Predicting At-Risk Patient Profiles from Big Prescription Data

Pierre Genevès, Thomas Calmant, Nabil Layaïda, Marion Lepelley, Svetlana Artemova, Jean-Luc Bosson

On the Prediction of At-Risk Patients with Big Data

Pierre Genevès[1], Thomas Calmant[1], Nabil Layaïda[1], Marion Lepelley[2], Svetlana Artemova[2], Jean-Luc Bosson[2]

[1] CNRS, Inria, Univ. Grenoble Alpes, LIG Laboratory UMR 5217, Tyrex, Grenoble, France

[2] Pôle Santé Publique, Centre Hospitalier Universitaire Grenoble Alpes, Grenoble, France

Abstract

We propose a system for detecting, on the day of hospital admission, patients at risk of developing complications during their hospital stay. We propose a method borrowing techniques from AI and supervised learning for the automatic detection of at-risk profiles based on a fine-grained analysis of prescription data at the time of admission. Our system relies on a distributed architecture adapted for processing large volumes of medical records and clinical data. We report on practical experiments with real data of millions of patients and hundreds of hospitals. We demonstrate how big data improves the detection of at-risk patients, making it possible to construct predictive models that benefit from volume and variety, while satisfying important criteria to be deployed in hospitals.

1 Introduction

A major challenge in healthcare is the prevention of complications and adverse effects during hospitalization. A complication is an unfavorable evolution or consequence of a disease, a health condition or a therapy; and an adverse effect is an undesired harmful effect resulting from a medication or other intervention. Typical examples include for instance pressure ulcers, nosocomial infections, admissions in Intensive Care Unit (ICU), and death.

From the perspective of complications, healthcare establishements can be considered as risky environments. For instance, in the USA, an estimated 13.5% of hospitalized Medicare beneficiaries experienced adverse effects during their hospital stays; and an additional 13.5% experienced temporary harm events during their stays[1] [Levinson, 2010]. However, physician reviewers determined that 44% of adverse and temporary harm events were clearly or likely preventable [Levinson, 2010]. Preventable events are often linked to the lack of patient monitoring and assessment.

One challenging and very interesting goal is to be able to predict the patients' outcomes and tailor the care that certain patients receive if it is believed that they will do poorly without additional intervention. In doing so, hospitals could prevent unnecessary readmissions, adverse events, or other delays in getting well [Schaeffer *et al.*, 2016]. For instance, if we can precisely identify groups of patients associated with a very high risk of requiring ICU treatment during their stay, then we can optimize their placement as soon as they are admitted, by affecting them e.g. to rooms closer to ICU, thereby drastically reducing transportation delay in life-critical situations in large hospitals. More generally, many complications could be avoided by immediate identification of at-risk patients upon admission and adapted prevention. A crucial prerequisite to any adapted and meaningful prevention is the precise identification of at-risk profiles.

The widespread adoption of Electronic Health Records (EHR) makes it possible to benefit from quality information provided by healthcare professionals [Hillestad *et al.*, 2005]. This opens the way for applying AI techniques in building helpful analytics systems for big medical data in which we can have a high level of trust – since drug prescriptions engage the responsibilities of healthcare professionals.

This paper aims to develop an automatic prediction system for identifying at-risk patients, based on a fine-grained analysis of large volumes of electronic health record data. This has long been viewed as a more challenging task than conventional prediction approaches with summary statistics and EHR-based scores [Schaeffer *et al.*, 2016; Frost and Sullivan, 2012]. We empirically demonstrate that medical big data can improve the automatic detection of at-risk patients.

Contributions. Our contributions are threefold:

1. we propose a method borrowing techniques from AI and machine learning for the automatic detection of at-risk patients based on a fine-grained analysis of prescription data at the time of hospital admission;
2. we develop a distributed architecture adapted for computing our fine-grained analysis on large volumes of medical records and clinical data;
3. we report on practical experiments with real data of millions of patients and hundreds of hospitals. We demonstrate how big data improves the detection of at-risk patients, making it possible to construct predictive models that benefit from volume and variety, while satisfying important criteria to be deployed in hospitals.

---

[1] Temporary harm events are those that require intervention but do not cause lasting harm.

2 Methods

We develop a supervised machine learning method that builds binary classification models to identify at-risk patients. Our method is implemented on top of a distributed architecture to ensure better scalability. The machine learning models built by our method yield predictions at hospital admission time.

