RC25627 (WAT1610-019) October 5, 2016
Computer Science

IBM Research Report

SemanticFind: Locating What You Want in a Patient Record, Not Just What You Ask For

John M. Prager, Jennifer J. Liang, Murthy V. Devarakonda
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598 USA

Research Division
Almaden – Austin – Beijing – Brazil – Cambridge – Dublin – Haifa – India – Kenya – Melbourne – T.J. Watson – Tokyo – Zurich

SemanticFind: Locating What You Want in a Patient Record, Not Just What You Ask For

John M. Prager, PhD, Jennifer J. Liang, MD, Murthy V. Devarakonda, PhD
IBM Research, Yorktown Heights, NY

Abstract

We present a new model of patient record search, called SemanticFind, which goes beyond traditional textual and medical synonym matches in locating patient data that a clinician would want to see rather than just what they ask for. The new model is implemented by making extensive use of the UMLS semantic network, distributional semantics, and NLP, to match query terms along several dimensions in a patient record, and the returned matches are organized accordingly. The new approach finds all clinically related concepts without the user having to ask for them. An evaluation of the accuracy of SemanticFind shows that it found twice as many relevant matches compared to those found by literal (traditional) search alone, along with very high precision and recall. These results suggest potential uses for SemanticFind in clinical practice, retrospective chart reviews, and in automatically extracting quality metrics.

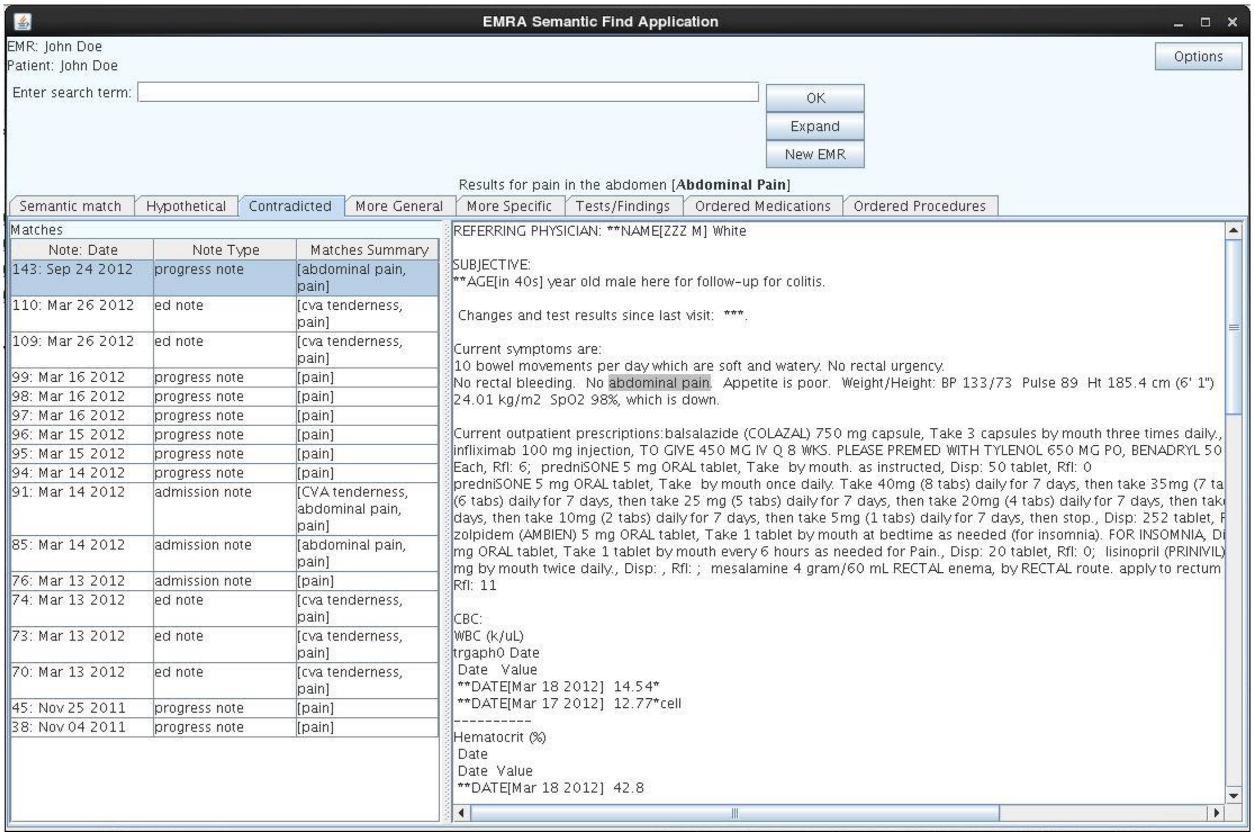
1. Introduction

The need for a search function within the patient record has been well-documented [1] even before the development of health information technology. With the advent of Electronic Health Records (EHRs), the amount of information that can be easily recorded in a patient record has increased. Traditionally, medical records were primarily used to document a patient's medical history and clinical care process to assist physicians in providing informed care. However, EHRs currently serve multiple functions, including patient scheduling, billing, coding, and documenting informed consent. As a result, physicians are finding it increasingly difficult to locate specific information within the EHR, making an effective EHR search utility all the more necessary.

