string t derived from a nonterminal A is the set of terminal strings semantically equivalent to t in the context of A.*

It follows then that the interpretation of a user query therefore requires interpretation of both nonterminals and terminals, whenever the latter contains equivalent interpretations.

Definition 5. *The translation $\Gamma(t)$ of a terminal t derivable from the nonterminal $A \in N^S$ in G = (N, T, P, S) is its contextual compilation $\Gamma(t) = cc(t, A)$.*

Definition 6. *The translation $\Gamma(A)$ of a nonterminal A in the grammar G = (N, T, P, S) is defined as the set of terminal strings that can be derived from A. That is, $\Gamma(A) = \{t \mid A \Rightarrow_G^* t\}$*

Note that this translation of terminals can be specified by domain experts or by search engine developers programmatically. For example, the translation of the 'greater than' operator is specified explicitly as: $\langle greaterThanOp \rangle \rightarrow >$ | **greater than | more than | above | in excess of | . . .** in the grammar. Given the definition of a user query, system query seeds can then be formally defined based on the translation of user query elements and the window-based ⟨RANGE⟩ token separator.

Definition 7. *The system query seeds of a user query $q = \alpha_1, \alpha_2, \ldots, \alpha_n$ where $\alpha_i \in (N^S \cup T)$, is the cross-product of a the translation of the terminals and nonterminals $\Gamma(\alpha_1) \times \Gamma(\alpha_2) \times \ldots \times \Gamma(\alpha_n)$ that comprise the user query.*

As noted, system query seeds become actual system queries when the ⟨RANGE⟩ operator is instantiated. For example, Table 1 shows how the two system queries: 1) "Subs I was taking 32mg a day" and "vicodin habit and I take 28mg of buprenorphine every day" could be derived from the production: ⟨TEMPLATE PATTERN⟩ → ⟨ENTITY⟩ ⟨PRONOUN⟩ ⟨DOSAGE⟩ ⟨INTENSITY⟩, based on the grammar. The production contains nonterminals in the RHS, which are in N^S called *template classes*. The user first selects the broad template pattern and then constructs a more specific user query, where appropriate. The grammar then generates the system query seeds, which become system queries after instantiating the ⟨RANGE⟩ values. Documents that contain textual patterns that match system queries in the corpus are considered as candidate matches for the user query.

Note that the system query for the second user query (Table 1, line 8) could also be considered a match for first user

Table 2: Template Class Classification

	Template Class Name	Class Source	Class Type
1	⟨INTERVAL⟩	Alphabet	Compound
2	⟨DOSAGE⟩	Alphabet	Compound
3	⟨FREQUENCY⟩	Alphabet	Compound
4	⟨ENTITY⟩	Ontology	Simple
5	⟨ROA⟩	Lexico-ontology	Simple
6	⟨DRUGFORM⟩	Lexico-ontology	Simple
7	⟨SIDEEFFECT⟩	Lexico-ontology	Simple
8	⟨EMOTION⟩	Lexicon	Simple
9	⟨PRONOUN⟩	Lexicon	Simple
10	⟨INTENSITY⟩	Lexicon	Simple
11	⟨SENTIMENT⟩	Lexicon	Simple

query in Table 1, line 2. This is because initially, all annotations are retrieved for a given template and then filtered by the query matcher. The combination of "vicodin" as an ⟨ENTITY⟩ in the first position, separated by a ⟨PERSONAL PRONOUN⟩ ("I"), then a ⟨DOSAGE⟩ exceeding the prescribed limit ("28mg") and then an ⟨INTENSITY⟩ ("every day"), could be a match because "buprenorphine" appears as a concept in one of the ⟨RANGE⟩ separators. In the next section we discuss how the four categories of data are represented in the knowledge model and interpreted by the grammar.

2.1.2. Knowledge Model

In Definition 1, we described the set of nonterminals N^S as the set of nonterminals in the RHS of productions that begin with the start symbol ⟨TEMPLATE PATTERN⟩. For example, in the production: ⟨TEMPLATE PATTERN⟩ → ⟨ENTITY⟩ ⟨PRONOUN⟩ ⟨DOSAGE⟩ ⟨INTENSITY⟩, the RHS elements ⟨ENTITY⟩, ⟨PRONOUN⟩, ⟨DOSAGE⟩ and ⟨INTENSITY⟩ are in the set N^S. According to the grammar, N^S consists of 11 nonterminals or template classes (shown in Table 2). These classes cover the four categories of data interpretable by the system: 1) concepts in ontologies; 2) concepts in lexicons; 3) lexico-ontology concepts and 4) intelligible constructs specified using rules. The ability to interpret these categories of data is a key contribution, which is crucial to the effectiveness of our system for domain specific information retrieval. In the next section, we begin with the interpretation of the template class, which are based on the ontology.

Ontology-based Query Interpretation: To facilitate ontology-based query interpretation, we utilize a Drug Abuse Ontology (DAO) (*pronounced dow*) [17], which was created as part of the PREDOSE project. The DAO consists of 43 classes and 20 properties, and serves two main purposes. First, it facilitates query interpretation, and second it serves as one of the annotation schemes for metadata extraction (discussed in Section 2.2). The DAO is important for query interpretation because it captures various mappings between slang terms and standard drug references. In a gold standard dataset consisting of 600 web forum posts, we observed a ratio of 33:1 slang references for the standard drug label for the prescription drug "Buprenorphine" and 24:1 for "Loperamide." The DAO is therefore of critical importance.

To perform such query interpretation based on the DAO, let the drug abuse ontology O be represented as a graph $O = (V, E)$, where V is the set of nodes, which formally represent real-world concepts $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots, e_m\}$ is the set of edges, which represent labeled edges (or relationships) between such concepts. The interpretation of a keyword k_i in q according to the ontology O, denoted $I(k_i)$, is some concept v_i in O, where the concept v_i may be a class c_i or an instance r_i. That is, $v_i \in (C \cup R)$, such that $c_i \in C$ and $r_i \in R$, where C is the set of all ontology classes and R is the set of all instances. The label of the class c_i is denoted $L(c_i)$ and the label of an instance is denoted $L(r_i)$. The set of all labels for all classes in O is denoted $L(C)$, while set of all slang terms for all classes is $L_s(C)$. Similarly the set of all labels for instances is $L(R)$ and their slang terms $L_S(R)$, respectively.

A keyword k_i in the corpus maps to a class or an instance level concept in the DAO. This interpretation is based on string matching using the labels and synonyms of instances and classes. String matching is sufficient, since the overlap in concept labels in the DAO is relatively small. Evidence for this comes from the evaluation of our entity identification approach in [17], in which 85% of slang term mentions in the gold standard could be easily reconciled to the correct ontology concept, without disambiguation. In the evaluation (see Section 3), we show explicitly that it is the ability to interpret intelligible constructs not captured by ontologies that is more crucial to domain specific information retrieval and concepts from the DAO are abstracted using the template class ⟨ENTITY⟩ (shown in Table 2, row 4).

Lexico-ontology-based Query Interpretation: Lexico-ontology concepts are those that have partial representation in ontologies and lexicons simultaneously. For example, the side effects "skin blisters that are itchy" and "skin blisters that are painful" are distinct "skin blisters." However, an ontology may only contain the side effect "skin blisters." This may be problematic if the distinction between itches and pain contribute new information about trends in drug abuse. Knowledge of mappings from lexicons, not present in the ontology, could therefore improve the effectiveness of the search system.

In practice, such discrepancies between lexicons and ontologies for the same concept, may arise as a natural consequence of ontology evolution. Various concepts (or additional attributes for existing ones), may eventually be added to the ontology, but should not be excluded from the search framework in the interim. Also, references in lexicons (such as the Urban Disctionary) may be unknown to domain experts altogether, and never come under consideration for formal representation in the ontology. Inclusion of such search terms in the system may bridge this knowledge gap between what is modeled and what is evident within the community. To address this, we introduce a category of concepts called *lexico-ontology* concepts. The three template classes: route of administration ⟨ROA⟩, ⟨DRUGFORM⟩ and ⟨SIDEEFFECT⟩ are lexico-ontology concepts in our system (Table 2, rows 5-7).

Lexicon-based Query Interpretation: The four template classes: ⟨SENTIMENT⟩, ⟨EMOTION⟩, ⟨PRONOUN⟩ and ⟨INTENSITY⟩ (Table 2, rows 8-11) are part of an ubiquitous class of non-

terminals present in many lexicons. These classes provide insights into self-disclosures, opinions, reports on mood changes and various other experiences in response to drug use. Sentiment expressions such as "didnt do sh*t," "not that great" and "felt pretty good" can help epidemiologists assess and evaluate user reaction and attitudes. In our application, ⟨SENTIMENT⟩ expressions are classified into three categories: ⟨POSITIVE⟩, ⟨NEGATIVE⟩ and ⟨NEUTRAL⟩ based on several lexicons, including LIWC[2] and MPQA[3]. The sentiment identification algorithm implemented by Chen et. al. in [18] is then used to link and annotate sentiment expressions in the corpus. To interpret ⟨EMOTION⟩, the online resource *ChangingMinds.org* that contains several categorizes, such as ⟨AFFECTION⟩, ⟨LUST⟩, ⟨LOVE⟩ and ⟨RAGE⟩ is used. Similarly, various online lexicons are used for categorization of such classes of ⟨PRONOUN⟩ including ⟨PERSONAL PRONOUN⟩, ⟨INTERROGATIVE PRONOUN⟩ and ⟨POSSESSIVE PRONOUN⟩. Distinguishing self references, which are of type ⟨PERSONAL PRONOUN⟩, are important in identifying drug users as the subject of discussions. The interpretation of the nonterminal ⟨INTENSITY⟩ is based on the domain. Expressions such as "low," "small" and "less" can convey ⟨LOW⟩ intensity with regards to drug usage, while "largest," "excessive" and "most" can convey ⟨HIGH⟩ intensity (see Appendix A).

