Evaluation of a generalizable approach to clinical information retrieval using the automated retrieval console (ARC)

Leonard W D'Avolio,[1,2,3] Thien M Nguyen,[1] Wildon R Farwell,[1,3,4] Yongming Chen,[1] Felicia Fitzmeyer,[1] Owen M Harris,[1] Louis D Fiore[1,5,6]

[1]Massachusetts Veterans Epidemiology Research and Information Center (MAVERIC) Cooperative Studies Coordinating Center, VA Boston Healthcare System, Jamaica Plain, Massachusetts, USA
[2]Center for Surgery and Public Health, Brigham and Women's Hospital, Boston, Massachusetts, USA
[3]Division of Ageing, Department of Medicine, Brigham and Women's Hospital and Harvard Medical School, Boston, Massachusetts, USA
[4]Department of Medicine, VA Boston Healthcare System, Boston, Massachusetts, USA
[5]Boston University School of Public Health, Boston, Massachusetts, USA
[6]Boston University School of Medicine, Boston, Massachusetts, USA

Correspondence to
Dr Leonard W D'Avolio, 150 S Huntington Ave, MAVERIC (151 MAV), VA Boston Healthcare System, Jamaica Plain, MA 02130, USA;
leonard.davolio@va.gov

Received 16 October 2009
Accepted 3 May 2010

ABSTRACT

Reducing custom software development effort is an important goal in information retrieval (IR). This study evaluated a generalizable approach involving with no custom software or rules development. The study used documents "consistent with cancer" to evaluate system performance in the domains of colorectal (CRC), prostate (PC), and lung (LC) cancer. Using an end-user-supplied reference set, the automated retrieval console (ARC) iteratively calculated performance of combinations of natural language processing-derived features and supervised classification algorithms. Training and testing involved 10-fold cross-validation for three sets of 500 documents each. Performance metrics included recall, precision, and F-measure. Annotation time for five physicians was also measured. Top performing algorithms had recall, precision, and F-measure values as follows: for CRC, 0.90, 0.92, and 0.89, respectively; for PC, 0.97, 0.95, and 0.94; and for LC, 0.76, 0.80, and 0.75. In all but one case, conditional random fields outperformed maximum entropy-based classifiers. Algorithms had good performance without custom code or rules development, but performance varied by specific application.

INTRODUCTION

Electronic medical record (EMR) data are becoming increasingly important for quality improvement,[1] comparative effectiveness research,[2] evidence-based medicine,[3] and establishing robust phenotypes for genomic analysis.[4] Unfortunately, most EMR implementations were designed to facilitate one-on-one interactions, not to support analysis of aggregated data as required by many secondary uses.[5 6] As a result, efforts to 'repurpose' clinical data must contend with few widely implemented data standards and large amounts of potentially useful information stored as unstructured free text.

Researchers have responded with the development and application of natural language processing (NLP), information extraction, and machine-learning algorithms—referred to here collectively as information retrieval (IR) technologies. Despite over 20 years of empirical demonstrations of capable IR performance, the complex nature of the challenge and technical barriers to entry have hindered widespread adoption and translation of clinical IR technologies. The Massachusetts Veterans Epidemiology Research and Information Center (MAVERIC) is addressing this challenge by attempting to deliver the benefits of IR technologies to non-technical end users. The automated retrieval console (ARC) is software designed to facilitate clinical IR translation by providing interfaces and workflows to automate many of the processes of clinical IR.

One process in particular that may be the most substantial barrier to adoption is the current reliance on custom software and rules or heuristic development for each individual application. In this study, we evaluate algorithms incorporated in ARC that were designed to be capable of achieving acceptable levels of performance without custom software development. We hypothesize that success in this regard will improve accessibility of IR technologies to non-technical users and afford system developers more time to focus on advancing the science and technologies of IR, rather than having to provide software as a service.

BACKGROUND

The application motivating this study is the retrieval of relevant documents from EMR systems. The identification of relevant documents is a prerequisite to most secondary data uses, such as automated quality measurement, medical record-based research, cohort identification, and comparative effectiveness research. Unfortunately, queries of structured data fields such as ICD-9 codes and Current Procedural Terminology (CPT) codes for secondary data use have proven less than ideal. The questionable quality of administrative code assignments has been documented extensively since the rise of administrative code-based reimbursement,[7–10] and custom case-finding algorithms can be time consuming to develop and must be evaluated for each application. A solution to this dilemma may be provided by clinical IR technologies.

In the past two decades, clinical IR has evolved from a field with few researchers working on even fewer systems[11–13] to the release of open-source components and libraries.[14–16] More recently, researchers in the fields of computer science and linguistics have released open-source software frameworks upon which IR methods can be developed.[17 18] Clinical IR researchers have capitalized on these frameworks, producing modular pipelines for specific retrieval applications.[18 19] One such pipeline for clinical NLP is the Clinical Text Analysis and Knowledge Extraction System (cTAKES).[20] The cTAKES maps free text to SNOMED concepts and is based on the open-source Unstructured Information Management Architecture (UIMA).[17]

Many approaches to clinical IR use open-source implementations of machine-learning classifiers to achieve high levels of performance.[21 22] Two supervised machine-learning classifiers used in this study are maximum entropy (MaxEnt) and conditional random fields (CRFs). MaxEnt is a framework for estimating probability distributions from a set of training data.[23] Maximum entropy models have been used in NLP to chunk phrases,[24] for part-of-speech tagging,[25] and in a number of biomedical applications.[26–28] A CRF is an undirected graphical model with edges representing dependencies between variables.[29] Peng and McCallum[30] showed that CRFs outperform the more commonly used support vector machines in extracting common fields from the headers and citations of literature. Wellner *et al*[21] showed the ability of CRFs to achieve high levels of performance in the deidentification of personal health identifiers, limiting customization to manual annotation of training sets.