Studies have identified various barriers to information search within EHRs in the patient care process, including lack of time and doubt about the existence of relevant information [2] [3]. Due to difficulties in locating relevant information within EHRs and the time constraints inherent to the clinical setting, physicians often leave questions unanswered at the point-of-care, with subsequent effects on clinical practice and patient care. By making information within the EHR more easily accessible, an efficient and effective search application could potentially reduce physicians' cognitive load, improve patient care and reduce medical costs. A University of Michigan study found that use of a search engine optimized for finding clinical information in free-text within EHRs resulted in significant time saved while maintaining reliability [4]. In a small study, semantic search has been shown to reduce both search time and number of clicks [5].

While a few EHR search tools are evolving, they place the burden on users to find the relevant and useful information [6] [7] [8]. A recent report discusses a graphical model and semantic inferencing over the graph for searching medical records [9], but it is evaluated as a document retrieval task, where we evaluate matches and present a new search application model. We discuss the use of UMLS and distributional semantics in a prototype application called SemanticFind that has the ability to perform sophisticated searches and places the burden on itself rather than on the user. The user simply enters the search term, while the system performs a number of appropriate searches and organizes the results on the screen in a tabbed interface, where each tab represents results of a particular kind of search (see Figure 1); the prototype user interface is described more fully later.

Building an application that is able to find not only instances of the search term, but others that are related to it in a clinically meaningful way, presents three kinds of challenges: (1) performing the necessary natural language understanding and matching, (2) using general medical knowledge in order to drive and control the finding of useful relationships, and (3) high usability: allowing the user to enter the search terms easily, and to present the resulting search hits in an organized and readily-understandable way. Because of these challenges the output produced may not always be what is expected, and therefore it makes sense to conduct an experimental evaluation of the system.



EMR: John Doe
Patient: John Doe
Enter search term:

Results for pain in the abdomen [**Abdominal Pain**]

Matches

Note: Date	Note Type	Matches Summary
143: Sep 24 2012	progress note	[abdominal pain, pain]
110: Mar 26 2012	ed note	[cva tenderness, pain]
109: Mar 26 2012	ed note	[cva tenderness, pain]
99: Mar 16 2012	progress note	[pain]
98: Mar 16 2012	progress note	[pain]
97: Mar 16 2012	progress note	[pain]
96: Mar 15 2012	progress note	[pain]
95: Mar 15 2012	progress note	[pain]
94: Mar 14 2012	progress note	[pain]
91: Mar 14 2012	admission note	[CVA tenderness, abdominal pain, pain]
85: Mar 14 2012	admission note	[abdominal pain, pain]
76: Mar 13 2012	admission note	[pain]
74: Mar 13 2012	ed note	[cva tenderness, pain]
73: Mar 13 2012	ed note	[cva tenderness, pain]
70: Mar 13 2012	ed note	[cva tenderness, pain]
45: Nov 25 2011	progress note	[pain]
38: Nov 04 2011	progress note	[pain]

```
REFERRING PHYSICIAN: **NAME[ZZZ M] White

SUBJECTIVE:
**AGE[in 40s] year old male here for follow-up for colitis.

 Changes and test results since last visit:  ***.

Current symptoms are:
10 bowel movements per day which are soft and watery. No rectal urgency.
No rectal bleeding.  No abdominal pain.  Appetite is poor.  Weight/Height: BP 133/73  Pulse 89  Ht 185.4 cm (6' 1")
24.01 kg/m2  SpO2 98%, which is down.

Current outpatient prescriptions:balsalazide (COLAZAL) 750 mg capsule, Take 3 capsules by mouth three times daily.,
infliximab 100 mg injection, TO GIVE 450 MG IV Q 8 WKS. PLEASE PREMED WITH TYLENOL 650 MG PO, BENADRYL 50
Each, Rfl: 6;  predniSONE 5 mg ORAL tablet, Take  by mouth. as instructed, Disp: 50 tablet, Rfl: 0
predniSONE 5 mg ORAL tablet, Take  by mouth once daily. Take 40mg (8 tabs) daily for 7 days, then take 35mg (7 ta
(6 tabs) daily for 7 days, then take 25 mg (5 tabs) daily for 7 days, then take 20mg (4 tabs) daily for 7 days, then tak
days, then take 10mg (2 tabs) daily for 7 days, then take 5mg (1 tabs) daily for 7 days, then stop., Disp: 252 tablet, R
zolpidem (AMBIEN) 5 mg ORAL tablet, Take 1 tablet by mouth at bedtime as needed (for insomnia). FOR INSOMNIA, Di
mg ORAL tablet, Take 1 tablet by mouth every 6 hours as needed for Pain., Disp: 20 tablet, Rfl: 0;  lisinopril (PRINIVIL)
mg by mouth twice daily., Disp: , Rfl: ;  mesalamine 4 gram/60 mL RECTAL enema, by RECTAL route. apply to rectum
Rfl: 11

CBC:
WBC (k/uL)
trgaph0 Date
 Date   Value
 **DATE[Mar 18 2012]  14.54*
 **DATE[Mar 17 2012]  12.77*cell
----------
Hematocrit (%)
 Date
 Date  Value
 **DATE[Mar 18 2012]  42.8
```

