Journal of Biomedical Informatics 57 (2015) 377–385

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Scaling and contextualizing personalized healthcare: A case study of disease prediction algorithm integration

Keith Feldman [a], Darcy Davis [b], Nitesh V. Chawla [a,*]

[a] University of Notre Dame, Notre Dame, IN, USA
[b] Advocate Healthcare Downers Grove, IL, USA

ARTICLE INFO

Article history:
Received 1 April 2015
Revised 21 July 2015
Accepted 26 July 2015
Available online 5 August 2015

Keywords:
Personalized healthcare
Big Data
Clinical informatics
Data mining

ABSTRACT

Today, advances in medical informatics brought on by the increasing availability of electronic medical records (EMR) have allowed for the proliferation of data-centric tools, especially in the context of personalized healthcare. While these tools have the potential to greatly improve the quality of patient care, the effective utilization of their techniques within clinical practice may encounter two significant challenges. First, the increasing amount of electronic data generated by clinical processes can impose scalability challenges for current computational tools, requiring parallel or distributed implementations of such tools to scale. Secondly, as technology becomes increasingly intertwined in clinical workflows these tools must not only operate efficiently, but also in an interpretable manner. Failure to identity areas of uncertainty or provide appropriate context creates a potentially complex situation for both physicians and patients. This paper will present a case study investigating the issues associated with first scaling a disease prediction algorithm to accommodate dataset sizes expected in large medical practices. It will then provide an analysis on the diagnoses predictions, attempting to provide contextual information to convey the certainty of the results to a physician. Finally it will investigate latent demographic features of the patient's themselves, which may have an impact on the accuracy of the diagnosis predictions.

1. Introduction

Over the past decade the digitization of healthcare records has provided a foundation for data scientists and clinicians alike to employ data mining and machine learning techniques on medical datasets [1]. These techniques have allowed for not only substantial improvements to existing clinical decision support systems, but also a platform for improved patient-centered outcomes through the development of personalized prediction models tailored to a patient's medical history and current condition [2–5]. While powerful, the integration of such tools into clinical workflows is a challenging endeavor. This paper will address two major components integral for the successful integration of analytical tools into a clinical workflow.

Of first concern is incorporating these tools within a clinical time frame and context. Due to the time sensitive nature of clinical scenarios, the machine learning models on which these tools are built must allow for execution within a relevant timeframe, which is often only the duration of a patient visit. As the data becomes increasingly available, drawn from multiple sources and encompasses multiple modalities, it also becomes critical for machine learning methods to process them both accurately and quickly. If a patient's current visit is used during a physician's office visit to suggest a series of personalized recommendations, then it is warranted that the back-end prediction engine is able to deliver within this timeframe. However the quantity of data necessary to build these models presents a significant issue for successful long-term utilization. It is important to remember that each medical encounter will result in additional data added to a patient's electronic health record.

Although for now this limitation can be overcome with algorithmic ingenuity, healthcare's "Big Data" may soon exceed the ability of standard data processing techniques, given its variety, veracity and volume. As a result we must look to other approaches, such as distributed computation, in order to scale personalized healthcare models. Failure to do so may result in the need to artificially restrict the data on which the models are built. This would typically be accomplished either through the process of feature or instance selection, the difficulties of which have already been well documented [6–8].

The ability to utilize these tools within a constrained time window is not the only obstacle to their deployment in clinical

* Corresponding author.
E-mail addresses: kfeldman@nd.edu (K. Feldman), davis.darcy.a@gmail.com (D. Davis), nchawla@nd.edu (N.V. Chawla).

http://dx.doi.org/10.1016/j.jbi.2015.07.017
1532-0464/

settings. The second concern stems from the realization that with the implementation of predictive models, patients are no longer the only individuals receiving treatment recommendations. Physicians will now begin to receive treatment recommendations personalized to their current patient. Although these tools and techniques are designed to augment the existing skills of the physician, expanding their clinical knowledge beyond their prior experience and education, they also introduce a new challenge of providing an appropriate narrative with the predictions or the analytics.

It is important to remember that while these predictions may be the result of advanced machine learning models, they have to be assessed and communicated within a clinical context. While physicians will likely be equipped to understand the clinical aspects of the recommendations they receive, as well as the risks associated with them, there is currently no process in place to ensure the algorithmic results are clinically interpretable. To date a substantial set of prior work has been done tuning the performance of these algorithms, and although these evaluations help to create functional and effective models, many fail to perform any medically focused evaluation of the predicted instances [9,10].

However due to the complexities of human disease, and the uniqueness of each patient, a deeper understanding of the algorithm producing the recommendation is critical for the successful integration of these tools into a clinical workflow. As an example the high probability of a frequently misdiagnosed disease may not be as diagnostically useful for a physician as would a slightly lower probability disease, that when predicted is almost always correct.

This paper will provide a case study addressing each of the integration challenges discussed above, walking through the process of bringing a disease prediction algorithm out of an academic setting and preparing it for the complexities of a clinical setting. For the study we will be utilizing the disease prediction algorithm CARE (Collaborative Assessment and Recommendation Engine) [11]. We have chosen CARE as the algorithm has already been shown to be effective, and as we will see CARE is a good proxy for an entire class of disease prediction algorithm utilizing *patient similarity techniques*. The scaling of CARE using distributed computing constructs can thus provide a possible template for integration with other existing disease prediction algorithms that leverage large-scale electronic health care records. Finally, as we will see throughout this paper, it is important to contextualize the outcome of any clinical decision making aid for patient as well as physician consumption in order to reach the goal of "patient empowerment and engagement".

The paper is structured as follows. We will begin with a back-end system-level investigation into the task of scaling the CARE algorithm to accommodate the patient datasets representative of true clinical databases. Next we include an in-depth analysis of the CARE algorithm from a clinical perspective, identifying those diagnoses that CARE can frequently predict correctly, and those that may present difficulty, and how these insights may translate to the patient. It will then evaluate patient demographic data, identifying latent features which may indicate the difficulty of correctly predicting an individual's future diseases as well the distribution of diagnosis across the highest and lowest performing individuals.