Rule-based Query Interpretation: The three nonterminals ⟨INTERVAL⟩, ⟨FREQUENCY⟩ and ⟨DOSAGE⟩ (Table 2, rows 1-3) require more complex interpretation, and are considered compound template classes. For example, the derivation of a system query from the class ⟨INTERVAL⟩ can be constructed as follows: ⟨INTERVAL⟩ → ⟨AMOUNT⟩ ⟨DURATION_INDICATOR⟩ ⟨PERIOD_DETERMINER⟩. In this production, a valid ⟨INTERVAL⟩ consists of any ⟨AMOUNT⟩ (numeric or textual), followed by a ⟨DURATION_INDICATOR⟩ (e.g. "days," "weeks," "years") and a ⟨PERIOD_DETERMINER⟩ (e.g. "now," "before," "ago"). The system queries "5 years ago" and "about nine months later" are therefore valid interval expressions. Similarly, the following production: ⟨FREQUENCY⟩ → ⟨AMOUNT⟩ ⟨PER_TIME_INDICATOR⟩, can be used to derive valid ⟨FREQUENCY⟩ expressions, such as "5 per min," "per hour" and "24 mg /min." ⟨DOSAGE⟩ system queries such as "1-5 grams" and "2 mcg" can be derived according to the grammar, based on the following production: ⟨DOSAGE⟩ → ⟨NUMBER_AMOUNT⟩ ⟨UNIT⟩ (see Appendix A) for a partial list of productions in the grammar.

The grammar consists of of 61 template patterns in the top-level CFG[4], consisting of template classes in N^S in the RHS of the productions. It also contains close to 150 productions in N^P (see partial list in Appendix A). Using this grammar, our search system in PREDOSE is able to perform domain specific information retrieval somewhat effectively compared with existing search systems (see Section 3). The top-level grammar enables query specification according to the direct information needs of epidemiologists, while the low-level grammar enables interpretation of four different categories of data, pertinent to the domain. In the next section we discuss the use of this grammar for metadata extraction and document annotation from the corpus.

2.2. Semantic Metadata Extractor

The semantic metadata extractor (Figure 1, right) identifies textual patterns (i.e., system queries) in the corpus that match the productions in the grammar. The extractor maintains a database of mappings between these textual patterns in the corpus and web forum posts, which contain them. All annotation extraction is performed offline in a pre-processing step. Given a system query, the *query matcher* (Figure 1, bottom left) retrieves the matching documents after applying various filters (discussed in Section 2.3).

Figure 3: Sample AQL queries

```
create dictionary Buprenorphine_dict as                    (a)
(
    'Buprel', 'Buprenex', 'Buprenorphine', 'Buprenorphine analgesic',
        'Buprenorphine opioid dependence',
        'Probuphine', 'Subbies', 'Suboxone',
        'Suboxone film', 'Suboxone tablet',
        'Subs', 'Subutex', 'Temgesic', 'film', 'films',
        'strip', 'strips', 'sub', 'tecs', 'tex', 'Zubsolv'
);
```

```
create view Buprenorphine_view as                          (b)
extract
    dictionaries 'Buprenorphine_dict'
on D.text as buprenorphine
from Document D;
```

```
create view ROA as ...                                     (c)
create view Entity as ...
Create view EntityROA as ...
```

```
create view DosageView as                                  (d)
(select N.match from NumberWithUnitView N)
  union all
(select D.match from DecimalCombinedView D);
```

```
create view EntityROADosageView as                         (e)
select CombineSpans(ER.match, D.match) as match
from EntityROAView ER, DosageView D
where
FollowsTok(ER.match, D.match, 0, 4);
```

To retrieve the semantic metadata from the corpus we rely on the SystemT [14, 15] framework, and its declarative language specification – AQL (Annotation Query Language). SystemT is a scalable algebraic framework for extracting structured information from unstructured text. It abstracts and manipulates textual data using relational operators such as *select, join, union and consolidate*. Queries can be formulated using AQL then executed with SystemT. AQL is based primarily on two constructs: the *view* and the *dictionary*. An AQL dictionary contains a list of strings and can be exposed as a view. For example, the 'Buprenorphine' dictionary (*Buprenorphine_dict*) shown in Figure 3(a) contains several synonyms for this concept obtained from the DAO. This dictionary can then be exposed

[2]LIWC Online – http://www.liwc.net/

[3]MPQA – http://mpqa.cs.pitt.edu/

[4]Top-Level Grammar Productions `http://wiki.knoesis.org/index.php/Knowledge-Aware-Search-Productions`

as the *Buprenorphine_view* as shown in Figure 3(b). More complex views (or patterns) can be derived by nesting existing views. Figure 3(e) shows the AQL query that extracts textual patterns for the production: ⟨TEMPLATE PATTERN⟩ → ⟨ENTITY⟩ ⟨ROA⟩ ⟨DOSAGE⟩, where ⟨ROA⟩ refers to route of administration (ROA). The *EntityROADosageView* is an annotator in which an "⟨ENTITY⟩ ⟨ROA⟩" pattern must occur within 4 tokens of a ⟨DOSAGE⟩ expression. As shown in Figure 3(d), a ⟨DOSAGE⟩ expression can be defined as any numerical value that co-occurs with a unit, or any decimal value combined with a unit (see Appendix A).

All AQL queries are written, compiled and published for execution on the corpus using the IBM BigInsights platform. Then the Unstructured Information Management Architecture (UIMA) [19] is used to execute the queries on the corpus. The annotated corpus contained 1,287,830 annotations from a corpus of approximately 1,026,502 web forum posts from three online forums[5]. The extracted metadata was indexed positionally (using Apache Lucene and Solr) for use by the query matcher, and also to provide highlighted annotations in the search results. Techniques for matching queries with documents based on extracted semantic metadata are discussed in the following section.

2.3. Query Matcher

The query matcher (Figure 1, bottom left) retrieves relevant documents based on a match between a system query and an annotation extracted using the semantic metadata extractor. To achieve this, the system adopts a two-step process. First, the query processor selects all documents indexed with the template pattern of the user query. For example, given the user query: ⟨q⟩ → ⟨Buprenorphine⟩ ⟨PERSONAL PRONOUN⟩ ">4mg" ⟨BY DAY⟩ | ⟨BY HOUR⟩, derived from template pattern: ⟨TEMPLATE PATTERN⟩ → ⟨ENTITY⟩ ⟨PRONOUN⟩ ⟨DOSAGE⟩ ⟨INTENSITY⟩, the query processor selects an initial set of documents (518) that contain the matching annotations for the given template pattern. Second, the query matcher applies various filters to prune the search results. The *EntityFilter* is first used to retain documents containing ⟨Buprenorphine⟩, specified by ontology-driven query interpretation in the grammar. The resulting document set is reduced (to 97). The query matcher then applies the *PronounFilter*, which restricts the result set to annotations containing only ⟨PERSONAL PRONOUN⟩ (resulting in 90 documents). The query matcher then applies the *DosageFilter*, which retains annotations that mention amounts greater than "4mg," according to the interpretation in the grammar. Recall that this translation involves interpreting of the contextual compilation of the expression ">4mg," which requires interpretation of the greater than ">" operator based on synonyms (e.g., "greater than" and "more than"), mapping the numeral "4" to the word "four," and also expanding the unit "mg" with its various semantically equivalent forms ("milligram," "mgs" and "milli-grams"). The resulting document set is then reduced (to 40). Finally, the query matcher applies the *IntervalFilter*, which restricts the document set to only those annotations that mention daily use ⟨BY DAY⟩ | ⟨BY HOUR⟩. The result is a document set consisting of 21 documents. The ability to extract such search results, based on the grammar, is key to effective domain specific search. In the next section we discuss the user-driven evaluation of our hybrid search system based on the application of our overall search paradigm to the domain of prescription drug abuse.

3. Evaluation

Search systems are typically evaluated using precision, recall and F-Score metrics computed against a baseline of relevant document, for various queries. However, in prescription drug abuse, gold standard datasets are unavailable. In general, the unavailability of standardized datasets for evaluating semantic search system is a common issue in the semantic web community [20, 21]. To evaluate our approach, we perform a comparative analysis of our system against existing search systems through a user-driven evaluation. We note that subjective differences in user agreement and relevance judgments may unduly impact the quality of the evaluation, as noted by Blanco et. al. [22]. Still, the expectation is that our domain specific information retrieval system will perform better than existing search systems, for these domain specific searches. Hence, the goal of the evaluation is to assess the shortcomings of existing systems and stress the need for richer systems for domain specific searches.

We selected three search systems for evaluation: 1) Hakia, 2) DuckDuckGo and 3) Google. Hakia was selected because it uses a SemanticRank algorithm together with background knowledge for search, and therefore fits the characteristics of a classic semantic search engine. DuckDuckGo was selected because it is uses crowd-sourced data from Wikipedia, Wolfram Alpha and Bing (formerly Powerset). The popular search engine Google was selected due to its prominence in general purpose search.

To conduct the evaluation we asked colleagues, not involved in this research but attached to the Kno.e.sis Center to participate in the user study[6]. Each query was executed on the same undisclosed web forum and provided *a priori* to evaluators after a short tutorial of the system. Each evaluator was asked to evaluate the relevance of retrieved documents across all four systems. Initial relevance judgements were based on the text snippet in the search result. If deemed interesting, document should then be explored to confirm or disprove relevance.

Two query scenarios were used in the evaluation (shown in Table 3). Each was then repeated once with different constraints. Thus, four scenarios were examined. These scenarios require interpretation of ontological concepts, concepts in lexicons, and rule-based derivations. Lexico-ontology concepts

[5]Please note that in compliance with the Institutional Review Board (IRB) protocol approved for the PREDOSE project at the Wright State University, to which we are required to adhere to, the names of the selected sources have not been disclosed.

[6]A live version of the search system is available online for option viewing – http://knoesis-hpco.cs.wright.edu/knowledge-aware-search; please refer to the accompanying video demo for a system overview)

Table 3: Evaluation: User Query Scenarios

What specific information is being shared by individuals in the corpus on the use of Buprenorphine in dosages exceeding 4mg daily?
What negative sentiments (or experiences) are being conveyed in the corpus by individuals towards the use of Buprenorphine?

are not included in the evaluation, however we note that several queries for which they are relevant, exist in PREDOSE[7].

