Assisted annotation of medical free text using RapTAT

Glenn T Gobbel,[1,2,3] Jennifer Garvin,[4,5,6,7] Ruth Reeves,[1,2] Robert M Cronin,[2,3] Julia Heavirland,[4] Jenifer Williams,[4] Allison Weaver,[4] Shrimalini Jayaramaraja,[3] Dario Giuse,[2] Theodore Speroff,[1,3,8] Steven H Brown,[1,2] Hua Xu,[9] Michael E Matheny[1,2,3,8]

▶ Additional material is published online only. To view please visit the journal online (http://dx.doi.org/10.1136/amiajnl-2013-002255).

For numbered affiliations see end of article.

Correspondence to
Dr Glenn T Gobbel, Department of Veterans Affairs, Tennessee Valley Healthcare, 1310 24th Ave South, 4th Floor GRECC, Nashville, TN 37212, USA; glenn.t.gobbel@vanderbilt.edu

Received 6 August 2013
Revised 13 December 2013
Accepted 17 December 2013
Published Online First 14 January 2014

ABSTRACT

Objective To determine whether assisted annotation using interactive training can reduce the time required to annotate a clinical document corpus without introducing bias.

Materials and methods A tool, RapTAT, was designed to assist annotation by iteratively pre-annotating probable phrases of interest within a document, presenting the annotations to a reviewer for correction, and then using the corrected annotations for further machine learning-based training before pre-annotating subsequent documents. Annotators reviewed 404 clinical notes either manually or using RapTAT assistance for concepts related to quality of care during heart failure treatment. Notes were divided into 20 batches of 19–21 documents for iterative annotation and training.

Results The number of correct RapTAT pre-annotations increased significantly and annotation time per batch decreased by ~50% over the course of annotation. Annotation rate increased from batch to batch for assisted but not manual reviewers. Pre-annotation F-measure increased from 0.5 to 0.6 to >0.80 (relative to both assisted reviewer and reference annotations) over the first three batches and more slowly thereafter. Overall inter-annotator agreement was significantly higher between RapTAT-assisted reviewers (0.89) than between manual reviewers (0.85).

Discussion The tool reduced workload by decreasing the number of annotations needing to be added and helping reviewers to annotate at an increased rate. Agreement between the pre-annotations and reference standard, and agreement between the pre-annotations and assisted annotations, were similar throughout the annotation process, which suggests that pre-annotation did not introduce bias.

Conclusions Pre-annotations generated by a tool capable of interactive training can reduce the time required to create an annotated document corpus by up to 50%.

INTRODUCTION

Natural language processing (NLP) systems can help to monitor patient care by automated processing of medical records and extraction of quality-of-care indicators.[1–5] Such systems are often designed to replace manual review but may still require a manually annotated corpus for initial training or formal evaluation because the structure of documents, syntax, and terminology used for expression can vary among domains and personnel; an existing system optimized for one medical specialty or organization may not work well for another. As a result, NLP systems that rely on machine learning may have to be trained using annotated documents from the intended medical domain, and both rule-based and machine learning-based NLP tools require testing and validation before deployment within a new medical field.

Because NLP can reduce the cost and increase the efficiency of data extraction relative to manual annotation, a recent Veterans Affairs (VA)-supported project has focused on developing an NLP system to support automated monitoring of care for patients with congestive heart failure (CHF). That system aims to detect clinical signs and treatments that can show the consistency with which providers adhere to American Heart Association (AHA) guidelines for CHF care. Following AHA guidelines has been shown to reduce hospital admissions, improve quality of life, and decrease mortality of patients with CHF,[6 7] so rapidly identifying discrepancies between guidelines and care can help to mitigate decreases in care quality. To achieve this aim, the NLP system will need to identify seven concepts within clinical notes: (1) mentions of ACE inhibitor administration; (2) mentions of angiotensin II receptor blocker (ARB) administration; (3) mentions of ejection fraction; (4) quantitative measures of ejection fraction; (5) mentions of left ventricular systolic function; (6) qualitative measures of left ventricular systolic function; and (7) documented reasons for not administering ACE inhibitors or ARBs when otherwise indicated. This study describes the design and evaluation of an assisted annotation tool designed to support the development of an annotated reference corpus, which will be used to train and test the machine learning-based NLP system.

BACKGROUND

The lack of annotated datasets can substantially hinder the development and use of NLP on clinical text.[8] Annotating clinical documents to create an annotated corpus is laborious and expensive. High cost and labor requirements are incurred owing to the requirement for reviewers with sufficient domain expertise to identify relevant text.[9] Annotation commonly employs two independent annotators for review and a third person to adjudicate disagreements.[10 11] Furthermore, the document corpus must be large enough to allow for accurate training and testing.

To cite: Gobbel GT, Garvin J, Reeves R, *et al*. *J Am Med Inform Assoc* 2014;**21**:833–841.

Given the importance of annotation to NLP system development, studies have focused largely on two primary methods of reducing the burden and cost of generating annotated corpora: active learning and pre-annotation. Active learning can decrease the cost of annotating text by actively involving the learning algorithm in the document selection process,[12] and its goal is to train the system while requiring as few samples as possible. It has been applied in a wide variety of language processing tasks[13]—for example, part-of-speech tagging,[14 15] text categorization,[16 17] named entity recognition,[18 19] and classification of assertions.[20] Active learning has been reported to reduce the number of training samples required by 38–63%.[18 20]

Because the focus is on reducing training sample size, active learning does not reduce the burden when annotating a set number of documents. In contrast, the goal of pre-annotation is to reduce the time and/or effort required to annotate a document by reducing the number of annotations a reviewer must add. Pre-annotation is generally carried out using a dictionary generated for the annotation task or an existing NLP system. Recently, Lingren *et al*[21] created a dictionary to generate pre-annotations in clinical trial announcements, focusing on the impact of pre-annotation on the ability of reviewers to label disease and symptom-related concepts. Pre-annotation reduced the time needed for review by 14–21% compared with fully manual annotation. Investigations using existing NLP systems for pre-annotation of non-medical documents reported reductions in annotation time of 50–58% for named entity recognition, part-of-speech tagging, and parsing within non-medical documents.[22–24]

Despite the reported benefits of pre-annotation, there are some potentially important considerations for its use. Inaccurate pre-annotations may require deletion or correction, and evidence indicates that time-savings correlate with pre-annotation accuracy.[25 26] For some tasks, pre-annotation may not alter annotation time,[27] and the presence of multiple, inaccurate pre-annotations may instead increase annotation time.[25 28] Also, pretrained systems capable of pre-annotating for a specific task or medical realm either may not exist or may not be sufficiently accurate when used within a new domain. Although it is possible to create task-specific pre-annotation systems,[21] doing so may require substantial effort and offset the time-savings afforded by pre-annotation. Furthermore, although some studies have found no evidence to suggest that pre-annotation induces bias or reduces quality of annotating text for biomedical concepts or part-of-speech,[21 29 30] Fort and Sagot suggest that pre-annotations can induce bias, leading to decreases in random errors but increases in systematic errors by reviewers.[25]

This study describes the design and evaluation of an assisted annotation tool that may serve as an alternative approach to previously described methods of pre-annotation. In it, we assess the impact of generating pre-annotations interactively using iterative machine learning as implemented in the Rapid Text Annotation Tool (RapTAT) on annotation burden. Specifically, the study evaluates whether RapTAT can support interactive, assisted annotation and reduce the time required for annotation without negatively affecting inter-annotator consistency or inducing annotation bias relative to manual review.