2. Disease prediction algorithm

Over recent years a number of disease prediction algorithms have been developed to accomplish a multitude of tasks. While some algorithms focus on modeling an individual's risk of developing specific diagnoses such as cardiac conditions, others can be utilized in a more general approach to identify individuals' high-risk future conditions [5,12–14]. The past few years have witnessed further development of these predictive tasks, creating systems to model tasks such as the progression of degenerative diseases as well as extensions into the genomic field, identifying target sites utilized in biomarker and drug discovery [15–17].

2.1. The CARE algorithm

Amongst the earliest general disease prediction models for personalized healthcare that leverage patient similarity is the CARE Algorithm. CARE uses collaborative filtering of an individual's medical history in order to identify high likelihood diagnoses in the patient's future. Collaborative filtering is traditionally a technique by which similar individuals are identified through a set of known shared preferences or attributes. The intent of collaborative filtering is to identify new preferences for an individual based on the non-shared preferences identified between other similar individuals [18–20]. While these techniques have been utilized for many years in online applications such as movie, book and product recommendations, they have recently shown promise in the healthcare domain as well. Beyond CARE, a number of recent algorithms have utilized collaborative filtering for applications such as nursing decision support, medical context identification, and identification of sudden deterioration for a patient's medical condition [21–23].

An architectural diagram of the standard CARE algorithm can be seen in Fig. 1 [11] and is comprised of three major steps. For a patient p, the algorithm begins with an initial filtering on all patients within the database, isolating only those patients who have at least one disease in common with p. This is done as totally disparate patients offer no potential similarity information, and will only serve to extend the computation time. Next utilizing this subset of patients the collaborative filtering step is performed. CARE's collaborative filtering algorithm incorporates a binary coding of diagnoses codes, with 1 representing a present diagnosis, and 0 one which is absent or undiagnosed. In addition, the inverse frequency of each diagnosis is used in order to give higher weight to less common diagnosis. This is particularly important as some diagnosis, such as hypertension, are present in 33.64% of all patients [11]. CARE also incorporates a time component representing when in the patient's medical history did they develop a disease. Next, an ensemble of such collaborative filtering models is generated for each similar set of patients identified for each disease of p. Finally the results are then aggregated, yielding a ranked list of high probability diseases for p.

3. Materials and methods

As mentioned prior the CARE algorithm has been previously shown to be accurate, and in an effort to maintain consistency with the published work the original dataset and source code were used in this case study as in the original CARE evaluation.

3.1. Data

The dataset utilized for this work contains approximately 32 million anonymized Medicare claims each representing a patient visit, accounting for just over 13 million unique patients. As per the original CARE work, in order to ensure sufficient diagnosis history during training only those patients with over 5 visits were considered for evaluation in this paper [11].

The Medicare claim itself contains 16 features as well as a unique identifier for patients with multiple visits. The claim is broken into two main sections, *patient demographics* and *diagnosis codes*. Patient demographics contains the date of the visit, the patient's

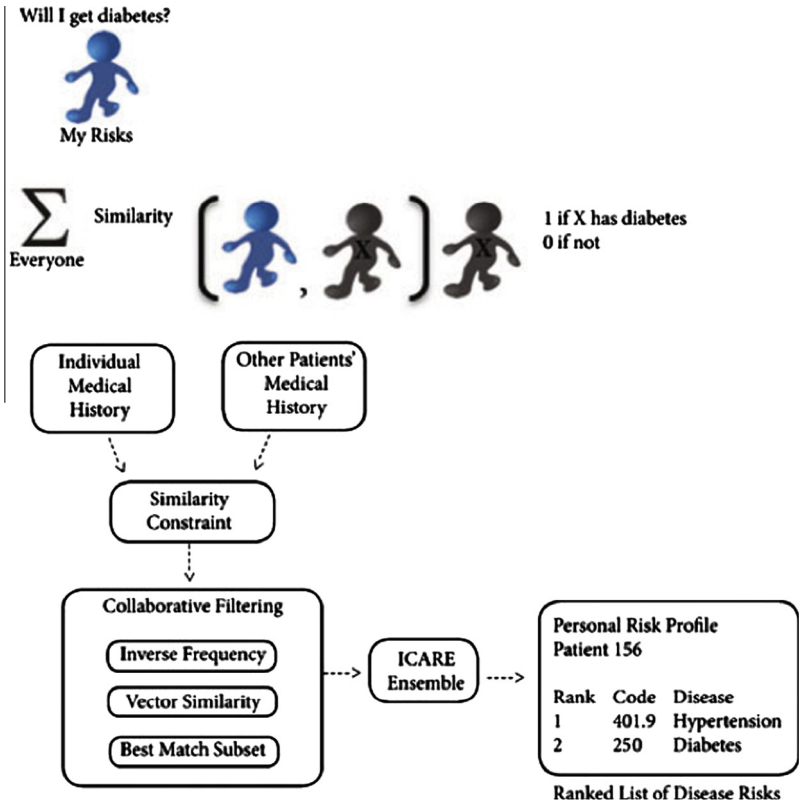

Fig. 1. Current CARE implementation [24].

age at the time of the visit, and a set of anonymized (masked) variables. These anonymous variables include the patient's race, geographic state code, as well as gender and poverty flags. The diagnosis codes section of the claim contains 1 primary, and up to 9 secondary diagnosis codes for each patients visit. All diagnoses codes are represented by the International Classification of Diseases, Ninth Revision, Clinical Modification (ICD-9-CM) [25].

One feature that requires further exposition is Age. As stated above the patients in the dataset must have at least 5 visits, which may span multiple years. As such, we decided to use their age at the time of their first recorded Medicare claim. This decision is supported by the unfortunate reality that the likelihood of health complications increases as an individual ages. Thus, synthetic age metrics such as a patient's mean and median claim age may be artificially right-skewed by the increasing frequency of claims at increasing ages. Further, the age of their first Medicare claim may be indicative of the early onset of health complications.