2.1 Considered Medical Data

We consider real data from United States Hospitals, as provided by the Premier Perspective™ database, which is the largest hospital clinical and financial database in the United States. The database features more than 33 million discharges from a representative group of 417 hospitals drawn by lot. Each hospital submits quarterly updates of aggregated data. Patient-level data go through 95 quality assurance and data validation checks. Once the data have been validated, patient-level information is available, comprising data consistent with the standard hospital discharge file, demographic and disease state information, and information on all billed services, including date-specific logs of medications, laboratory, diagnostics, and therapeutic services. This database is developed and maintained by Premier Inc. and is well-known in clinical research (see e.g. [Delaney *et al.*, 2008]). The raw data for the year 2006 contains 33 048 852 admissions, and more than three billion patient charge records, representing 2.8 Tb of data.

For our study, we focused on basically two kinds of data: (1) population characteristics (age, gender, marital status, etc.) and (2) clinical data including all drug prescriptions (dosage, route of administration of each drug, etc.) for all admissions.

Filters. We selected adult and adolescent patients (between 15 and 89 years old[2]), hospitalized for more than 3 days. We chose this minimal length of stay of 3 days in order to ensure enough time for manifestation and detection of complications during the stay. Other exclusion criteria for the patients were:

- patients hospitalized in surgery, because in surgery medical prescription and its complexity varies considerably according to preoperative, operative and postoperative phase as described in [Lepelley *et al.*, 2016] and this information was not available in the database);
- out-patients and consultations;
- those with no drug prescription at admission; without which we cannot apply our analysis.

A total of 1 487 867 eligible admissions were retained. We also filtered clinical data to keep only drugs served.

Considered Complications and Ground Truth. To build the complication prediction system, we need labeled data for training and evaluation purposes. We consider four complications:

- death during hospital stay;

- admission to ICU on or after the second day (excluding patients directly admitted to ICU on the first day);
- pressure ulcers that were not present at admission time but developed during the stay;
- nosocomial infections developed during the stay.

Medical experts guided us to label these complications and establish a ground truth based on the International Classification of Diseases, Ninth Revision, Clinical Modification (ICD-9-CM) codes [Quan *et al.*, 2005] that are used by the database.

Participants and Occurrence of Complications. Figure 1 illustrates the repartition of eligible admissions by age and gender. Among this population, there were 44 667 cases of
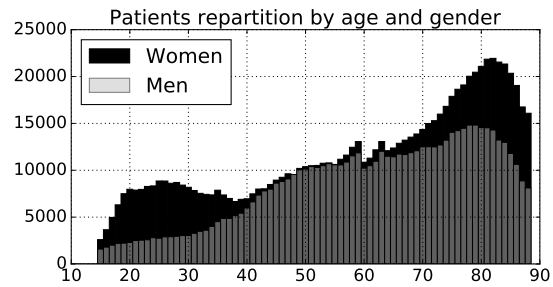

Figure 1: Admissions by Age and Gender.

hospital death (3.00%), 38 040 cases of pressure ulcers complications (2.55%), 50 876 cases of ICU admission on or after the second day (3.42%), and 37 866 cases of nosocomial infections (2.54%). On average, the probability that a patient experiences at least one complication among the ones that we consider during his hospital stay is 10.08%.

2.2 Prediction System

Distributed Architecture. The structure of our prediction system is illustrated in Fig. 2. Initial data consist in a set of raw relational tables, that we store in a NFS distributed file system. This file system communicates with Spark SQL [Armbrust *et al.*, 2015] that we use for data preprocessing, integration, and filtering. The feature engineering and classification components are hand coded in Spark [Zaharia *et*
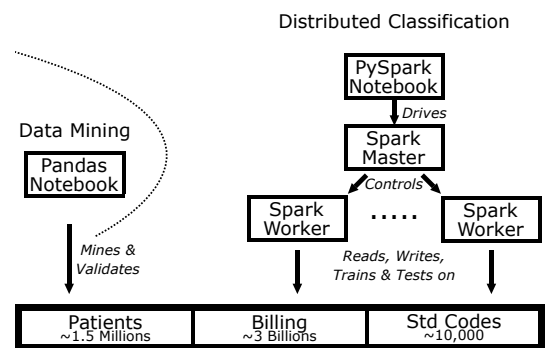

Figure 2: Architecture of Prediction System.

---

[2]We filtered out other ages because this information was biased in the database, i.e. age 89 denoting in fact age category 89+.

al., 2016] and MLlib [Meng *et al.*, 2015] and are based on distributed implementations of some widely used algorithms such as Logistic Regression (LR) [Cox, 1958], Linear Support Vector Machines (LSVM) [Suykens and Vandewalle, 1999], Decision Trees (DT) [Quinlan, 1986] and FP-Growth [Han *et al.*, 2004]. We also used TSNE [Maaten and Hinton, 2008] and facilities provided by Pandas and scikit-learn libraries on smaller excerpts of data that were extracted and preprocessed with Spark. For optimizing the representation of features, we also use a library for perfect hashing, that we modified and upgraded for use in our Spark and Python environments, based on the work found in [Czech *et al.*, 1997].