METHODS

Sampling and population

The study corpus consisted of notes on patients with CHF, including discharge summaries, emergency department triage and nursing notes, internal medicine attending notes, neurology resident notes, physician discharge notes, physician history and physical notes, and primary care outpatient notes. Documents were selected from a larger corpus consisting of a random sample of documents generated between September 2007 and September of 2008 by six independent VA medical centers from the western USA. Patients were excluded if they (a) had participated in trials related to ACE inhibitors or ARBs; (b) had comfort measure advanced directives; (c) were fitted with heart assist devices (except pacemakers or defibrillators); or (d) had had a heart or heart/lung transplant. The final study corpus contained 404 documents from 171 patients. The Tennessee Valley and Salt Lake City Health System VA and University of Utah institutional review boards and research and development committees approved the study and granted a waiver regarding the need to obtain informed consent and Health Insurance Portability and Accountability Act authorization.

Schema development

A cardiology expert and three experienced annotators designed the annotation schema using an iterative process involving schema generation, annotation of a document sample, review of the annotations, and schema revision. The schema development process defined the key concepts that occur within the medical record and that relate to clinical care guidelines for patients with CHF. According to the guidelines, patients in systolic heart failure with an ejection fraction of ≤40% should be treated with ACE inhibitors or, alternatively, ARBs.[31] The schema was designed to provide annotations so that the NLP tool could identify (1) evidence of heart failure, (2) whether the patient was receiving ACE inhibitors or ARBs, and (3) if a reason was provided for not prescribing ACE inhibitors or ARBs to patients with heart failure. The final schema contained seven concepts (table 1), and the task of annotators was to identify phrases in the text that express those concepts.

Annotator training

Four reviewers, all experienced in clinical note annotation, were responsible for annotation. All annotators were provided with annotation guidelines specific to the schema. Two were responsible for manual annotations only, and the other two carried out only RapTAT-assisted annotation. To train all reviewers with respect to the annotation schema, the creators of the schema used consensus annotation to generate a training set of 30 documents distinct from the study corpus. Reviewers annotated the training set in batches of 10 using the Knowtator annotation tool (figure 1).[32] They were required to achieve an agreement score exceeding 80% between their annotations and the adjudicated training set before proceeding with review of documents in the study corpus, where

$$\text{Agreement} = \frac{\text{Matches}}{\text{Matches} + \text{Non-Matches}} \quad (1)$$

Annotation of the study corpus

Each document in the corpus was randomly assigned to one of 20 batches, and each batch contained 19–21 documents (figure 2). The batches were used as units of analysis for statistical purposes and to identify document sets for training RapTAT during assisted annotation. Assisted reviewers annotated the first document batch without any pre-annotation to provide the initial training of the machine learning algorithms within RapTAT. The next batch was pre-annotated by RapTAT based on this training,

Table 1 Schema for the seven concepts annotated within the corpus and text samples demonstrating phrases that should be annotated

Concept	Number of documents containing concept	Number of patients with concept	Sample text*
ACEI	272	132	"**ACEI**," "**ACE inhibitor**", "**Altace**", "**Vaseretic**," "**Captopril**," "**Lisinopril**"
Angiotensin II receptor blocker	107	53	"**ARB**," "**Angiotensin receptor blocker**," "**Sartans**," "**Losartan**"
EF	201	118	"**Estimated ejection fraction**," "**EF**", "**LVEF**", "**Ejection fraction**"
EF quantitation	197	116	"EF=**60–70%**," "EF is **about 30%**," "Ejection fraction in the range of **40 to 50%**"
LV systolic function/dysfunction	79	51	"**LV systolic function**," "**Systolic dysfunction**," "**LV function**," "Normal LV size and **function**,"
LV systolic function value	76	48	"**Mild** systolic dysfunction," "Systolic function is **borderline normal**"
Reason not on ACEI/ARB	40	26	"**Elevated creatinine** levels," "**Developing sepsis**," "**Patient refuses to take ACEI**", "**Renal disease**"

*Annotated phrases in bold. Examples corresponding to each concept were provided for reviewers as part of the annotation guidelines, but they were not meant to comprehensively represent all phrases that might refer to a given concept. For the concept 'Reason not on ACE inhibitor/ARB,' reviewers were instructed to annotate a phrase only when it was provided as an explicit reason for not prescribing one of the drugs.
ACE, angiotensin converting enzyme; ACEI, ACE inhibitor; ARB, angiotensin II receptor blocker; EF, ejection fraction; LV, left ventricular.

displayed within Knowtator for review and correction by the assisted annotators, and the corrected annotations were entered into RapTAT to update its training before pre-annotating the subsequent batch. This iterative process of pre-annotation, correction, and updating of RapTAT training was carried out by separate instances of RapTAT for each of the two assisted reviewers, and it continued until the final batch had been corrected following pre-annotation. Manual annotators also used Knowtator for annotating each batch, but the documents were not pre-annotated. An adjudicator who was neither a manual nor assisted annotator reviewed the manual annotations to produce the reference standard. Inter-annotator agreement (IAA) was calculated using equation 1.

