Published in final edited form as:

Methods Inf Med. 2015 ; 54(2): 164–170. doi:10.3414/ME13-01-0130.

Adaptive Semantic Tag Mining from Heterogeneous Clinical Research Texts

Tianyong Hao, PhD^{a,b,*} and Chunhua Weng, PhD^a

^aDepartment of Biomedical Informatics, Columbia University, New York, NY, United States

^bKey Laboratory of Language Engineering and Computing of Guangdong Province, Guangdong University of Foreign Studies, Guangzhou, China

Summary

Objectives—To develop an adaptive approach to mine frequent semantic tags (FSTs) from heterogeneous clinical research texts.

Methods—We develop a “plug-n-play” framework that integrates replaceable unsupervised kernel algorithms with formatting, functional, and utility wrappers for FST mining. Temporal information identification and semantic equivalence detection were two example functional wrappers. We first compared this approach's recall and efficiency for mining FSTs from ClinicalTrials.gov to that of a recently published tag-mining algorithm. Then we assessed this approach's adaptability to two other types of clinical research texts: clinical data requests and clinical trial protocols, by comparing the prevalence trends of FSTs across three texts.

Results—Our approach increased the average recall and speed by 12.8% and 47.02% respectively upon the baseline when mining FSTs from ClinicalTrials.gov, and maintained an overlap in relevant FSTs with the baseline ranging between 76.9% and 100% for varying FST frequency thresholds. The FSTs saturated when the data size reached 200 documents. Consistent trends in the prevalence of FST were observed across the three texts as the data size or frequency threshold changed.

Conclusions—This paper contributes an adaptive tag-mining framework that is scalable and adaptable without sacrificing its recall. This component-based architectural design can be potentially generalizable to improve the adaptability of other clinical text mining methods.

Keywords

Medical Informatics; Text Mining; Clinical Trials; Semantic Tags; Component-based Architecture

1. Introduction

Adapting text-mining algorithms to various domains is full of challenges [1]. Extra cost and development are often inevitable when applying text-mining algorithms developed for one type of text to another. Therefore, adaptability, one of the key features for widely adopted

*: Corresponding Author: Chunhua Weng, Ph.D., Associate Professor, Department of Biomedical Informatics, Columbia University, 622 W 168 Street, PH-20, New York, NY, 10032, USA, cw2384@columbia.edu.

software systems [2], has been recognized as a prerequisite for enabling the reuse and portability of text-mining algorithms [3]. Adaptive algorithms have been developed for various domains [3-7], e.g., energy management [8], global optimization [9], and health care [10]. This paper aims to explore an adaptive design for extracting common data elements (CDEs) from a variety of clinical research texts without compromising the efficiency and recall that a specialized CDE-mining algorithm designed for a particular type of clinical research text can achieve.

Standardization of CDEs is a high-priority task for clinical research in different disease domains [11, 12]. CDEs have been recommended to standardize data collection for clinical trials, patient registries, and other types of human subject research at different phases in the clinical research life cycle, including eligibility screening and research data reporting [12, 13]. CDEs are useful for (1) enabling data sharing and knowledge reuse, and easing clinical research designs [13, 14]; (2) promoting standard data collection, e.g., for eligibility determination [15, 16]; (3) facilitating scalable systematic reviews or meta-analyses [17, 18]; and (4) supporting the development of evidenced-based guidelines [19]. Workshops have been held to advance the use of CDEs for collaboration, data exchange, and research advancement [20]. The Clinical Data Interchange Standards Consortium (CDISC) has recommended pan-disease or cancer-specific CDEs for clinical trial eligibility determination [21].

However, current CDE development heavily relies on the manual work of domain experts and hence is time-consuming and costly. For example, it took multiple international experts more than one year and numerous teleconferences to complete one CDE definition project led by the National Institute of Neurological Disorders and Stroke in 2009 [19, 22, 23]. One potential solution to accelerate the development of CDEs is to leverage informatics methods to augment domain experts during CDE development. Our previous study has shown that recommendation of CDE candidates to domain experts can significantly save the effort on CDE identification so that domain experts can focus on defining the detailed data model for each CDE rather than searching for CDEs and harmonizing their different semantic representations [24].

