Building an Automated Problem List Based on Natural Language Processing: Lessons Learned in the Early Phase of Development

Imre Solti, MD, PhD[1], Barry Aaronson, MD[1], Grant Fletcher, MD, MPH[1], Magdolna Solti, MD, MSc[2], John H. Gennari, PhD[1], Melissa Cooper[1], Thomas Payne, MD[1]
[1]University of Washington, Seattle, WA; [2]Northwest Cancer Specialists, Vancouver, WA

Abstract

Detailed problem lists that comply with JCAHO requirements are important components of electronic health records. Besides improving continuity of care electronic problem lists could serve as foundation infrastructure for clinical trial recruitment, research, biosurveillance and billing informatics modules. However, physicians rarely maintain problem lists. Our team is building a system using MetaMap and UMLS to automatically populate the problem list. We report our early results evaluating the application. Three physicians generated gold standard problem lists for 100 cardiology ambulatory progress notes. Our application had 88% sensitivity and 66% precision using a non-modified UMLS dataset. The system's misses concentrated in the group of ambiguous problem list entries (Chi-square=27.12 p<0.0001). In addition to the explicit entries, the notes included 10% implicit entry candidates. MetaMap and UMLS are readily applicable to automate the problem list. Ambiguity in medical documents has consequences for performance evaluation of automated systems.

Introduction

Problem lists are important components of the medical record.[1,2] They contribute to the readability of the record in an emergency situation and consequently to the safety of the patient. Well maintained and comprehensive problem lists ensure continuity of care and reduce redundancy and cost. The Institute of Medicine report on the Computerized Patient Record recommends that: "The CPR contains a problem list that clearly delineates the patient's clinical problems and the current status of each".[3]

An appropriately granular problem list is a valuable foundation for any automated or semi-automated clinical trial eligibility and recruitment software. In order to build a decision support tool to automatically monitor or search the electronic medical record for patient eligibility in clinical trials the medical record should list all significant medical diagnoses and conditions. At the present state of our institution's medical record system, some medical conditions are available as structured data but far more are "buried" in narrative text notes. Our problem list application will "unearth" the "buried" conditions and make them discoverable as semi-structured entries in the medical record under the "Problem List" heading. Collecting all the significant medical conditions in one easy to digest list will make feasible to build a clinical trial eligibility assessment tool and serve other purposes in the future.

The Joint Commission on Accreditation of Health Care Organizations (JCAHO) requires a problem list (Summary List) for all ambulatory patients who are participating in continuous care process to facilitate the assessment of the patients' status in emergency situations.[4] The lists assist in continuity of care and increase care quality and safety. However, the intended consequences of the problem list will be present only if the lists can be trusted to be comprehensive and regularly maintained. List entries that are not generated at the level of physicians' required granularity (for example using aggregating billing code) or are not regularly updated cannot be trusted in the care giving process. Similarly, lists that include the same entries multiple times are too redundant and are less likely to be used by physicians.

Despite the policy requirements, and despite the potential benefits to clinical care and research physicians rarely create or maintain problem lists. As Meystre and Haug reported the sensitivity of physician created and maintained non-automated list is about nine percent.[5] Structured data stored in the records and billing information was used first to automatically generate the problem list.[6] It was quickly realized that UMLS and SNOMED perform better in capturing the clinical content of the problem lists than administrative or billing codes.[7] The physician's note is the most authoritative representation of the physician's cognitive process. Clinical or translational research that needs to capture the result of that cognitive process is best served by a problem list based on the physician's note.

Recent maturing of the UMLS knowledge base and publicly available Natural Language Processing

(NLP) software components created new opportunities for realizing the automated problem list. Our goal is to integrate public domain NLP tools into internally developed software components for an automated electronic problem list. The problem list will be the foundation of an informatics infrastructure to support translational and clinical research and to comply with JCAHO requirements in our institution. In the paper we report the early results evaluating the proof of concept application as a milestone of our development process.

Background

Using NLP to automate problem lists is an emerging technology. There were numerous publications in the 1990's to evaluate and show the effectiveness of various coding schemes to capture the clinical concepts stored in medical records.[6, 7] However, these efforts centered on either manual coding of documents or using computerized pick lists, dictionaries and search algorithms.[8] Since 2000 research teams have started to fully automate or semi-automate (with humans as final arbiter) the coding process based on NLP of biomedical documents. The major efforts to automate medical diagnoses indexing that are most related to our work include MedLEE at Columbia, MetaMap at the National Library of Medicine, and Meystre's and Haug's work at Utah.[9, 10, 11, 12]

We have several goals for the research reported here. First is to extend the work of the Utah team beyond their limited set of 80 medical problems in the cardiovascular domain.[11] Our second goal is to test the hypothesis that high sensitivity and precision results similar to theirs could be achieved in a different organizational setting. Finally, we wanted to evaluate the performance of our proof of concept application.