Figure 1. *Shows results for issuing "pain in the abdomen" for a certain patient. The UMLS preferred name for this concept is "abdominal pain", which is present in many locations in this patient record, but the term as typed in is not present, so there is no Literal Match. For this screenshot, the Contradicted Match tab and the first note in its hit list were selected; the matched terms in all matching notes for this search type are summarized on the left and highlighted within the selected note on the right. Note that the highlighted match is in a negated context.*

2. SemanticFind Functionality

SemanticFind extends the common keyword-search paradigm to help the clinician find what he/she means by the meaning of the keywords, rather than solely by the literal input. To that goal, SemanticFind performs a variety of searches on a single query, including synonyms and paraphrases of the search term, negated and hypothetical mentions, and matches to other related medications, labs and procedures over both structured and unstructured data [11].

Even outside of the medical domain, users of keyword search systems run into difficulties when the material they are searching represents the concepts they are seeking using synonyms or paraphrases. In health care, the same applies but in more extreme ways; for example, some concepts are "latent", requiring interpretation of test results. While we are not suggesting that SemanticFind perform diagnosis (or that necessarily any keyword-search system should do it), we do think it is reasonable that a search for "hyperkalemia" should match "K 6.1".

Negation is also a property of medical texts in general and medical records in particular that is critical for the correct understanding of the content, and by extension should be handled properly by a search application [11]. Much of the EHR can be viewed as assertions of what is true of the patient (at the time of the recording); some of these assertions are of conditions that the patient is found not to have. Some medical properties are mutually exclusive (such as hyperX and hypoX for any X), so an assertion of one also asserts the absence of the other. It is useful for a search application to find not only what a user is looking for, but also any direct indications that it is absent.

Conditions are often present in an EHR in a hypothetical context, where it is not asserted that the patient does or does not have a condition, but it may be mentioned as something to test for, that the patient must be aware of the risks of it, that a medication or procedure is being given as a preventative measure, and so on. These will be valid matches when the condition is a search term, but should not be mixed in with the positively or negatively asserted matches. In a similar vein, search terms might be more general or more specific than a term in the record: again matching should occur, but the results organized so as not to conflate the results. We describe in this section the different kinds of matches that SemanticFind supports, and how they are organized and presented to the user.

Table 1. *Searches performed in SemanticFind, with examples of search terms and corresponding matched text. Under More Specific and Assoc. Tests/Findings, several matches are shown, separated by commas*

Search Type/Tab	**Description**	**Search Term(s)**	**Example Match(es)**
Literal Match	The exact text in unstructured data, except case and singular/plural differences are disregarded	Hypertension	hypertension
Semantic Match	Terms in unstructured data that are medically synonymous, regardless of textual representation	Normal blood pressure Leg pain	BP 120/79 Pain in the lower limb
Hypothetical	Terms in unstructured data with same semantic meaning that are presented in a hypothetical context	DVT	DVT prophylaxis
Contradicted	Terms in unstructured data that are in negative context, incompatible with, or opposite of what was searched for	Normal blood pressure Smoking	Hypertension Patient denies smoking
More General	Ontological hypernyms in unstructured data	PTSD	Anxiety disorder
More Specific	Ontological hyponyms in unstructured data	Lung cancer	NSCLC, non-small cell lung cancer, pulmonary and hepatic metastasis, squamous cell carcinoma of the lung
Assoc. Tests/Findings	Terms of this type in unstructured data that co-occur in the medical literature.	Asthma	Wheezing, spirometry
Assoc. Treatments	Terms of this type in unstructured data that co-occur in the medical literature.	Antihypertensives	Blood pressure management
Assoc. Medications	Terms of this type in unstructured data that co-occur in the medical literature.	Asthma Antihypertensives	Albuterol Spironolactone
Ordered Medications	Entities in the structured data that are logically related, e.g. via “treats” or “prevents”	Hashimoto Disease ACE-I	Synthroid Lisinopril
Contraindicated Ordered Medications	Entities in the structured data that are logically related, e.g. via “causes”	Hypotension	Lisinopril
Ordered Procedures	Entities in the structured data that are logically related, e.g. via “diagnoses” or “treats”	HTN	EKG
Ordered Labs	Entities in the structured data that are logically related, e.g. via “measures”	Kidney	Renin Activity Plasma

2.1 Search Types

We implemented a total of 13 different types of searches, which can be grouped into four classes, based on the technology used and the meaning of the corresponding matches. Most of the matching (in fact, all of the "conceptual" matching) is mediated by UMLS **[12]** Concept Unique Identifiers (CUIs), as well as our own natural language processing tools (described in the next section). There are currently about 3 million distinct CUIs in UMLS, representing a large proportion of the distinct concepts in the medical domain. UMLS also maintains collections of relations between CUIs which we exploit (and in some cases have extended). Associated with each CUI is a "preferred name" and a set of "variants", representing likely ways the concept will be expressed in text. We describe below ways we use to increase the number of variations of the concept. Our general approach to search is to annotate the EHR contents with the CUIs associated with the contained medical terms, to similarly annotate the input search phrase with CUIs, and then to match them in several ways. The results of each kind of match are displayed in a separate "tab" in the graphical user interface – see Table 1.