We use Docker [Merkel, 2014] to improve the runtime performance of our distributed architecture (mainly input/output) compared to a traditional approach with virtual machines. We automatically deploy custom Docker images on each machine of the cluster. On the application layer, we use Jupyter Notebooks as a prototyping frontend.

Feature Engineering. For each admission, we retained the following categories of features:

- A list `B` of basic features including patient age, gender, and admission type (e.g. whether the patient is admitted from a doctor's office and requiring acute care for e.g. pneumonia or dehydration; or whether the patient in life-threatening condition such as accident victim, etc.).
- A score `M` that corresponds to MRCI at admission. The Medication Regimen Complexity Index (MRCI) [George *et al.*, 2004] is one of the most valid and reliable scale for assessing regimen complexity [Paquin *et al.*, 2013]. It is a global score aggregating 65 sub-items for the purpose of indicating the complexity of a prescribed medication regimen. The MRCI has 3 sections giving information on the dosage form (section A), dosing frequency (section B) and additional instructions (section C) with 32, 23 and 10 items respectively. Each section reflects a different aspect of the complexity of prescription regimen. The total MRCI score is the sum of subscores for the 3 sections[3]. While the minimum total score is 2 (e.g. one tablet taken once a day as needed), there is no maximum score. MRCI is frequently used and readily available in hospital EHRs.
- A list `C` of clinical quantities associated with each drug served during the first day. Since there are many (more than 10 thousands) drugs possibly prescribed during the first day, we adopt a sparse representation when preprocessing features.

We performed data mining on the database and used feature ranking to select basic features among the population characteristics. A 7.9% overall correlation was found between patient's age and occurence of death during hospital stay. A 4.7% overall correlation was found between the MRCI value at admission and occurence of death during hospital stay. We retained MRCI as a marker of risk, and also include it for the purpose of comparison with earlier works

---

[3]The considered database lacks data required for computing the subscore for section C, which we thus arbitrarily set to zero. In the sequel, the total MRCI score is thus the sum of sections A and B.

Feature Index	Feature Description	Feature Value	Standard Charge Master Code
0	Age	15	
1	Gender (1 for male, 0 for female)	1	
2	MRCI	24	
8024	DEXTROSE/NACL SOLUTION 1000ML	1.00	250258000970000
7955	NACL SOLUTION 100ML	2.50	250258000220000
7949	NACL SOLUTION 1000ML	1.00	250258000160000
7084	DOCUSATE NA CAP 100MG	1.00	250257020020000
6654	ACETAMIN TAB 325MG (EA)	2.00	250257000530000
4869	SOD BICARB INJ 8.4% 50MEQ 50ML	1.00	250250058740000
4332	POT CHL VL 20MEQ 10ML	0.50	250250053100000
3566	MORPHINE TAB SR 30MG	0.50	250250044450000
5871	PEG-ES, MIRALAX PWDR 255GM	0.07	250250100890000
3483	MISC TOPICAL	1.00	250250043560000
1563	DIPHENHYD INJ 50MG	1.00	250250019540000
1292	CYCLOBENZAPRINE TAB 10MG	2.00	250250016100000
2882	LANSOPRAZOLE CAP 30MG	1.00	250250036600000
784	CEFEPIME VL 1GM	2.00	250250010280000
3093	LORATADINE TAB 10MG	1.00	250250038770000
134	ALLOPURINOL TAB 100MG	1.00	250250001580000

Table 1: Sparse feature vector for a sample patient who was served 16 drugs on the day of admission.
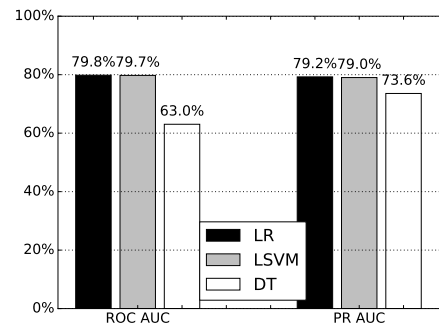

Figure 3: Comparison of Different Classifiers.

such as [Lepelley *et al.*, 2016]. In the sequel we investigate and report to which extent these correlations can actually be exploited for prediction purposes. We also investigate the benefits of considering all clinical quantities associated with drugs served on the first day, which is made possible by our distributed architecture.

Table 1 illustrates a sample feature vector for a 15 years-old patient, who was served 16 drugs on the day of admission.

Classifiers. For experiments we report, we choose a distributed LR classifier to make predictions. Specifically, we use the MLlib distributed implementation [Meng *et al.*, 2015; Zaharia *et al.*, 2016] of the LR classifier [Cox, 1958]. The reason of this choice is that LR yields the best predictive performance among several widely-used classifiers (See § 3.1 for comparative metrics and Fig. 3 for a comparison). Notice that LSVM also yields a very similar predictive performance. When predictive performances are equivalent, we favor LR because its raw output has a probabilistic interpretation.