Finally, it is important to note the concept of *code collapse* as it relates to the hierarchical nature of ICD-9 diagnosis codes. A complete ICD-9 code is typically 5 digits in length, with the leading digits representing the medical diagnosis family, while the trailing offer specifics about the condition such as the location or severity. As a result these codes can be collapsed into more general representations. Three digit codes are known as the "Major Category", four digit as "Intermediate Category" and five digit codes as the "Minor Category" representing the level of specificity detailed at each level [26]. Multiple code collapse levels from 5 digits, to 4, to 3 digits will be used within the disease evaluation portion of this paper, while the patient evaluation will utilize the fully detailed 5-digit code.

3.2. Algorithmic procedure

The analyses required for the case study are broken down into two main components. First we address the scaling issue, performing an exploration of execution bottlenecks and developing a scalable distributed implementation of the CARE algorithm, which can be extended to any frameworks that compute patient similarity. Secondly we detail the strengths and weaknesses of the algorithm in terms of specific diagnosis codes as well as demographic attributes of the patients themselves. The major contribution for each of these analyses will be denoted within the section headers "Scaling" and "Contextualizing" respectively.

3.3. Scaling CARE

Through a detailed analysis of the CARE algorithm we were able to identify parallelizable components within the patient similarity process, greatly reducing the execution time required to obtain the disease rankings. We will demonstrate the ability to scale CARE's usage on Big Data to achieve execution times at approximately *one sixth* of a day. We will also examine the implications of execution time when accounting for the limited duration of a typical clinical encounter.

3.3.1. Evaluation metrics and environment

As the scaling analysis is at its core a systems level analysis, it is important to first define the metrics and computing environment used to evaluate the CARE algorithm. For our evaluation we utilize two primary metrics, total execution time, and the total number of patients on which the algorithm is trained. As noted prior, due to the time sensitive nature of clinical environments the execution time of the algorithm is a critical component for the successful integration of informatics tools into a clinical workflow, and as such total execution time will be utilized as the principle evaluation metric. Further, as a result of the collaborative filtering performed as part of the CARE algorithm the total number of patients on which the algorithm is trained is also an important statistic. As demonstrated empirically by Breese et al. the size of

the training set is directly related to the accuracy of collaborative filtering algorithms [20].

To benchmark the standard CARE algorithm we utilized a machine containing 4, 16-core 2.3 GHz AMD Opteron processors with 128 GB RAM. All simulations were run 5 times, and the results averaged to obtain the reported performance statistics. Additionally, prior to each benchmark CARE was executed in a single non-timed run to warm the cache with the respective patient and disease records. Finally, CARE was compiled with the -O0 flag to disable any architecture specific complier optimizations. While this may cause a slight increase in overall execution time, the performance will be significantly more stable between multiple machines; a particularly important element for the distributed version detailed below.

It should be noted that due to the variability in execution time all evaluations were performed with at least 25 patients. This is a reasonable requirement, as even the smallest medical practices can be expected to have medical records for 25 unique patients.

3.3.2. Identifying scaling opportunities in CARE

Our first step was to perform a sweep of CARE over an increasing number of patients, analyzing the time spent within each internal function. As CARE utilizes collaborative filtering, it was logical to see the all-pairs similarity comparison identified as the primary bottleneck. From here we then investigated the individual components, which comprise this portion of the algorithm, identifying three main opportunities: the time needed to aggregate the diagnoses codes from each patient's visit set (*visit aggregation*); the similarity calculation between the diagnosis vectors of two patients (*vector similarity*); and the system calls used to perform the computation (*system computation*).

3.3.3. Distributed analysis

Using the execution statistics obtained from the standard CARE execution we then created a distributed version of the algorithm, the goal of which was to perform within a reasonable clinical timeframe as well as increase the size of the patient training set.

Although all patients' medical histories are required to train a complete model, each patient's disease similarity is calculated independently, providing an ideal scenario for parallelization. First the set of all patients is partitioned into equal size subsets, and then using the WorkQueue algorithm created by the Cooperative Computing Lab (CCL) at Notre Dame, we distribute the execution across a compute grid [27]. This process is detailed in Fig. 2.

However, while parallelization is an effective method of scaling these algorithms, due to the highly sensitive nature of healthcare data the distribution of data can provide a challenge. Prior work Berkvosky el. in [28] has evaluated the issue of privacy when using collaborative filtering techniques on distributed data sets. In their implementation, Berkvosky details a method where each calculation's subset of data contains only a minimal amount of identifiable information. However, we aim to take the method one step further and distribute the computation to each data site, where each worker receives a copy of the CARE algorithm as well as information about the subset it is tasked to compute. Further, the workqueue framework allows for access to a centrally located set of patient records though a shared file system. This setup would address privacy concerns as only result of the similarity calculations are then transmitted over the network, a process more akin to the MapReduce framework [29].

3.4. Contextualizing CARE

Once we had established that the CARE algorithm could be augmented to execute on realistically sized datasets, we looked to the second integration challenge of generating context around the diagnosis predictions. This analysis explores two main aspects of this interpretability, pertaining to the patient's diagnosis rankings, and the demographics of the patient themselves. An example of the contextualization workflow can be found in Fig. 3. The workflow begins with the probabilistic ranked list of diagnosis predictions provided by the CARE algorithm. Next through the analyses techniques found within this paper we provide contextual information

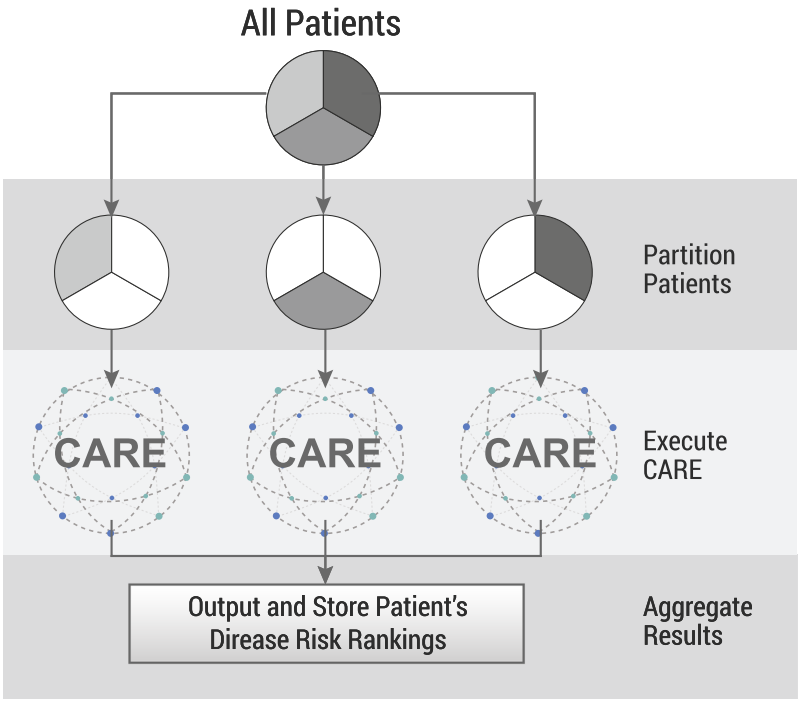

Fig. 2. Distributed CARE implementation.