In the first query, which is "What specific information is being shared by individuals in the corpus on the use of Buprenorphine in dosages exceeding 4mg daily?", our system interprets the keyword "buprenorphine" using the ontology, the keyword "daily" using the lexicon, the keyword "individuals" using a lexicon, and the phrase "dosages exceeding 4mg" through rules in the grammar. In the second query, the keyword "buprenorphine" is again interpreted from the ontology, the keyword "individuals" is interpreted using a lexicon and the keyword phrase "negative sentiments" is interpreted using the sentiment lexicon (and the method by Chen et. al. [18]). The evaluators were asked to perform their evaluation by first dynamically formulating a query of their choice, for use in Google, Hakia and DuckDuckGo, but using a static query in PREDOSE. This dynamic query requirement was intended to capture the subjective viewpoints of the various evaluators. All measures are based on the top 20 hits in each search system.

Scenario 1: Six evaluators completed the evaluation by formulating an appropriate query of their choice for the web searches (i.e., Google, Hakia and DuckDuckGo), but using the following specific user query to search PREDOSE: ⟨Buprenorphine⟩ ⟨PERSONAL PRONOUN⟩ ">4mg" ⟨BY DAY⟩ | ⟨BY HOUR⟩. Table 4 shows that among the redacted search queries for each user, each contained a mention of the keyword buprenorphine and various expressions for excessive dosage, including the greater than operator ">," "more than," "dosage excess," and "over."

Figure 4 (top left) shows the results across the four systems. Our system retrieved 16/20 relevant results across the six evaluators. Google performed second best, by retrieving 14/20 relevant (but different) documents according to the human judgments. The Google search results showed that it was indeed able to retrieve documents with semantic equivalents for 'buprenorphine' (namely 'Suboxone' and 'bupe'). However, the variability in our system was much greater. In particular, Google did not highlight any search result which contained a dosage greater than 4mg. Instead, greater amounts occurred serendipitously in the snippet of search results. This is in stark contrast to our system in which all documents met this constraint. Furthermore, the result set in our system contained a few mentions of dosage in excess of 32mg, which is considered the known limit by epidemiologists. Hakia retrieved

[7]Numerous search queries are available online for optional viewing – http://wiki.knoesis.org/index.php/Knowledge-Aware-Search-Evaluation

Table 4: User Queries for Scenario 1

	Freeform User Queries Scenario 1 (for Google, Hakia, DuckDuckGo)
1	site:domain.name daily buprenorphine dosage > 4mg
2	site:domain.name using bupe more than 4mg
3	site:domain.name buprenorphine dosage excess 4mg daily
4	site:domain.name buprenorphine dosage more than 4mg daily
5	site:domain.name buprenorphine dosage over 4mg daily
6	site:domain.name (buprenorphine OR bup) dosage daily ("above 4mg" OR "over 4mg" OR "more than 4mg")

4/20 relevant search results, which were not very informative. So were the search results from DuckDuckGo, which retrieved only 3/20 relevant results. On close inspection, we observed that their poor performance was largely due to an inability to interpret the semantics of the greater than ('>') operator. Most search results were retrieved because they contained the label 'buprenorphine' itself and other query elements.

Figure 5 shows a screenshot of the results from our hybrid search system for this scenario. The selected web forum is indicated as Site Y under the Data Sources(s) panel (top left). As shown in the Template Query Builder interface (Figure 5, top right), to construct the user query for the information need, the evaluator must first select the template class ⟨ENTITY⟩, and then select the nonterminal 'Buprenorphine' from the list. This concept is expanded according to the grammar production $\langle ENTITY \rangle \rightarrow L(C) \cup L_s(C) \cup L(R) \cup L_s(R)$, which includes all slang terms and labels for all subclasses and individuals. The respondent then selected the template class ⟨PRONOUN⟩ and selected the set of all ⟨PERSONAL PRONOUN⟩, which is interpreted according to a lexicon of pronouns. The evaluator then selected the template class ⟨DOSAGE⟩, which is interpreted according to the rules applied to the alphabet in the grammar (see Appendix A). Finally, the evaluator selected the template class ⟨INTERVAL⟩ and then the nonterminals ⟨BY_DAY⟩ and ⟨BY_HOUR⟩.

The search results are shown in the Template Pattern Search Results Grid (Figure 5, bottom left) and the Integrated Template Pattern Content Viewer (Figure 5, bottom right). Among the search results, note first that all documents contained an amount greater than 4mg. Second, there were only 2 search results that contained the actual label 'buprenorphine' in their annotation. This list of synonyms is as follows: subs–12, sub–2, Subutex–2, Suboxone–3, Buprenorphine–2. Both events can be attributed to the level of interpretation performed by the grammar.

We note that a search result is considered if at least 4/6 evaluators agreed the result was relevant. This is reasonable because the significance of kappa scores across multiple users may diminish, but not necessarily indicate a major disagreement. Although Fleiss' kappa may be used instead, the results from this slight majority seems reasonable. And since each user query could be (and indeed was) different among the 6 evaluators, as

Figure 4: Evaluation Scenarios: 1(top left), 2(top right), 3(bottom left), 4(bottom right)

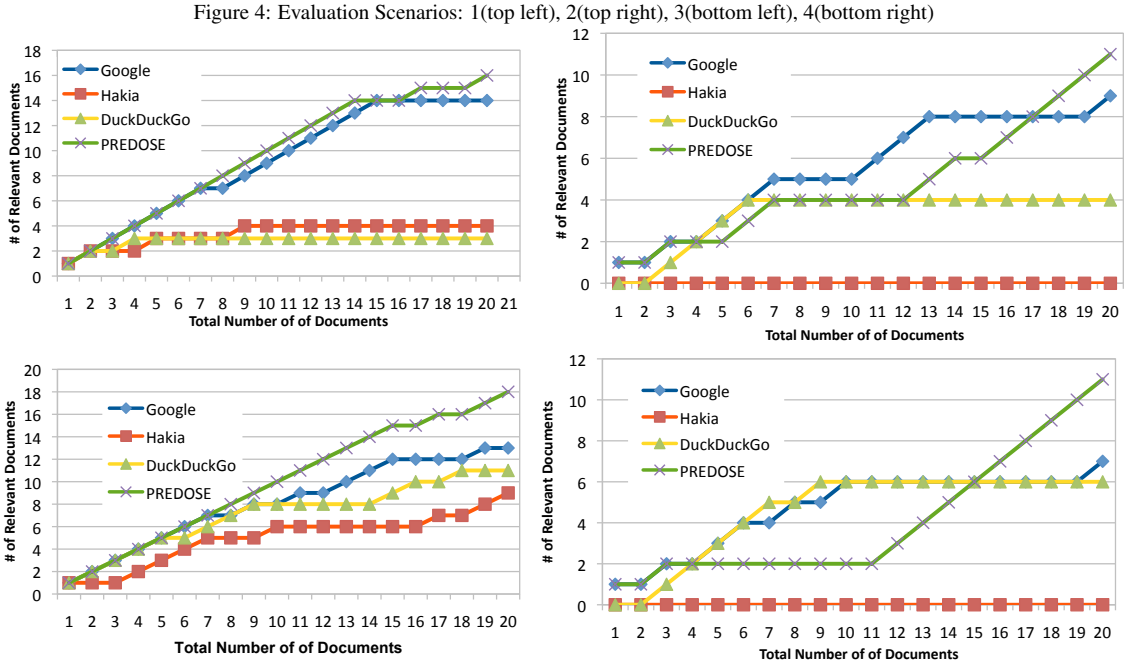


Table 5: User Queries for Scenario 2

	Freeform User Queries Scenario 2 (for Google, Hakia, DuckDuckGo)
1	site:domain.name side effects for buprenorphine
2	site:domain.name side effect for buprenophine | bup | bupe | bupes | suboxone
3	site:domain.name buprenorphine bad experience
4	site:domain.name “buprenorphine” “horrible experience”
5	site:domain.name “buprenorphine” “horrible
6	site:domain.name “buprenorphine” (“horrible” OR “side effect”) “buprenorphine” (“bad experience” OR “side effect”)

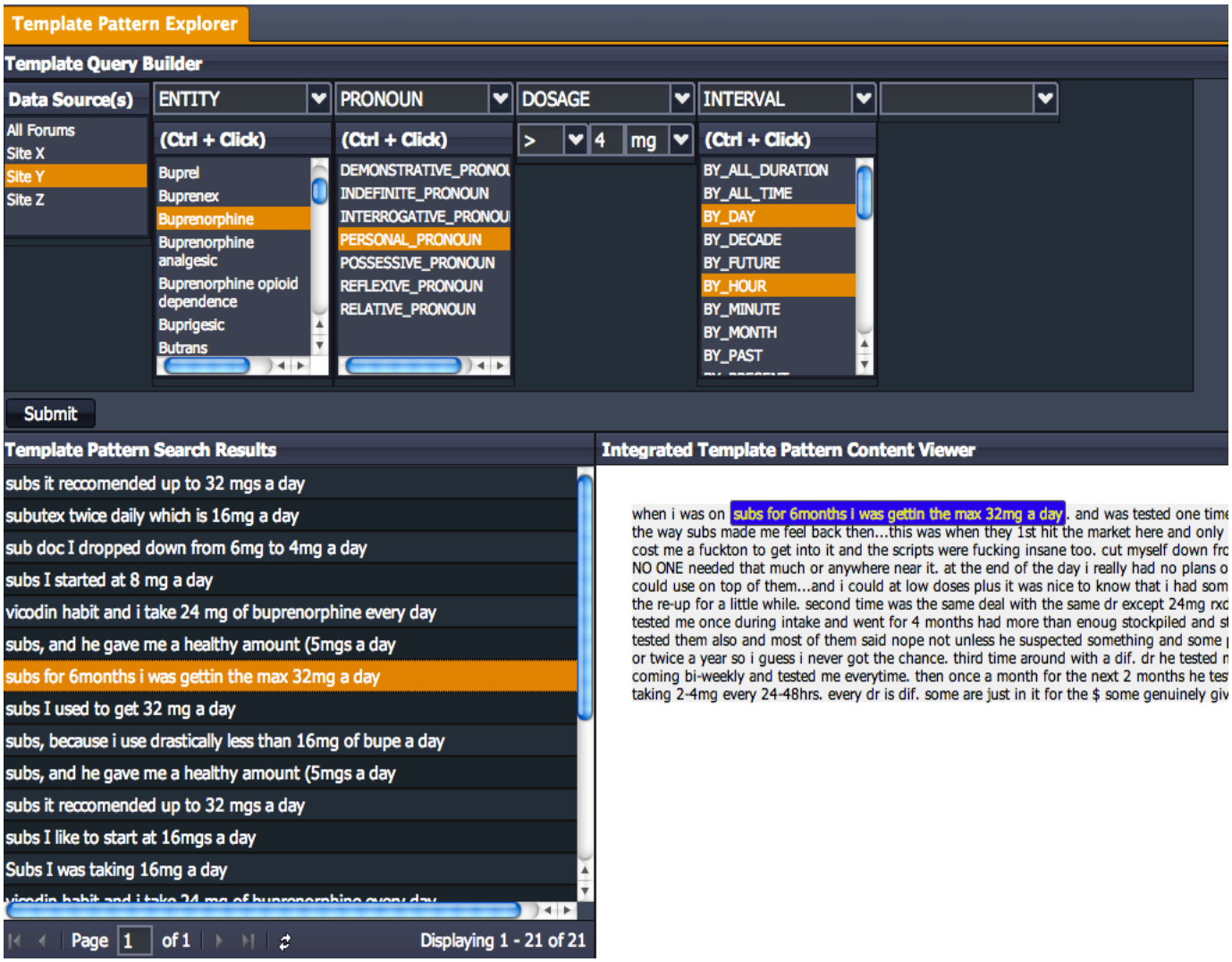
shown in Table 4, relevance agreement for specific search results among the other sources (i.e., Google, Hakia and DuckDuckGo) is of little meaning. To compute the relevance of search results in the top 20, we considered agreement positionally, on the relevance of each results across the separate lists. That is, among the evaluators how many agreed their first, second, third result and so on, were relevant. Given the absence of a goal standard dataset for evaluation, as noted in [20, 21, 22], was deemed a reasonable compromise.