Hereinafter we refer to such a CDE candidate as a Frequent Semantic Tag (FST). A Semantic Tag (ST) is a concept associated with syntactically different but semantically equivalent textual strings. For example, “p-450aldo” and “aldosterone-synthesizing enzyme” are different semantic representations for the same ST, “cyp11b2 protein human”, whose semantic type is “enzyme” according to The Unified Medical Language System (UMLS) [25]. An FST is a ST whose occurrence frequency exceeds a specified threshold, e.g., occurring in 5% of all the clinical trials on ClinicalTrials.gov. In this context, FST is exchangeable with *common data element candidate* [24] and *tag* [26-28] used previously. FSTs have served a range of applications. For example, FSTs mined from the eligibility criteria text of clinical trials have provided effective support for the indexing, search, and clustering of clinical trials [26, 29] and for dynamic filtering of clinical trial search results [28]. Analyses of FSTs used in different diseases have also informed knowledge discovery for disease relatedness [30].

Semi- and fully-automated methods for FST mining have been proposed, such as [23, 31, 32]. Nevertheless, most of them are developed for a particular clinical text format and have not been tested among different clinical texts, such as a recently published tag mining method [26] (“BaselineM” hereinafter). BaselineM specializes in processing eligibility criteria text from ClinicalTrials.gov and generates FSTs for clinical trial indexing and search [24, 26]. This paper describes an extension to BaselineM using an adaptive text-mining framework.

2. Materials and Methods

2.1. Data Preparation

We considered three representative clinical research texts: de-identified clinical data requests submitted to the Clinical Data Warehouse (CDW) of our institution, the New York Presbyterian Hospital, clinical trial summaries (free-text eligibility criteria section) from ClinicalTrials.gov, and full-text clinical trial protocols. A sample of clinical data requests is “*positive Candida all species and positive yeast cultures from all sites in our patient population with MRN, date of cx, site of cx, organism isolated time period XXX - XXX*”. A sample of clinical trial eligibility criteria summary from ClinicalTrials.gov is shown in the “Eligibility Criteria” section of trial NCT00955773 (<http://clinicaltrials.gov/ct2/show/NCT00955773>). The three texts differ in their language formality, context-dependency, and disease-specificity, based on researchers' judgment on the sample texts. For example, a typical clinical data request is context-dependent and can contain informal words (e.g., “I m”), abbreviations (e.g., CT), and misspelling (e.g., “*wehre*”). Appendix Table 1 shows detailed comparisons of these texts with regard to their average counts of words, temporal expressions, and abbreviations.

We retrieved all 145,745 clinical trials from ClinicalTrials.gov on May 17, 2013. After excluding the trials whose eligibility criteria were missing or inadequate (i.e., with only the phrase “please contact site for information”), 142,948 trials were retrained as our testing dataset. We identified 2,770,746 sentences, from which 5,508,491 semantic tags (459,936 unique) were extracted (on average 38.5 semantic tags per trial) using the syntax-based kernel for leveraging its feature of finding self-contained noun phrases. We randomly selected 500 clinical trials, 500 paragraphs from 12 proprietary clinical trial protocols, and 500 clinical data requests for methodology development and testing.

2.2 The Kernel-Wrapper Framework

Our approach is based on a kernel-wrapper framework, as shown in Appendix Figure 1. The framework consists of replaceable and extensible kernels, wrappers, and knowledge bases. The kernels are tag-mining algorithms. The wrappers process different text formats and supply functionalities and utilities. Specifically, text-format wrappers focus on processing the input and output of different text formats, while functional wrappers provide functionalities, such as semantic equivalence detection or temporal processing. Wrapper utilities acquire data from heterogeneous data sources, such as the Web or a database. The wrappers are organized as three layers surrounding kernels. At the outmost layer are text-format wrappers, in the middle are functional wrappers, and at the innermost layer are

wrapper utilities. A text-format wrapper can invoke multiple functional wrappers, and vice versa. Each functional wrapper provides a unique functionality. Each utility or kernel is an independent module that can be plugged into wrappers for reuse. The peripheral knowledge modules supply decision support rules, patterns, controlled vocabularies, thesaurus, and knowledge alike to wrappers and kernels.

The framework differs from configurations that are commonly used in software application programming interfaces (APIs), such as the APIs of MetaMap [33]. The framework goes beyond routine configurations by enabling flexible reassembling of task-dependent while standalone functionalities, such as our functional wrappers. Another major difference is that this reusable framework enables users to define or extend the functionality. For example, a user can specify the output content (e.g., capturing terms unmapped to UMLS for analyzing potentially missed important concepts) for their tasks rather than selecting fixed candidate output formats.