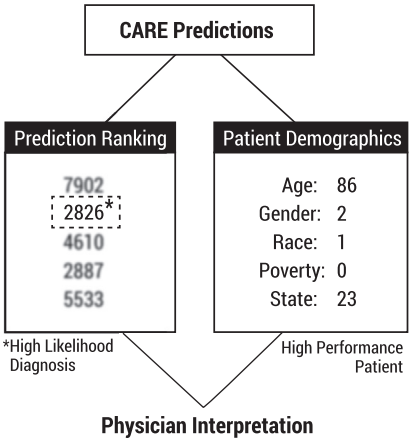

Fig. 3. Contextualization workflow.

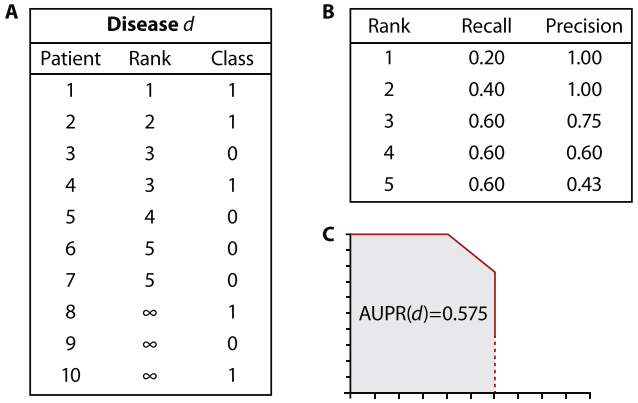

Fig. 4. Example CARE ranking.

about the confidence in the patients predictions based on their personalized demographic data, as well as confidence based on the specific diagnosis codes predicted. As noted earlier this diagnosis code level analysis is pertinent as the identification of specific diagnosis codes that when found on the ranked predicted list, are almost always correct would be extremely useful for a physician to know when planning their course of action, despite being ranked somewhat lower than expected. Finally these analyses lead back into physician's interpretation of the results, as our goal remains to provide tools to better inform their clinical decisions not make them autonomously. We then provide a compelling case study into how a physician may identify a patient at risk for prostate cancer by utilizing both their prior clinical knowledge, and a contextual understanding of the algorithm's high confidence diagnoses. Finally we demonstrate that difficult to predict patients may in fact have a statically different demographic profile from those in the general population.

3.4.1. Diagnosis analysis

The goal of the diagnosis analysis was to provide a physician with insight into the various codes recommended for each patient. Since CARE creates personalized diagnosis predictions for each patient, we designed a metric order to objectively evaluate the overall predictability of each diagnosis code. The evaluation was performed for each disease d, with each patient p treated as an instance. The predictability metric utilizes both the diagnosis code ranking from p's CARE prediction, as well as an associated class variable to indicate the correctness of the prediction (1 if the diagnosis truly occurs in p's future, 0 if it does not). If the current code was not present on the ranked list of diagnoses produced by CARE for patient p it receives a value of ∞. It is important to note that the instance p is removed for a particular d if that diagnosis occurs both in the patient's history as well as their future diagnosis. This is to help prevent any potential bias in the resulting predictability score for diagnosis such as chronic conditions that may occur many times though a patient's medical history, and thus would be easily predicted artificially increasing the algorithms accuracy.

Using the augmented dataset containing diagnosis codes and correctness, the precision and recall are calculated over a threshold of 1–50. The predictability score of d is then computed as the area under the precision-recall curve (AUPR). A toy example of this process over the top 5 ranks is provided in Fig. 4. Finally it is important to note, that while some diagnosis may always be associated with a single ICD-9 code, others may be distributed across multiple 5-digit codes based on slight variations in diagnosis details. To account for these potential variations we perform the same analysis across 5, 4 and 3-digit ICD-9 code groupings.

3.4.2. Patient analysis

The goal of the patient analysis was to identify demographic characteristics at the patient level that influence CARE's overall predictive success. Insights such as these would prove invaluable to a physician attempting to identify a patient for whom external factors such as age or race may influence the models predicted conditions. As with the diagnosis predictability, we created a metric to quantify the predictability of a particular patient. Since each patient has a unique set and number of diagnoses we decided to utilize the Jaccard coefficient set similarity metric for the predictability score [30]. The Jaccard coefficient is calculated as follows $J(A,B) = \frac{|A \cap B|}{|A \cup B|}$, where we can define our sets with respect to an individual patient p. Set A is defined as the set of diagnoses CARE correctly predicts across all visits for p, and set B is defined as the set of all diagnoses recorded across all visits for p. The Jaccard coefficient then results in a standardized metric of the overall effectiveness of CARE's diagnosis predictions with respect to an individual patient.

It should also be noted that precision was not included as factor in the patient's predictability score. This is due to the fact that the absence of a diagnosis in claims data cannot be viewed as a confirmation to the lack of that particular the diagnosis for a patient. For clarity the absence of a diagnosis in claims data fails to differentiate a patient who will receive that diagnosis after the data collection period was finished (a correct future prediction outside of the study window), a patient who was not diagnosed despite having the condition, or those who truly never had the condition. Thus the coverage of that patient's future diagnosis set becomes more important than the total number of diagnoses that do not occur.