Automated IR approaches have proven capable of high levels of performance across a number of applications, as evidenced by the results of 10 years of the Message Understanding Conferences (MUCs),[31] more than 15 years of the Text REtrieval Conference (TREC),[32] and in the clinical domain, three i2b2 'shared task' challenges.[33–35] Despite empirical evidence of its potential, widespread adoption of clinical IR remains elusive. A small number of systems have proven capable of migrating beyond empirical evaluation to actual implementation. Fewer have been adopted beyond the home institution of their developers,[36–40] and we know of no clinical IR systems that can be applied for different retrieval applications without custom software or rules development.

METHODS

Design of ARC

Current use of clinical IR technologies is heavily dependent on the system developer. With ARC, we are attempting to either automate or shift to the end user as many of the processes of clinical IR as possible. Figure 1 shows the current processes of clinical IR versus the proposed shift in responsibilities we are attempting to achieve with ARC.

The ARC design is based on the hypothesis that supervised machine learning with robust enough feature sets is capable of delivering acceptable performance across a number of clinical IR applications. This approach allows us to reduce end-user input to a reference set that can be used as both the training and test sets for any one application. Proceeding with this hypothesis, the challenge becomes how best to enable the end user to perform the remaining processes of clinical IR use, including annotation, training versus test set partitioning, performance calculation, storage of models and results, and deployment on the larger corpus.

Toward this end, ARC features several interfaces to enable greater end-user control over the processes of clinical IR. The ARC menu from which each of the interfaces is launched is shown in figure 2.

The 'Create New Project' interface is used to establish a workspace and import samples. This workspace is used to save the state of any project including models and performance results across the various interfaces. Annotation can be a bottleneck in applying IR technologies. The 'Judge' interface shown in figure 3 was therefore designed to be simple and fast, featuring one click and shortcut key labeling ('Y', 'N') and document advancement (left arrow, right arrow). The reference set created in the Judge interface is saved to the workspace and used for model creation and performance calculations. The 'Kappa' interface supports the calculation inter-rater reliability by presenting totals of agreement among judges that can be exported to statistical packages. The 'Feature Blast' interface iteratively calculates the performance (ie, recall, precision, F-measure) of different combinations of feature types and

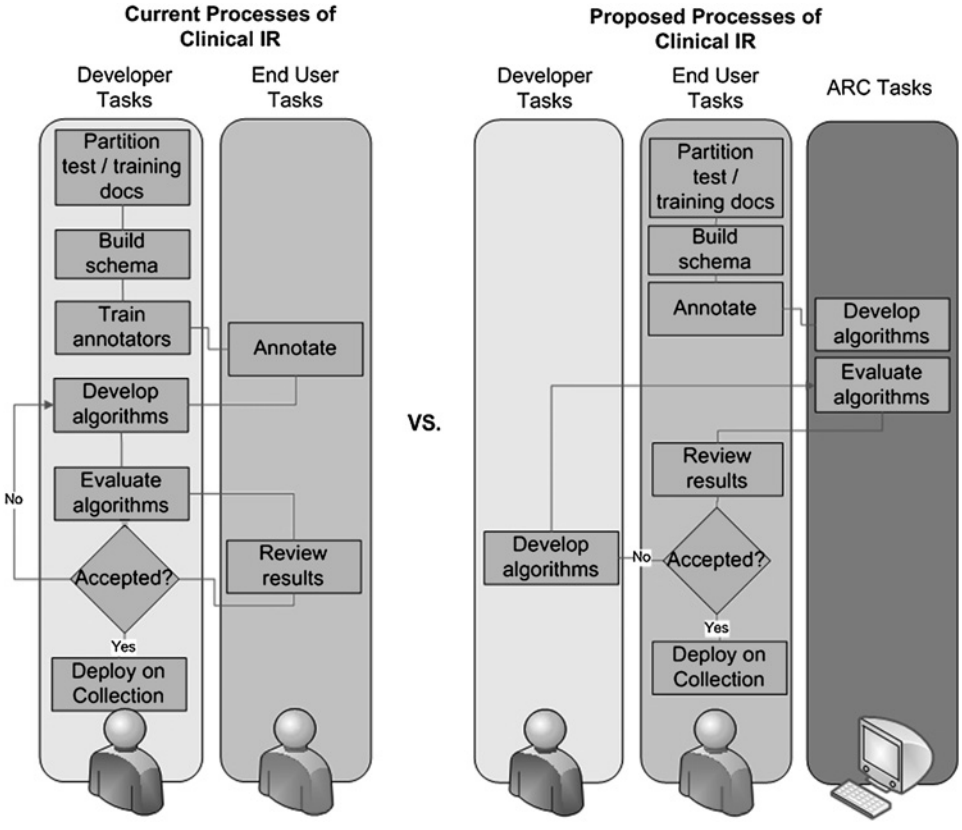

Figure 1 Current processes of clinical information retrieval (IR) versus those proposed in the design of the automated retrieval console (ARC).