Methods

We used Meystre's and Haug's design to develop our backend NLP application as described in their publications.[11, 12] We followed their blueprint of pre-processing, problems detection using MetaMap Transfer (MMTx 2.4.C release) and unmodified UMLS dataset (2006 strict version). Finally we post-processed the MMTx machine output file and harvested UMLS Concept Unique Identifier (CUI) codes and corresponding phrases as returned by MetaMap. We used only CUIs that belonged to semantic types that were part of the "Disorders" semantic group ("Finding" excluded). We also collected "Therapeutic or Preventive Procedure", in total we used 12 semantic types.

In the "training" notes the investigators studied three sections (Impression, Plan and Problems). We focused our problem list extraction on these three sections. Our conclusion was that negation detection was not a major issue for the three sections. Consequently, we decided to deviate from the Utah's team design for the proof of concept application and did not include negation detection in the system for this study. We understood that this would probably reduce our precision/specificity and we intend to implement the NegEx algorithm in the production quality application.[13]

Our system harvested the CUIs and corresponding phrases as they were returned by MetaMap for the three sections (Impression, Plan and Problems) of the notes. To generate the final problem list multiple entries were collapsed based on having the same CUIs. Finally, we used a list of "filtering phrases" to reduce noise in the problem list. To build this list, we extracted phrases that were not medical problems but were returned by MetaMap. For example, the phrase "thinking" was returned by MetaMap as "Mental or Behavioral Disorder" semantic type.

One investigator (IS) compiled the list of "filtering phrases" from all 100 notes and phrases represented on the list were deleted from the proposed problem lists. We used a "rolling" method to build the filter list. We counted the number of new filter phrases added per five notes. Our hypothesis is that the growth of the list will slow as notes are processed.

We understand that using the 100 "test" notes for building the filter list to eliminate "noise" in the final problem list makes our precision value incomparable to the Utah team's. However, our system is a proof of concept application and we believe that the filter list will stabilize after processing a larger number of notes (per medical specialty) so in the production quality application we will have a finite and definitive filter list. We wanted to test if the "rolling" method of building filter list would work when we move to the pilot and production phase of development. The "rolling" method of building the filter list does not affect our sensitivity results. The sensitivity value is fully comparable to other studies.

To build the proof of concept application we used ten General Medicine ambulatory care notes as "training" notes. We asked two physicians (BA, GF) to create problem lists based on each note. The ten notes were not included in the 100 "test" notes that were used for evaluation of the system performance.

In order to accomplish our first goal of testing NLP based problem list automation capabilities beyond the set of Utah's 80 medical problems, we designed our evaluation study to closely follow the everyday work process of physicians. In clinical practice physicians have unlimited number of medical diagnoses to choose from when they are generating the problem list. The physicians also have flexibility in their style and vocabulary to use when describing diagnoses and therapeutic processes. Consequently, we did not limit either our system or our annotators to a predefined set of diagnoses or vocabulary.

To generate the gold standard for evaluation we randomly selected 100 cardiology clinic ambulatory progress notes from one week's worth of notes. We had IRB approval and notes were de-identified before being used in this study. Three physicians with General Internal medicine board certification were asked to create problem lists for the notes, individually (BA, GF, MS). The physicians received only sections of the "test" notes (Impression, Plan and Problems). Most of the notes included only two of the above section headings so the physicians received only the corresponding sections of each note in Word files.

The physicians used implicit review technique where reviewers rely only on their knowledge and beliefs to make judgments. The Utah team used explicit review as they were looking for a predefined set of disease categories. We believed that applying implicit review (listing all relevant problems) for evaluating our system that replicated their design could add further support to their claim of the design's usefulness (if we find high sensitivity and precision) because the implicit review is a good approximation of the care giving workflow process. One of the investigators (IS) generated the gold standard based on majority decision of the three physicians' collated lists.

None of the physicians participated actively in the development of the system beyond generating the problem lists manually. In general we tried to follow Friedman's recommendations for evaluating Natural Language Processors.[14] None of the investigators or the developers had access to the 100 "test" notes before the development phase of the proof of concept application was over.

To measure the gold standard's reliability we used positive specific agreement (p_{pos}) as it is described by Hripcsak and Rothschild.[15, 16] We believe that in our case p_{pos} is a better measure of inter rater agreement than Kappa because the universe of terms to choose from is both undefined (from a practical point of view) and very large.

Agreement between the system and gold standard did not require agreement for the qualifiers of the gold standard. For example, the most frequent qualifier was "history of" and our proof of concept application did not look for qualifiers but "raw" medical problems only. That is if the physicians listed "history of Diabetes" in their problem lists, and the system returned "Diabetes" without the qualifier "history of" we accepted the system's result as true positive. In future research we will work on implementing qualifier detection but it could be a highly challenging task.