Text processing

RapTAT learns to pre-annotate documents with the likely annotations of a reviewer based on iterative feedback from that same reviewer. The tool used two different probabilistic models to estimate the likelihood of a reviewer (1) annotating a particular phrase and (2) mapping that phrase to a particular schema concept (table 1). For both models, we defined a token as a contiguous group of characters that corresponded to a word, value, or unit of measure, and a phrase as a contiguous sequence of one or more tokens that is representative of one of the schema concepts. Considering only token sequences (S) in a phrase without regard to context, the probability of annotation (A) of a given sequence is

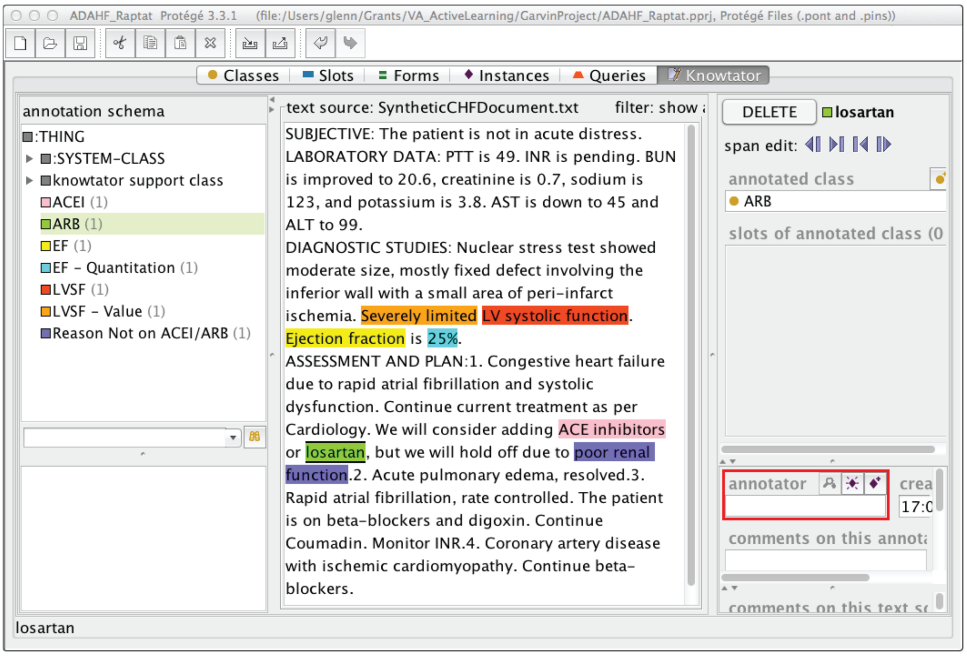

Figure 1 Screen capture of the Knowtator annotation plug-in within the Protégé application. The displayed document is synthetic but contains text representative of that found within the study corpus. Schema concepts are listed on the left. For each corpus document, reviewers use the input device of the computer to highlight all phrases mapping to one of the schema concepts and to select the concept associated with each highlighted phrase.

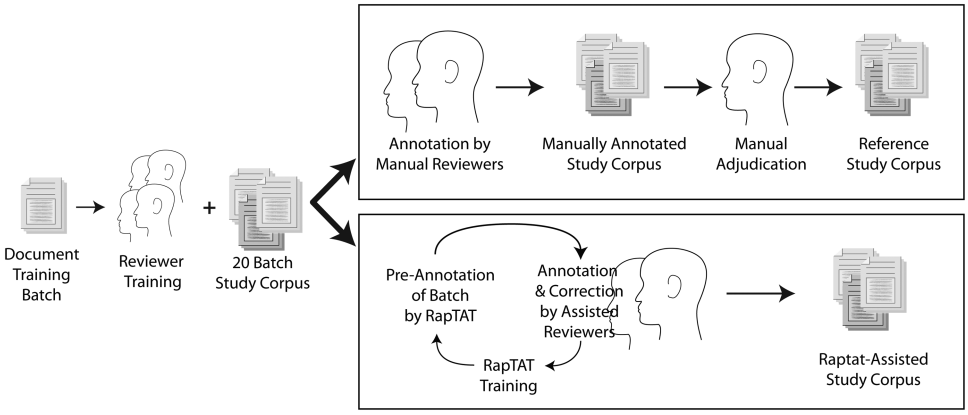

Figure 2 Document flow for generation of the annotated study corpus using manual review and adjudication or RapTAT-assisted review.

$$P(A|S) = \frac{\text{Number of Annotations of S}}{\text{Number of Occurrences of S}} \quad (2)$$

We modified this equation for use in RapTAT because, if this simple phrase identification model is used, subsequences shorter than the complete annotated phrase do not enter into probability calculations. For example, if 'high fever of unknown origin' was annotated, the probability of annotating the subsequence 'high fever' would not increase. Such a model could reduce recall by underestimating the probability of annotating token sequences that occur infrequently as complete annotated phrases even though they might occur frequently as subsequences. We therefore adjusted RapTAT to give partial credit to subsequences (table 2). Each subsequence within an annotated phrase of length i in a sequence of length j was credited with an annotation count of i/j (numerator, equation 2). Thus, the credited count was lower for sub-sequences that were particularly short relative to the length of the complete annotated phrase. All token sequences whose first token was not the first token in an annotated phrase were considered unlabeled and contributed equally to the number of sequence occurrences (denominator, equation 2).

Estimating the likelihood of mapping a phrase to a concept was accomplished using a multinomial naïve Bayes classifier. The classifier calculated the most probable concept for a given phrase, using the equation

$$P(C_i|T_1, \ldots, T_k) = \frac{P(C_i) \bullet P(T_1|C_i) \bullet \cdots \bullet P(T_k|C_i)}{P(T_1, \ldots, T_k)} \quad (3)$$

Table 2 Examples demonstrating how annotated phrases and their subsequences are counted during training, where n represents the number of tokens in the phrase

Sequence length	Phrase	Tokens	Number of annotations credited to sequence
Full, annotated phrase	"LV systolic function"	3	1.0
n−1 subsequence	"LV systolic"	2	0.67
n−2 subsequence	"LV"	1	0.33
Full, annotated phrase	"Renal disease"	2	1.0
n−1 subsequence	"Renal"	1	0.5

LV, Left venticular.

where $P(C_i)$ refers to the probability of occurrence of the ith concept, k is the number of tokens in the phrase and T_k refers to the token at the kth position in the sequence. The value of $P(T_k|C_i)$ is provided by the equation

$$P(T_k|C_i) = \frac{\text{Occurrences of Token T at position k when a Phrase Maps to Concept } C_i}{\text{Occurrences of Concept } C_i} \quad (4)$$

Because the denominator in equation 3, $P(T_1, \ldots, T_k)$, is constant when mapping a given phrase, finding the most probable concept for mapping is reduced to identifying the one that maximizes the numerator. Laplace smoothing adjusted for the occurrence of tokens missing from the training data.[33] Multiple studies have used this multinomial naïve Bayes models for text classification,[34] although, to the best of our knowledge, the use of token position as a feature for medical concept mapping is unique to RapTAT.[35]

RapTAT system design

The RapTAT system was programed in Java, and consisted of one module that determined the likelihood of phrase annotation and a second that determined the likelihood of a given phrase mapping to a particular concept (figure 3). Phrases analyzed by the system were limited to contiguous sequences of ≤7 tokens. Before analysis by the two RapTAT modules, the text was preprocessed, which consisted of detecting sentence boundaries, dividing each sentence into tokens, removing 'stop word' tokens ("and," "by," "for," "in," "nos," "of," "on," "the," "to," and "with"), and identifying and adding the appropriate part of speech to the token as a suffix. The preprocessing steps were carried out using the OpenNLP libraries (Apache Software Foundation). All versions of RapTAT are available at http://code.google.com/p/raptat/, and V.0.6a was used for this study.