2.2 Concept (CUI) Annotation

The recognition of CUIs in text is performed by an annotation pipeline using UIMA [13][14]. This pipeline performs annotation of recognized medical concepts with annotations that record, amongst other data, the UMLS semantic type and the CUI of the concept. These semantic types include *Finding, Diagnostic Procedure, ClinicalDrug, DiseaseOrSyndrome* and about 130 others.

The pipeline consists of the following stages: **tokenization, parsing and predicate-argument structure generation**, **dictionary lookup of concepts in our Medical Concept Dictionary (MCD)**[i,] **lab value analysis and annotation**, **further concept recognition via** a "**concept transformer"**, and **negation & hypothetical detection**.

The lab value analysis uses pattern and parse-based analysis to detect measurements of quantities, normalizing units where appropriate. If the quantity is found in our tables of reference ranges goes ahead and asserts the appropriate condition, including normality. Thus "potassium was 8 mEq/L" gets annotated as "hyperkalemia", but "potassium 4 mEq/L" gets annotated as "normal potassium".

Concept Transformer understands the different ways that are used to express modification in English (such as adjectives in front of the noun, prepositional phrases following), use of synonyms, and also how to convert between nouns and adjectives and vice-versa (via WordNet **[16]**). For example, "pain in the abdomen" = "abdominal pain", "swelling of the ankle" = "edematous ankle", "heart problems" = "cardiac problems" and so on.

Negation Detection is the process by which assertions (typically findings and diseases) are detected in the scope of a negation trigger. Triggers include "no", "not", "never", "without", "absent", "denies", amongst others. Unlike the commonly used NegEx **[17]** our component uses the parse tree to determine which are the actual concepts that are asserted not to hold. Thus "swelling of the leg was not present" negates the swelling, not the leg. Hypothetical detection works in a similar way, except with its own set of triggers ("if", "risk of", "rule out", "prophylaxis", etc.).

2.3 Traditional Search

This search, which is the only one which directly compares text strings, is very similar to the common "Control-F" functionality found in document reading/editing systems. The differences are

1. Case is ignored. EHR contents are not subject to editorial review, so casing is an unreliable indicator.
2. Both singular and plural forms of the search term are searched for, regardless of whether the input is singular or plural. This is particularly important for Latin and Greek words whose singular form is often not a substring of the plural (e.g. uterus/uteri).

Results are shown in the Literal Match tab.

2.4 Conceptual Search

This search matches CUIs in the input with CUIs in the EHR document text. Depending on the nature of the match, the result gets added to the appropriate tab: equal or synonymous matches are listed under Semantic Match, unless one is in a negative context or they are opposite or incompatible, when the Contradicted tab is used, or if the context is hypothetical or un-asserted, when Hypothetical is used. If an ISA relation or chain exists between the concepts, the More Specific or More General tab is used.

[i] This is similar conceptually to MetaMap [15], but only exact matches against UMLS concept preferred names and variants are allowed.

2.5 Associative Search

This search uses Latent Semantic Analysis (LSA) **[18]** which from a corpus of training data quantifies the degree to which two concepts have similar distributional properties. We know the semantic types of the concepts involved, but LSA does not give us the kind of relation, just a score representing the probability it exists. We use a threshold of 0.5, on a scale of 0 to 1. The UMLS semantic type of the concept in the document determines the tab that the match is associated with.

2.6 Inferential Search

In addition to clinical notes, EHRs contain structured collections of quantities related to the patient's care. These include ordered medications, procedures and lab tests, amongst others. Our inferential search makes connections between search terms and these quantities by chaining together one or more relations in medically- and logically-meaningful ways.

UMLS contains many instances of the *treats, prevents* and *causes* relations associating medications or procedures with symptoms and diseases. When the search term is a condition, it may be useful to match it against medications and procedures ordered for the patient which are related to it in one of these ways, but because of mismatched granularity/specificity the appropriate UMLS relation will often not directly apply. Our inferential search employs general rules such as:

- If drug D treats condition C, then medication M that contains D also treats C.
- If medication M treats condition C, then it treats more specific kinds of C
- If a test T is used to measure quantity Q, then a panel P that includes T also measures Q

To allow for looser terminology when specifying search terms and maximum recall during matches, these rules are meant to uncover all possible correct relations rather than only identifying chains that are always true. In addition, note that these rules do not take account of conventional practice or whether the found connection (such as medication treating a condition) is the best possible. Rather they find chains that are logically correct and of potential interest, given that one end is the search term, and the other is a quantity associated with the patient.