In addition to explicit problem list entries, physician annotators also extracted implicit medical problems when reading the notes. We defined implicit medical problems similarly to Wicentowski and Sydes.[17] That is we considered an entry potentially implicit if it could be inferred from the context of the note in the absence of explicit mentioning of the medical problem. The physicians were instructed to present the implicit entries separately from the "regular" explicit problem list entries and if present the implicit entries were not included in the gold standard for the study.

Results

After eliminating typographic or case differences among physicians the gold standard included 154 different problem list entries. 54 of the problem lists' entries appeared more than once and 100 entries were listed only once in the gold standard. 12 medical problems covered 50 percent of the entries in the gold standard. Hyperlipidemia, Ischemic Heart Disease and Hypertension covered a third of the entries. Altogether there were 442 problem list entries in the gold standard for the 100 notes.

Out of the 80 medical diagnoses listed by Meystre's and Haug's study our gold standard included 35 (44%).[11] Out of the 37 most frequent diagnoses at Intermountain Health Care, our 100 notes included 24 (65%).

The positive specific agreement (p_{pos}) between physicians ranged from 0.94 to 0.96. The agreement between the individual physicians and the system's findings for medical problems that were made into the gold standard ranged from 0.92 to 0.93. Even if entries of the manual problem lists that did not make it into the gold standard are included in the p_{pos} calculation, the agreement between physicians ranges from 0.91 to 0.93. There were 56 entries that only one physician found. However, the 56 entries were not part of the gold standard.

Our automated proof of concept system found 389 gold standard problem list entries or their equivalent phrase (88 percent sensitivity). After eliminating duplicates and using the "filtering phrases" from the filter list the system produced 591 "findings" for the 100 notes. The precision of our proof of concept application was 66 percent.

We tested if our "rolling" method of building the filtering phrases is accompanied with a decreasing trend of accumulating phrases from the notes per five-note subgroups. If our assumption is correct then as we progress building the filter list the number of phrases added will decrease. To find contrast we grouped the fifty notes we processed first into a separate group from the second set of fifty notes and compared the means of number of filtering phrases added, per five notes. The result of the t-test showed that in case of the first fifty notes we added 4.2 filtering phrases per five notes versus to 2.3 phrases in case of the second fifty notes. The p-value for the t-statistics was 0.05.

We drilled further down into the distribution of system detected problem list entries. In the first step we created a variable to represent ambiguity for gold standard medical problems. The value of the new variable reflected ambiguity for the gold standard entry if any of the three physicians missed the particular medical problem. We found 58 entries (13 percent of the total gold standard entries) where one physician did not include the problem on her/his manual list. In the next step we analyzed the distribution of system detected medical problems by crosstabulating the system's findings against the new ambiguity variable. The results of the analysis showed 27.28 Chi-square statistics on a $p<0.0001$ value and 0.062 Goodman and Kruskal tau statistics for directional measures on a $p<0.0001$ value.

Ambiguity reduced the overall sensitivity. For ambiguous gold standard entries the system's sensitivity was 67 percent. For non-ambiguous entries the system sensitivity was 91 percent. When ambiguous and non-ambiguous gold standard entries were combined, sensitivity was 88 percent.

The three physicians extracted 13, 18, and 28 implicit medical problems specifically. Altogether 46 implicit entries were extracted and ten of them were found by more than one physician. That is ten entries would qualify to be part of the gold standard of medical problems based on majority decision.

Discussion

We fulfilled the goal of extending Meystre's and Haug's work beyond the set of 80 medical problems in the system evaluation. Our gold standard had 154 different problem list entries with no limitation for physician annotators or the system to extract information from the notes. As the descriptive statistics indicated we had a good representation of the Utah team's medical problem set with 65 percent of their most important medical problems also appearing in our gold standard of 100 notes. The agreement between physicians for entries of the gold standard and the agreement between individual physicians and the system's findings were high.

Based on the representativeness of Utah's medical problem set in our gold standard, the high reliability of our gold standard, and the wide variety of discoverable problems in our notes, we can conclude that the 88 percent sensitivity value provides valid support to the Utah team's findings. The 88 percent sensitivity is high and is a good match of their sensitivity findings but in our case it was achieved on an unlimited set of problems. We also fulfilled the goal of replicating their study in a different institutional setting and our results contributed to the generalizability of their claim of the usefulness of MetaMap and UMLS in problem list automation.

With the results of the t-test for building the "filtering phrase" list we also proved that it is realistic to get a finite set of filter phrases. We can expect to succeed generating the (sub)specialty specific filter set to improve precision of our planned production application.