Like several other standard machine learning methods, LR can be formulated as a convex optimization problem, i.e. the task of finding a minimizer of a convex function f that depends on a variable vector w which has d entries. More for-

mally this can be written as the optimization problem

$$\min_{\boldsymbol{w} \in \mathbb{R}^d} f(\boldsymbol{w})$$

in which the objective function f is of the form:

$$f(\boldsymbol{w}) = \lambda R(\boldsymbol{w}) + \frac{1}{n} \sum_{i=1}^{n} L(\boldsymbol{w}; \boldsymbol{x_i}, y_i)$$

where the vectors $\boldsymbol{x_i} \in \mathbb{R}^d$ are the training data examples, for $1 \le i \le n$, and $y_i \in \mathbb{R}$ are their corresponding labels, which we want to predict; and the logistic loss function L is of the form:

$$L(\boldsymbol{w}; \boldsymbol{x}, y) = \log(1 + \exp(-y\boldsymbol{w}^T \boldsymbol{x}))$$

Given a new data point, denoted by $\boldsymbol{x}$, the LR model makes predictions by applying the logistic function:

$$f(z) = \frac{1}{1 + e^{-z}}$$

where $z = \boldsymbol{w}^T \boldsymbol{x}$. We eventually use a threshold t such that if $f(\boldsymbol{w}^T x) > t$, the outcome is predicted as positive, or negative otherwise. By default the Spark/MLlib implementation sets $t = 0.5$. We make t vary to compute ROC curves and report AUC in the sequel. Notice however that the raw output of the logistic regression model, $f(z)$, already has a probabilistic interpretation (i.e. the probability that x is positive).

Cross-Validation, Rebalancing and Normalization. We perform cross-validation: we separate training and testing subsets and we use only the training subset to fit the model and only the testing subset to evaluate the accuracy of the model. We pick the training and testing subsets randomly.

There are many more patients without complication than patients experiencing complications during their stays (hopefully). To deal with this class imbalance, we apply downsampling on patients with no complication in order to rebalance classes before the random selection of the training subset. We apply feature normalization for the linear models.

3 Results

We now report on practical experiments with our system for predicting complications, while illustrating the benefits brought by volume and variety of considered medical data.

3.1 Performance Metrics

The system outputs a boolean prediction (either positive or negative) for each admission and for each complication. To evaluate prediction results, we use recall, precision and other standard metrics computed from confusion matrices [Fawcett, 2006; Powers, 2011]. In particular, we use the area under the ROC curve (AUC) evaluated on the test data, which is the standard scientific accuracy indicator [Guyon *et al.*, 2009]. The higher AUC indicates the better prediction performance. Intuitively, when using normalized units, AUC is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one [Fawcett, 2006]. We also use the area under the precision-recall curve (PR-AUC) as an additional insight
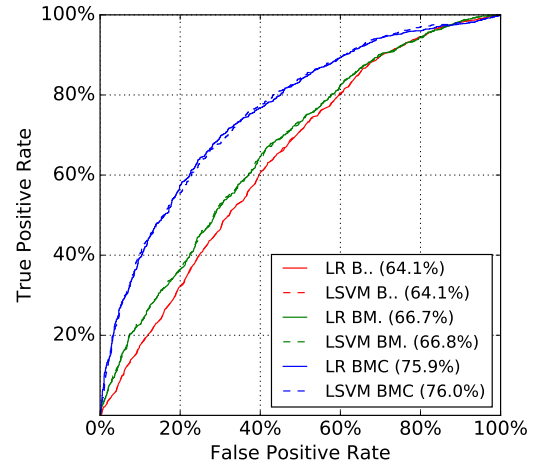

Figure 4: Impact of More Finer-Grained Features (Variety) on AUC for Predicting Mortality.

(though for any dataset, the ROC curve and PR curve for a given algorithm contain the same points [Davis and Goadrich, 2006]).

We use precision, recall, AUC, and PR-AUC to evaluate the overall predictive performance in terms of a large variety of features and a large volume of training data.

3.2 Variety

We investigate the impact of considering more finer-grained features on accuracy when predicting complications. In other terms, we examine whether considering more features (Variety) per instance yields a better predictive performance. Fig. 4 presents ROC curves and AUC results for mortality prediction when we consider the list of basic features (B) for each patient, the MRCI score (M), clinical quantities (C), and combinations of them. The more finer-grained features we consider the better predictive performance we obtain; the best predictive performance being obtained with the combination of all features (B+M+C). In particular, we observe that the detailed clinical quantities yield a significant increase in predictive performance compared to just MRCI (Fig. 4). We obtain similar gains when predicting other complications.