For the patient analysis the CARE algorithm was trained across all visits for 10,000 unique patients, and evaluated on a separate 10,000 patient set. Predictability scores were then calculated for each of the test patients as detailed prior. Using these scores those patients with the top 10% were classified as high performers, while those with the bottom 10% were classified as low performers. These high performer and low performer groups form the populations for an additional diagnosis evaluation, the goal of which is to identify frequent diagnosis subgroups indicative of a high/low performing patient.

In order to identify the demographic features that impact CARE's diagnosis prediction coverage for a patient, we evaluated the distribution of each of the 5 demographic features *age, race, geographic state code, gender, and poverty level flag* using a Chi-Squared goodness of fit test.

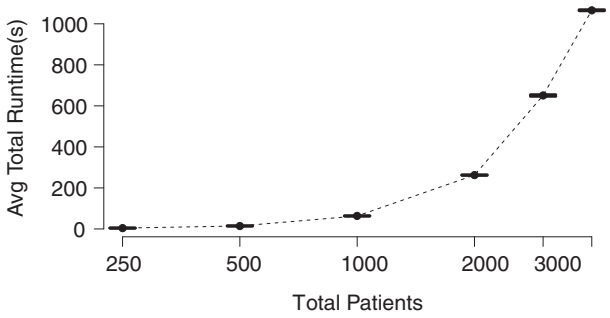

Fig. 5. Standard CARE runtime.

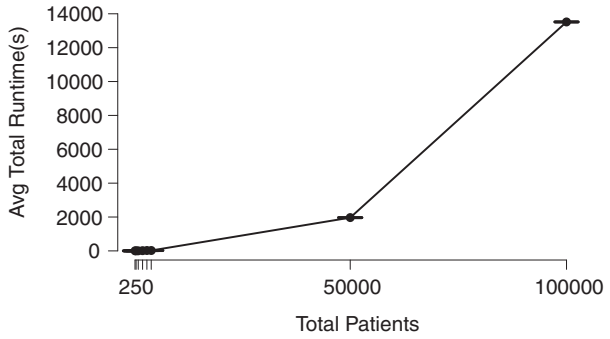

Fig. 6. Distributed CARE runtime.

For the goodness of fit the baseline distributions were drawn from the full patient population, and the sample distributions drawn from the high and low performing patient groups. The Chi-Squared statistic was computed using K-1 degrees of freedom, where K is the number of levels of each demographic variable. In the case of the continuous variable *Age* the data was discretized into 7 bins each spanning 5 years, with the final bin pooled all ages over 95.

4. Results

4.1. Scaling CARE

The execution timing results for the unmodified CARE algorithm can be found in Fig. 5. As noted above, while trivial, we isolated the I/O and CPU bound elements of the algorithm noting that CARE spent 93% of it's execution time in CPU based components for datasets containing as low as 250 patients. From here we found that CARE spent greater than 99% of all CPU time in a single function, known as best match, which executed the collaborative filtering operations. Breaking down the Best Match function further we found that although the execution time was exponential (Fig. 5), the percent of time spent in each of these step of the function remained fairly steady over increasing numbers of patients. With the *vector similarity*, *visit aggregation*, and *system computation* comprising approximately 42, 8, and 50 percent of the total execution time respectively, with an average standard deviation of 1.61%. The results of the distributed version of CARE can be found in Fig. 6, with more detailed statistics on the WorkQueue workers performance found in Table 1.

4.2. Contextualizing diagnosis analysis

Tables 2 and 3 detail CARE's top 5 most and least predictable diagnosis codes respectively, with the rankings calculated for each of the three code collapse levels (5,4 and 3 digits ICD-9 codes). For

Table 1
Avg total runtime per task of distributed CARE.

Task count	Avg IO overhead (s)	Avg. CPU time (s)	Num. worker cycles	Derived execution time (h)
10000	0.017	134.358	200	7.465
25000	0.021	43.811	500	6.088
50000	0.098	19.412	1000	5.419
100000	0.101	6.658	2000	3.755

Table 2
Top 5 most predictable codes.

3-Digit codes		4-Digit codes		5-Digit codes	
Top 5 most predictable codes					
401	Hypertension	2826	Sickle-cell disease	25002	Diabetes w/out complication uncontrolled
272	Disorders of lipoid metabolism	4019	Unspec. Essential hypertension	60001	Hypertrophy of prostate w/ obstruction
250	Diabetes	2500	Diabetes w/out complication	60000	Hypertrophy of prostate w/out obstruction
786	Symptoms resp. system	6000	Hypertrophy of prostate	60010	Nodular prostate w/out obstruction
780	General symptoms	7194	Pain in joint	25000	Diabetes w/out complication controlled

Table 3
Top 5 least predictable codes.

3-Digit codes		4-Digit codes		5-Digit codes	
Top 5 least predictable codes					
553	Hernia of abdominal cavity	7886	Abnormality of urination	72887	Muscle weakness
389	Hearing loss	5246	TMJ disorders	78909	Abdominal pain other specified site
560	Intestinal obstruction w/out hernia	3899	Unspec. hearing loss	78863	Urgency of urination
536	Disorders of stomach function	5368	Dyspepsia	68110	Cellulitis/abscess of toe
410	Acute myocardial infarction	6010	Acute prostatitis	80700	Closed fracture of rib(s)

Table 4
Chi-Squared results.

High performing patients Chi-Squared p-value	Demographic feature	Low performing patients Chi-Squared p-value
.16	Age	$4.40e^{-3}$ *
.42	Gender	.11
.38/.42	Race	$1.36e^{-6}/2.41e^{-7}$ *
$2.67e^{-7}$ *	Poverty line flag	$1.06e^{-7}$ *
$3.90e^{-3}/5.74e^{-3}$ *	Geographic state	$7.64e^{-4}/1.81e^{-3}$ *

clarity the term most predictable corresponds to those diagnoses which receive the highest AUPR values. However, the least predictable diagnoses are calculated slightly differently. Due to the number of unique diagnoses (5,949) it is likely that not every disease is represented within each test set. As AUPR is bounded in the range [0,1], this creates the situation where many diagnoses may occur with low AUPR values and minimal variation, making differentiation difficult. To account for this we utilize the concept of diagnosis prevalence to the calculation of the least predictable rankings. The least predictable diagnoses are then calculated utilizing the count of patient instances p that actually receive the future diagnosis d, despite never having d receive a ranking in the CARE results.