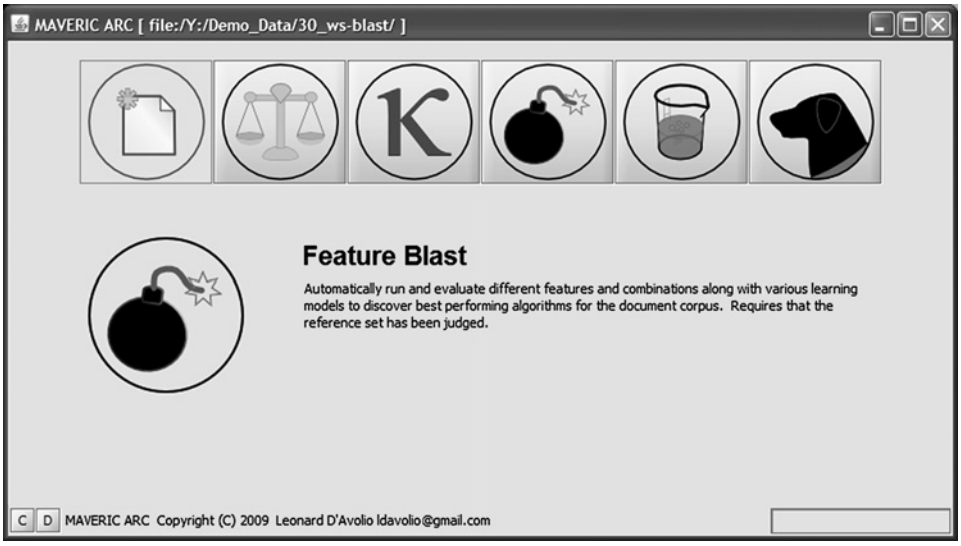

Figure 2 Automated retrieval console (ARC) menu, showing the various ARC interfaces.

classifiers to determine appropriate combinations for a given application. The 'Laboratory' interface enables developers to explore and evaluate different approaches to IR. Developers can use the Laboratory interface to select which feature types and models to experiment with, tracking the performance of each combination. The 'Retrieve' interface shows the performance of all models created as part of a project and facilitates deployment of saved models on larger collections.

The ARC was used to manage all of the processes involved in this study from sample creation to algorithm evaluation. It was developed in Java and is available as open-source software at http://research.maveric.org/mig/arc.html. Users can download ARC or, thanks to the generous cooperation of the National Library of Medicine and Dr Guergana Savova, users can download a 'full' version of ARC with cTAKES and its UMLS-based knowledge base installed. The site also features html and video tutorials designed to use a small collection of simulated radiology reports.

Approach

The focus of this study was the evaluation of the algorithms used within the Feature Blast interface to retrieve relevant documents across a number of different applications with no custom software development. Building on the collection of currently available open-source clinical IR software, ARC combines open-source NLP pipelines with machine learning.

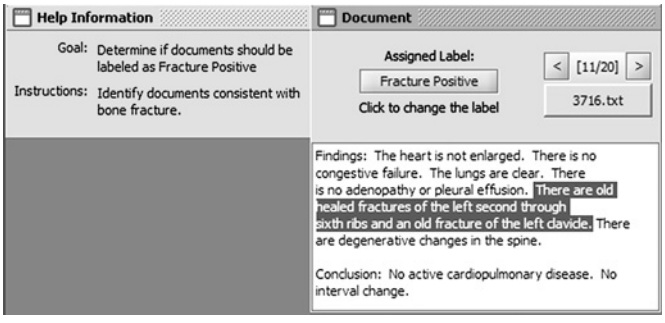

Figure 3 A screen shot of the Judge interface. The annotation instructions shown in the 'Help Information' window is populated as part of the creation of a new project.

The ARC uses UIMA-based pipelines for NLP. The UIMA pipelines can be launched to process text from within ARC, or complete UIMA project files can be loaded into ARC. Each pipeline created in UIMA has an XML-formatted configuration file that describes the structured output the pipeline produces. The ARC reads the XML configuration file and exposes NLP-structured output as feature types for machine learning classification. As a result, any UIMA-based pipeline can be used by ARC. However, the goal of this study is to design and evaluate the ability of our approach to perform well across different applications with no custom code or rules development. We therefore chose cTAKES, a general concept-mapping clinical pipeline.[20] The transforms performed on clinical data using cTAKES result in more than 90 different types of structured output (eg, noun phrases, tokens, sentences, SNOMED codes).

The version of cTAKES available for this study uses a section boundary detector that is based on the HL7 Clinical Document Architecture (CDA), which is not widely implemented by the VA Healthcare System. Therefore one minor modification made to cTAKES was the removal of the CDA-based section boundary detector and the addition of a regular expression-based section boundary detector. The ability to make such modifications easily is a function of the modular design of open-source NLP frameworks such as UIMA and GATE. An abbreviated list of some of the structured results produced by cTAKES is provided in table 1.

For supervised machine learning, ARC integrates the open-source Application Programming Initiative (API) exposed by the MAchine Learning for Language Toolkit (MALLET).[41] In this study, two particular classifiers from MALLET are used: a MaxEnt classifier and a classifier based on CRFs.

The ability of ARC to reduce developer involvement in the clinical IR process is predicated on the capacity of the system to 'learn' effective approaches to solving a given IR problem. After a user provides ARC with a reference set, ARC's Feature Blast algorithm uses the following steps to identify which types of NLP output and machine-learning classifiers to combine for a given application. Firstly, it processes text documents with the cTAKES NLP pipeline, exposing more than 90 NLP-derived feature types (eg, noun phrases, tokens, SNOMED concepts) for supervised classification. Using 10-fold cross-validation, the system partitions both the training and test sets and calculates the performance of each individual

Table 1 Abbreviated list of cTAKES structured output

UMLS concept (CUI)	Sentence number	Measurement annotation
Named entity	Roman numeral annotation	UMLS concept semantic type (TUI)
Word token	New line token	Negated named entities
Canonical form of word token	Range expression	Negated CUIs
Noun phrase	Location expression	Number token
Verb phrase	Time annotation	Person/title annotation
Adjective phrase	Fraction annotation	Size expression
Adverb phrase	Symbol token	Unit expression

NLP-produced feature type using all available machine-learning classifiers. The performance of each of the individual feature types and classifier combinations is stored to the workspace.