3.3 Volume

The experiments reported above confirm that Variety (the number and granularity of features) can improve the predictive modeling accuracy. An alternative perspective on increasing the number of features is to increase the number of training instances (Volume) used to construct the model.

Figure 5 presents AUC, Recall and Precision results with train datasets of increasing sizes. We evaluate our models on the same randomly chosen test subset of 3200 instances, while we increase the train dataset size, as reported on the x-axes of the graphs of Fig. 5 that indicate the number of instances in the training set. Increased volume tends to improve predictive performance.
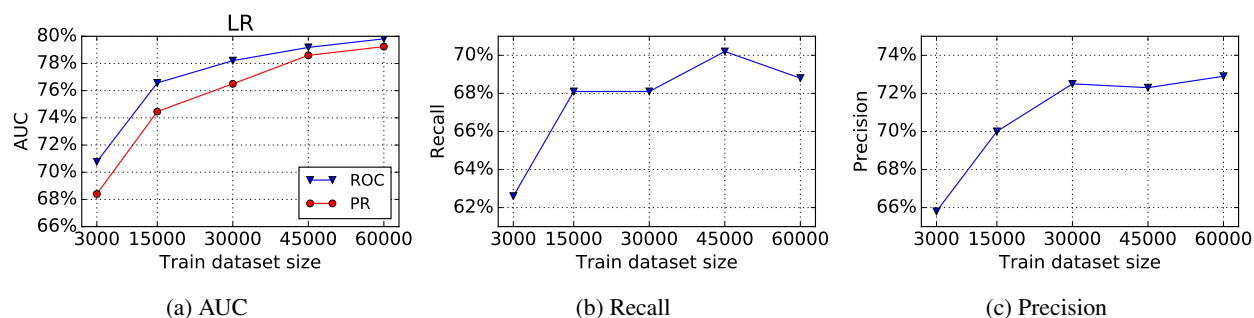
(a) AUC   (b) Recall   (c) Precision

Figure 5: Increased Volume Leads to Better Predictive Performance.
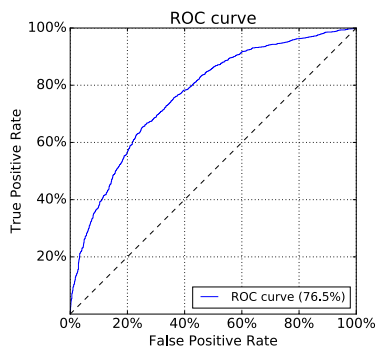

Figure 6: Predicting Hospital Death (AUC > 76%).

3.4 Predictions of Complications

Our system builds (learns) a specific model for each complication. We now examine and evaluate the predictions for the different complications that we consider. We performed extensive tests using cross-validation methodology (see § 2.2), and we report on accuracy obtained from randomly choosen training and testing subsets.

Figure 6 shows a ROC curve obtained when predicting hospital death. We obtain an AUC greater than 76%. Fig. 7 shows ROC curves obtained when predicting occurrence of the other complications during hospital stay, respectively: pressure ulcers, ICU admissions, and nosocomial infections.

Overall, the system exhibits best performance for predicting the occurrence of nosocomial infections, pressure ulcers, and hospital deaths. Table 2 further illustrates detailed metrics on randomly selected datasets, with a threshold $t = 0.5$. On average, the time spent in constructing the LR model is 4.5 minutes (excluding data preparation time).

3.5 System Deployment: Lessons Learned

An advantage of our system is that of being adapted for deployment in hospitals, because it is implementable in-house with off-the-shelf big data frameworks (e.g. Spark) running with modest hardware. In particular, our system does not require neither external storage of sensitive medical data (thus avoiding additional issues of confidential data leakage), nor very expensive hardware (thanks to the distribution of data and computations).

Distribution and Cost-Effectiveness. All the computations have been performed on a cluster composed of 2 commodity machines totalizing 48 cores and 218Gb of RAM[4]. We have configured Spark so that data are split into 400 partitions and each Spark executor uses 20Gb of RAM.

Two tasks were the most computationally-intensive: (i) prefiltering and transformation of data (as explained in § 2.1 and 2.2), and (ii) the construction (learning) of models. Still, our system manages to compute the models in less than 300 seconds, as shown on Fig. 8. Figure 8 also illustrates how the distribution of data and computations in our system is beneficial for constructing models and normalizing features. Once the model is generated, it can be saved and transmitted (e.g. in PMML standard format) so that it can be executed on a different single machine for computing predictions.

It is worth noticing that the distribution of data and computations was instrumental in achieving all the aforementioned results[5]. If equivalent computations were to be performed on a single machine, then this machine would have to be an order of magnitude more powerful (and expensive), making the whole system less easily deployable in the setting of an hospital. The use of Docker also facilitates deployment on commodity and heterogenous machines.