By appropriate chaining of relations (from UMLS) according to rules such as those above (developed by us) we get inference chains such as the following when the search term is *infection*:

> Infection <includes> Lower respiratory tract infection <treated by> Amoxicillin <is ingredient of> Augmentin 875 mg-125 mg tablet

which is in the patient's ordered medications list. Note that we are not asserting that this medication was ordered for this purpose: such treatment connections are not always explicitly made in the EHR.

2.7 SemanticFind Demonstration User Interface

The user interface is intended to demonstrate how the search results could be organized in understanding the functionality of SemanticFind, and as mentioned earlier, the assessment or effectiveness of the UI is not a goal of this study.

SemanticFind organizes the results on the screen in a tabbed interface, where each tab represents results of a particular kind of search (see Figure 1). The contents of each tabbed pane depend on whether the search was on unstructured (clinical notes) or structured content (such as ordered medications, procedures, or labs). Each tabbed pane is split horizontally into two parts; one part summarizes all results of this particular kind of search, while the other part allows detailed view of the matched content within the context of the original unstructured document or structured table. For any given search, only tabs containing at least one search result are presented.

For unstructured search results, on the left is a hit list of matched clinical notes, on the right the contents of the currently selected note. In the hit list each row corresponds to a clinical note with one or more matched terms, and these are arranged in reverse chronological order (most recent at the top). Each entry is in three columns, representing the date/id of the note, the kind of note, and a summary of the text of the matches found in the note. In the content panel on the right, the matches are highlighted within the selected note.

Structured search results are similar, but with a couple of differences. Instead of the note type, an explanation is given of the reason for the match between the search term and the structured item. This is felt to be useful to the user since the connection might not always be obvious. In the right-hand pane, the corresponding entry from the structured repository is given. We are experimenting with alternate displays, such as a timeline of medication orders against dosage.

The screenshot in Figure 1 shows the result of issuing the query “pain in the abdomen”. There is no Literal Match of the string, so that tab is absent. Under Contradicted there are three notes, each with a match to one or more of the following terms: “abdominal pain”, “cva tenderness”, “pain”. Even though “pain” is a more general concept than “pain in the abdomen”, it is located in the Contradicted match since negating the former logically negates the latter. The Contradicted tab is selected in this screenshot, as is one of the notes in the hit-list on the left. In the right panel, the note content is shown, with the matching terms highlighted.

Other tab panes show matches such as “periumbilical pain” in More Specific, “constipation” in Assoc. Tests/Findings, and “acetaminophen 325 mg tablet” in Ordered Medications. We do not show screenshots of these other tabs for reasons of space; instead we summarize in Table 2 the matches found across tabs.

Unlike traditional information retrieval systems, which display search “hits” based on a relevancy score (the highest at the top), SemanticFind organizes its results under various categories and in reverse chronological order. Categorization of the results helps users easily navigate to what they want to know about the search term. The reverse chronological ordering is common to displaying patient information in the clinical setting and familiar to users of most EHR systems, where temporal proximity to the present time may be more useful. For example, a ten-year-old clinical note with a well-documented discussion of the treatment of hypertension may have the highest relevancy score in a traditional search system for a search on “hypertension”, but may not be very relevant to the patient’s care today.

3. Methods

A study was conducted to evaluate the performance of some of the conceptual matches performed by our system. Semantic Match, Contradicted and More Specific were evaluated. Hypothetical was not operative at the time. More General was excluded due to anticipated difficulties in judging matched concepts that may be too broad to be of use. Other matches that SemanticFind can make (see Table 1) are of an associative or logical nature, and are also not as straightforward to evaluate [19], so are omitted from the current evaluation.

The traditional way to evaluate search systems [20] is to perform document relevance judgments on the returned document lists. We used a different methodology for assessing SemanticFind because:

1. We are assessing mentions, not documents.
2. We are distinguishing between types of matches (e.g. whether more specific or negated), which has not been a concern of traditional systems.

These differences imply a qualitatively different form of evaluation than in traditional Information Retrieval. The latter generates a ranked list of documents whose relevance is judged, and measures are calculated based on the scores of the top N documents for some N, or for all documents that pass a given score threshold. Our judgments are on individual mention matches, which are binary, so ranking-based metrics are meaningless. Instead, as described in the *Metrics* section later, we count matches that are correct (true positives), matches that are incorrect (false positives) and misses (false negatives), and compute evaluation scores from these counts.

True and false positives are evaluated by showing evaluators the matches in context (see *Evaluation Interface*). Misses can be determined in principle by tasking annotators to carefully read each clinical note and, for each search term, note any textual expressions that represent it (via any of the match types of interest). Any of these that the system does not find automatically is a miss. However, this human process is extremely time-consuming and error-prone, and for that reason we designed an alternative approach involving the generation of paraphrases. We reasoned that the search-term equivalents that an annotator would identify in text would be the same as any paraphrases they could generate from the search term in a stand-alone task.