The optimal combination of feature types and classification algorithms could be determined by calculating all possible variations. However, with greater than 90 different feature types and two classifiers, the cost in time would be prohibitive. Instead, we explored the performance of two different algorithms designed to identify favorable combinations more efficiently. The two algorithms used to determine those combinations are described below.

1. Algorithm 1: top scoring combinations

The first algorithm used by Feature Blast to determine optimal combinations evaluates all combinations of the five top scoring feature types or classes (eg, noun phrases, concepts) using either selected or all available classification algorithms. Algorithm 1 reduced the process to a manageable 52 iterations (26 combinations of feature types multiplied by two classifiers). The five top scoring feature types are defined as:

Configuration	Feature type combinations
1	Highest F-measure
2	2nd highest F-measure
3	3rd highest F-measure
4	Highest recall not already included
5	Highest precision not already included

2. Algorithm 2: top score + negation

A limitation of the first algorithm is its exclusion of feature types that score poorly as the only feature types in consideration but may add value as part of a combination of feature types. The one feature type that most obviously falls into this category is negated concepts or phrases. For example, in classifying imaging reports consistent with cancer, evidence of negated concepts (eg, 'no evidence of cancer') may add value. The cTAKES assigns negation to both named entities and UMLS concept unique identifiers (CUIs). A named entity is an atomic element or 'thing' found in the text, usually mapped from a noun phrase (eg, 'heart attack'). Several different named entities can mean the same thing (eg, heart attack, myocardial infarction, MI), and therefore named entities are often mapped to unique concepts such as UMLS CUIs (eg, heart attack = CUI C0027051). The ARC supports the conversion of negated entities and concepts to features by allowing the user to specify a prefix or suffix to any feature type through the user interface. For example, by adding the prefix 'neg' to all negated named entities (eg, 'cancer'), ARC will pass 'neg-cancer' as a feature to the classifier. In each case, we chose the highest scoring configuration of negation, selecting either the negated named entity or the negated CUI based on the highest F-measure.

Our second algorithm, combining top scoring feature types and negation is defined as:

Configuration	Feature type combinations
1–5	Algorithm 1 combinations
6	Highest recall + highest precision
7	Highest recall + negated text
8	Highest precision + negated text
9	Highest recall + highest precision + negated text

Data collection and sampling

In this study, we evaluate the ability of ARC to retrieve relevant documents from the collection of relevant and irrelevant documents returned from ICD-9 code-based queries. To test the ability of our approach to generalize across different applications, three samples and targets for retrieval were used: (1) imaging reports consistent with lung cancer; (2) pathology reports consistent with colorectal cancer (CRC); (3) pathology reports consistent with prostate cancer. For each sample, 500 documents were chosen at random from documents created between 1997 and 2007 at hospitals within the New England Veterans Integrated Service Network (VISN 1). Our original case finding queries for identifying the collections from which samples were selected were as follows.

For CRC:

- Select all pathology reports within 60 days before and 60 days after the first appearance of ICD-9 codes 153.x, 154.x.

For prostate cancer:

- Select all pathology reports within 60 days before and 60 days after the first appearance of ICD-9 codes 185.x.

For lung cancer:

- Select all imaging reports within 60 days before and 60 days after the first appearance of ICD-9 codes 162.x.

We considered only the first appearance of a targeted ICD-9 code, regardless of assignment position (primary code, secondary code, etc). These samples were used to create 'gold standard' reference sets for both training and testing the algorithms.

Creation of reference sets

For each of the three samples, two physician judges assigned values of 'relevant' or 'irrelevant' to each of the 500 documents. A third physician judge served as final adjudicator for any disagreements. A total of five physicians participated in the creation of the three reference sets. Reviewers were instructed to base their assessment of relevance on whether each document was 'consistent with a diagnosis of cancer.' They were instructed to ignore any clinical history and instead focus on the immediate report of the pathologist or radiologist. In-situ cancers in the colon or rectum were counted as CRC, and prostate intraepithelial neoplasia was counted as prostate cancer. For CRC and prostate cancer, even if the subject of the report was tissue outside of the organ of interest, if the pathologist recorded CRC or prostate cancer, the reviewers were instructed to classify the document as consistent with the particular cancer of interest.

Whereas the pathology report is the primary document for recording a diagnosis of prostate cancer and CRC, imaging reports are less likely to contain conclusive evidence of a lung cancer diagnosis. Instead, lung cancer diagnoses may be determined by a combination of imaging studies, biopsies, and/or laboratory results. Despite the potential inconclusiveness of imaging reports for lung cancer, they are considered important documents for finding lung cancer cases and monitoring cancer progression.

They also provide the opportunity to test the performance of our approach on a sample of documents with less structure and with less agreement between judges. The imaging reports in this study were generated from a number of study modalities including x-rays, CT scans, and MRI.