4 Related Work

With the broad adoption of EHRs systems, the development of techniques for improving the quality of clinical care has received considerable interest recently, especially from the AI community [Lee *et al.*, 2016; Kuang *et al.*, 2016; Li *et al.*, 2016; Luo *et al.*, 2016; Zhang *et al.*, 2016].

The work found in [Luo *et al.*, 2016] adresses ICU mortality risk prediction with unsupervised feature learning techniques from timeseries of physiologic measurements (whereas we consider supervised techniques on prescribed drug data at admission). In [Lee *et al.*, 2016] a method is introduced for the purpose of extracting phenotype information from EHRs and for providing analyses on phenotypes.

---

[4]Each machine is equipped with 2 Intel(R) Xeon(R) 1.90GHz with 24 cores and either 156Gb RAM or 62 Gb RAM.

[5]Initial attempts with the Pandas library on a single machine with 160 Gb of RAM were non-conclusive. Only a fraction of the dataset was fitting in memory (after joining and filtering made in Spark) and yet no transformation requiring copies (e.g. joins) was possible. We tried to compute joins by chunks and finally stopped the computation after 3 days (for an estimated time of at least 6 days).
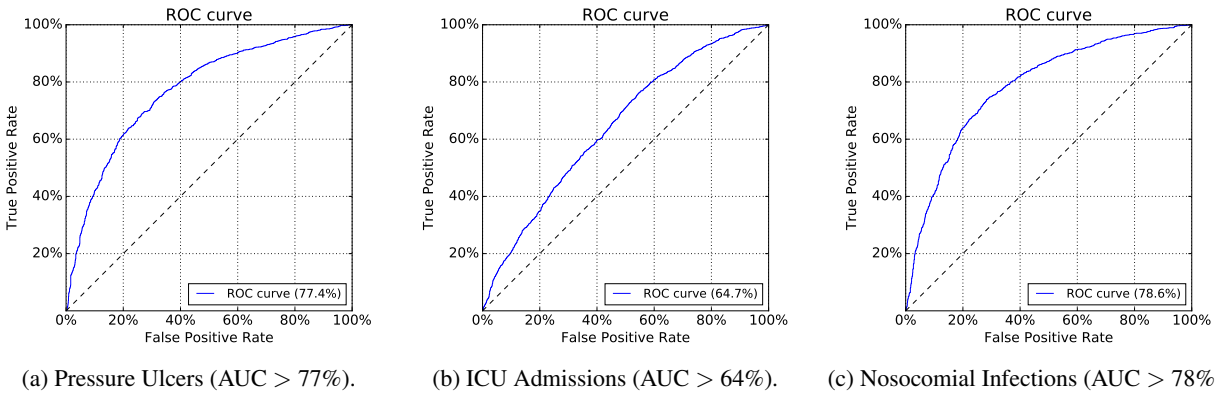
(a) Pressure Ulcers (AUC > 77%). (b) ICU Admissions (AUC > 64%). (c) Nosocomial Infections (AUC > 78%).

Figure 7: Predictions of Complications with `BMC` Features.

Metric	Death	ICU	Pressure Ulcers	N. Infection	Metric Definition	Metric Aliases
True Positive Rate	66.9%	60.6%	61.9%	65.1%	TP/P	Hit Rate, Recall, Sensitivity
True Negative Rate	72.8%	58.2%	79.7%	78.9%	TN/N	Specificity
False Positive Rate	27.2%	41.8%	20.3%	21.1%	FP/N	Fall-out
False Negative Rate	33.1%	39.4%	38.1%	34.9%	FN/P	Miss Rate
Negative Predictive Value	67.8%	59.5%	66.9%	68.8%	TN/(TN+FN)	
Positive Predictive Value	71.9%	59.3%	75.9%	76.0%	TP/(TP+FP)	Precision
False Discovery Rate	28.1%	40.7%	24.1%	24.0%	FP/(TP+FP)	
Accuracy	69.8%	59.4%	70.6%	71.9%	(TP+TN)/(P+N)	
Error	30.2%	40.6%	29.4%	28.1%	(FP+FN)/(P+N)	

Table 2: Detailed LR prediction metrics on random train and test subsets (threshold=0.5), with notations adopted from [Fawcett, 2006]: TP is the number of true positives, FP: false positives, TN: true negatives, FN: false negatives, P=TP+FN and N=FP+TN.
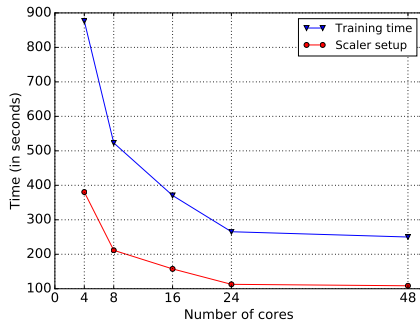

Figure 8: Elapsed Times and Number of Cores.