Our overall FST mining procedure includes four steps: 1) candidate tag extraction, 2) ST normalization, 3) FST filtering, and 4) UMLS semantic type assignment. With preprocessed clinical research texts being input, the algorithm recognizes UMLS concepts as tag candidates, which are normalized as ST candidates. Those ST candidates whose frequencies exceed a predefined threshold are retained as FSTs, which are assigned with a distinctive UMLS semantic types and sent to text-format wrappers for formatting. We adopted two kernels for FST mining: i.e., *n*-gram-based and syntax-based (using NLTK [34]) algorithms. The *n*-gram-based kernel used a widely used *n*-gram model [35] for candidate tag extraction, which is used in BaselineM [26]. The syntax-based algorithm performs sentence boundary detection, part-of-speech (POS) tagging, key phrase extracting (e.g., using an open library with NLTK [34]), and stop-word removing. It differs from the *n*-gram-based algorithm during candidate tag extraction by not containing C-value-based filtering [36] for removing nested FSTs. To improve the performance of FST mining algorithm, we further added temporal expression identification and semantic equivalence detection, whose details are provided below.

2.3 Temporal Expression Identification

Temporal expressions commonly exist in clinical research texts [37, 38]. For example, according to our informal study of randomly selected 100 clinical data requests submitted to the CDW, 61% of the requests collectively contained a total of 155 temporal expressions [39]. However, only 14.4% were detected using BaselineM. Therefore, we embedded a temporal information processing wrapper - TEXer, whose details have been previously reported [39]. It employed heuristic rules for pattern matching. An example rule was “(?<!(\\w\\d|</>|/))(' + month + r' \\d{1,2}(-|/| -|/|)\\d{1,2}(/|,|/|)\\d{2,4})(?!\\w\\d|</>|/))”, where “month” was a set of standard month names or abbreviations, identifying the format such as “December 1-7, 2003”. The heuristic rules were applied first to recognize temporal expression candidates. Afterwards validated text patterns (e.g., “between <TI> and <TI>”) were matched with texts to recognize temporal candidates missed by the heuristic rules. These text patterns were learned from the contextual information at sentence level. TEXer

normalized the extracted temporal expressions using TimeML, a widely-adopted temporal markup language [40, 41].

2.4 Semantic Equivalence Detection

Clinical research texts contain semantically equivalent but lexically different words or phrases. For example, “icd9”, “icd-9”, “icd9s”, “516.3”, and “428.xx” are expressions of ICD-9 codes in the clinical data requests to our CDW. From our manual investigation on randomly selected 100 data requests, there were a total of 147 expressions including ICD-9 codes. However, most of these codes were in numerical format and failed to be detected by BaselineM. To address the problem, we first leveraged the concept-oriented design [42] of UMLS Metathesaurus, where each concept has exactly one meaning and links all atoms that have the same meaning [25], for semantic equivalence detection. Then we extended this design with an important feature currently unavailable in UMLS. This feature is to detect the similarities among different representations for ICD, CPT, ADT codes and trial registry numbers, which are frequently available in clinical research texts. We first defined regular expressions and rules to match texts for identifying semantic equivalent expressions based on distributional similarity [43] and validated the results using the commonly used measures for information retrieval, i.e., precision, recall, and F1 [44]. Identified equivalent expressions were tagged with users preferred annotation markers. For example, Appendix Table 2 lists 12 expressions for ICD-9 codes.

2.5 Adaptability Evaluation Design

We first used ClinicalTrial.gov for FST mining and compared the resulting FSTs' overlap, precision and recall, at different frequency thresholds, against the BaselineM. One medical doctor and one informatics researcher reviewed the results from the two methods and reached consensus afterwards. They examined if the FSTs were meaningful concepts or represent certain aspects of disease, treatment, lab test, etc., by refereeing to the general common data elements for all diseases recommended by the National Institute of Neurological Disorders and Stroke [45]. Precision was calculated by the count of correct FSTs divided by the total extracted FSTs. Recall was calculated by the percentage of correct FSTs deemed by the previous reference standard used in the BaselineM study that can be extracted by our method. Adaptability measures whether a method is capable of accommodating variable resources and stakeholders' changing requirements [2]. A big barrier we faced for adaptability evaluation was that we only had the reference standard for the clinical trials text, but not for the other two texts, i.e., clinical research protocols and clinical data requests. Currently there is no well-accepted definition for FSTs; therefore, the determination of the FSTs remains a contextual, task-dependent decision according to the feedback we collected from some medical doctors. Therefore, following the strategies used in [46, 47], we resorted to comparing the trends (1) in the percentage of retrieved relevant FSTs over the maximum number of potential FSTs for a certain data size and (2) in the computational speed (i.e., the data size divided by the execution time) for various data sizes across the three text types.