Evaluation measures

RapTAT was evaluated based on the number of true positives (TPs), false negatives (FNs), and false positives (FPs) within the pre-annotations. Precision, recall, and F-measure provided measures of performance of the RapTAT tool and were calculated for both the corrected annotations from the RapTAT-assisted annotators and the reference standard described above. A TP was defined as an overlap of one or more tokens between the RapTAT-generated and reference standard that mapped to the same concept. RapTAT automatically scored TPs, FPs, and FNs

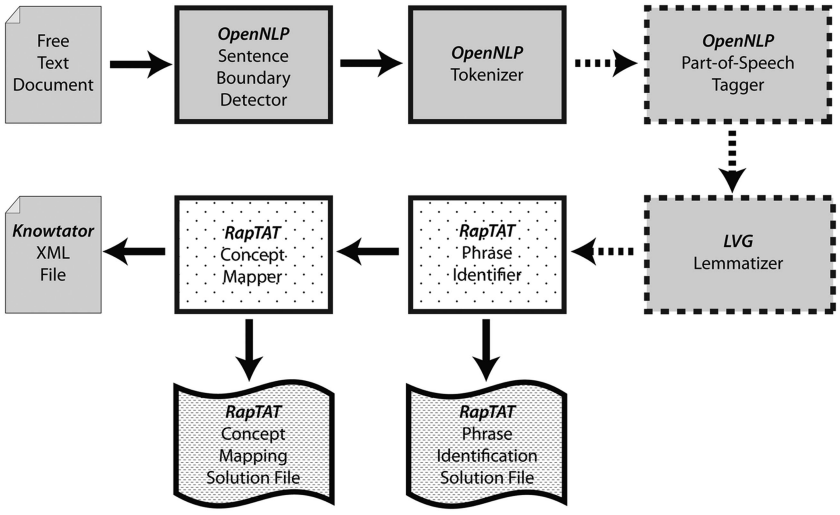

Figure 3 Data flow during training and pre-annotation by the RapTAT interactive machine learning system. Dotted lines and arrows represent optional parts of the system that are available but were not used in this study, such as Lexical Variant Generation (LVG) lemmatization. Stippled patterns represent RapTAT-specific modules (light stippling) and files (dense stippling).

and calculated precision, recall, and F-measure according to the equations

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (5)$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (6)$$

$$\text{F} - \text{Measure} = 2 \bullet \text{Precision} \bullet \text{Recall}/(\text{Precision} + \text{Recall}) \quad (7)$$

We used leave-one-out cross-validation to estimate the performance of RapTAT with respect to each of the schema concepts. Cross-validation consisted of training RapTAT using all but one of the annotated documents from a given reviewer; RapTAT then generated annotations for the 'left-out' document, which were compared with those of the reference standard. This process was repeated for each document and reviewer. Precision, recall, and F-measure for a given concept were calculated by combining the TPs, FPs, and FNs for that concept.

Reviewer annotation time and rate

To assess batch-to-batch changes in annotation time, each RapTAT-assisted reviewer recorded the time required to review each document. Time for each batch was normalized to batch size in kilobytes. Because correct pre-annotations might decrease, and incorrect annotations might increase, annotation time, we also calculated annotation rate of both manual and assisted reviewers with respect to only those annotations that were added or corrected. Correction was defined as either modifying the beginning or end offsets of the annotation or changing the concept to which the phrase mapped. We defined the annotation rate as the number of annotations added or corrected per minute based on timestamps generated by Knowtator for each annotation. Knowtator did not create timestamps for FP pre-annotations that were removed during assisted review, so the occurrence of, and the time taken for, such corrections were not explicitly included in the calculations of annotation rate. Because annotation rates were not normally distributed, we determined the median rates for each reviewer and batch, and those data were used for statistical evaluations of the change in annotation rate as a function of batch number.

RapTAT system training and annotation rates

To evaluate the training rate of the RapTAT system, we measured the time required to process the first 10 document batches. To evaluate the annotation rate, the corpus was divided into two independent training and test groups with 10 batches of documents in each. After processing the training documents, annotation rate of RapTAT was calculated based on the time spent pre-annotating the test documents. Times were normalized to document corpus size in kilobytes. Time required to read the corpus from disk into computer memory and read and write data structures before and after training was excluded from all rate calculations. Heap size of the Java virtual machine was ≤1 GB. Training and testing were carried out on the VA informatics and computing infrastructure (VINCI) server, which ran on an Intel Xeon quad-core processor running at 2.27 GHz and supplied with 128 GB of RAM. The operating system was Windows Server, 2008 R2 Enterprise.

Statistical analysis

The study used simple linear regression to evaluate the statistical significance of changes in F-measure, annotation rate, and fraction of annotations correctly provided by RapTAT as a function of document batch. A 'correct' RapTAT annotation was defined as a pre-annotation that was neither added nor corrected by the reviewer. To compare the similarity of RapTAT-generated pre-annotations with the assisted and reference standard annotations, we ran paired t tests on estimates of precision, recall, and F-measure across all batches. A Student's t test was used to compare the number of annotations added or corrected by assisted versus manual reviewers. A two-sample proportion test was employed to identify statistical differences for single measures of IAA. All statistical analyses were carried out using Stata/IC V.11.2 for Mac (Stata Corp, College Station, Texas, USA), and p values <0.05 were considered significant.