4.3. Contextualizing patient demographic analysis

As detailed in the Methods section the patient analysis was broken down into high performer and low performer groups, containing 1335 and 1179 patients respectively. Both patient groups are of sufficient sample size, where the resulting Chi-Squared p-value has been shown to become essentially equivalent to that of the Fisher's exact test [31].

The results for the significance tests on both groups can be seen in Table 4, with significant $(p<0.05)$ values indicated with an asterisk. Two features, Race and Geographic State, have additional p-values recorded, as these features contained distributions which resulted in expected values less than 5. As this is a requirement for the Chi-Squared test, those levels with expected values fewer than 5 were pooled into a new *Other* category and the Chi-Squared statistic recalculated. The format for these features is as follows: *unmodified distribution/pooled distribution*.

Continuing with the patient analysis, we investigated the prevalence of diagnosis across both the high and low groups. The top 5 most prevalent diagnoses for each can be seen in the leftmost columns in Tables 5 and 6 respectively. As high prevalence diagnoses are likely to bias the top 5 rankings regardless of performance group, we extended the analysis to examine the top 5 diagnosis codes which are unique to each performance group. These results can be found in the rightmost columns of Tables 5 and 6.

As a point of reference the high performers contained 2084 different diagnosis codes, compared to the 2401 of the low performers. The groups shared 1558 codes, leaving the high and low performing groups with 526 and 843 unique diagnosis codes respectively.

5. Discussion

5.1. Scaling CARE in a distributed fashion

The benefit of the distributed CARE algorithm can best be conveyed with an example. Utilizing the execution times of the standard version of CARE we were able to fit the results with the exponential function $60.3882e^{0.00073x}$. Please note that there are opportunities to further optimize this code itself, but we wanted to use the exact codebase as used in the original paper for consistency. However, gains achieved from this scaling will be reflected in both the sequential and distributed version, so using the original codebase was sufficient for this paper. From this curve we can expect a dataset of 100,000 patients, a total patient size easily obtained and eclipsed by larger medical practice or hospitals, to take approximately $7.04e^{25}$ years. However, looking at the distributed CARE's execution time on the same set of 100,000 patients, we see a runtime of approximately 4 h. This could be utilized in one of two ways. First the algorithm could be utilized to preprocess the data within a full clinical dataset each night for the existing patient records, and presented to the physician per

Table 5
High performing patient top diagnosis.

Top 5 overall diagnosis		Top 5 unique diagnosis	
Code	Description	Code	Description
4280	Unspec. congestive heart failure	5723	Portal hypertension
4140	Coronary atherosclerosis	1622	Malignant neoplasm of main bronchus
496	Chronic airway obstruction	3942	Mitral stenosis w/insufficiency
4019	Unspec. hypertension	2532	Panhypopituitarism
4111	Intermediate coronary syndrome	45621	Esophageal varices in diseases classified elsewhere w/out mention of bleeding

Table 6
Low performing patient top diagnosis.

Top 5 overall diagnosis		Top 5 unique diagnosis	
Code	Description	Code	Description
4019	Unspec. hypertension	1530	Malignant neoplasm of hepatic flexure
496	Chronic airway obstruction	45111	Phlebitis and thrombophlebitis of femoral vein (Deep) (Superficial)
5990	Unspec. site urinary tract infection	3014	Obsessive–compulsive personality disorder
4280	Unspec. congestive heart failure	44621	Goodpastures syndrome
4140	Coronary atherosclerosis	220	Benign neoplasm of ovary

Table 7
Avg IO overhead for patient dataset in μs.

Task count	1000	10,000	25,000	50,000
Patients per task	50	5	2	1
No cache	43618.23	43609.78	46276.27	57924.65
Cache first access	58597.37	50521.59	57008.15	54396.94
Cache subsequent access	29.92	32.11	29.89	31.43

his or her schedule of patient appointments. Secondly, for a new patient the algorithm could be run on a 4000 patient subset in around 23 s. To contrast unmodified CARE takes approximately 17 min to process the same 4000 patient dataset. While this may not seem to be an exorbitant amount of time, the issue comes clearer when presented with the fact that prior studies have shown an entire clinical encounter lasts around 17 min [32]. A total time that includes ancillary activities such as recording a patient's vital signs, thus our distributed version leaves significantly more time for the physician to review and interpret the results.

As our overarching goal was clinical integration we also aimed to demonstrate that this complexity could be further reduced. We investigated the benefits of caching input files, such as the CARE algorithm and patient data, on each of the worker machines the results of which can be seen in Table 7. As you can see, the cache access effectively alleviates the majority of the system overhead. Although there may be a diminishing benefit with increasing number of workers we were not able to reach a point at which communication overhead affected overall execution time, even on the full 5-visit patient dataset. This supports the idea that regardless of the system setup a medical provider could simply utilize the full set of computational resources available without the need for complex optimizations.

5.2. Contextualizing diagnosis analysis

The evaluation of diagnosis predictability yielded two distinct sets for interpretation. Beginning with the most predictable diagnoses, we find they fall predominantly into two main categories. The first category contained diagnoses common with elderly patients, such as hypertension, diabetes and osteoporosis. This is logical, as the minimum age for patients in the Medicare data was 65, with a mean and median of 76.83 and 76 respectively. The second category contained common co-morbid diagnoses such as fatigue and joint pain, as well as respiratory conditions and secondary infections, such as bronchitis and sinusitis, which often develop as a result of other illnesses.

Overall the high performing diagnoses seem promising, identifying frequently co-morbid conditions. However, transitioning these predictions to actionable insight for a physician remains a critical step for successful integration into a clinical workflow. An example of how this insight may be accomplished can be illustrated with an example scenario. Looking at the results of the 5-digit most predictable codes we see the code 60010. 60010 represents nodular prostate without urinary obstruction, which falls under the diagnosis category *Hyperplasia of prostate*. It has been shown that most prostate cancer (PCA) arise concomitantly with nodular hyperplasia [33]. The high performance of this diagnosis provides an ideal example of where a physician aware of both the medical implications of a diagnosis and the strengths of the prediction algorithm may be alerted to a promising avenue for investigation, furthering the goal of providing of preventative medicine to a potentially at risk patient.