Study design

In order to evaluate the effectiveness of the proposed approach, we captured the performance of individual feature types and both classifiers for all three samples as well as the performance of algorithms 1 and 2 using both classifiers. In all experiments, performance was measured in terms of recall, precision, and F-measure using 10-fold cross-validation. The performance of the NLP system has a direct effect on the quality of the features produced for classification. However, the focus of this study does not include a specific evaluation of cTAKES' performance on the samples used. Figure 4 illustrates the design of the study.

RESULTS

The percentage of documents in the samples found to be consistent with CRC, prostate cancer, and lung cancer by the judges was 16.6%, 18.8%, and 28.6%, respectively. Reference set creation and distribution information, including annotation time, kappa scores, self-reported time to annotate 500 documents, and the number of documents adjudicated by a third judge is provided in table 2.

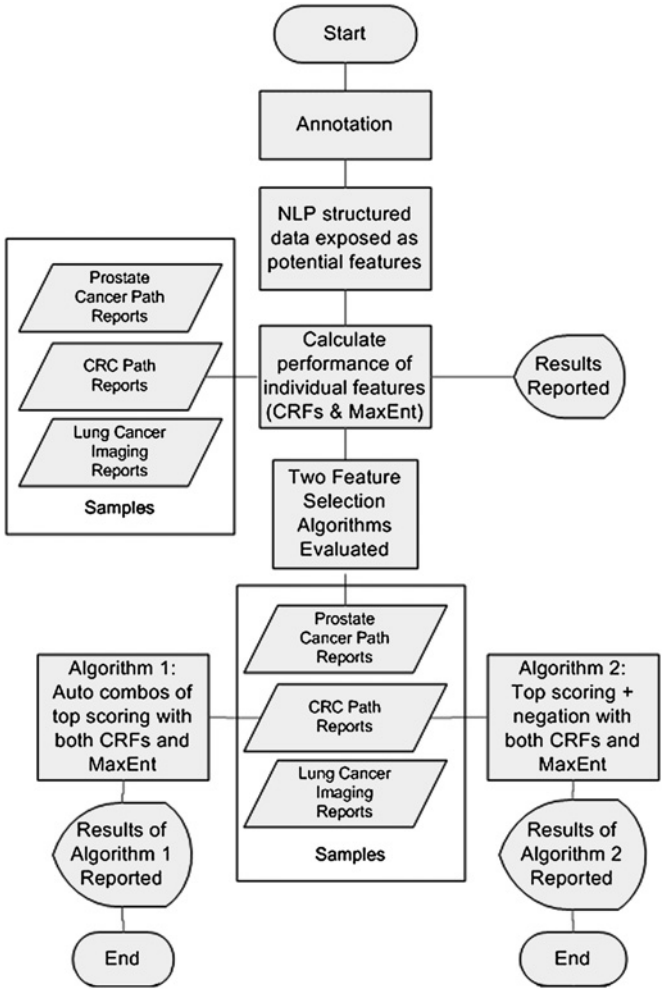

Figure 4 A graphical representation of the study design. CRC, colorectal cancer; CRF, conditional random field; MaxEnt, maximum entropy; NLP, natural language processing.

The top recall, precision, and F-measure for each sample and the classifier/feature type combinations with which they were achieved are shown in table 3.

A total of 98 different types of structured output were produced by cTAKES. In all cases except in the precision of lung cancer document retrieval, CRFs outperformed MaxEnt. In most cases, the canonical form of word tokens, named entities, and CUIs were among the top scoring feature types. The top scoring feature types varied depending on the application and, in some cases, the classification algorithm used. For example, using MaxEnt to classify prostate cancer pathology reports, named entities were the top scoring feature type in recall, precision, and F-measure. However, named entities scored second in recall, fourth in precision, and fourth in F-measure for the same application using CRFs as a classifier. Certain feature types scored strongly in either recall or precision (eg, recall of CUIs for prostate cancer reports), suggesting that their inclusion in a model may be advantageous, depending on the clinical use-case.

Algorithm 1, which combined top scoring feature types (eg, CUI + noun phrases), matched or outperformed classification attempts using individual feature types (eg, CUIs) in all cases but one. For example, algorithm 1 achieved an improvement in F-measure of approximately three points compared with the top scoring individual feature in CRC classification (0.89 vs 0.86). The exception was the recall performance of the individual feature CUIs in classifying CRC (0.90 vs 0.88). Algorithm 1 also promoted negated named entities into consideration, resulting in the top precision score for all attempts at lung cancer document identification.

The addition of negation in algorithm 2 had an adverse effect on performance in some cases. For example, the recall of CRC reports using CRFs experienced a greater than two point drop when CUI was combined with negated CUIs. A three point drop in F-measure was experienced with the addition of negated CUIs to CUIs for the same CRC sample using MaxEnt (0.85 to 0.82). The few gains realized from the addition of negation were minimal.

DISCUSSION

Overall performance in context

The assessment of what is considered acceptable performance is dependent on the intended secondary use of the data. That said, we see promise in the ability to create, evaluate, and deploy clinical IR across different applications at the performance levels achieved in a matter of hours rather than days or weeks. For the retrieval of CRC and prostate cancer reports consistent with cancer, the proposed approach was able to identify cases with F-measures of greater than 0.88 and 0.93, despite a collection with relatively few true positives to train on (83 for CRC; 94 for prostate). Classification of radiology reports consistent with lung cancer proved to be more challenging to both our algorithms and our physician judges, as indicated by the inter-rater reliability among the physician judges (κ=0.73). An F-measure of 0.75 is not unexpected in light of the disagreement among the physician judges. A more appropriate approach to imaging report classification may be the inclusion of a third class to represent 'not enough information.'