RESULTS

There was a notable decrease in annotation time from batch to batch for the RapTAT-assisted reviewers, especially over the first six to seven batches, followed by a slower apparent decrease over batches 14–20 (figure 4; top). Annotation time decreased by about 50% from the first to the last batch. Part of this decrease may be accounted for by the gradual decrease in the number of annotations that had to be added or corrected by the annotators over the course of annotation (figure 4; bottom). Averaged over the entire corpus, the two manual annotators added 100±18 (mean±SD) annotations per batch. The assisted annotators added or corrected significantly fewer; 78±12 annotations per batch, and 21±9 annotations per batch were

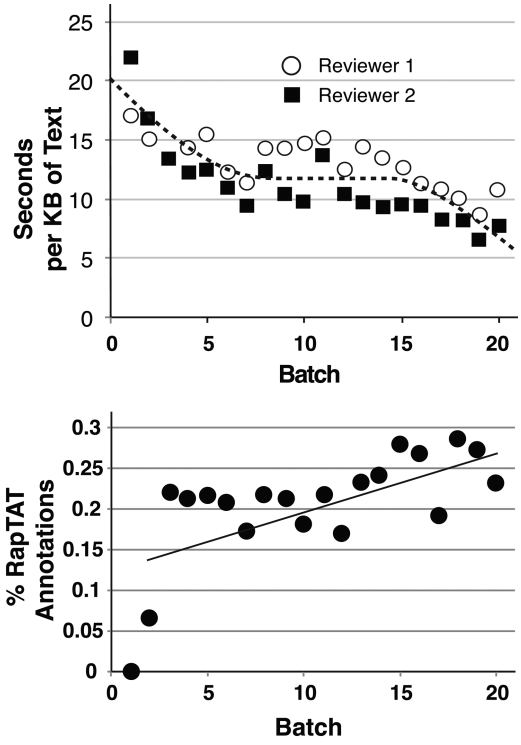

Figure 4 Time required to annotate one kilobyte of text as a function of the number of document batches reviewed (top), and the fraction of all annotations that were uncorrected by the reviewers and added only by RapTAT. For the annotation time plot (top), each symbol represents the time taken by a single RapTAT-assisted annotator for a particular batch of documents from the study corpus, and the dashed line represents the apparent, batch-to-batch trend in annotation time. For the plot of the fraction of annotations generated by RapTAT alone with no correction by the annotators (bottom), each symbol represents the total number of uncorrected annotations generated by RapTAT for each batch divided by the total number of annotated phrases in the batch; the least squares line of regression is also included, and the slope is significantly different from zero ($p<0.01$).

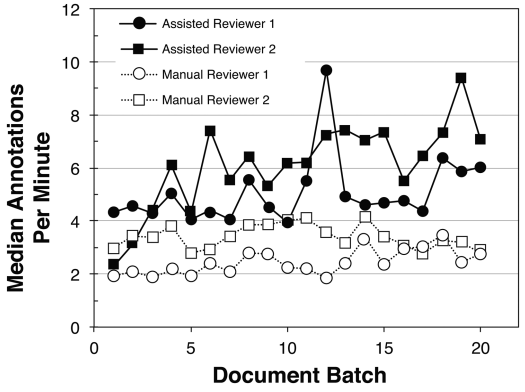

Figure 5 Annotation rate as a function of the number of document batches reviewed. Each symbol represents the rate for a single reviewer for a particular batch of documents from the study corpus. The rate represents the inverse of the time between adding or correcting annotations for both manual (open symbols) and RapTAT-assisted (closed symbols) reviewers.

generated as pre-annotations by RapTAT during assisted annotation and did not require correction. To determine if the decrease in annotation number alone accounted for the marked decrease in annotation time (figure 4; top), the rate of adding or correcting only annotations while excluding correct annotations generated by RapTAT was evaluated. The annotation rate of the assisted reviewers significantly increased over the course of annotation (+0.145 added or corrected annotations per minute per batch; 95% CI 0.07 to 0.22) and approximately doubled from the first to the last batch (figure 5). In contrast, the batch-to-batch change in annotation rate for the manual reviewers was significantly lower than that of the assisted annotators and did not change significantly over the course of annotation (+0.022 annotations per minute per batch; 95% CI −0.004 to 0.048).

The F-measure of the RapTAT pre-annotations relative to the assisted reviewer annotations increased steeply over the initial five to six batches. After a single batch of training, the F-measure was 0.5–0.6 and increased to >0.80 after three batches. Precision and recall increased similarly. Linear regression analysis of the performance scores after the initial five batches showed a non-significant trend towards a continuing increase in F-measure ($p=0.0623$ for slope > 0 by linear regression analysis). There was no evidence that pre-annotation introduced bias. Precision, recall, and F-measure (data not shown) increased in a similar fashion through the course of annotation regardless of whether the pre-annotation performance measures were calculated relative to the reviewer annotations (figure 6, left) or the reference standard (figure 6, right). Furthermore, although the RapTAT pre-annotations were more similar to the annotations of the assisted reviewers than the reference standard based on significantly increased precision, recall, and F-measure across all batches (paired t test; $p<0.05$), the average increases were generally slight (≤0.046) and consistent from batch to batch. This finding was expected because the tool was specifically trained using the annotations of each assisted reviewer. There was no evidence that pre-annotation adversely affected IAA, which was significantly greater for assisted than manual annotation for certain concepts as well as overall (table 3).

The performance of RapTAT with respect to its ability to annotate phrases accurately was concept dependent (table 4). The four highest F-measures ranged from 0.80 to 0.97 and corresponded to the most highly prevalent concepts in the corpus, and the lowest F-measure was for the least prevalent concept, 'Reason not on ACE inhibitor/ARB' concept (table 1).

The processing speed of the RapTAT tool during annotation was 132.0 ms/kb of text. 'Preprocessing,' which we define as sentence boundary detection, tokenization, part-of-speech detection, and stop word removal, took most of the time (123 ms); only 9 ms were required for training once the text was read into computer memory and preprocessed. Annotation rate by the tool was 116.6 ms, which consisted of 116 ms for preprocessing and 0.55 ms for phrase identification and concept mapping.

DISCUSSION

This study demonstrates that pre-annotation based on interactive, iterative machine learning can reduce the burden associated with creating an annotated corpus. Considering the annotation time and rate of the two assisted reviewers compared with the manual reviewers, we estimate that using assisted rather than manual annotation would have saved each reviewer roughly 16 h for annotation of the entire 404-document corpus. Also, our study found no evidence to suggest that pre-annotation introduces bias. Before the study, we were concerned that the closed feedback loop between RapTAT and each

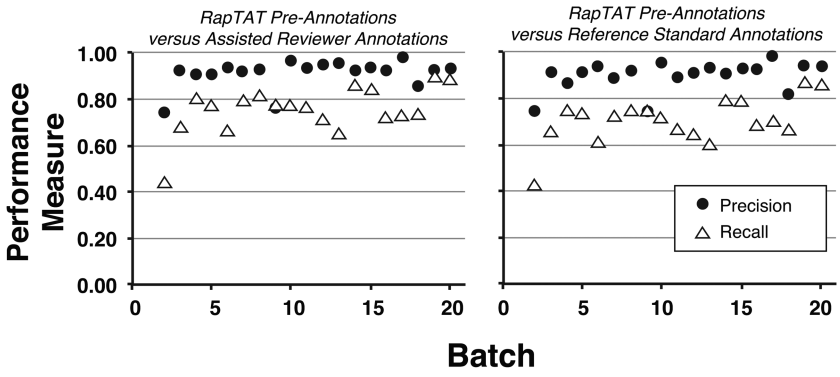

Figure 6 Performance of the RapTAT tool as measured by precision and recall as a function of the number of document batches used for training. Pre-annotations provided by the RapTAT tool were scored for performance versus either the assisted reviewer annotations (left) or the reference standard annotations (right).

reviewer might induce a drift in the annotations, so that pre-annotations might closely match annotations of each reviewer but increasingly deviate from those of the reference or other annotator over the course of annotation. However, the IAA for the assisted annotators was equal to or higher than that of the manual annotators. Also, the precision, recall, and F-measure for the pre-annotations relative to the assisted annotations and for the pre-annotations relative to the reference standard remained similar throughout the course of annotation.