3. Results

Appendix Figure 2 shows the graphic user interface of our method that enable users to set up data paths, select wrappers and kernel, and assign a group of parameters and options.

3.1. Adaptability and scalability

Our approach using the n -gram-based kernel extracted between 73 and 2957 relevant FSTs with corresponding frequency threshold ranging from 0.2% to 2% (i.e., 10 times increase), as shown in Figure 1. The change patterns of the relevant FSTs on the three text formats were consistent. The top 100 relevant FSTs for each type of text were shown as Appendix Table 3.

Then we set the frequency threshold to be 1.8% so that the resulting FSTs were manageable for manual review. We also randomly selected between 50 and 500 documents for three times for each text format for dataset generation in order to reduce random FST variations. The FST prevalence appeared to be relatively stable when the data size was greater than 200 documents for all the text formats. Therefore, based on the data sets (> 200 documents), we calculated the percentage of the correct FSTs, as determined by manual review by one medical doctor and one informatics researcher as mentioned in evaluation design, over the corresponding maximum retrieved relevant FSTs in each round for each text format. As shown in Figure 2, the percentages varied slightly for all the three texts, with the minimum variances among them being $6.5\% \pm 2.3\%$ and the maximum variances being $9.8\% \pm 4.2\%$, at a 95% confidence interval.

We measured FST mining speed trends as the data size increased from 1 to 7,614 sentences for clinical protocol texts. Running our method using both n -gram-based and syntax-based kernel except input/output handling, we recorded the execution time with the same computer environment (CPU: Intel Core 2 Duo E8400; RAM: 4GB; OS: 32-bit Win7). Because our kernel reused certain intermediate results (e.g., POS tagged phrases) within a FST mining task and across similar tasks, we compared three strategies: 1) reuse within a task, 2) reuse within and across tasks, and 3) reference, with the kernel without any reuse being the baseline. From the result shown in Appendix Figure 3, the running time of “reuse within a task” was improved (dashed lines) when compared with the baseline (dotted lines), e.g., 46.7s by “with reuse” as opposed to 70.2s without it when the data size was 3500 sentences, leading to 47.02% execution time improvement. A 95% confidence interval of the improvement upon the baseline was $28.6\% \pm 4.2\%$. The “reuse within and across tasks” further presented execution time improvement (i.e., solid lines) for the all data sizes, e.g., 5.1s used compared with 46.7s by “with reuse” when the data size was 3500 sentences, leading to 89.1% execution time improvement. A 95% confidence interval of the improvement compared to “reuse within a task” was $66.5\% \pm 14.3\%$. The execution time first grew as the data size increased from 0 to 1500 sentences, and dropped afterwards. After 4000 sentences, the computation time increased slightly (4.2s for 4000 sentences and 5.5s for 6000 sentences) despite the fast increase in data size. The speed kept increasing as data size increases for our method within and across FST mining tasks.

3.2. Comparison with BaselineM

Both our approach and the BaselineM used the n -gram-based kernel to extract FSTs from ClinicalTrials.gov. As shown in Table 1, our approach retrieved more relevant FSTs than BaselineM under all frequency thresholds. The counts of overlapping FSTs increased as the threshold increased, and stabilized at 100% when the threshold was equal to or larger than 8%. All the FSTs identified by each algorithm were deemed meaningful based on the consensus of one medical doctor and one informatics researcher, though several FSTs with identical meaning were detected. Therefore, the two algorithms had the same precision, but the proposed approach retrieved more relevant FSTs than BaselineM at all thresholds, with an average improvement of 12.8%. Since the count of identified FSTs was different for each threshold, recall improvement ranged from 6.4% to 21.9%.