Our study provided interesting results about the relationship of ambiguous gold standard entries and misses of the automated system and the potential role of implicit medical problems in evaluation of automated systems. There was a relationship between the ambiguity of the particular gold standard entry and system misses for that entry that could not be explained by chance only. The system missed entries more likely if the entry was missed by any of the three physicians, too. The relationship was not strong, the ambiguity reduced the error rate of the system misses only by 6.2 percent but it was statistically significant.

The extracted implicit medical problems had the potential of adding 10 percent extra problem list candidates that would not be found by an automated system unless the system accounts for inferences. Consequently, implicit entries and ambiguities could reduce the reported sensitivity of automated systems. We feel that these findings justify further research efforts in this area.

The limitations of our study included sampling, that was not truly random. We focused only on one subspecialty and by that we decreased generalizability of the results. Same set of documents was used to test the system and build the list of “filtering phrases”. We used a limited set of notes (100) that is acceptable for proof of concept applications but clearly more notes and a new set of notes will need to be studied.

We did not use negation detection and although it was not a major issue it certainly reduced our precision. We focused only on 12 semantic types and that reduced our sensitivity. Although we demonstrated that there is a statistically significant decreasing trend in the accumulation of “filtering phrases”, the filtering list did not stabilize, yet. We worked only on the back end NLP system and did not test integration with workflow or physician acceptance.

In future work we will address the limitations mentioned above. We will also extend our system to include a front end application and run usability studies. Furthermore, we plan to investigate the consequences of ambiguity and inferences in problem list automation and system evaluation.

Conclusion

MetaMap and UMLS are readily applicable to automate the problem list with high accuracy (88 percent sensitivity and 66 percent precision in a proof of concept application). Ambiguity and implicit medical problems in the clinical notes have consequences for performance evaluation of automated systems.

Acknowledgement

We would greatly appreciate guidance and suggestions of Dr. Peter Haug in the early stages of our research. This research was funded in part by NLM training grant: NIH NLM #T15 LM07442.

References

1. Worthley LI. A system-structured medical record for intensive care patient documentation. Critical Care Medicine 1975;3:188-191.
2. Weed LL. Medical records that guide and teach. N Engl J Med. 1968;278:593-600.
3. Institute of Medicine (US): Committee on Improving the Patient Record. In the computer-based patient record: an essential technology for health care. Revised edition. Edited by: Dick RS, Steen EB, Detmer DE. Washington, DC: National Academy Press; 1997.
4. Joint Commission on Accreditation of Healthcare Organizations. Standard I.M. 6.40. http://www.jointcommission.org/NR/rdonlyres/DD06404B-66AE-4C7F-9D67-6419734651D7/0/06_cah_im.pdf Accessed at: January 27, 2008.
5. Meystre S, Haug PJ. Randomized controlled trial of an automated problem list with improved sensitivity. Int J Med Inform. 2008;doi:10.1016/j.ijmedinf.2007.12.001.
6. Payne TH, Murphy GR, Salazar AA. How well does ICD9 represent phrases used in the medical record problem list? Proc Annu Symp Comput Appl Med Care. 1992;:654-657.
7. Campbell JR, Payne TH. A comparison of four schemes for codifications of problem lists. Proc Annu Symp Comput Appl Med Care. 1994;:201-205.
8. Wang SJ, Bates DW, Chueh HC, et al. Automated coded ambulatory problem lists: evaluation of a vocabulary and a data entry tool. Int J Med Inform. 2003;72:17-28.
9. 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.
10. Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Proc AMIA Symp. 2001;:17-21.
11. Meystre S, Haug PJ. Automation of a problem list using natural language processing. BMC Med Inform Decis Mak. 2005; 5:30.
12. Meystre S, Haug PJ. Natural language processing to extract medical problems form electronic clinical documents: performance evaluation. J Biomed Inform. 2006;39:589-599.
13. Chapman WW, Bridewell W, Hanbury P, et al. A simple algorithm for identifying negated findings and diseases in discharge summaries. J Biomed Inform. 2001;34:301-310.
14. Friedman C, Hripcsak G. Evaluating natural language processors in the clinical domain. Methods Inf Med. 1998;37:334-344.
15. Hripcsak G, Rothschild AS. Agreement, the F-measure, and reliability in information retrieval. J Am Med Inform Assoc. 2005;12:296-298.
16. Rothschild AS, Lehmann HP, Hripcsak G. Inter-rater agreement in physician-coded problem lists. AMIA Annu Symp Proc. 2005;:644-648.
17. Wicentowski R, Sydes MR. Using implicit information to identify smoking status in smoke-blind medical discharge summaries. J Am Med Inform Assoc. 2008;15:29-31.