Scenario 2: The same evaluators repeated the evaluation for a different query scenario. This time, to find relevant documents that discuss: “negative sentiments/experiences resulting from the use of Buprenorphine” (Table 3, row 2). The selected free form queries used by the evaluators are listed in Table 5. The following specific user query was provided for use in our system: ⟨Buprenorphine⟩ ⟨PERSONAL PRONOUN⟩ ⟨NEGATIVE⟩ for the template pattern ⟨ENTITY⟩ ⟨PRONOUN⟩ ⟨SENTIMENT⟩.

Figure 4, top right, shows that our system retrieved relevant 11/20 search results. This drop is likely because of the difficulty in interpreting the nonterminal ⟨NEGATIVE SENTIMENT⟩. Google retrieved 9/20 relevant results. However, we noticed that their search results did not contain sentiment-conveying keywords other than those specified by the user. Our system retrieved the following sentiment expressions in the results (f*up–1, sh*t–2, weird–2, disappointed–1, f*ing weird–2, hated–1, rough–1, nauseous–1), which were not specified in the user query, but based on the interpretation of ⟨NEGATIVE⟩ sentiment in the grammar (see Appendix A). Hakia and DuckDuckGo also performed less effectively, due to an inability to interpret ⟨NEGATIVE⟩ sentiment as formulated in the user queries.

Scenario 3: Unlike the previous two scenarios, we asked 6 evaluators to search Google, Hakia and DuckDuckGo using the following specific search query: ‘site:domain.name buprenorphine dosage excess 4mg daily.’ We performed this evaluation to assess the objective relevance judgments across the same search results. In this set, 3 evaluators were the same from Scenarios 1 and 2, and there were 3 new. Figure 5 (bottom left) shows that this set of evaluators agreed that slightly more documents were relevant to the information need for the given query. A total of 18 out of 20 results were considered relevant in PREDOSE (an increase of 2), compared with 13 for Google (a decrease of 1). Within this set of 18 relevant search results, a few contained a mention of usage in the region of 32mg, which is considered an upper bound. We also observed that among the 13 relevant Google results, none of the highlighted amounts was greater than 4mg, but rather serendipitously contained greater amounts in the surrounding text. This reaffirms what is already known; Google does not interpret keywords to any significant degree, but rather performs keyword-based query expansion. DuckDuckGo showed a striking increase in the overall number of relevant results, increasing from 3/20 to 11/20. This suggests

Figure 5: Screenshot of search results from our hybrid information retrieval system for scenario1



that the search query provided by our team for this evaluation was more effective in retrieving search results. Still however, it was observed that only few document snippets highlighted amounts that were greater than "4mg." The relevant search results in Hakia also increased from 4/20 to 9/20 with only 3 highlighted amounts greater than "4mg."

Scenario 4: Finally, the same procedure was repeated using the following specific query ('site:domain.name buprenorphine bad experience') provided by our team for the same information need from Scenario 2. That is, find relevant documents that discuss: "negative sentiments/experiences resulting from the use of Buprenorphine." Figure 4 (bottom right) shows that our system retrieved 11/20 relevant search results and again outperformed Google (7/20), DuckDuckGo (6/20) and Hakia (which notably did not retrieve any relevant results for the query).

4. Related Work

In this section, we provide an overview of semantic search and hybrid information retrieval systems, which rely on semantic web technologies. Semantic Web offers to create machine-processable representation of real-world concepts, whose meaning can be exploited for various tasks across heterogeneous information environments [23]. Semantic search, as an application of semantic web technologies, is intended to enhance the retrieval of more accurate and high quality search results, when compared with traditional keyword-based search models and their various enhancements. An early realization of this idea has been the Semantic Content Organization and Retrieval Engine (SCORE) [11, 24, 12], which uses both ontologies and rules derived using regular expressions for search. SCORE and its successor Semagix FREEDOM [25, 26] were early platforms that integrated structured knowledge and additional intelligible constructs to support real-world and commercial knowledge-driven applications.

In this work we go beyond the functionality provided by SCORE, by providing search support for: 1) classes of query elements modeled almost exclusively in lexicons (such as positive and negative sentiment expressions, and varying types and degrees of emotions), 2) information in lexicons, with only

partial ontology representation, such as side effects, route-of-administration (ROA) and drug form and 3) elements that belong to broad classes (including certain parts-of-speech), levels of intensity (high, low, average), and fuzzy interval references (past, present, future, etc). Moreover, we perform a deeper level of interpretation of certain rule-based constructs (such as '>4mg'), through a low-level CFG for query interpretation. We perform these tasks in addition to ontology-driven and rule-based search as was implemented in SCORE.

Popov et. al. [13, 27] developed KIM that supports semantic annotation and search for entities and entity-patterns from the ontology that are also present in the corpus. The search is enhanced with support for lexical resources (such as currency, date, location, aliases, abbreviations etc) not typically represented in ontologies. To achieve this, Popov uses a modified pattern-matching grammar based on GATE , which recognizes relations in text, by gleaning entity associations from predicates in the ontology schema. While KIM is similar to our approach, we provide a more inclusive hybrid search system, which supports the retrieval of two additional types of data not considered by KIM: 1) from lexicons and 2) lexico-ontology concepts. Moreover, our system is more loosely coupled to accommodate query elements not in the ontology, while providing a broader range of pattern-based search through a top-level CFG. Additionally, we evaluate the relevance of search results for specific complex information needs in a domain specific setting. Popov et. al. [13, 27] evaluate the precision and recall of annotations types (elements in our second-level grammar) rather than actual results of semantic search. Although a search evaluation on television and radio news articles was conducted in [28] using KIM, but based on ontology and keyword-based query interpretation.

Guha et. al. [4] presented a prototype semantic search system called TAP, that interprets keywords according to real-world concepts modeled in background knowledge. Various heuristics were used to find matching subgraphs for single keyword queries and keyword pairs. The retrieved structured data was then rendered as an augmentation of search results from the document list, in a Google-style search interface. A key issue is that while the information gleaned from background knowledge may complement the search results in the document list, there is an assumption that query elements can be mapped to the ontology in the first place. This assumption will not always hold, as the authors themselves note, if "the search term does not denote anything known to the Semantic Web, then we are not able to contribute anything to the search results."

Thirunarayan and Immaneni in [29] also developed a hybrid query language to unify web of data and web of documents, This approach improves both: 1) information retrieval from Semantic Web through keyword-based search and 2) semantic search of hyperlinked web documents through the exploitation of inheritance hierarchy. Their lucene-based SITAR (Semantic InformaTion Analysis and Retrieval) system provided enhanced retrieval from combined data sources such as AIFB SEAL. SITAR contains information about researchers that combines both structured and unstructured data.

Lei et. al. [3] developed a semantic search engine called SemSearch, which is another ontology-driven system for keyword-based search over documents. However, SemSearch provides flexibility in query interpretation by also providing search support using Lucene, for keywords not present in the ontology. In the case of complex queries that contain multiple keywords, facts from the ontology are used as templates for query interpretation, similar to the approach in KIM [13].

Keyword-based semantic search systems offer a tradeoff between query expressiveness and accuracy of query interpretation, both of which affect the quality of the retrieved search results. Hence, there is a body of research focused on natural language query interfaces for semantic search [30, 31, 32, 1, 6, 7, 8], to provide more query expressiveness. Lopez et. al. developed AquaLog [30] and PowerAqua [32], which translate natural language user queries into binary relational (triple) format, consistent with ontological representations. Fernandez et. al. [31, 1] utilize PowerAqua for ontology-driven natural language query interpretation over documents. Queries expressed in natural language are translated into a formal representation using SPARQL. The document corpus is annotated based on entities, which populate the ontology knowledgebase of instances. In this way, the knowledgebase is a representation of the corpus, through the association of its annotations. While this approach is plausible (like SCORE), it requires that the corpus be represented in the ontology. The technique used for corpus annotation must necessarily be aware of the various types of query elements, extract them and represent them formally in the ontology. This is a challenging problem, as not all data types are suited for ontology representation, let alone querying using ontology query languages.