For the 500 randomly selected clinical trial summaries, 7,932 sentences were extracted, with 12.84 words per sentence on average. We calculated the percentages of the n -grams ($n=8$) that were FSTs, as shown in Appendix Table 4. Using the frequency threshold 1.8%, we extracted 1,438 unique frequent n -grams, in which 1,201 (83.5%) were FSTs. The FSTs covered more than 80% of all n -grams and 74%-79% of all the occurrences. Of these unique frequent n -grams, 60.1% contained more than one word. Then we compared the FSTs extracted from 500 clinical data requests using our approach and BaselineM, respectively, using the frequency threshold 2%. Our approach extracted 77 FSTs, while BaselineM extracted 63 FSTs, with an overlap rate at 87.3%. We aligned the top 20 FSTs from both results for comparison (Appendix Figure 4). The FST “data in time” was identified with a frequency of 58.6% using our approach, indicating 58.6% requests contain temporal expressions. In contrast, only 5.8% “data in time” were detected and another 6.2% were normalized as “month” by BaselineM.

We compared the efficiency of the two methods on the same 500 clinical data requests. We tested our method using n -gram-based kernel and BaselineM on the same computer with the same software environment. As ways for reading input documents were different, we did not count time for file operation parts (e.g., data loading and result output) for both methods. Each method ran 5 rounds and the average time was calculated. The results are shown in Appendix Table 5. Our approach shortened the execution time by 8.5%, from 22.3s to 20.4s on average.

4. Discussion

We presented an adaptive kernel-wrapper framework for mining FSTs from heterogeneous clinical research texts. It outperformed the baseline and demonstrated satisfactory adaptability and scalability to different texts. The improvement was largely due to the refinement in the framework on word tokenization, sentence splitting, and phrase identification strategies. This multi-layer framework is extensible. All modules including kernel algorithms can be customized for different tasks. For example, the candidate tag generation function in current the syntax-based kernel can be replaced by the Stanford parser [48] or MetaMap [33]. Functional wrapper can also be used independently. Our study also demonstrates that by decomposing a text-mining method into modular functionalities at different logical layers and integrating them in a “plug-and-play” framework, we can

achieve algorithm adaptability without significant redesign while keeping flexible module extensions and upgrading. cTAKES [49] and Unstructured Information Management Architecture (UIMA) [50] both use such a component-based software architecture, although cTAKES works primarily with clinical notes and UIMA is for general-purpose text processing. Therefore, the framework presented in this study is one of the few open-architecture for extracting FSTs from clinical research texts.

There are three major limitations in this study. First, word stemming can help detect strings of the same meaning to get more precise frequency ranking of FST, but may impair string matching because the UMLS Metathesaurus contains a large number of non-stemmed terms. We therefore have not applied any stemming function but prefer to use an adaptive stemming algorithm in accordance with the UMLS term mapping in the future. Second, the rule-based semantic equivalence detection method does not include all possible rules and can be further improved. Without domain-specific knowledge, it is hard to identify all equivalent cases. Moreover, each newly defined rule need to be evaluated towards a gold standard through human annotation, which is labor intensive and time consuming. Third, our adaptability evaluation lacks a proper reference standard for FSTs from clinical research protocols and clinical data requests. This is due to the difficulty of identifying and accessing a CDE expert since currently CDEs have not been widely adopted except for being recommended by NIH. We found most of the FSTs were noteworthy to be CDE candidates, e.g., “electrocardiogram”, “gravidity or nursing therapy”, “skin carcinoma”, “medical history of hypersensitivity”, and “depressed mood”. Ideally a domain expert, such as someone from CDISC, could evaluate these FSTs and create a reference standard. Therefore, in the future we plan to partner with CDISC researchers to evaluate and disseminate the results from this research. In spite of these limitations, given the scarcity of semi-automated or automated methods for extracting CDEs from heterogeneous clinical research texts, this study represents an important stepping stone in this critical research area by contributing a large datasets and a piece of novel FST mining software framework to enable more researchers to join the collaborative design and evaluation of future CDE mining methods.

5. Conclusions

We developed a kernel-wrapper framework to mine frequent semantic tags from different clinical research texts. We compared the method with a recently published baseline and achieved better performance, computational efficiency, and adaptability. The methodology reported here can be potentially applied to improve the adaptability of other information extraction systems.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

This work was supported by National Library of Medicine grant **R01LM009886 (PI: Weng)**, Food and Drug Administration (FDA) grant **R24 FD004656-01 (PI: Nahm)**, and National Center for Advancing Translational

Sciences grant **UL1TR000040 (PI: Ginsberg)**. Thanks to Riccardo Miotto for his generosity on sharing the baseline algorithm used in the research. We also thank Dr. Meredith Nahm for providing the clinical trial protocols for this study and thank Dr. Alexander Rusanov for evaluating the clinical meaningfulness of the tags.