In [Kuang *et al.*, 2016], a baseline regularization model is proposed for the task of finding new indications for existing drugs leveraging heterogeneous drug-related data based on EHRs. In [Li *et al.*, 2016] joint models are explored for extracting mentions of drugs and their side effects, such as diseases that they cause. In [Zhang *et al.*, 2016] survival prediction is explored with a focus on intermittently varying data.

Our work fundamentally differs from previous works by the initial assumed postulate from which we start. We formulate the hypothesis that the information required for identifying at-risk profiles is available in the initial patients' drug prescription data at the time of hospital admission. In other terms, our system exclusively relies on analysing prescribed drug data of the day of admission. Previous studies that seeked to exploit EHR information with a similar postulate have mainly been developing score-based techniques.

For instance the works found in [Schoonover *et al.*, 2014; Willson *et al.*, 2014; Yam *et al.*, 2016; Lepelley *et al.*, 2016] also assume that the complexity of the patient's medication regimen is a good indicator of the complexity of the patient's condition. The existence of correlations between MRCI at admission and occurence of complications is empirically demonstrated in [Lepelley *et al.*, 2016]. Compared to these works, we go further by (1) exploring how this information can be leveraged for predictive purposes (on large datasets) and (2) by considering finer-grained features, thanks to our distributed architecture, which allows to improve prediction accuracy. A simple score such as MRCI constitutes a rough approximation. For example, the same MRCI value may denote different situations with radically different evolution perspectives. Our fine-grained approach is more adapted to capture these differences. We showed that this leads to increased prediction accuracy.

5 Conclusion

We proposed a novel method for identifying patients at risk of complications during their hospital stay, which is based exclusively on data of the day of admission. We tested our method with the largest US hospital clinical and financial database. Experimental results suggest that our system might be especially useful for detecting patient profiles at risk of nosocomial infections, pressure ulcers and death. We also illustrated how the volume and variety perspectives of big medical data can facilitate the automatic identification of at-risk patients, for the purpose of developing adapted prevention.

References

[Armbrust *et al.*, 2015] M. Armbrust, R. S. Xin, C. L., Y. H., D. L., J. K. Bradley, X. Meng, T. Kaftan, M.J. Franklin, A. Ghodsi, and M. Zaharia. Spark SQL: relational data processing in Spark. In *SIGMOD*, pages 1383–1394, 2015. url.

[Cox, 1958] D.R. Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, 20(2):215–242, 1958. doi.

[Czech *et al.*, 1997] Z.J. Czech, G. Havas, and B.S. Majewski. Perfect hashing. *Theor. Comput. Sci.*, 182(1-2):1–143, 1997. url.

[Davis and Goadrich, 2006] J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. In *ICML*, pages 233–240, 2006. doi.

[Delaney *et al.*, 2008] C.P. Delaney, E. Chang, A.J. Senagore, and M. Broder. Clinical outcomes and resource utilization associated with laparoscopic and open colectomy using a large national database. *Annals of Surgery*, 247(5), 2008. url.

[Fawcett, 2006] T. Fawcett. An introduction to roc analysis. *Pattern Recogn. Lett.*, 27(8):861–874, jun 2006. doi.

[Frost and Sullivan, 2012] Frost and Sullivan. Drowning in big data? reducing information technology complexities and costs for healthcare organizations. Technical report, Frost & Sullivan Company White Paper, 2012. url.

[George *et al.*, 2004] J. George, Y.T. Phun, M. J Bailey, D. CM Kong, and K. Stewart. Development and validation of the medication regimen complexity index. *Annals of Pharmacotherapy*, 38(9):1369–1376, 2004. url.

[Guyon *et al.*, 2009] I. Guyon, V. Lemaire, M. Boullé, G. Dror, and D. Vogel. Analysis of the KDD cup 2009: Fast scoring on a large orange customer database. In *Proc. of KDD-Cup competition*, volume 7 of *JMLR Proceedings*, pages 1–22, 2009. url.

[Han *et al.*, 2004] J. Han, J. Pei, Y. Yin, and R. Mao. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Min. Knowl. Discov.*, 8(1):53–87, 2004. url.

[Hillestad *et al.*, 2005] R. Hillestad, J. Bigelow, A. Bower, F. Girosi, R. Meili, R. Scoville, and R. Taylor. Can electronic medical record systems transform health care? potential health benefits, savings, and costs. *Health Affairs*, 24(5):1103–1117, September 2005. url.

[Kuang *et al.*, 2016] Z. Kuang, J. A. Thomson, M. Caldwell, P. L. Peissig, R. M. Stewart, and D. Page. Baseline regularization for computational drug repositioning with longitudinal observational data. In *IJCAI*, pages 2521–2528, 2016. url.