5. Discussion

In this paper we present a hybrid information retrieval system based on a CFG, to enable domain specific information retrieval for the domain of prescription drug abuse. The need for external resources to complement ontologies when addressing complex information needs is not isolated. In the development of Watson, the DeepQA research project [33] at IBM Research exploited a range of data sources, including "unstructured data (e.g., typical web pages, blog posts) and semi-structured data (e.g., Wikipedia) to completely structured data (facts mined from the Web or [from] pre-existing databases)" [34] for question answering (QA) in the Jeopardy! Challenge. Our approach is consistent with this view, and other semantic search applications [11, 13] capable of interpreting multi-faceted queries involving ontological concepts and additional intelligible constructs. There continues to be a growing realization that to effectively address practical information retrieval problems, current knowledge models must be enhanced, by incorporating semantic enrichment modules capable of interpreting heterogeneous data. The implemented *Template Pattern Explorer* is currently in use by epidemiologists at the Center for Interventions, Treatment and Addictions Research (CITAR) at the Wright State University. Further, the existing grammar is being extended for application to other social media and unstructured clinical notes. These include: 1) clinical notes

[35], from which the grammar was inherited, 2) eDrugTrends – http://wiki.knoesis.org/index.php/EDrugTrends, 3) biomaterials and materials science – http://wiki.knoesis.org/index.php/MaterialWays and 4) knowledge acquisition from EMRs, which are project at Kno.e.sis.

While the approach outlined in this application shows early progress, there are several limitations. The first limitation is that the manual specification of the grammar for each application is cumbersome and not scalable. General-purpose search engines perform well on the web due to the implementation of generic search algorithms. While it is difficult to implement a generic algorithm that can effectively retrieve data for specific domains, a greater degree of automation in grammar composition is needed. Second, template-based query specification also requires considerable domain expertise and familiarity with the search application. A less restrictive query specification interface, such as those based on keywords or natural language queries is under consideration. Another critical issue is the need for entity disambiguation to filter out spurious results. False positives impact the overall accuracy of the system, and the relevance of the generated search results. While our earlier results, reported in [17], showed that approximately 85% of entities are correctly identified in our system, entities such as "alcohol," "cannibis" and "oxycontin" are highly ambiguous. A context-aware methodology for entity disambiguation, such as that implemented by Mendes et. al. [36] could be beneficial. Likewise, ranking search results, which is currently not provided, may be crucial in search scenarios where many search results exist. Popular concepts such as 'methadone' and 'heroin,' which occur with high frequency in the corpus can generate many search results, which may be overwhelming for domain experts to explore. Additionally, a method that can dynamically identify new and frequently occurring template patterns in text could improve the scalability of our approach. The 60 template patterns used in this study were created manually, based on information needs provided by domain experts. However, such resources may not be available in other domains. In spite of these and many other limitations, this research is an early effort to develop a search system which addresses complex domain specific information retrieval in prescription drug abuse. We believe that overcoming the aforementioned limitations will only serve to improve the overall quality of the search system.

6. Conclusion

In this paper we presented a hybrid information retrieval system for domain specific information retrieval, applied to prescription drug abuse, which uses a context-free grammar to specify the query language of expressions interpretable by the system. Our hybrid approach is capable of interpreting four types of query elements: 1) ontological knowledge, 2) concepts in lexicons, 3) concepts in lexicons with partial ontology representation (i.e., lexico-ontology) and 4) intelligible constructs defined exclusively through rules. The system uses template-based query specification to facilitate interpretation of domain specific query elements. In an evaluation against the contemporary semantic search system (Hakia), the crowd sourcing-based search system (DuckDuckGo) and the popular search engine Google, our system performed satisfactorily in retrieving relevant results for complex information needs. A live web application is currently available and in use by epidemiologists conducting research on emerging patterns and trends in prescription drug abuse using social media. The search system is also available online for option viewing – http://knoesis-hpco.cs.wright.edu/knowledge-aware-search, together with an accompanying video demo http://bit.ly/kasdemo.

7. Author Contributions

Delroy Cameron assisted in writing the manuscript and developed key aspects of the overall hybrid information retrieval system, and also contributed many ideas for the overall research. Amit P. Sheth established the interdisciplinary collaboration of the PREDOSE project, guided the development of the project, while providing ideas for its positioning within semantic search and also contributed to the writing. Krishnaprasad Thirunarayan developed key aspects of the context-free grammar and also provided crucial research ideas. Nishita Jaykumar, Gaurish Anand and Gary A. Smith assisted with many aspects of the system, including system implementation and evaluation, and the provision of various supporting online resources.

8. Conflict of Interest

The authors would like to assert that there are no conflicts of interest regarding this manuscript.

9. Acknowledgment

The research is funded in part by the National Institute on Drug abuse (NIDA) Grant No. 5R01DA039454-02 grant titled: "Trending: Social media analysis to monitor cannabis and synthetic cannabinoid use" and by Grant No. R21 DA030571-01A1. Any opinions, findings, conclusions or recommendations expressed in this material are those of the investigator(s) and do not necessarily reflect the views of the National Institutes of Health. We would also like to thank other members of the research team, including Raminta Daniulaityte, Drashti Dave, Revathy Krishnamurthy, Swapnil Soni and Kera Z. Watkins.

10. References

[1] M. Fernández, I. Cantador, V. Lopez, D. Vallet, P. Castells, E. Motta, Semantically enhanced information retrieval: An ontology-based approach, J. Web Sem. 9 (4) (2011) 434–452.

[2] T. Tran, P. Cimiano, S. Rudolph, R. Studer, Ontology-based interpretation of keywords for semantic search, in: ISWC/ASWC, 2007, pp. 523–536.

[3] Y. Lei, V. S. Uren, E. Motta, Semsearch: A search engine for the semantic web, in: EKAW, 2006, pp. 238–245.

[4] R. V. Guha, R. McCool, E. Miller, Semantic search, in: WWW, 2003, pp. 700–709.

[5] C. Rocha, D. Schwabe, M. P. Aragao, A hybrid approach for searching in the semantic web, in: Proceedings of the 13th international conference on World Wide Web, WWW '04, ACM, New York, NY, USA, 2004, pp. 374–383. doi:10.1145/988672.988723.
URL http://doi.acm.org/10.1145/988672.988723
[6] D. Vallet, M. Fernández, P. Castells, An ontology-based information retrieval model, in: ESWC, 2005, pp. 455–470.
[7] P. Castells, M. Fernández, D. Vallet, An adaptation of the vector-space model for ontology-based information retrieval, IEEE Trans. Knowl. Data Eng. 19 (2) (2007) 261–272.
[8] T. Ruotsalo, Domain specific data retrieval on the semantic web, in: ESWC, 2012, pp. 422–436.
[9] A. P. Sheth, S. Agrawal, J. Lathem, N. Oldham, H. Wingate, P. Yadav, K. Gallagher, Active semantic electronic medical records, in: The Semantic Web: Real-World Applications from Industry, 2007, pp. 123–140.
[10] A. Seaborne, Rdql - a query language for rdf. w3c member submission 9 january 2004, http://www.w3.org/submission/rdql/ (2004).
[11] B. Hammond, A. Sheth, K. Kochut, S. Inc, Semantic enhancement engine: A modular document enhancement platform for semantic applications over heterogeneous content, in: in Real World Semantic Web Applications, V. Kashyap and L. Shklar, Eds., IOS, Press, 2002, pp. 29–49.
[12] A. Sheth, D. Avant, C. Bertram, System and method for creating a semantic web and its applications in browsing, searching, profiling, personalization and advertising, uS Patent 6,311,194 (Oct. 30 2001).
URL http://www.google.com/patents/US6311194
[13] B. Popov, A. Kiryakov, D. Ognyanoff, D. Manov, A. Kirilov, Kim - a semantic platform for information extraction and retrieval, Natural Language Engineering 10 (3-4) (2004) 375–392.
[14] L. Chiticariu, R. Krishnamurthy, Y. Li, S. Raghavan, F. R. Reiss, S. Vaithyanathan, Systemt: an algebraic approach to declarative information extraction, in: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10, Association for Computational Linguistics, Stroudsburg, PA, USA, 2010, pp. 128–137.
[15] R. Krishnamurthy, Y. Li, S. Raghavan, F. Reiss, S. Vaithyanathan, H. Zhu, Systemt: a system for declarative information extraction, SIGMOD Record 37 (4) (2008) 7–13.
[16] A. Broder, A taxonomy of web search, SIGIR Forum 36 (2) (2002) 3–10.
[17] D. Cameron, G. A. Smith, R. Daniulaityte, A. P. Sheth, D. Dave, L. Chen, G. Anand, R. Carlson, K. Z. Watkins, R. Falck, Predose: A semantic web platform for drug abuse epidemiology using social media, Journal of Biomedical Informatics 46 (6) (2013) 985–997.
[18] L. Chen, W. Wang, M. Nagarajan, S. Wang, A. P. Sheth, Extracting diverse sentiment expressions with target-dependent polarity from twitter, in: ICWSM, 2012.
[19] UIMA, Unstructured information management architecture.
URL http://uima.apache.org
[20] Y. Sure, V. Iosif, First results of a semantic web technologies evaluation, in: Proceedings of the Common Industry Program at ODBASE 2002, 2002.
[21] R. McCool, A. Cowell, D. Thurman, End-user evaluations of semantic web technologies, in: Proceedings of the ISWC 2005 Workshop on End User Semantic Web Interaction, 2005.
[22] R. Blanco, H. Halpin, D. M. Herzig, P. Mika, J. Pound, H. S. Thompson, T. Tran, Repeatable and reliable semantic search evaluation, J. Web Sem. 21 (2013) 14–29.
[23] T. Berners-Lee, M. Fischetti, Weaving the web - the original design and ultimate destiny of the World Wide Web by its inventor, HarperBusiness, 2000.
[24] A. Sheth, C. Bertram, D. Avant, B. Hammond, K. Kochut, Y. Warke, Managing semantic content for the web, Internet Computing, IEEE 6 (4) (2002) 80–87.
[25] A. Sheth, Enterprise applications of semantic web: The sweet spot of risk and compliance, in: In IFIP International Conference on Industrial Applications of Semantic Web, Springer, 2005, pp. 25–27.
[26] A. P. Sheth, B. Aleman-Meza, I. B. Arpinar, C. Bertram, Y. S. Warke, C. Ramakrishnan, C. Halaschek, K. Anyanwu, D. Avant, F. S. Arpinar, K. Kochut, Semantic association identification and knowledge discovery for national security applications, J. Database Manag. 16 (1) (2005) 33–53.
[27] A. Kiryakov, B. Popov, I. Terziev, D. Manov, D. Ognyanoff, Semantic annotation, indexing, and retrieval, J. Web Sem. 2 (1) (2004) 49–79.
[28] M. Dowman, V. Tablan, H. Cunningham, B. Popov, Web-assisted annotation, semantic indexing and search of television and radio news, in: WWW, 2005, pp. 225–234.
[29] K. Thirunarayan, T. Immaneni, Integrated retrieval from web of documents and data, in: Z. Ras, A. Dardzinska (Eds.), Advances in Data Management, Vol. 223 of Studies in Computational Intelligence, Springer Berlin Heidelberg, 2009, pp. 25–48.
[30] V. Lopez, M. Pasin, E. Motta, Aqualog: An ontology-portable question answering system for the semantic web, in: ESWC, 2005, pp. 546–562.
[31] M. Fernández, V. Lopez, M. Sabou, V. S. Uren, D. Vallet, E. Motta, P. Castells, Semantic search meets the web, in: ICSC, 2008, pp. 253–260.
[32] V. Lopez, E. Motta, V. S. Uren, Poweraqua: Fishing the semantic web, in: ESWC, 2006, pp. 393–410.
[33] D. A. Ferrucci, E. W. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. M. Prager, N. Schlaefer, C. A. Welty, Building watson: An overview of the deepqa project, AI Magazine 31 (3) (2010) 59–79.
[34] D. Ferrucci, E. Nyberg, J. Allen, K. Barker, E. W. Brown, J. Chu-Carroll, A. Ciccolo, P. A. Duboue, J. Fan, D. Gondek, E. Hovy, B. Katz, A. Lally, M. McCord, P. Morarescu, J. W. Murdock, B. Porter, J. M. Prager, T. Strzalkowski, C. Welty, W. Zadrozny, Towards the open advancement of question answering systems (2008).
[35] D. Cameron, V. Bhagwan, A. P. Sheth, Towards comprehensive longitudinal healthcare data capture, in: BIBM Workshops, 2012, pp. 240–247.
[36] P. N. Mendes, M. Jakob, A. García-Silva, C. Bizer, Dbpedia spotlight: shedding light on the web of documents, in: I-SEMANTICS, 2011, pp. 1–8.