References

1. López-Paz, D.; Hernández-Lobato, JM.; Schölkopf, B. Semi-Supervised Domain Adaptation with Non-Parametric Copulas. In: Bartlett, PL., et al., editors. NIPS. 2012. p. 674-682.
2. Tarvainen, P. Adaptability Evaluation of Software Architectures; A Case Study. 31st Annual International Computer Software and Applications Conference; 2007;
3. Benveniste, A.; Metivier, M.; Priouret, P. Adaptive Algorithms and Stochastic Approximations. Springer Publishing Company, Incorporated; 2012. p. 376
4. Flora S, Tsai, A, TK.; Wenyin HS, Tang; Kap Luk, Chan. Adaptable Services for Novelty Mining. Systems and Service-Oriented Engineering. 2010; 1(2):17.
5. Xu Q, Quan Y, Yang L, He J. An adaptive algorithm for the determination of the onset and offset of muscle contraction by EMG signal processing. IEEE Trans Neural Syst Rehabil Eng. 2013; 21(1): 65–73. [PubMed: 23193462]
6. Sandin M, Ali A, Hansson K, Månsson O, Andreasson E, Resjö S, Levander F. An adaptive alignment algorithm for quality-controlled label-free LC-MS. Mol Cell Proteomics. 2013; 12(5): 1407–20. [PubMed: 23306530]
7. Li YR, Shen L, Suter BW. Adaptive inpainting algorithm based on DCT induced wavelet regularization. IEEE Trans Image Process. 2013; 22(2):752–63. [PubMed: 23060331]
8. Alippi C, Anastasi G, Francesco MD, Roveri M. An Adaptive Sampling Algorithm for Effective Energy Management in Wireless Sensor Networks With Energy-Hungry Sensors. IEEE Transactions on Instrumentation and Measurement. 2010; 59(2):335–344.
9. Das S, Mandal A, Mukherjee R. An Adaptive Differential Evolution Algorithm for Global Optimization in Dynamic Environments. IEEE Trans Cybern. 2013
10. Ellis B. Complexity in practice: understanding primary care as a complex adaptive system. Inform Prim Care. 2010; 18(2):135–40. [PubMed: 21078236]
11. Maas AI, Harrison-Felix CL, Menon D, Adelson PD, Balkin T, Bullock R, Engel DC, Gordon W, Orman JL, Lew HL, Robertson C, Temkin N, Valadka A, Verfaellie M, Wainwright M, Wright DW, Schwab K. Common data elements for traumatic brain injury: recommendations from the interagency working group on demographics and clinical assessment. Arch Phys Med Rehabil. 2010; 91(11):1641–9. [PubMed: 21044707]
12. Nadkarni PM, Brandt CA. The Common Data Elements for Cancer Research: Remarks on Functions and Structure. Methods of Information in Medicine. 2006; 45(6):594–601. [PubMed: 17149500]
13. Gennari JH, Sklar D, Silva J. Cross-tool communication: from protocol authoring to eligibility determination. Proc AMIA Symp. 2001:199–203. [PubMed: 11825180]
14. Lynch DR, Pandolfo M, Schulz JB, Perlman S, Delatycki MB, Payne RM, Shaddy R, Fischbeck KH, Farmer J, Kantor P, Raman SV, Hunegs L, Odenkirchen J, Miller K, Kaufmann P. Common data elements for clinical research in Friedreich's ataxia. Mov Disord. 2013; 28(2):190–5. [PubMed: 23239403]
15. Fink E, Kokku PK, Nikiforou S, Hall LO, Goldgof DB, Krischer JP. Selection of patients for clinical trials: an interactive web-based system. Artif Intell Med. 2004; 31(3):241–54. [PubMed: 15302090]
16. Respondek G, Roeber S, Kretschmar H, Troakes C, Al-Sarraj S, Gelpi E, Gaig C, Chiu WZ, van Swieten JC, Oertel WH, Höglinger GU. Accuracy of the National Institute for Neurological Disorders and Stroke/Society for Progressive Supranuclear Palsy and neuroprotection and natural history in Parkinson plus syndromes criteria for the diagnosis of progressive supranuclear palsy. Mov Disord. 2013; 28(4):504–9. [PubMed: 23436751]
17. Sim I, Olasov B, Carini S. An ontology of randomized controlled trials for evidence-based practice: content specification and evaluation using the competency decomposition method. J Biomed Inform. 2004; 37(2):108–19. [PubMed: 15120657]