Through a research collaboration agreement with the Cleveland Clinic, we acquired a number of de-identified EHRs available for our use. Ten of these were selected at random. For each patient record, an MD (Liang) generated a set of 12 or more search terms pertinent to that EHR by reviewing the patient’s last progress note and problem list [21]. This method of generating search terms was meant to simulate what healthcare practitioners might ask at the point-of-care, where they may review the patient’s most recent progress note to get a quick overview of the patient prior to the patient encounter. These search terms were then run through SemanticFind and the output reviewed by two fourth-year medical students recruited to participate in the evaluation.

***Table 2**. Terms in an EHR matched against the search "pain in the abdomen", organized by search type, displayed alphabetically. The columns reflect the typical situation where for a given patient and search term, only some of the 13 search types produce matches. Contradicted matches occur in contexts such as "not …", "negative for …", or "denies …". For reasons of space, the Ordered Medications have been truncated to display only the medication name, leaving out the strength and formulation. Note that some of the Ordered Medications are connected indirectly to the search term, for example "lisinopril" treats hypertension, which can cause "pain in the abdomen".*

Conceptual				Associative	Inferential	
Semantic	**Contradicted**	**More General**	**More Specific**	**Assoc. Tests/Findings**	**Ordered Medications**	**Ordered Procedures**
abd pain abdominal pain	abdominal pain cva tenderness pain	neurological pain sensation	pain around umbilicus periumbilical pain tenesmus	abd pain abdominal discomfort abdominal pain bloating …	Acetaminophen Balsalazide Diphenhydramine Fentanyl Lisinopril …	CT Enterography w contrast

3.1 Search terms used in the Evaluation

A total of 169 search terms were generated over 10 EHRs, ranging from 13 to 32 search terms per EHR. Search terms included a variety of semantic types (e.g. diseases, findings, labs, medications), single and multi-word concepts, different parts of speech, as well as commonly accepted medical abbreviations. Of the 169 search terms, 134 were unique.

To aid in computing system misses, the assessors were then asked to generate paraphrases for each of the 134 unique search terms. They were instructed to generate as many paraphrases as they could with a few minutes' consideration (including none if they felt it was not possible for a particular term), where, according to their judgment, the paraphrases meant clinically the same as the original term. We explain how these paraphrases were used in evaluation in the *Metrics* section below.

A total of 652 paraphrases were generated, ranging from 0 to 13 paraphrases per search term. These alternatives were generally in the form of abbreviations, abbreviation expansions, definitions and general English paraphrases. Table 3 shows the paraphrase for some of the designated search terms for a particular EHR.

3.2 Evaluation Interface

For evaluation, the interface is enhanced with widgets to allow the user to enter judgments (not shown for reasons of space). In evaluation mode, the assessor will issue a search term, choose a search type, then proceed to examine each note that contains matches of that type, click on those matches in the note view panel, and enter an assessment of GOOD or BAD.

3.3 Metrics

In this evaluation, Precision and Recall were computed. Precision is the number of true positives found divided by the total number of system assertions (i.e. true positives plus false positives), and thus is equivalent to Positive Predictive Value. Recall is the number of true positives found divided by the number of positives "in truth" (i.e. true positives plus false negatives), and thus is equivalent to Sensitivity.

For Precision, assessors were asked to review the output from SemanticFind for the three search types being evaluated and mark each matched instance as GOOD or BAD using the context available in the full clinical note. If a search term was in a positive context but showed up as a Contradicted match, or was in a negative context and showed up as a Semantic/More Specific match, it was marked as BAD.

At the end of the judgment session, the total number of GOOD judgments gives us the true positive (TP) count, the total number of BAD judgments the false positive (FP) count.

For Recall, the assessors were asked to come up with as many clinically-equivalent paraphrases as they could (including none) for each of the search terms, as described earlier. SemanticFind processed these expansions and counted how often the GOOD matches in the Precision experiment failed to match instances of the paraphrases in the EHRs, which were automatically located via Literal Match – these are false negatives (FN). Since Precision = TP/(TP+FP) and Recall = TP/(TP + FN), having the counts for Precision plus FN allows us to compute Recall.

Table 3. *Sample of search terms with (possibly empty) lists of assessor-generated paraphrases.*

Search Term	Paraphrases
EGD	esophagogastroduodenoscopy, upper endoscopy, panendoscopy, OGD, oesophagogastroduodenoscopy, upper GI endoscopy, upper gastrointestinal endoscopy
GERD	gastroesophageal reflux disease, gastric reflux
HTN	hypertension, high blood pressure
constipation	not pooping, backed up, dyschezia, costiveness, no bowel movements, infrequent bowel movements
diabetes	diabetes mellitus, DM, high blood sugar
diverticulitis	

4. Results

A total of 13579 matches over 10 EHRs were found by SemanticFind and evaluated by two assessors, with an overall Precision of 0.87 over the entire dataset, as shown in Table 4.

We report two values for Recall as calculated using the search term paraphrases generated by the assessors. This is to show the difference between including and excluding search term paraphrases that are not valid UMLS concepts (either UMLS variants or variants found by our additional processing). When used interactively, SemanticFind tells the user if the input term is not a known concept, and gives the user an opportunity to rephrase. When run in this batch evaluation mode, this interaction was absent. Table 5 shows the evaluation results and Recall values. The constrained numbers represent Recall when the unknown concepts are dropped, while the unconstrained numbers represent Recall when considering all paraphrases provided including those that are not valid UMLS concepts.