The F-measure of the pre-annotations relative to the assisted reviewer annotations was <0.8 for the first few annotation batches, so the tool may provide only slight assistance in the early stages. This is a limitation of the iterative training needed for RapTAT compared with prior approaches that initially pre-annotate all documents using existing tools or ones created for the task. As RapTAT learns and improves, the number of annotations that must be added or corrected decreases and the annotation time of the reviewers correspondingly decreases. Fort and Sagot examined the impact of pre-annotation accuracy on annotation time and found that increasing accuracy from 66.5% to 81.6% was associated with an ~50% decrease in annotation time.[25] In our study, the F-measure reached 81% after three document batches, which suggests that about 60 documents may be required for training RapTAT to a level of accuracy such that its pre-annotations substantially reduce annotation time. The impact of training on F-measure was concept dependent, which may be partially related to concept prevalence, so the rate of increase in annotation speed as a function of the number of documents annotated may be slower for infrequent concepts.

In this study, while RapTAT was used to generate the pre-annotations, annotators added and corrected pre-annotations using the separate Knowtator tool. Our goal is to eventually embed RapTAT within an annotation tool. This will allow annotators to update the machine learning algorithms after each document and obviate the import and export of data that was required in this study. When designing RapTAT, we were concerned that existing language models, such as maximum entropy Markov and conditional random fields, might not be sufficiently rapid to support iterative training and pre-annotation in a way that would avoid delays during annotation. We therefore used language models and worked to implement algorithms that would be sufficiently fast to support the interactive annotation process described in this study. Based on the annotation and system training rate determined in this study, RapTAT should be readily capable of supporting real-time, interactive annotation. The rate-limiting factor is disk access. Since 1 kb of text equals about half a page, the current RapTAT system should take about 1 s to train on four pages or annotate eight pages once the documents are read from disk and stored in computer memory.

The impact of the interactive approach to pre-annotation described here on annotation time appears to be within the range reported in other similar studies, which decreased annotation time by 14–58%.[21–24 30] Interactive, assisted pre-annotation in our study approximately doubled the annotation rate relative to that of manual reviewers. Studies examining changes in IAA due to pre-annotation have been less consistent, with some studies reporting no change and another reporting an increase of 11%.[21 28 30] Interactive assisted annotation in this study improved IAA by ~27%. Although some of the decrease in annotation time in our study was expected and probably due to the increased fraction of annotations correctly labeled by RapTAT, there was an unexpected increase in annotation rate unrelated to annotation number. Our calculation of annotation rate did not explicitly include the time required for

Table 3 Inter-annotator agreement (IAA) between the two manual and between the two RapTAT-assisted reviewers

	Average IAA (95% CI)	
Concept	**Manual**	**Assisted**
Angiotensin converting enzyme inhibitor	0.89 (0.86 to 0.93)	0.93* (0.91 to 0.96)
Angiotensin II receptor blocker	0.81 (0.72 to 0.89)	0.97* (0.95 to 1.00)
Ejection fraction	0.86 (0.80 to 0.93)	0.97* (0.95 to 1.00)
Ejection fraction quantitation	0.90 (0.85 to 0.94)	0.88 (0.83 to 0.92)
Left ventricular systolic function/dysfunction	0.82 (0.73 to 0.91)	0.76 (0.62 to 0.89)
Left ventricular systolic function value	0.85 (0.78 to 0.93)	0.77 (0.64 to 0.90)
Reason not on ACE inhibitor/ARB	0.58 (0.46 to 0.70)	0.54 (0.45 to 0.64)
Total (combined over all concepts)	0.85 (0.81 to 0.88)	0.89* (0.87 to 0.91)

*Indicates significant difference when comparing the IAA of the manual reviewers with that of the RapTAT-assisted reviewers.
ACE, angiotensin converting enzyme; ARB, angiotensin II receptor blocker.

Table 4 Performance of the RapTAT tool for the various schema concepts as measured by precision, recall, and F-measure

	Performance Measure		
Concept	**Precision**	**Recall**	**F**
Angiotensin converting enzyme inhibitor	0.97	0.94	0.95
Angiotensin II receptor blocker	0.99	0.96	0.97
Ejection fraction	0.96	0.95	0.96
Ejection fraction quantitation	0.77	0.82	0.80
Left ventricular systolic function/dysfunction	0.61	0.82	0.70
Left ventricular systolic function value	0.83	0.37	0.51
Reason not on ACE inhibitor/ARB	0.36	0.12	0.18

ACE, angiotensin converting enzyme; ARB, angiotensin II receptor blocker.

removal of FP pre-annotations. When annotators review a single document, our observation has been that they continuously alternate between removing FPs, adding FNs, and correction of inaccurate text spans or concept mapping. Based on this observation, the time for correcting FPs would have been added to the time taken between adding or correcting annotations, and thus reduced the annotation rate of the assisted annotators. Therefore, exclusion of the time taken for correcting FPs does not account for the increased annotation rate of the assisted reviewers. One possible explanation is that correcting annotations may take less time than adding missing annotations. The existence of pre-annotations may also reduce the cognitive burden by decreasing the number of annotations that have to be identified in each document or helping to delineate document sections. With respect to the increase in IAA for assisted annotators, we theorize that pre-annotation by RapTAT may help reviewers to identify and annotate phrases that they might otherwise overlook, thus reducing inter-annotator discrepancies. A potential benefit of increased IAA is a decrease in the adjudication workload.

Although previous studies have suggested that pre-annotation can reduce annotation burden, the iterative, machine learning-based approach to pre-annotations described here has some important advantages. First, there is no need to identify or create a pre-annotation system because such a system is generated during the annotation process. RapTAT can be used without the linguistic and computational experience that might otherwise be required to implement a pre-annotation system. Second, the system carrying out pre-annotation is automatically optimized for the schema and intended domain via machine learning during annotation. Considering that low pre-annotation accuracy can slow the annotation process,[25 28] correctly tailoring the pre-annotation to the domain is important, and non-optimized pre-annotation tools, such as pre-existing systems or dictionaries developed for a task, may not be sufficient.