The least predictable diagnoses can also be separated into two clear groups, injuries and infections. The difficulty encountered in predicting these diagnoses is logical as the prediction of injuries such as hernia and fractures would prove to be nearly impossible based solely on a patient's medical history without a physical evaluation. This further highlights the need to make the physician aware of the limitations of the algorithm producing the recommendations. As an example high risk conditions, such as internal bleeding, may be ranked lower for a patient suffering from trauma after an accident due to algorithmic affinities.

There was however one anomaly in the least predicted set, myocardial infarction was amongst the top 10 in both the 3-digit code 410 and the 4-digit 4109 (ranked 6th and not shown in Table 3). We believe that while there are many warning signs for myocardial infarctions, such as obesity, tobacco use and high cholesterol, which should be identified by CARE, but the availability of such data itself presents an issue. Although the conditions that precede a myocardial infarction are well documented, they are not well reported. Non-acute risk factors such as the observation of obesity or tobacco use are often not documented in a patient's medical record. This is reflected in our dataset, as the prevalence of documented obesity is clearly under-reported at 1.3%. As a point of reference the Centers for Disease Control and Prevention (CDC) collected obesity prevalence data in same region, during the same time period that our dataset was being compiled. Their estimated rate of obesity fell somewhere between 26.3% and 29.7% [34]. This highlights another important point, that the predictions produced by any algorithm can only be as good as the data being utilized to make them.

5.3. Contextualizing patient demographic analysis

Prior to detailing the results of the patient analysis it is important to note that in good faith with the data sharing agreement the authors have not attempted to discern the underlying values of any anonymized demographic feature. As such the analysis focuses on the identification of *which* demographic features are indicative of a patient's predictability.

5.3.1. Demographic profile

Once we had identified the high and low performer group, we were able to perform a detailed analysis of the demographic profile of associated with each group.

High performing individuals follow a fairly similar demographic profile to the overall Medicare population, with the exception of *states codes* and the *poverty line flag*. As the state codes are anonymized we cannot comment on how geographic location of a patient affects the predictability of diagnosis. However the presence of both geographic location and poverty as significant features provides evidence that socioeconomic condition may not only be indicative of an individual's overall health, but potentially of precisely which health conditions they may be at risk for.

On the other hand low performing individuals had a largely different overall demographic profile; having significantly different distributions in 4 of the 5 demographic features, with the exception of gender. Again we find poverty-line flag and geographic state code as significant features, likely due to the same reasons discussed above.

Looking further we note that the age distribution is significantly different than the standard Medicare population within the low performer group, elevated from the baseline population expected value in 3 of the 4, 5-year age bins age ranging from 65 to 80. This lends credence to the idea that early onset diagnosis may produce more difficult diagnosis patterns over course of a patient's lifetime.

5.3.2. Performance group diagnosis

After completing the demographic profile analysis we then moved into an investigation into the diagnosis subgroups present within each performance category, and how they could be utilized

by a medical provider. It is unsurprising that the high prevalence diagnoses of hypertension and atherosclerosis appear in the top 5 of both performance groups as they are the top 2 most prevalent diagnosis reported in 33.64 and 21.16 percent of all patients respectively [11].

However it is the unique diagnosis codes that offer some interesting insights into the differentiation of these groups. The high performer patient group contains the diagnosis panhypopituitarism (2532), which is a form of hypopituitarism. Panhypopituitarism is a typically lifelong condition where the pituitary gland does not produce normal amounts of some or all of its hormones [35]. It is interesting that lifelong conditions occur frequently amongst those patients on which CARE performs well. Although these diagnoses may not be among the most prevalent or predictable, they likely have well defined co-morbid diagnoses. Thus a well-informed physician may make use of their appearance in a patient's algorithmic predictions as a signal to screen for known co-morbid conditions.

On the other end the top 5 unique diagnoses for the low performer patients also present some intriguing codes, such as Goodpastures Syndrome. Goodpastures Syndrome is a rare autoimmune disorder, which has a particularly quick onset [36]. The National Library of Medicine notes that symptoms may occur very slowly over months or even years. This extended time frame, along with the generic nature of the symptoms makes early of this diagnosis extremely difficult.

6. Conclusion

This paper provides an extensive discussion of the integration challenges for informatics tools into clinical workflows. The case study of the CARE algorithm has provided an informative look at how these tools can be effectively utilized to provide both a computational and diagnostic advantage to clinicians utilizing them.

We have demonstrated how the rate at which healthcare data is growing will soon necessitate the ability to scale these tools to maintain their effectiveness and accuracy outside of an academic setting. Further we provided two compelling scenarios in which an understanding of the algorithms' strengths along with a clinical understanding of the conditions predicted could be used to improve patient care; either by preventative prostate cancer testing or early identification of co-morbid conditions lifelong conditions such as panhypopituitarism. Additionally, we have shown that predictive performance is not only a byproduct of the algorithm itself, but may also be linked the demographic profile of the individuals whose diagnosis are being predicted.

Finally it is important to remember that regardless of the diagnostic tools used; ultimately a patient's treatment course will be decided by their physician. As such, we hope this paper has conveyed the importance of creating tools that scale and are clinically interpretable to serve as a decision aid or assistant for physicians.

Conflict of interest

The CARE IP (Patent No. 8,504,343) has been licensed to iCareAnalytics, LLC by the University of Notre Dame. Nitesh Chawla serves as a scientific advisor to iCareAnalytics and also has an equity share.

References

[1] H. Chen, S.S. Fuller, C. Friedman, W. Hersh, Medical Informatics: Knowledge Management and Data Mining in Biomedicine, vol. 8, Springer Science & Business Media, 2006.

[2] A. Belle, M.A. Kon, K. Najarian, Biomedical informatics for computer-aided decision support systems: a survey, Sci. World J. (2013).