18. Sim I, Olasov B, Carini S. The Trial Bank system: capturing randomized trials for evidence-based medicine. *AMIA Annu Symp Proc.* 2003
19. Loring DW, Lowenstein DH, Barbaro NM, Fureman BE, Odenkirchen J, Jacobs MP, Austin JK, Dlugos DJ, French JA, Gaillard WD, Hermann BP, Hesdorffer DC, Roper SN, Van Cott AC, Grinnon S, Stout A. Common data elements in epilepsy research: development and implementation of the NINDS epilepsy CDE project. *Epilepsia.* 2011; 52(6):1186–91. [PubMed: 21426327]
20. Marton, Maryann; F, TL.; Kisler, Bron; Pathak, Jyotishman; Haendel, Melissa; Bretz, Jennifer; Haas, Magali. Surveying and Navigating the CDE Landscape. *CDISC Journal.*
21. Niland, JC. ASPIRE: Agreement on Standardized Protocol Inclusion Requirements for Eligibility. 2008. Available from: http://wiki.hl7.org/images/7/7c/March_5,_2008.pdf
22. Stone K. NINDS common data element project: a long-awaited breakthrough in streamlining trials. *Ann Neurol.* 2010; 68(1):A11–3.
23. Grinnon ST, Miller K, Marler JR, Lu Y, Stout A, Odenkirchen J, Kunitz S. National Institute of Neurological Disorders and Stroke Common Data Element Project - approach and methods. *Clin Trials.* 2012; 9(3):322–9. [PubMed: 22371630]
24. Luo Z, Miotto R, Weng C. A human-computer collaborative approach to identifying common data elements in clinical trial eligibility criteria. *J Biomed Inform.* 2013; 46(1):33–9. [PubMed: 22846169]
25. Humphreys BL, Lindberg DAB, Schoolman HMG, Barnett O. Technical Milestone: The Unified Medical Language System: An Informatics Research Collaboration. *J Am Med Inform Assoc.* 1998; 5(1):1–11. [PubMed: 9452981]
26. Miotto R, Weng C. Unsupervised mining of frequent tags for clinical eligibility text indexing. *J Biomed Inform.* 2013 Dec; 46(6):1145–51. 2013. [PubMed: 24036004]
27. Lee-Smeltzer KH. Finding the needle: controlled vocabularies, resource discovery, and Dublin Core. *Library Collections Acquisitions & Technical Services.* 2000; 24(2):205–215.
28. Miotto R, Jiang S, Weng C. eTACTS: A method for dynamically filtering clinical trial search results. *J Biomed Inform.* 2013 Dec; 46(6):1060–7. 2013. [PubMed: 23916863]
29. Boland MR, Miotto R, Weng C. Feasibility of Feature-based Indexing, Clustering, and Search of Clinical Trials. A Case Study of Breast Cancer Trials from ClinicalTrials.gov. *Methods Inf Med.* 2013; 52(5):382–94. [PubMed: 23666475]
30. Boland MR, Miotto R, Weng C. A method for probing disease relatedness using common clinical eligibility criteria. *Stud Health Technol Inform.* 2013; 192:481–5. [PubMed: 23920601]
31. Mougin F, Burgun A, Bodenreider O. Mapping data elements to terminological resources for integrating biomedical data sources. *BMC Bioinformatics.* 2006; 7(Suppl 3):S6. [PubMed: 17134479]
32. Korkontzelos I, Mu T, Ananiadou S. ASCOT: a text mining-based web-service for efficient search and assisted creation of clinical trials. *BMC Med Inform Decis Mak.* 2012; 12(Suppl 1):S3. [PubMed: 22595088]
33. Aronson AR, Lang FM. An overview of MetaMap: historical perspective and recent advances. *J Am Med Inform Assoc.* 2010; 17(3):229–36. [PubMed: 20442139]
34. GitHub Gist. Extracting key phrases with NLTK in Python. Available from: <https://gist.github.com/alexbowe/879414>
35. Brown, Peter F.; d, PV.; Mercer, Robert L.; Della Pietra, Vincent J.; Lai, Jenifer C. Class-based n-gram models of natural language. *Computational Linguistics.* 1992; 18(4):12.
36. Frantzi K, Ananiadou S, Mima H. Automatic recognition of multi-word terms: the C-value/NC-value method. *International Journal on Digital Libraries.* 2000; 3(2):115–130.
37. Boland MR, Tu SW, Carini S, Sim I, Weng C. EliXR-TIME: A Temporal Knowledge Representation for Clinical Research Eligibility Criteria. *AMIA Summits Transl Sci Proc.* 2012:71–80. 2012. [PubMed: 22779055]
38. Luo Z, Johnson SB, Lai AM, Weng C. Extracting temporal constraints from clinical research eligibility criteria using conditional random fields. *AMIA Annu Symp Proc.* 2011:843–52. 2011. [PubMed: 22195142]