There have been previous reports on the use of machine learning-based pre-annotations for assisted annotation. Kors *et al*[36] used the assisted annotation function of the BRAT tool to generate multilingual corpora of documents annotated for multiple biomedical semantic types.[37] BRAT is a web browser-based tool that can use external web services for text processing and generation of pre-annotations.[37] Culotta *et al*[9] described an iterative approach similar to the one described for RapTAT for training a named entity recognition system. Using simulations, they reported that their approach reduced the number of 'actions' required by an annotator by 42%. The MIST tool has been used to annotate protected health information within medical documents, and it can be trained to identify other concepts.[38] Another annotation tool, BOEMIE, is reported to have the ability to use a similar interactive approach to assist with text annotation.[39] To the best of our knowledge, the impact of using MIST or BOEMIE on annotation time and IAA and their ability to support real-time interactive annotation have not been reported.

CONCLUSION

This study demonstrates that interactive, iterative machine learning as provided by RapTAT can assist with the annotation of text by gradually learning to produce accurate pre-annotations. Doing so substantially reduces the annotation time by decreasing the number of annotations that must be added by reviewers and helping to accelerate the rate at which reviewers are able to add missing annotations and correct inaccurate ones. RapTAT also improves IAA, which should accelerate adjudication when using multiple reviewers for annotation. Integration of RapTAT or a similar system with an annotation tool could help to mitigate an important barrier to implementing NLP systems in the medical field.

Author affiliations
[1]Department of Veterans Affairs Medical Center, Geriatric Research, Education and Clinical Center (GRECC), Nashville, Tennessee, USA
[2]Department of Biomedical Informatics, Vanderbilt University School of Medicine, Nashville, Tennessee, USA
[3]Division of General Internal Medicine & Public Health, Department of Medicine, Vanderbilt University School of Medicine, Nashville, Tennessee, USA
[4]IDEAS Center SLC VA Healthcare System, Salt Lake City, Utah, USA
[5]Division of Epidemiology, University of Utah School of Medicine, Salt Lake City, Utah, USA
[6]Department of Biomedical Informatics, University of Utah School of Medicine, Salt Lake City, Utah, USA
[7]Department of Veterans Affairs Medical Center, Geriatric Research, Education and Clinical Center (GRECC), Salt Lake City, Utah, USA
[8]Department of Biostatistics, Vanderbilt University School of Medicine, Nashville, Tennessee, USA
[9]School of Biomedical Informatics, University of Texas Health Science Center, Houston, Texas, USA

Acknowledgements We thank Vincent Messina for technological assistance and Stephane Meystre for a careful review of the manuscript.

Contributors GTG developed the algorithms, conceived and designed the study, acquired, analyzed, and interpreted the data, and wrote and revised the manuscript. RR conceived and designed the study and also reviewed and revised the manuscript. RMC, JH, JW, and AW designed the study, acquired the data, and reviewed the manuscript. SJ developed the algorithms and acquired and analyzed the data. JG, TS, and DG designed the study and reviewed and revised the manuscript. SHB conceived and designed the study and reviewed and revised the manuscript. HX analyzed and interpreted the data and reviewed and revised the manuscript. MEM developed the algorithms, conceived and designed the study, analyzed and interpreted the data, and reviewed and revised the manuscript.

Funding This material is based upon work funded by the Department of Veterans Affairs (VA), Veterans Health Administration, Office of Research and Development, Health Services Research and Development (HSR&D) program. The work was supported with resources and the use of facilities at the VA Tennessee Valley Healthcare System (TVHS). Funding for this study was provided through VA grant SAF-03-223 and HSR&D IBE 09-069.

Competing interests The VA Consortium for Health Informatics Research (CHIR) HIR 09-001 and HIR 09-003 also provided support to GTG, TS, and MEM. The Department of Veterans Affairs Health Administration HSR&D Career Development Award CDA-08–020 provided additional support to MEM; GTG and RR were supported by the Department of Veterans Affairs Medical Informatics Fellowship Program (sponsored by Office of Academic Affiliations, Office of Health Information, and HSR&D). GTG and RR performed the work in this study while serving as medical informatics fellows within the Department of Veterans Affairs Medical Center, Nashville, Tennessee. MEM is a physician researcher at the Geriatrics Research Education and Clinical Center (GRECC) at the Department of Veterans Affairs Medical Center, Nashville, Tennessee. SHB is a staff physician at the Department of Veterans Affairs Medical Center, Nashville, Tennessee and Director of

Knowledge-Based System, Health Informatics, Office of Informatics and Analytics, Department of Veterans Affairs. TS is chief of TVHS Center for Health Services Research, GRECC, Department of Veterans Affairs Medical Center, Nashville, Tennessee.

Ethics approval Institutional review boards of the Tennessee Valley VA, Salt Lake City VA, and University of Utah.

Provenance and peer review Not commissioned; externally peer reviewed.