[3] Y. Zhang, S. Fong, J. Fiaidhi, S. Mohammed, Real-time clinical decision support system with data stream mining, BioMed. Res. Int. (2012).

[4] J.M. Hardin, D.C. Chhieng, Data mining and clinical decision support systems, in: Clinical Decision Support Systems, Springer, 2007, pp. 44–63.

[5] P.B. Jensen, L.J. Jensen, S. Brunak, Mining electronic health records: towards better research applications and clinical care, Nat. Rev. Genet. 13 (6) (2012) 395–405.

[6] T.-H. Cheng, C.-P. Wei, V.S. Tseng, Feature selection for medical data mining: comparisons of expert judgment and automatic approaches, in: 19th IEEE International Symposium on Computer-Based Medical Systems, 2006 (CBMS 2006), IEEE, 2006, pp. 165–170.

[7] H. Liu, H. Motoda, Feature Extraction, Construction and Selection: A Data Mining Perspective, Springer Science & Business Media, 1998.

[8] T. Reinartz, A unifying view on instance selection, Data Min. Knowl. Discovery 6 (2) (2002) 191–210.

[9] I. Yoo, P. Alafaireet, M. Marinov, K. Pena-Hernandez, R. Gopidi, J.-F. Chang, L. Hua, Data mining in healthcare and biomedicine: a survey of the literature, J. Med. Syst. 36 (4) (2012) 2431–2448.

[10] M.A. Jabbar, B. Deekshatulu, P. Chandra, Classification of heart disease using artificial neural network and feature subset selection, GJCST 13 (3) (2013).

[11] D.A. Davis, N.V. Chawla, N.A. Christakis, A.-L. Barabási, Time to care: a collaborative engine for practical disease prediction, Data Min. Knowl. Discovery 20 (3) (2010) 388–415.

[12] E. AbuKhousa, P. Campbell, Predictive data mining to support clinical decisions: an overview of heart disease prediction systems, in: 2012 International Conference on Innovations in Information Technology (IIT), IEEE, 2012, pp. 267–272.

[13] P.C. Austin, J.V. Tu, J.E. Ho, D. Levy, D.S. Lee, Using methods from the data-mining and machine-learning literature for disease classification and prediction: a case study examining classification of heart failure subtypes, J. Clin. Epidemiol. 66 (4) (2013) 398–407.

[14] T.H. McCormick, C. Rudin, D. Madigan, et al., Bayesian hierarchical rule modeling for predicting medical conditions, Ann. Appl. Stat. 6 (2) (2012) 652–668.

[15] J. Zhou, J. Liu, V.A. Narayan, J. Ye, A.D.N. Initiative, et al., Modeling disease progression via multi-task learning, NeuroImage 78 (2013) 233–248.

[16] H.M. Fonteijn, M.J. Clarkson, M. Modat, J. Barnes, M. Lehmann, S. Ourselin, N.C. Fox, D.C. Alexander, An event-based disease progression model and its application to familial alzheimers disease, in: Information Processing in Medical Imaging, Springer, 2011, pp. 748–759.

[17] Y. Yang, S.J. Adelstein, A.I. Kassis, Target discovery from data mining approaches, Drug Discovery Today 17 (2012) S16–S23.

[18] D. Goldberg, D. Nichols, B.M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, Commun. ACM 35 (12) (1992) 61–70.

[19] P. Resnick, H.R. Varian, Recommender systems, Commun. ACM 40 (3) (1997) 56–58.

[20] J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., 1998, pp. 43–52.

[21] L. Duan, W.N. Street, E. Xu, Healthcare information systems: data mining methods in the creation of a clinical recommender system, Enterprise Inform. Syst. 5 (2) (2011) 169–181.

[22] J. Kim, D. Lee, K.-Y. Chung, Item recommendation based on context-aware model for personalized u-healthcare service, Multimedia Tools Appl. 71 (2) (2014) 855–872.

[23] E. Vlahu-Gjorgievska, V. Trajkovik, Personal healthcare system model using collaborative filtering techniques, Adv. Inform. Sci. Service Sci. 3 (3) (2011).

[24] N.V. Chawla, D.A. Davis, Bringing big data to personalized healthcare: a patient-centered framework, J. General Internal Med. 28 (3) (2013) 660–665.

[25] V.N. Slee, The international classification of diseases: ninth revision icd-9, Ann. Internal Med. 88 (3) (1978) 424–426.

[26] Centers for Disease Control and Prevention, International classification of diseases – 9, abbreviated titles, 2014. <http://wonder.cdc.gov/wonder/sci_data/codes/icd9/type_txt/icd9abb.asp>.

[27] P. Bui, D. Rajan, B. Abdul-Wahid, J. Izaguirre, D. Thain, Work queue+ python: a framework for scalable scientific ensemble applications, in: Workshop on Python for High Performance and Scientific Computing at SC11, 2011.

[28] S. Berkovsky, Y. Eytani, T. Kuflik, F. Ricci, Enhancing privacy and preserving accuracy of a distributed collaborative filtering, in: Proceedings of the 2007 ACM Conference on Recommender Systems, ACM, 2007, pp. 9–16.

[29] J. Dean, S. Ghemawat, Mapreduce: simplified data processing on large clusters, Commun. ACM 51 (1) (2008) 107–113.

[30] G. Salton, M.J. McGill, Introduction to Modern Information Retrieval, 1983.

[31] J.H. McDonald, Handbook of Biological Statistics, vol. 2, Sparky House Publishing, Baltimore, MD, 2009.

[32] D. Mechanic, D.D. McAlpine, M. Rosenthal, Are patients' office visits with physicians getting shorter?, New England J Med. 344 (3) (2001) 198–204.

[33] A. Bachmann, J. de la Rosette, Benign Prostatic Hyperplasia and Lower Urinary Tract Symptoms in Men, Oxford University Press, 2011.

[34] Centers for Disease Control and Prevention, National diabetes surveillance system. <http://apps.nccd.cdc.gov/DDTSTRS/default.aspx>.

[35] A.D.A.M. Medical Encyclopedia, Hypopituitarism. <http://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0001383/>.

[36] A.D.A.M. Medical Encyclopedia, Goodpasture syndrome. <http://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0001197/>.