Finally, we analyzed how many additional GOOD matches, beyond those found by Literal Match, were achieved by Semantic Match, More Specific and Contradicted Match. The Semantic Matches include all Literal Matches that are recognized concepts; matches associated with Semantic Match, More Specific and Contradicted Match are exclusive (i.e. a given text string will be matched by at most one of those). The results are presented in Table 6.

Table 4. *Column 2 shows the overall precision of the system across all test conditions.*

Precision Batch	Overall
True Positives	11851
False Positives	1728
#matches judged	13579
Precision	0.87

Table 5. *Recall evaluated when evaluators were free to choose any expansion terms (Unconstrained), or were constrained to use recognized concepts (in UMLS).*

Recall Mode	Unconstrained	Constrained
True Positives	11851	11851
False Negatives	1704	297
Recall	0.87	0.98

5. Discussion

Table 6 shows, in an incremental fashion, how Semantic Match, More Specific and Contradicted augment those found by Literal Match, which represents traditional Control/F matching. We see that all three judged matches together found twice as many matches as the baseline Literal Match. On the assumption that literal matches are by definition correct[ii], we conclude that in our data, half of desired matches are being missed by performing traditional search alone. A stronger baseline would be the Semantic Match result, corresponding roughly to a system that did traditional search with synonym expansion. Relative to that, the combined total of good matches is 68% greater. Either way, we are in good agreement with the work of Koopman et al. [9], who (despite theirs being a document retrieval task) also showed that addressing the semantic gap uncovered many more match results.

[ii] This is not technically true if we consider a match of a homograph (a different word spelled the same way) to be incorrect.

Table 6. *Analysis of extra GOOD matches by replacing Literal Match (LM) with, cumulatively, Semantic Match (SM), More Specific (MS) and Contradicted Match.*

Search types considered	**% GOOD matches as compared to Literal Match alone**
Literal Match alone	100%
Semantic Match, relative to LM	121%
SM + More Specific, relative to LM	190%
SM + MS + Contradicted Match, relative to LM	203%

5.1 Precision and recall

Overall, as compared to the traditional literal match (Precision up to 1.0 and Recall 0.49), SemanticFind has a lower precision but higher recall. By informing the user on entering a search term whether the input term is a known concept and allowing the user the opportunity to rephrase, search terms can be constrained to be recognized concepts, resulting in an 11-point improvement in Recall from 0.87 to 0.98. Error analysis showed that the errors are spread amongst a variety of causes, mostly tokenization errors and user spelling errors. Precision is also high (0.87) with a substantial percentage of the errors being due to a sentence-end detection problem, which we now explain.

We discovered that the somewhat informal formatting of manual note writing had been causing the system to concatenate consecutive lines as a single sentence when there was no terminal period in the earlier one, causing many instances of incorrect concept detection, specifically incorrect negation analysis. For example, the following text

```
alcohol use : no
smoking : yes
```

would give rise to recognition of "no smoking", which would generate the wrong polarity match when the search term was "smoking". From sampling about half of the error cases, we estimate that 30% of the system's false positives (precision errors) were due to this problem.

5.2 Applications in clinical practice and research

In clinical practice, the different types of search performed by SemanticFind may be used to address various information needs that arise at the point-of-care. Depending on the question the user has in mind, one or more types of search performed may be of interest. This highlights another advantage of SemanticFind, the ability to help the user locate relevant information without requiring construction of a complex query. It does so by taking any search term and returning clinically meaningful matches on multiple dimensions. So for a patient who comes in with a vague description of his or her medical history but does not know specific details of it, for example a patient with "heart disease", issuing "heart disease" on SemanticFind will match general references to cardiac disorders as well as specific types of cardiac problems such as cardiomyopathies and arrhythmias, thereby helping the user determine what specific type of heart problem the patient has. The same search will also return matches on related cardiac medications, tests, and procedures in both structured and unstructured data, providing further information on past and current management of the patient's heart problem.

5.4 Limitations

This study has several limitations. First, our medical records all originate from a single institution, Cleveland Clinic. Although health care professionals in general have a common shared vocabulary and terminology, it is likely that health care practitioners within a specific institution may have a shared bias for preferred terminology or syntax for clinical documentation. However, our system was built based on UMLS concept ontology and not tuned specifically to medical records from a specific institution, and therefore should perform just as well on medical records from any other institution. Second, the search terms used to evaluate the system were generated in an artificial setting, although efforts were made to simulate what a user at a point-of-care setting would want to search for.

6. Conclusion

For a number of reasons, there is considerable "dark matter" in an EHR when it comes to text search. Terms that the user is interested in finding can be expressed as synonyms, paraphrases, more specific or more general variants, logical equivalents and even in terms of their opposites – all of these can hinder the process of locating them when traditional Find or Control/F searches are used. In this paper we describe and evaluate SemanticFind, an application that supports the location of medical concepts in both the clinical notes and structured areas of a health record by employing various types of search. An evaluation of three specific search types, Semantic Match, More Specific and Contradicted, showed that SemanticFind performed with very high precision and recall, and, in our data uncovered twice as many potentially useful matches in the clinical notes as would be found by traditional search alone.