Appendix A. Context-Free Grammar

This appendix provides a partial listing of the context-free grammar (CFG) in *Backus-Naur Form (BNF)*, which is used to interpret the query language U of our hybrid information retrieval system H. Further details are available online: http://wiki.knoesis.org/index.php/Knowledge-Aware-Search

Appendix A.1. Start Symbol

The Start Symbol of the grammar is the nonterminal ⟨TEMPLATE PATTERN⟩. This start symbol supports productions containing sequences of ⟨TEMPLATE CLASS⟩ nonterminals from the set N^S. There are 61 specific sequences of template classes supported by the start symbol[8]. To avoid listing these in detail the *kleene start* operator is used in the first production below. The completelist of productions is available in the online supplementary materials.

1. ⟨TEMPLATE PATTERN⟩ → ⟨ TEMPLATE CLASS⟩*

Appendix A.2. Template Classes

There are 11 nonterminals in the set of template classes in N^S that comprise the top-level grammar.

2. ⟨TEMPLATE CLASS⟩ → ⟨*INTERVAL*⟩|⟨*FREQUENCY*⟩ | ⟨*DOSAGE*⟩ |⟨*ENTITY*⟩|⟨*ROA*⟩ | ⟨*DRUGFORM*⟩ | ⟨*SIDEEFFECT*⟩ | ⟨*EMOTION*⟩ | ⟨*PRONOUN*⟩ | ⟨*INTENSITY*⟩ | ⟨*SENTIMENT*⟩

[8] http://wiki.knoesis.org/index.php/Knowledge-Aware-Search-Productions

Appendix A.3. Productions

The set of nonterminals of in G are shown in the following productions.

The $\langle INTERVAL \rangle$ template class is defined as follows:

3. $\langle INTERVAL \rangle \rightarrow \langle DURATION\_PERIOD | \langle PERIOD\_DURATION \rangle | \langle TIME\_PERIOD \rangle | \langle PERIOD\_TIME \rangle | \langle AMOUNT\_TIME\_PERIOD \rangle | \langle AMOUNT\_TIME \rangle | \langle PERIOD\_AMOUNT\_TIME \rangle | \langle AMOUNT\_DURATION\_PERIOD \rangle | \langle AMOUNT\_DURATION \rangle | \langle PERIOD\_AMOUNT\_DURATION \rangle$
4. $\langle DURATION\_PERIOD \rangle \rightarrow \langle PAST\_DETERMINER \rangle \langle RANGE \rangle \langle PERIOD \rangle | \langle PRESENT\_DETERMINER \rangle \langle RANGE \rangle \langle PERIOD \rangle | \langle FUTURE\_DETERMINER \rangle \langle RANGE \rangle \langle PERIOD \rangle$
5. $\langle PERIOD\_DURATION \rangle \rightarrow \langle PAST\_DETERMINER \rangle \langle RANGE \rangle \langle DURATION\_INDICATOR \rangle | \langle PRESENT\_DETERMINER \rangle \langle RANGE \rangle \langle DURATION\_INDICATOR \rangle | \langle FUTURE\_DETERMINER \rangle \langle RANGE \rangle \langle DURATION\_INDICATOR \rangle$
6. $\langle TIME\_PERIOD \rangle \rightarrow \langle TIME\_INDICATOR \rangle \langle RANGE \rangle \langle PAST\_DETERMINER \rangle | \langle TIME\_INDICATOR \rangle \langle RANGE \rangle \langle PRESENT\_DETERMINER \rangle | \langle TIME\_INDICATOR \rangle \langle RANGE \rangle \langle FUTURE\_DETERMINER \rangle$
7. $\langle PERIOD\_TIME \rangle \rightarrow \langle PAST\_DETERMINER \rangle \langle RANGE \rangle \langle TIME\_INDICATOR \rangle | \langle PRESENT\_DETERMINER \rangle \langle RANGE \rangle \langle TIME\_INDICATOR \rangle | \langle FUTURE\_DETERMINER \rangle \langle RANGE \rangle \langle TIME\_INDICATOR \rangle$
8. $\langle AMOUNT\_TIME\_PERIOD \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle TIME\_PAST\_PERIOD \rangle | \langle AMOUNT \rangle \langle RANGE \rangle \langle TIME\_PRESENT\_PERIOD \rangle | \langle AMOUNT \rangle \langle RANGE \rangle \langle TIME\_FUTURE\_PERIOD \rangle$
9. $\langle AMOUNT\_TIME \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_PAST\_PERIOD \rangle | \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_PRESENT\_PERIOD \rangle | \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_FUTURE\_PERIOD \rangle$
10. $\langle PERIOD\_AMOUNT\_TIME \rangle \rightarrow \langle PAST\_DETERMINER \rangle \langle RANGE \rangle \langle AMOUNT\_TIME \rangle | \langle PRESENT\_DETERMINER \rangle \langle RANGE \rangle \langle AMOUNT\_TIME \rangle | \langle FUTURE\_DETERMINER \rangle \langle RANGE \rangle \langle AMOUNT\_TIME \rangle$
11. $\langle AMOUNT\_DURATION\_PERIOD \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_PAST\_PERIOD \rangle | \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_PRESENT\_PERIOD \rangle | \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_FUTURE\_PERIOD \rangle$
12. $\langle AMOUNT\_DURATION \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle DURATION\_INDICATOR \rangle$
13. $\langle PERIOD\_AMOUNT\_DURATION \rangle \rightarrow \langle PAST\_DETERMINER \rangle \langle RANGE \rangle \langle AMOUNT\_DURATION \rangle | \langle PRESENT\_DETERMINER \rangle \langle RANGE \rangle \langle AMOUNT\_DURATION \rangle | \langle FUTURE\_DETERMINER \rangle \langle RANGE \rangle \langle AMOUNT\_DURATION \rangle$
14. $\langle TIME\_INDICATOR \rangle \rightarrow \langle HOUR \rangle | \langle MINUTE \rangle | \langle SECOND \rangle$
15. $\langle DURATION\_INDICATOR \rangle \rightarrow \langle DECADE \rangle | \langle YEAR \rangle | \langle MONTH \rangle | \langle WEEK \rangle$
16. $\langle DECADE \rangle \rightarrow$ **day | night**
17. $\langle YEAR \rangle \rightarrow$ **year | years | yr | yrs | annum**
18. $\langle MONTH \rangle \rightarrow$ **month | months | mth | mths | mo**
19. $\langle WEEK \rangle \rightarrow$ **week | weeks | wk | wks**
20. $\langle PERIOD \rangle \rightarrow \langle PAST\_DETERMINER \rangle | \langle PRESENT\_DETERMINER \rangle | \langle FUTURE\_DETERMINER \rangle$
21. $\langle PAST\_DETERMINER \rangle \rightarrow$ **ago | prior | previous | since | before | ...**
22. $\langle PRESENT\_DETERMINER \rangle \rightarrow$ **now | about | around | several | ...**
23. $\langle FUTURE\_DETERMINER \rangle \rightarrow$ **next | later | after**
24. $\langle SECOND \rangle \rightarrow$ **second | seconds | sec | secs**
25. $\langle MINUTE \rangle \rightarrow$ **minute | minutes | min | mins**
26. $\langle HOUR \rangle \rightarrow$ **hour | hours | hr | hrs**
27. $\langle NUMBER \rangle \rightarrow \mathbb{N}$
28. $\langle NUMERIC\_AMOUNT \rangle \rightarrow \mathbb{R}$
29. $\langle WORDED\_AMOUNT \rangle \rightarrow$ **one | once | two | twice | three | thrice | four | five | six | seven | eight | nine | ten | eleven | twelve | thirteen | fourteen | fifteen | sixteen | seventeen | eighteen | nineteen | twenty | thirty | forty | fifty | sixty | seventy | eighty | nintey | hundred**
30. $\langle AMOUNT \rangle \rightarrow \langle NUMBER \rangle | \langle WORDED\_AMOUNT \rangle$
31. $\langle RANGE \rangle \rightarrow [0 - \langle NUMBER \rangle]$