The performance degradation resulting from the inclusion of negated named entities and CUIs may indicate that negation is not a valuable contribution to such classification applications. It may also be due to poor performance of the NegEx-based negation detector included in cTAKES. The version of cTAKES

Table 2 Reference sets for colorectal cancer (CRC), prostate cancer, and lung cancer samples

CRC (n=500)	Physician A	Physician B	Cohen's kappa	Physician D (adjudicator)	Final reference set
Positives/reviewed	83/500	89/500	0.92	3/12	83/500=16.6%
Estimated time to annotate (min)	90	90			
Prostate cancer (n=500)	Physician A	Physician C	Cohen's kappa	Physician D (adjudicator)	
Positives/reviewed	94/500	94/500	0.99	1/2	94/500=18.8%
Estimated time to annotate (min)	60	80			
Lung cancer (n=500)	Physician A	Physician D	Cohen's kappa	Physician E (adjudicator)	
Positives/reviewed	157/500	142/500	0.73	22/57	143/500=28.6%
Estimated time to annotate (min)	60	90			

used in this study uses an older version of NegEx, which has since been improved upon. A cursory review of cases did not indicate poor performance of the negation module. However, a thorough analysis of the performance of the individual feature types is an important topic for future investigation.

By focusing on streamlining processes through the development of generalizable algorithms, we do not anticipate best possible performance for all applications. Instead, we expect to sacrifice some performance that might otherwise be realized through code customization in exchange for the ability to move from one application to another with manual annotation as the only requisite input. While our focus in this study is the evaluation of solutions that require no custom code, ARC incorporates the structured output of NLP. Therefore the results of any custom code written for inclusion in a UIMA pipeline can be used as features for classification by ARC. For example, in a follow-up experiment, we incorporated a lymph node annotator component from IBM's open-source UIMA-based MedKAT pipeline[42] and realized an approximately 0.1 point improvement in recall, precision, and F-measure for classifying prostate cancer cases using MaxEnt.

The development of a one-click annotation interface helped keep annotation times for all five participating doctors between 60 and 90 min for 500 document samples. The reduction of annotation time from 90 to 60 min for the one physician annotating multiple samples indicates some benefit from familiarity. Total processing time per sample, including generating NLP-derived features and calculating iteration performance, was approximately 1.5 hours. All models created are serialized by ARC and can be deployed on other collections using the Retrieve interface. Maintaining short annotation times will be more challenging as we shift from document-level to concept-level IR.

Table 3 Top scoring combinations for each sample

Cancer		Classifier	Feature(s)
Colorectal cancer			
Recall	0.90	CRFs	Single feature (CUI)
Precision	0.92	CRFs	Algorithm 1 (token+named entity)
F-measure	0.89	CRFs	Algorithm 1 (token+named entity)
Prostate cancer			
Recall	0.97	CRFs	Single feature (CUI) Algorithm 1 (CUI+date) Algorithm 2 (CUI+neg. CUI)
Precision	0.95	CRFs	Single feature (token) Algorithm 1 (token+date) Algorithm 2 (token+neg. CUI)
F-measure	0.94	CRFs	Single feature (token) Algorithm 1 (token+date) Algorithm 2 (token+neg. CUI)
Lung cancer			
Recall	0.76	CRFs	Single feature (canonical) Algorithm 1 (canonical+measurement)
Precision	0.80	MaxEnt	Algorithm 1 (canonical+token+negated named entity)
F-measure	0.75	CRFs	Algorithm 1 (canonical+measurement)

CRF, conditional random field; MaxEnt, maximum entropy; CUI, concept unique identifier; neg. CUI, negated concept unique identifier; canonical, canonical form of a word.

No single 'best feature' or 'best model' for clinical IR

There was a trend toward strong performance of individual feature types such as tokens or their canonical form. This reinforces the findings of Salton and others decades ago who showed the power of simple tokens as features for document retrieval.[43 44] However, the results also showed that different feature types, different feature type combinations, and different classification algorithms performed best depending on the application. Some unexpected feature types proved valuable for achieving top classification scores. For example, canonical form + punctuation or measurement annotation was an unexpected combination that scored the highest precision for lung cancer retrieval. Also unexpected was the one case in which MaxEnt outperformed CRFs after consistently performing several percentage points lower in most other applications. This variation occurred despite the similar nature of the application and, in the case of prostate cancer and CRC classification, similar document types. These findings imply that there is no optimal configuration for all clinical IR applications and offers support for our attempt to learn favorable combinations from multiple feature types and classifiers.

Benefits of open-source clinical IR

The approach to clinical IR explored in this study capitalizes on the efforts of those that have previously developed and released open-source IR software. As a result of packages such as MALLET, UIMA, and cTAKES, we were able to focus on improving the processes involved in clinical IR and produce an open-source product in the relatively short span of six months. We expect that ARC will continue to benefit from the model of open-source software development. As new NLP components, pipelines, or machine-learning classifiers are released, they can be easily incorporated, extending their advantages to ARC users. Similarly, we hope that others will find ways to improve the processes currently exposed by ARC.

Quality of administrative code assignment

The focus of this study is the evaluation of algorithms that we hypothesize can be used as part of an effort to streamline the processes of clinical IR to lower the cost of adopting this important technology. The questionable quality of ICD-9 code assignment and the challenges it presents to secondary data use provoked the choice of this particular clinical IR use-case. While this study was not designed to answer questions pertaining to the quality of ICD-9 code assignment, we did not expect true positive rates of only 17–29% based on the case-finding technique used.