[Lee *et al.*, 2016] W. Lee, Y. Lee, H. Kim, and Il-C. Moon. Bayesian nonparametric collaborative topic poisson factorization for electronic health records-based phenotyping. In *IJCAI*, pages 2544–2552, 2016. url.

[Lepelley *et al.*, 2016] M. Lepelley, C. Genty, A. Lecoanet, B. Allenet, P. Bedouch, M.R. Mallaret, P. Gillois, and J.L. Bosson. Use of the electronic medication regimen complexity index at admission to predict complications during hospitalization in medical wards. Technical report, Univ. Grenoble Alpes and CHU Grenoble, Pôle Santé Publique, Centre de pharmacovigilance, 2016.

[Levinson, 2010] D.R. Levinson. Adverse events in hospitals: National incidence among medicare beneficiaries. Technical report, Department of Health and Human Services, USA, 2010. url.

[Li *et al.*, 2016] F. Li, Y. Zhang, M. Zhang, and D. Ji. Joint models for extracting adverse drug events from biomedical text. In *IJCAI 2016*, pages 2838–2844, 2016. url.

[Luo *et al.*, 2016] Y. Luo, Y. Xin, R. Joshi, L. Celi, and P. Szolovits. Predicting ICU mortality risk by grouping temporal trends from a multivariate panel of physiologic measurements. In *AAAI*, pages 42–50, 2016. doi.

[Maaten and Hinton, 2008] L.v.d. Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[Meng *et al.*, 2015] X. Meng, J.K. Bradley, B. Yavuz, E.R. Sparks, S. Venkataraman, D. Liu, J. Freeman, D.B. Tsai, M. Amde, S. Owen, D. Xin, R. Xin, M.J. Franklin, R. Zadeh, M. Zaharia, and A. Talwalkar. Mllib: Machine learning in Apache Spark. *CoRR*, abs/1505.06807, 2015. url.

[Merkel, 2014] D. Merkel. Docker: Lightweight linux containers for consistent development and deployment. *Linux J.*, 2014(239), March 2014. url.

[Paquin *et al.*, 2013] A.M. Paquin, K.M. Zimmerman, T.R. Kostas, L. Pelletier, A. Hwang, M. Simone, L.M. Skarf, and J.L. Rudolph. Complexity perplexity: a systematic review to describe the measurement of medication regimen complexity. *Expert Opinion on Drug Safety*, 12(6):829–840, 2013.

[Powers, 2011] D.M.W. Powers. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[Quan *et al.*, 2005] H. Quan, V. Sundararajan, P. Halfon, A. Fong, B. Burnand, J.-C. Luthi, L. Duncan Saunders, C.A. Beck, T.E. Feasby, and W.A. Ghali. Coding algorithms for defining comorbidities in ICD-9-CM and ICD-10 administrative data. *Med Care*, 43(11):1130–1139, Nov 2005. url.

[Quinlan, 1986] J.R. Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[Schaeffer *et al.*, 2016] C. Schaeffer, A. Haque, L. Booton, J. Halleck, and A. Coustasse. Big data management in united states hospitals: Benefits and barriers. In *Proceedings of the Business and Health Administration Association Annual Conference*. J. Sanchez (Ed.), April 2016. url.

[Schoonover *et al.*, 2014] H. Schoonover, CF. Corbett, DL. Weeks, MN. Willson, and SM. Setter. Predicting potential postdischarge adverse drug events and 30-day unplanned hospital readmissions from medication regimen complexity. *J Patient Saf.*, 10(4):186–91, 2014. url.

[Suykens and Vandewalle, 1999] J.A.K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[Willson *et al.*, 2014] MN. Willson, CL.Greer, and DL. Weeks. Medication regimen complexity and hospital readmission for an adverse drug event. *Ann Pharmacother.*, 48(1):26–32, 2014. url.

[Yam *et al.*, 2016] FK. Yam, T. Lew, S.A. Eraly, H.W. Lin, J.D. Hirsch, and M. Devor. Changes in medication regimen complexity and the risk for 90-day hospital readmission and/or emergency department visits in U.S. veterans with heart failure. *Res Social Adm Pharm.*, 12(5):713–21, 2016. url.

[Zaharia *et al.*, 2016] M. Zaharia, R. S. Xin, P. Wendell, T. Das, M. Armbrust, A. Dave, X. Meng, J. Rosen, S. Venkataraman, M.J. Franklin, A. Ghodsi, J. Gonzalez, S. Shenker, and I. Stoica. Apache spark: a unified engine for big data processing. *Commun. ACM*, 59(11):56–65, 2016.

[Zhang *et al.*, 2016] J. Zhang, L. Chen, A. Vanasse, J. Courteau, and S. Wang. Survival prediction by an integrated learning criterion on intermittently varying healthcare data. In *AAAI*, pages 72–78, 2016. doi.