REFERENCES

1 Matheny ME, Fitzhenry F, Speroff T, *et al*. Detection of infectious symptoms from VA emergency department and primary care clinical documentation. *Int J Med Inform* 2012;81:143–56.
2 Murff HJ, FitzHenry F, Matheny ME, *et al*. Automated identification of postoperative complications within an electronic medical record using natural language processing. *JAMA* 2011;306:848–55.
3 Chiang JH, Lin JW, Yang CW. Automated evaluation of electronic discharge notes to assess quality of care for cardiovascular diseases using Medical Language Extraction and Encoding System (MedLEE). *J Am Med Inform Assoc* 2010;17:245–52.
4 Harkema H, Chapman WW, Saul M, *et al*. Developing a natural language processing application for measuring the quality of colonoscopy procedures. *J Am Med Inform Assoc* 2011;18(Suppl 1):i150–6.
5 Greenberg JO, Vakharia N, Szent-Gyorgyi LE, *et al*. Meaningful measurement: developing a measurement system to improve blood pressure control in patients with chronic kidney disease. *J Am Med Inform Assoc* 2013;20:e97–101.
6 Bonow RO, Bennett S, Casey DE Jr, *et al*. ACC/AHA clinical performance measures for adults with chronic heart failure: a report of the American College of Cardiology/American Heart Association Task Force on Performance Measures (Writing Committee to Develop Heart Failure Clinical Performance Measures): endorsed by the Heart Failure Society of America. *Circulation* 2005;112:1853–87.
7 Juckett D. A method for determining the number of documents needed for a gold standard corpus. *J Biomed Inform* 2012;45:460–70.
8 Chapman WW, Nadkarni PM, Hirschman L, *et al*. Overcoming barriers to NLP for clinical text: the role of shared tasks and the need for additional creative solutions. *J Am Med Inform Assoc* 2011;18:540–3.
9 Culotta A, Kristjansson T, McCallum A, *et al*. Corrective feedback and persistent learning for information extraction. *Artif Intell* 2006;170:1101–22.
10 Roberts A, Gaizauskas R, Hepple M, *et al*. Building a semantically annotated corpus of clinical texts. *J Biomed Inform* 2009;42:950–66.
11 South BR, Shen S, Leng J, *et al*. A prototype tool set to support machine-assisted annotation. *2012 Workshop on Biomedical Natural Language Processing*; Montreal, Canada: Association for Computational Linguistics, 2012.
12 Thompson CA, Califf ME, Mooney RJ. Active learning for natural language parsing and information extraction. *Sixteenth International Conference on Machine Learning*; Morgan Kaufmann Publishers Inc, 1999.
13 Olsson F. *A literature survey of active machine learning in the context of natural language processing: Swedish Institute of Computer Science (SICS) Technical Report; 2009*. Report No: T2009:06.
14 Dagan I, Engleson SP. Committee-based sampling for training probabilistic classifiers. *Twelfth International Conference on Machine Learning*; Tahoe City, California: Morgan Kaufmann, 1995.
15 Ringger E, McClanahan P, Haertel R, *et al*. Active learning for part-of-speech tagging: accelerating corpus annotation. *Linguistic Annotation Workshop*; Prague, Czech Republic: Association for Computational Linguistics, 2007.
16 McCallum A, Nigam K. Employing EM and pool-based active learning for text classification. *Fifteenth International Conference on Machine Learning*; Morgan Kaufmann Publishers Inc, 1998.
17 Lewis DD, Gale WA. A sequential algorithm for training text classifiers. *17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*; Dublin, Ireland: Springer-Verlag New York, Inc, 1994.
18 Hachey B, Beatrice A, Becker M. Investigating the effects of selective sampling on the annotation task. *Ninth Conference on Computational Natural Language Learning*; Ann Arbor, Michigan: Association for Computational Linguistics, 2005.
19 Vlachos A. Active annotation. *Workshop on Adaptive Text Extraction and Mining (ATEM 2006)*; Trento, Italy; 2006.
20 Chen Y, Mani S, Xu H. Applying active learning to assertion classification of concepts in clinical text. *J Biomed Inform* 2012;45:265–72.
21 Lingren T, Deleger L, Molnar K, *et al*. Evaluating the impact of pre-annotation on annotation speed and potential bias: natural language processing gold standard development for clinical named entity recognition in clinical trial announcements. *J Am Med Inform Assoc* 2014;21:406–13.
22 Chiou F-D, Chiang D, Palmer M. Facilitating treebank annotation using a statistical parser. *First International Conference on Human Language Technology Research*; San Diego: Association for Computational Linguistics, 2001.
23 Ganchev K, Pereira F, Mandel M, *et al*. Semi-automated named entity annotation. *Linguistic Annotation Workshop*; Prague, Czech Republic: Association for Computational Linguistics, 2007.
24 Marcus MP, Marcinkiewicz MA, Santorini B. Building a large annotated corpus of English: the Penn treebank. *Comput Linguist* 1993;19:313–30.
25 Fort K, Sagot B. Influence of pre-annotation on POS-tagged corpus development. *Fourth Linguistic Annotation Workshop*; Uppsala, Sweden: Association for Computational Linguistics, 2010.
26 Ringger E, Carmen M, Haertel R, *et al*. Assessing the costs of machine-assisted corpus annotation through a user study. *Sixth International Conference on Language Resources and Evaluation (LREC'08)*; Marrakech, Morocco: European Language Resources Association (ELRA), 2008.
27 Rehbein I, Ruppenhofer J, Sporleder C. Assessing the benefits of partial automatic pre-labeling for frame-semantic annotation. *Third Linguistic Annotation Workshop*; Suntec, Singapore: Association for Computational Linguistics, 2009.
28 Ogren PV, Savova G, Chute C. Constructing evaluation corpora for automated clinical named entity recognition. *Language Resources and Evaluation Conference (LREC)*; 2008.
29 Dandapat S, Biswas P, Choudhury M, *et al*. Complex linguistic annotation—no easy way out!: a case from Bangla and Hindi POS labeling tasks. *Third Linguistic Annotation Workshop*; Suntec, Singapore: Association for Computational Linguistics, 2009.
30 Névéol A, Islamaj Doğan R, Lu Z. Semi-automatic semantic annotation of PubMed queries: a study on quality, efficiency, satisfaction. *J Biomed Inform* 2011;44:310–18.
31 Yancy CW, Jessup M, Bozkurt B, *et al*. 2013 ACCF/AHA guideline for the management of heart failure: a report of the American College of Cardiology Foundation/American Heart Association Task Force on Practice Guidelines. *Circulation* 2013;128:e240–327.
32 Ogren PV. Knowtator: a Protégé plug-in for annotated corpus construction. *North American Chapter of the Association for Computational Linguistics on Human Language Technology*; New York, New York: Association for Computation Linguistics, 2006.
33 Manning CD, Raghavan P, Schütze H. *Introduction to information retrieval*. New York: Cambridge University Press, 2008.
34 Schneider K-M. Techniques for improving the performance of naive Bayes for text classification. *6th International Conference on Computational Linguistics and Intelligent Text Processing*; Mexico City, Mexico: Springer-Verlag, 2005.
35 Gobbel GT, Reeves R, Jayaramaraja S, *et al*. Development and evaluation of RapTAT: a machine learning system for concept mapping of phrases from medical narratives. *J Biomed Inform* http://dx.doi.org/10.1016/j.jbi.2013.11.008.
36 Kors JA, Clematide S, Akhondi SA, *et al*. Creating multilingual gold standard corpora for biomedical concept recognition. *CLEF 2013 Conference*; Valencia, Span: 2013.
37 Stenetorp P, Pyysalo S, Topic G, *et al*. BRAT: a web-based tool for NLP-assisted text annotation. *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*; Association for Computational Linguistics, 2012.
38 Aberdeen J, Bayer S, Yeniterzi R, *et al*. The MITRE identification scrubber toolkit: design, training, and assessment. *Int J Med Inform* 2010;79:849–59.
39 Fragkou P, Petasis G, Theodorakos A, *et al*. Boemie ontology-based text annotation tool. *6th International Conference on Language Resources and Evaluation (LREC)*; Marrakech, Morocco, 2008.