With these results, a user can be confident that when using SemanticFind, very few desired matches will be missed, and very few false matches will be found. Considering the variability in clinical language and abundance of content within EHRs, we believe that SemanticFind can be a great boon to productivity for any tasks requiring EHR search, whether that is in a point-of-care clinical setting or in retrospective chart reviews for research purposes.

References

[1] P. Tang, D. Fafchamps and E. Shortliffe, "Traditional medical records as a source of clinical data in the outpatient setting," in *Annual Symposium on Computer Application in Medical Care*, 1994.

[2] J. W. Ely, J. A. Osheroff, M. L. Chambliss, M. H. Ebell and M. E. Rosenbaum, "Answering physicians' clinical questions: obstacles and potential solutions," *Journal of the American Medical Informatics Association : JAMIA,* vol. 12, no. 2, pp. 217-224, 2005.

[3] K. Davies and J. Harrison, "The information-seeking behaviour of doctors: a review of the evidence," *Health Informatics Library Journal,* vol. 24, no. 2, pp. 78-94, 2007.

[4] A. Tawfik, K. Kochendorfer, D. Saparova, S. Ak Ghenaimi and J. Moore, "'I don't have time to dig back through this': the role of semantic search in supporting physician information seeking in an electronic health record," *Performance Improvement Quarterly,* vol. 26, pp. 75-91, 2014.

[5] S. Schulz, P. Daumke, P. Fischer and M. Muller, "Evaluation of a document search engine in a clinical department system," in *AMIA Annual Symposium,* 2008.

[6] L. Seyfried, D. Hanauer, D. Nease, R. Albeiruti, J. Kavanagh and H. Kales, "Enhanced identification of eligibility for depression research using an electronic medical record search engine.," *International Journal of Medical Informatics,* vol. 78, no. 12, 2009.

[7] D. Hanauer, "EMERSE: the electronic medical record search engine," in *AMIA Annual Symposium*, 2006.

[8] D. Hanauer, Q. Mei, J. Law, R. Khanna and K. Zheng , "Supporting information retrieval from electronic health records: A report of University of Michigan's nine-year experience in developing and using the Electronic Medical Record Search Engine (EMERSE)," *Journal of Biomedical Informatics,* vol. 55, pp. 290-300, 2015.

[9] B. Koopman, G. Zuccon, P. Bruza, L. Sitbon and M. Lawley, "Information Retrieval as Semantic Interface: A Graph Inference Model applied to Medical Search," *Information retrieval Journal,* vol. 19, no. 1, pp. 6-37, 2016.

[10] M. Devarakonda, D. Zhang, T. Ching-Huei and M. Bornea, "Problem-Oriented Patient Record Summary: An Early Report on a Watson Application," in *IEEE HealthCom*, Natal, Brazil, 2014.

[11] T. Edinger, A. Cohen , S. Bedrick, K. Ambert and W. Hersh, "Barriers to retrieving patient information from electronic health record data: failure analysis from the TREC medical records track," in *AMIA Annual Symposium*, 2012.

[12] US National Library of Medicine, "UMLS Reference Manual," National Library of Medicine (US), September 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK9675/. [Accessed 15 04 2014].

[13] D. Ferrucci and A. Lally, "Building an example application with the unstructured information management architecture," *IBM Systems Journal,* vol. 43, no. 3, pp. 455-475, 2004.

[14] Apache, "Apache UIMA," 2016. [Online]. Available: http://uima.apache.org/. [Accessed August 2016].

[15] A. R. Aronson and F.-M. Lang, "An overview of MetaMap: historical perspective and recent advances. JAMIA 2010 17: 229-236," *Journal of AMIA,* vol. 17, pp. 229-236, 2010.

[16] G. Miller, "WordNet: A Lexical Database for English," *Communications of ACM,* vol. 38, no. 11, pp. 39-41, 1995.

[17] W. Chapman, W. Bridewell, P. Hanbury, G. Cooper and B. Buchanan, "A simple algorithm for identifying negated findings and diseases in discharge summaries," *Journal of Biomedical Informatics,* vol. 34, no. 5, pp. 301-310, 2001.

[18] S. Deerwester, D. T. Susan, G. W. Furnas, T. K. Landauer and R. Harshman, "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science,* vol. 41, no. 6, pp. 391-407, September 1990.

[19] S. Simmons and Z. Estes, "Using latent semantic analysis to estimate similarity," Hillsdale, NJ, 2006.

[20] C. Manning , P. Raghavan and H. Schutz, Introduction to Information Retrieval, Cambridge University Press, 2008.

[21] M. Devarakonda and C.-H. Tsou, "Automated Problem List Generation from Electronic Medical Records in IBM Watson," in *Proceedings of the Twenty-Seventh Conference on Innovative Applications of Artificial Intelligence*, Autin, TX, 2015.