Concerned that we had an error in our ICD-9 code-based case-finding algorithm, we conducted reviews of 30 randomly selected false positives in each of the three samples for a total of 90 reports. The reviews showed that many of the false positives were reports related to the appropriate anatomy but without evidence of a cancer of interest (lung 43%, CRC 30%, prostate 1%). Dermatological analyses (skin lesions, biopsies, etc) comprised 30% of the total false-positive pathology reports. In many cases, the reports were focused on anatomy within close proximity of the anatomy of interest (eg, 23% of prostate assignments were for colorectal anatomy). In some false positives, the reports indicated a prior history of cancer. As a result, these numbers do not indicate that only 17–29% of the patients with the targeted ICD-9 codes ever had cancer. Instead, the numbers indicate that 17–29% of pathology or imaging reports appearing within 120 days of cancer-related ICD-9 code assignment were consistent with cancer. The low rates of true positives does emphasize the need for careful consideration of the quality of electronic medical data in light of the growing number of proposed secondary uses.

CONCLUSION

We theorize that greater adoption and translation of clinical IR can be achieved by reducing several of the dependencies of clinical IR on IR researchers and system developers. This study is a first step toward streamlining the processes of clinical IR in an effort to facilitate translation. In the process we achieved encouraging levels of performance with minimal time between applications and with no custom code or rules development. Our results show that the performance of various combinations of feature types and even classification algorithms is contingent on the application, supporting the potential of our approach.

There are limitations to this overall approach and the specific study conducted. Firstly, this study was an evaluation of technical feasibility, with performance measured in terms of recall, precision, and F-measure. Our goal of increased translation of clinical IR technology is not only dependent on performance in terms of system accuracy but also on usability. This study does not measure that critical aspect of system design. In addition, while document retrieval is an important prerequisite of most efforts at secondary data use, ARC will remain of limited utility until it is extended to perform concept-level IR (eg, retrieval of tumor stage from pathology reports).

Having explored the potential of an approach to document retrieval without custom code or rules development, we are in the process of extending ARC to address both concept-level and patient-level IR. This requires a rethinking of the document- and concept-oriented data structures and workflows of current IR to allow patient-level inference. A significant challenge will be providing such robust functionality while maintaining our emphasis on delivering the capabilities of IR to non-technical end users. Future work will also include the incorporation of alternative approaches for optimal feature type selection and the addition of other proven classifiers such as support vector machines.

Acknowledgments We thank Guergana Savova, PhD and James Masanz of the Mayo Clinic as well as David Mimno and Fernando Pereira, PhD of the University of Massachusetts for their assistance in incorporating the open-source tools cTAKES and MALLET. We thank Jan Willis and the National Library of Medicine for working with us to make the UMLS available with ARC. We would also like to acknowledge the dedicated staff of MAVERIC for their assistance in this project.

Funding This work was supported by VA Cooperative Studies Program as well as the Veterans Affairs Health Services Research and Development grant, Consortium for Health Informatics Research (CHIR), grant HIR 09-007. Other funders: VA Cooperative Studies Program; Veterans Affairs Health Services Research and Development; Consortium for Health Informatics Research. The views expressed here are those of the authors, and not necessarily those of the Department of Veterans Affairs.

Competing interests None.

Ethics approval This study was conducted with the approval of the VA Boston Healthcare System.

Provenance and peer review Not commissioned; externally peer reviewed.