The $\langle FREQUENCY \rangle$ template class is defined as follows:

32. $\langle FREQUENCY \rangle \rightarrow \langle PER\_TIME\_INDICATOR \rangle | \langle PER\_DURATION\_INDICATOR \rangle | \langle AMOUNT\_FREQUENCY\_DURATION \rangle | \langle PERIOD\_FREQUENCY\_DURATION \rangle | \langle PERIOD\_FREQUENCY\_TIME \rangle | \langle AMOUNT\_FREQUENCY\_TIME \rangle | \langle AMOUNT\_PER\_TIME \rangle | \langle AMOUNT\_PER\_DURATION \rangle | \langle FREQUENCY\_ITEM \rangle$
33. $\langle AMOUNT\_FREQUENCY\_DURATION \rangle \rightarrow \langle AMOUNT\_FREQUENCY \rangle \langle RANGE \rangle \langle DURATION\_INDICATOR \rangle$
34. $\langle FREQUENCY\_DURATION \rangle \rightarrow \langle FREQUENCY\_INDICATOR \rangle | \langle RANGE \rangle \langle DURATION\_INDICATOR \rangle$
35. $\langle FREQUENCY\_TIME \rangle \rightarrow \langle FREQUENCY\_INDICATOR \rangle \langle RANGE \rangle \langle TIME\_INDICATOR \rangle$
36. $\langle PERIOD\_FREQUENCY\_DURATION \rangle \rightarrow \langle PERIOD\_DETERMINER \rangle \langle RANGE \rangle \langle FREQUENCY\_INDICATOR \rangle$
37. $\langle PERIOD\_FREQUENCY\_TIME \rangle \rightarrow \langle PERIOD\_DETERMINER \rangle \langle RANGE \rangle \langle FREQUENCY\_TIME \rangle$
38. $\langle AMOUNT\_FREQUENCY\_TIME \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle FREQUENCY\_TIME \rangle$
39. $\langle AMOUNT\_PER\_TIME \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle PER\_TIME\_INDICATOR \rangle$
40. $\langle AMOUNT\_PER\_DURATION \rangle \rightarrow \langle AMOUNT \rangle \langle RANGE \rangle \langle FREQUENCY\_TIME \rangle$
41. $\langle FREQUENCY\_ITEM \rangle \rightarrow$ **hourly | daily | weekly | bi-weekly | bi-weekly | monthly | yearly | annually**
42. $\langle PER\_INDICATOR \rangle \rightarrow$ **per | / |** $\langle FREQUENCY\_INDICATOR \rangle$
43. $\langle PER\_SECOND \rangle \rightarrow \langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle SECOND \rangle$
44. $\langle PER\_MINUTE \rangle \rightarrow \langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle MINUTE \rangle$
45. $\langle PER\_HOUR \rangle \rightarrow$ **hourly |** $\langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle HOUR \rangle$
46. $\langle PER\_DAY \rangle \rightarrow$ **daily | nightly |** $\langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle DAY \rangle$
47. $\langle PER\_WEEK \rangle \rightarrow$ **weekly | bi-weekly | biweekly |** $\langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle WEEK \rangle$
48. $\langle PER\_MONTH \rangle \rightarrow$ **monthly—**$\langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle MONTH \rangle$
49. $\langle PER\_YEAR \rangle \rightarrow$ **yearly | annually |** $\langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle YEAR \rangle$
50. $\langle PER\_DECADE \rangle \rightarrow \langle PER\_INDICATOR \rangle \langle RANGE \rangle \langle DECADE \rangle$

51. ⟨*PER_TIME_INDICATOR*⟩ → ⟨*PER_SECOND*⟩ | ⟨*PER_MINUTE*⟩ | ⟨*PER_HOUR*⟩ | ⟨*PER_DAY*⟩ | ⟨*PER_WEEK*⟩ | ⟨*PER_MONTH*⟩ | ⟨*PER_YEAR*⟩ | ⟨*PER_DECADE*⟩
52. ⟨*PER_DURATION_INDICATOR*⟩ → ⟨*PER_INDICATOR*⟩ ⟨*RANGE*⟩ ⟨*DURATION_INDICATOR*⟩
53. ⟨*AMOUNT_PER_TIME_INDICATOR*⟩ → ⟨*AMOUNT*⟩ ⟨*RANGE*⟩ ⟨*PER_TIME_INDICATOR*⟩
54. ⟨*AMOUNT_FREQUENCY*⟩ → ⟨*AMOUNT*⟩⟨*RANGE*⟩ ⟨*FREQUENCY_INDICATOR*⟩
55. ⟨*FREQUENCY_INDICATOR*⟩ → **times** | **times a** | **times an** | **both times**
56. ⟨*DAY*⟩ → **day** | **days** | **night** | **nights** | **nite** | **nites** | **morning** | **mornings** | **mornin** | **evening** | **evenin** | **evenings** | **afternoon** | **noon**

The ⟨*DOSAGE*⟩ template class is as follows:

57. ⟨*DOSAGE*⟩ → ⟨*NUMERIC_AMOUNT_UNIT*⟩ |⟨*WORDED_NUMERIC_AMOUNT_UNIT*⟩

The ⟨*ENTITY*⟩ template class is as follows:

58. ⟨*ENTITY*⟩ → $L(C) \cup L_s(C) \cup L(R) \cup L_s(R)$

The route-of-administration ⟨*ROA*⟩ template class is defined as follows:

59. ⟨*ROA*⟩ → ⟨*ENTERAL*⟩|⟨*EPIDURAL*⟩|⟨*INTRAARTERIAL*⟩ |⟨*INTRACARDIAC*⟩|⟨*INTRACEREBRAL*⟩|⟨*INTRADERMAL*⟩ |⟨*INTRAMUSCULAR*⟩|⟨*INTRAVENOUS*⟩|⟨*INHALATIONAL*⟩ |⟨*INTRAPERITONEAL*⟩|⟨*INTRATHECAL*⟩|⟨*INTRAOSSEOUS INFUSION*⟩|⟨*NASAL*⟩|⟨*PARENTERAL*⟩|⟨*TRANSDERMAL*⟩ |⟨*TRANSMUCOSAL*⟩ | ⟨*TOPICAL*⟩ | ⟨*SUBCUTANEOUS*⟩
60. ⟨*INTRAPERITONEAL*⟩ → ⟨*INTRACEREBRAL*⟩
61. ⟨*INTRATHECAL*⟩ → ⟨*INTRACEREBRAL*⟩
62. ⟨*INTRAOSSEOUS INFUSION*⟩ → ⟨*INTRACEREBRAL*⟩
63. ⟨*PARENTERAL*⟩ → ⟨*EPIDURAL*⟩
64. ⟨*INTRAMUSCULAR*⟩ → ⟨*EPIDURAL*⟩ | **skin poppin**
65. ⟨*INTRAVENOUS*⟩ → ⟨*INTRAARTERIAL*⟩
66. ⟨*TOPICAL*⟩ → ⟨*TRANSDERMAL*⟩
67. ⟨*SUBCUTANEOUS*⟩ → ⟨*INTRACEREBRAL*⟩
68. ⟨*INTRADERMAL*⟩ → ⟨*INTRAARTERIAL*⟩ | **sniff** | **snort** | **snorting** | **bumping** | **railing** | **doozing**
69. ⟨*ENTERAL*⟩ → **ate** | **chewing** | **drink** | **eat** | **insufflate** | **plug** | **plugged** | **smoke** | **smoked** | **sniff** | **snort** | **...**
70. ⟨*EPIDURAL*⟩ → **inject** | **injected** | **injection**
71. ⟨*INTRAARTERIAL*⟩ → **IV** | **IVed** | **IVing** |**inject** | **injected** | **...**
72. ⟨*INTRACARDIAC*⟩ → ⟨*EPIDURAL*⟩
73. ⟨*INTRACEREBRAL*⟩ → ⟨*EPIDURAL*⟩
74. ⟨*INHALATIONAL*⟩ → **smoke** | **smokes** | **smoked** | **smoking** | **sniff** | **sniffed** | **sniffing** | **snort** | **snorted** | **snorting** | **bumping** | **railing** | **doozing**
75. ⟨*NASAL*⟩ → **sniff** | **snort** | **snorting** | **bumping** | **railing** | **doozing**
76. ⟨*TRANSDERMAL*⟩ → **patch** | **patches**
77. ⟨*TRANSMUCOSAL*⟩ → **snort** | **snorted** | **snorting** | **sniff** | **sniffed** | **sniffing** | **bumping** | **railing** | **doozing**

The ⟨*DRUGFORM*⟩ template class is defined as follows:

78. ⟨*DRUGFORM*⟩ → ⟨*LIQUID*⟩ | ⟨*SOLID*⟩
79. ⟨*LIQUID*⟩ → **syrups** | **elixirs** | **suspensions** | **ointment** |**...**
80. ⟨*SOLID*⟩ → **powder** | **tablet** | **tablets** | **tab** | **tabs** | **pill** | **...**

The ⟨*SIDEEFFECT*⟩ template class is defined as follows:

81. ⟨*SIDEEFFECT*⟩ → ⟨*MILD*⟩ | ⟨*MODERATE*⟩ | ⟨*SEVERE*⟩
82. ⟨*MILD*⟩ → **bruising** | **itching** | **itching of skin** | **tingling** | **...**
83. ⟨*MODERATE*⟩ → **blisters** | **blistering** | **skin blisters that are itchy** | **skin blisters that are painful** | **skin discoloration** | **...**
84. ⟨*SEVERE*⟩ → **abnormal heartbeat** | **bone pain** | **chest pain** | **chest discomfort** | **chest tightness** | **chills** | **coma** | **...**

The ⟨*EMOTION*⟩ template class is defined as follows:

85. ⟨*EMOTION*⟩ → ⟨*AFFECTION*⟩ | ⟨*LUST*⟩ | ⟨*LONGING*⟩ | ⟨*CHEERFULNESS*⟩ | ⟨*ZEST*⟩ | ⟨*CONTENTMENT*⟩ | ⟨*PRIDE*⟩ | ⟨*OPTIMISM*⟩ | ⟨*ENTHRALLMENT*⟩ | ⟨*RELIEF*⟩ | ⟨*SURPRISE*⟩ | ⟨*IRRITATION*⟩ | ⟨*EXASPERATION*⟩ | ⟨*RAGE*⟩ | ⟨*DISGUST*⟩ | ⟨*ENVY*⟩ | ⟨*TORMENT*⟩ | ⟨*SUFFERING*⟩ | ⟨*DEPRESSION*⟩ | ⟨*DISAPPOINTMENT*⟩ | ⟨*SHAME*⟩ | ⟨*NEGLECT*⟩ | ⟨*SYMPATHY*⟩ | ⟨*HORROR*⟩ | ⟨*CONFUSE*⟩ | ⟨*DISCONTENTMENT*⟩ | ⟨*EMBARRASSMENT*⟩ | ⟨*FORGIVENESS*⟩ | ⟨*THANKFULNESS*⟩ | ⟨*BLAME*⟩ | ⟨*NERVOUSNESS*⟩ | ⟨*LOVE*⟩ | ⟨*JOY*⟩ | ⟨*ANGER*⟩ | ⟨*SADNESS*⟩ | ⟨*FEAR*⟩
86. ⟨*LOVE*⟩ → ⟨*AFFECTION*⟩ ⟨*LUST*⟩ ⟨*LONGING*⟩
87. ⟨*JOY*⟩ → ⟨*CHEERFULNESS*⟩ ⟨*ZEST*⟩ ⟨*CONTENTMENT*⟩ ⟨*PRIDE*⟩ ⟨*OPTIMISM*⟩ ⟨*ENTHRALLMENT*⟩ ⟨*RELIEF*⟩
88. ⟨*ANGER*⟩ → ⟨*IRRITATION*⟩ ⟨*EXASPERATION*⟩ ⟨*RAGE*⟩ ⟨*DISGUST*⟩ ⟨*ENVY*⟩ ⟨*TORMENT*⟩
89. ⟨*SADNESS*⟩ → ⟨*SUFFERING*⟩|⟨*DEPRESSION*⟩|⟨*DISAPPOINTMENT*⟩ |⟨*SHAME*⟩|⟨*NEGLECT*⟩|⟨*SYMPATHY*⟩
90. ⟨*FEAR*⟩ → ⟨*HORROR*⟩ | ⟨*NERVOUSNESS*⟩
91. ⟨*AFFECTION*⟩ → **adoration** | **affection** | **love** | **fondness** | **liking** | **attraction** | **caring** | **...**
92. ⟨*LUST*⟩ → **arousal** | **desire** | **lust** | **lusting** | **passion** | **infatuation**
93. ⟨*LONGING*⟩ → **longing**
94. ⟨*CHEERFULNESS*⟩ → **amused** | **amusement** | **bliss** | **blithe** | **...**
95. ⟨*ZEST*⟩ → **enthusiasm** | **zeal** | **zest** | **excited** | **exciting** | **excitement** | **thrill** | **thrilling** | **exhilaration**
96. ⟨*CONTENTMENT*⟩ → **contented** | **contentedness** | **contentment** | **pleasure** | **satisfied** | **satisfaction** | **gratified** | **gratification**
97. ⟨*PRIDE*⟩ → **pride** | **proud** | **prideful** | **pridefulness** | **triumph**
98. ⟨*OPTIMISM*⟩ → **eagerness** | **expecting** | **hope** | **hopeful** | **hoping** | **hopefulness** | **optimistic** | **optimism**
99. ⟨*ENTHRALLMENT*⟩ → **enthrallment** | **enthrall** | **rapture**
100. ⟨*RELIEF*⟩ → **relief** | **ease** | **relaxation** | **alleviation**
101. ⟨*SURPRISE*⟩ → **amazement** | **amazed** | **surprise** | **surprised** | **surprising** | **astonished** | **astonishment** | **astounded** | **unexpected**
102. ⟨*IRRITATION*⟩ → **aggravation** | **irritation** | **irritated** | **irritating** | **agitation** | **annoyed** | **annoyance** | **disturbing** | **grouchiness** | **grumpiness**
103. ⟨*EXASPERATION*⟩ → **exasperation** | **frustration**
104. ⟨*RAGE*⟩ → **anger** | **rage** | **outrage** | **fury** | **wrath** | **hostility** | **...**
105. ⟨*DISGUST*⟩ → **disgust** | **revulsion** | **contempt** | **disgusting** | **disgusted**
106. ⟨*ENVY*⟩ → **envy** | **jealousy** | **jealous** | **envying**
107. ⟨*TORMENT*⟩ → **torment** | **tormented**
108. ⟨*SUFFERING*⟩ → **aggravation** | **irritation** | **irritated** | **irritating** | **...**
109. ⟨*DEPRESSION*⟩ → **depressed** | **depression** | **depressing** | **cheerless** | **despair** | **despairing** | **...**

110. ⟨*DISAPPOINTMENT*⟩ → **dismay** | **disappointment** | **disappointed** | **disappointing** | **displeasure** | **letdown**
111. ⟨*SHAME*⟩ → **ashamed** | **shame** | **regret** | **regretful** | **regretting** | **remorseful** | **guilt** | **remorse** | **guilty** | **...**
112. ⟨*NEGLECT*⟩ → **alienation** | **isolation** | **neglect** | **loneliness** | **...**
113. ⟨*SYMPATHY*⟩ → **pity** | **sympathy** | **compassion** | **compassionate** | **...**
114. ⟨*HORROR*⟩ → **alarm** | **shock** | **hysteria** | **mortification** | **...**
115. ⟨*CONFUSE*⟩ → **confused** | **confusing** | **confusion** | **confuse**
116. ⟨*DISCONTENTMENT*⟩ → **discontent** | **discontented** | **...**
117. ⟨*EMBARRASSMENT*⟩ → **embarrassment** | **embarrass** | **...**
118. ⟨*FORGIVENESS*⟩ → **forgiveness** | **forgive** | **pardon** | **forgiving**
119. ⟨*THANKFULNESS*⟩ → **thankfulness** | **thankful** | **appreciation** | **...**
120. ⟨*BLAME*⟩ → **blame** | **blamed** | **blaming** | **...**
121. ⟨*NERVOUSNESS*⟩ → **anxiety** | **nervousness** | **tenseness** | **...**

The ⟨*PRONOUN*⟩ template class is defined as follows:

122. ⟨*PRONOUN*⟩ → ⟨*DEMONSTRATIVE_PRONOUN*⟩ |⟨*PERSONAL_PRONOUN*⟩|⟨*POSSESSIVE_PRONOUN*⟩ |⟨*REFLEXIVE_PRONOUN*⟩|⟨*RELATIVE_PRONOUN*⟩ |⟨*INDEFINITE_PRONOUN*⟩|⟨*INTERROGATIVE_PRONOUN*⟩
123. ⟨*DEMONSTRATIVE_PRONOUN*⟩ → **this** | **that** | **these** | **those** | **...**
124. ⟨*PERSONAL_PRONOUN*⟩ → **i** | **me** | **you** | **she** | **her** | **he** | **...**
125. ⟨*POSSESSIVE_PRONOUN*⟩ → **my** | **our** | **ours** | **your** | **yours** | **...**
126. ⟨*REFLEXIVE_PRONOUN*⟩ → **myself** | **ourselves** | **yourself** | **...**
127. ⟨*RELATIVE_PRONOUN*⟩ → **that** | **which** | **who** | **whom** | **whose** | **...**
128. ⟨*INDEFINITE_PRONOUN*⟩ → **anybody** | **anyone** | **anything** | **...**
129. ⟨*INTERROGATIVE_PRONOUN*⟩ → **what** | **who** | **which** | **whom** | **...**

The ⟨*INTENSITY*⟩ template class is defined as follows:

130. ⟨*INTENSITY*⟩ → ⟨*LOW*⟩ | ⟨*AVERAGE*⟩ | ⟨*HIGH*⟩
131. ⟨*LOW*⟩ → **low** | **very low** | **lower** | **lower than** | **lowest** | **...**
132. ⟨*AVERAGE*⟩ → **average** | **ideal** | **...**
133. ⟨*HIGH*⟩ → **high** | **very high** | **higher** | **highest** | **large** | **...**

The ⟨*SENTIMENT*⟩ template class is defined as follows:

134. ⟨*SENTIMENT*⟩ → ⟨*POSITIVE*⟩ | ⟨*NEGATIVE*⟩ | ⟨*NEUTRAL*⟩
135. ⟨*POSITIVE*⟩ → **Im glad** | **luckily** | **awesome** | **benefit** | **best choices** | **best for me** | **best** | **...**
136. ⟨*NEGATIVE*⟩ → **big f*cking mistake** | **threw up** | **It was bad** | **Its really rough** | **...**
137. ⟨*NEUTRAL*⟩ → **hope** | **longer** | **well** | **as well** | **...**

The set of contextual compilation defines additional semantics for common constructs, which can be easily extended and reused across different domains.

138. ⟨*greaterThanOp*⟩ → > | **greater than** | **more than** | **above** | **in excess of** | **slightly above** | **little more**| **bit more** | **slightly more** | **high** |...
139. ⟨*lessThanOp*⟩ → < | **less than** | **lower than** | **below** | **in lack of** | **slightly below** | **little less** | **bit less** | **slightly less** |...
140. ⟨*equalToOp*⟩ → = | **exactly** | **precisely** | ...
141. ⟨*greaterThanEqualToOp*⟩ → >= | **greater than** | **greater than or equal to** | **more than** | **above** | **in excess of** | **slightly above** | **little more** | **bit more** | **slightly more** | **exactly** | **precisely** | **high** | ...
142. ⟨*lessThanEqualToOp*⟩ → <= | **less than** | **less than or equal to** | **lower than** | **below** | **in lack of** | **slightly below** | **little less** | **bit less** | **slightly less** | **exactly** | **precisely** | ...