39. Hao T, Rusanov A, Weng C. Extracting and Normalizing Temporal Expressions in Clinical Data Requests from Researchers. *Lecture Notes in Computer Science*. 2013; 8040:10.
40. Pustejovsky, J.; Ingria, R.; Sau, R.; Castaño, J.; Littman, J.; Gaizauskas, R.; Setzer, A.; Katz, G.; Mani, I. *The Language of Time: A Reader*. Oxford University Press; 2007. The Specification Language TimeML.
41. Branimir, B.; Rie Kubota, A. TimeML-compliant text analysis for temporal reasoning. *Proceedings of the 19th international joint conference on Artificial intelligence*; 2005; Edinburgh, Scotland. p. 997-1003.
42. Cimino JJ. Desiderata for controlled medical vocabularies in the twenty-first century. *Methods Inf Med*. 1998; 37(4-5):394-403. [PubMed: 9865037]
43. Fan JW, Friedman C. Semantic classification of biomedical concepts using distributional similarity. *Journal of the American Medical Informatics Association*. 2007; 14(4):467-477. [PubMed: 17460124]
44. Manning, CD.; Raghavan, P.; Schütze, H. *Introduction to information retrieval*. Cambridge ; New York: Cambridge University Press; 2008. p. xxi-482.
45. National institute of health. NINDS Common Data Elements. Available from: http://www.commondataelements.ninds.nih.gov/General.aspx#tab=Data_Standards
46. Sun, XH. The Relation of Scalability and Execution Time. *Proceedings of the 10th International Parallel Processing Symposium*; 1996; p. 457-462.
47. Sun XH. Scalability versus execution time in scalable systems. *Journal of Parallel and Distributed Computing*. 2002; 62(2):173-192.
48. de Marneffe, Marie-Catherine; Ma, B.; M, CD. Generating Typed Dependency Parses from Phrase Structure Parses. in *LREC*. 2006
49. Savova GK, Masanz JJ, Ogren PV, Zheng J, Sohn S, Kipper-Schuler KC, Chute CG. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*. 2010; 17(5):507-513. [PubMed: 20819853]
50. UIMA (Unstructured Information Management Architecture). Available from: <http://uima.apache.org>

Fig 1.
Changes in the counts of relevant FSTs as the frequency threshold grows in the three texts.

Fig 2. Changes in the percentage of the retrieved relevant FSTs over maximum retrieved relevant FSTs as the data size grew from 200 to 500 clinical trials summaries, clinical research protocol paragraphs, and clinical data requests, respectively.

Table 1
Comparison of the FST overlap and recall between our approach and BaselineM for all the trials (N=145,745) from ClinicalTrials.gov using frequency thresholds ranging from 1% (i.e., an FST occurs in 1% of all the sample trials) to 8%

Frequency threshold	#Relevant FSTs			Overlap with BaselineM	Recall improvement upon BaselineM
	BaselineM	kernel-wrapper	Shared		
0.01	316	349	243	76.9%	10.4%
0.02	233	248	187	80.3%	6.4%
0.03	133	142	117	88.0%	6.8%
0.04	96	106	88	91.7%	10.4%
0.05	77	85	71	92.2%	10.4%
0.06	49	57	47	95.9%	16.3%
0.07	40	48	39	97.5%	20.0%
0.08	32	39	32	100.0%	21.9%