REFERENCES

1. **U.S. Department of Health and Human Services.** About healthy people, 2009. http://www.healthypeople.gov/About/.
2. **Committee on Comparative Effectiveness Research Prioritization.** *Initial priorities for comparative effectiveness research*. Washington DC: Institute of Medicine, 2009.
3. **Bates D,** Kuperman G, Wang S, *et al*. Ten commandments for effective clinical decision support: Making the practice of evidence-based medicine a reality. *J Am Med Inform Assoc* 2003;**10**:523–30.
4. **Murphy S,** Churchill S, Bry L, *et al*. Instrumenting the health care enterprise for discovery research in the genomic era. *Genome Res* 2009;**19**:1675–81.
5. **Berg M,** Goorman E. The contextual nature of medical information. *Int J Med Inform* 1999;**56**:51–60.
6. **D'Avolio L.** Electronic medical records at a crossroads: impetus for change or missed opportunity. *J Am Med Assoc* 2009;**302**:1109–10.
7. **Burnum J.** The misinformation era: the fall of the medical record. *Ann Int Med* 1989;**110**:482–4.
8. **Musen M.** The strained quality of medical data. *Methods Inf Med* 1989;**28**:123–5.
9. **Peabody J,** Luck J, Jain S, *et al*. Assessing the accuracy of administrative data in health information systems. *Med Care* 2005;**42**:1066–72.
10. **Iezzoni L.** Assessing quality using administrative data. *Ann Int Med* 1997;**127**:666–74.
11. **Hersh W,** Hickam D. Information retrieval in medicine: The SAPHIRE experience. *J Am Soc Inf Sci* 1995;**46**:743–7.
12. **Friedman C,** Alderson P, Austin J, *et al*. A general natural language text processor for clinical radiology. *J Am Med Inform Assoc* 1994;**1**:161–74.
13. **Sager N,** Friedman C, Lyman M, *et al*. *Medical language processing: Computer management of narrative data*. Reading, MA: Addison-Wesley, 1987.
14. **Witten I,** Frank E. *Data mining: practical machine learning tools and techniques*. 2nd edn. San Francisco: Morgan Kaufmann, 2005.
15. **Marcus MP,** Marcinkiewicz MA, Santorini B. Building a large annotated corpus of English: the penn treebank. *Comput Linguist* 1993;**19**:313–30.
16. **Toutanova K,** Klein D, Manning C, *et al*. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. *Proceedings of HLT-NAACL* 2003:252–9.
17. **Ferrucci D,** Lally A. UIMA: an architectural approach to unstructured information processing in the corporate research environment. *NLE* 2004;**10**:327–48.
18. **Cunningham H.** GATE, a general architecure for text engineering. *Comput Hum* 2004;**36**:223–54.
19. **Zeng Q,** Goryachev S, Weiss S, *et al*. Extracting principle diagnosis, co-morbidity, and smoking status for asthma research: evaluation of a natural language processing system. *BMC Med Inform Decis Mak* 2006;**6**:30.
20. **Savova G,** Kipper-Schuler K, Buntrok J, *et al*. *UIMA-based clinical information extraction system*. Language Resources and Evaluation Conference, Morocco, 2008.
21. **Wellner B,** Huyck M, Mardis S, *et al*. Rapidly retargetable approaches to de-identification. *J Am Med Inform Assoc* 2007;**14**:564–73.
22. **Savova G,** Clark C, Zheng J, *et al*. The Mayo/MITRE system for discovery of obesity and its comorbidities. *Proceedings of the i2b2 Workshop in Challenges in Natural Language Processing for Clinical Data*; 2009.
23. **Harremoës P,** Topsøe F. Maximum entropy fundamentals. *Entropy* 2001;**3**:191–226.
24. **Koeling R.** Chunking with maximum entropy models. *Proceedings of CoNLL-2000 and LLL-2000*, 2000:139–41.
25. **Ratnaparkhi A.** A maximum entropy part-of-speech tagger. *Proceedings of the Empirical Methods in Natural Language Processing; 1996*; University of Pennsylvania, May 1996:133–42.
26. **Taira R,** Soderland S. A statistical natural language processing for medical reports. *Proceedings of the Annual Meeting of the American Medical Informatics Association*, Washington DC, 1999:970–4.
27. **Savova G,** Kipper-Schuler K, Buntrock J, *et al*. *UIMA-based clinical information extraction system. LREC 2008: Towards enhanced interoperability for large HLT systems: UIMA for NLP*, Marrakech, Morocco, 2008.
28. **Taira RK,** Soderland SG. A statistical natural language processor for medical reports. *Proc AMIA Symp* 1999:970–4.
29. **Lafferty J,** McCallum A, Pereira F. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data. International Conference on Machine Learning*, San Francisco: Morgan Kaufman, 2001:282–9.
30. **Peng F,** McCallum A. Information extraction from research papers using conditional random fields. *Inform Process Manag* 2006;**42**:963–79.
31. **National Institute of Standards and Technology.** Message Understanding Conference Proceedings MUC-7 Table of Contents, 2001. http://www.itl.nist.gov/

iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html (accessed 08 June 2010).
32. **National Institute of Standards and Technology.** Introduction to information extraction, 2005. http://www-nlpir.nist.gov/related_projects/muc/ (accessed 31 Aug 2009).
33. **Uzuner O,** Goldstein I, Luo Y, *et al*. Identifying patient smoking status from medical discharge records. *J Am Med Inform Assoc* 2008;**15**:14—24.
34. **Uzuner O.** Recognizing obesity and comorbidities in sparse data. *J Am Med Inform Assoc* 2009;**16**:561—70.
35. **Uzuner O,** Luo Y, Szolovits P. Evaluating the state-of-the-art in automatic de-identification. *J Am Med Inform Assoc* 2007;**14**:550—63.
36. **Friedman C,** Shagina L, Lussier Y, *et al*. Automated encoding of clinical documents based on natural language processing. *J Am Med Inform Assoc* 2004;**11**:392—402.
37. **Demner-Fushman D,** Seckman C, Fisher C, *et al*. A Prototype System to Support Evidence-based Practice. *Proceedings of the 2008 Annual Symposium of the American Medical Information Association;* Washington, DC, 2008.
38. **Dreyer K,** Mannudeep K, Hurier A, *et al*. Application of a recently developed algorithm for automatic classification of unstructured radiology reports: validation study. *Radiology* 2005;**234**:323—9.
39. **Denny JC,** Smithers JD, Miller RA, *et al*. Understanding medical school curriculum content using KnowledgeMap. *J Am Med Inform Assoc* 2003;**10**:351—62.
40. **Aronson A.** Effective mapping of biomedical text to the UMLS metathesaurus: the MetaMap program. *Proc AMIA Symp* 2001:17—21.
41. **McCallum A.** MALLET: a machine learning for language toolkit, 2002. http://mallet.cs.umass.edu/.
42. **IBM.** OHNLP documentation and downloads. caBIG Knowledge center, 2009. https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/OHNLP_Documentation_and_Downloads.
43. **Salton G.** *The SMART retrieval system*. Englewood Cliffs: Prentice Hall, 1971.
44. **Salton G,** Buckley C. Term-weighing approaches in automatic text retrieval. In: Spark Jones K, Willett P, eds. *Readings in information retrieval*. San Francisco: Morgan Kaufmann Publishers, Inc, 1987:323—7.

Evaluation of a generalizable approach to clinical information retrieval using the automated retrieval console (ARC)

Leonard W D'Avolio, Thien M Nguyen, Wildon R Farwell, et al.

J Am Med Inform Assoc 2010 17: 375-382
doi: 10.1136/jamia.2009.001412