Medication Recommendation for Critical Care Patients Using Patient Similarity in Clinical Records

Mohammad M. Masud[1*], Muhsin Cheratta[1], Kadhim Hayawi[2]

[1] College of Information Technology, United Arab Emirates University, PO Box 15551, Al Ain, UAE.
[2] College of Technological Innovation, Zayed University, P.O. Box 144534, Abu Dhabi, UAE.

* Corresponding author. Tel.: +97137135547; email: m.masud@uaeu.ac.ae
Manuscript submitted April 23, 2019; accepted October 29, 2019.
doi: 10.17706/ijbbb.2020.10.1.1-14

Abstract: This paper presents an effective clinical decision support technique for recommending medications for critical care patients by combining various clinical data of the patients. Specifically, our goal is to infer necessity of medications for intensive care unit (ICU) patients by looking into similar patients' medication. The patient similarity is inferred utilizing their first 24-hour clinical data. Modern ICUs are equipped with numerous monitoring devices, including devices that monitor vital signs such as heart rate, blood pressure, Oxygen saturation and so on. Patients also undergo different pathological (i.e., laboratory) tests, medications and nursing. In this work, we utilize a combination of several of these data for inferring the necessity of prescribing or administering certain medication. We encounter several challenges associated with this task, including heterogeneous sources of features such as lab test results, prescription, fluid intake and fluid output, missing values, class imbalance and high dimensionality. We address some of these challenges by feature selection, missing value imputation and class balancing. We investigate the effectiveness of using patient similarity network and fusion of these networks for the recommendation task. We also study the effectiveness of ensemble classification models, where each model is trained from a specific set of features. When recommending certain medication for a new patient, we first find the most similar patients of the query patient based on the first 24-hour clinical data and recommend or decline based on the medication data of those patients. The proposed technique has been evaluated on a real ICU patients' database and exhibits AUROC score of up to 0.83 for certain prescription drugs.

Key words: Healthcare decision support, patient similarity network, patient grouping, survival prediction.

1. Introduction

Recent advances in intelligent data analytics, availability of high-performance computing and abundance of clinical data have encouraged multitude of research works in clinical decision support systems. Healthcare data are generated from many different sources and in different formats such as electronic health records (EHR), medical imaging, medical text data (e.g. nurse notes), and public health data. In this work, we are interested mainly on utilizing various clinical records of intensive care unit (ICU) patients for recommending medications. Modern ICUs are equipped with various sensors and monitoring devices, and therefore ICU patients are usually always under intensive observation. Therefore, many different forms of data are generated by the patients, such as time series data (e.g. vital signs), pathological test results, periodic administration of medication, and fluid intake and output data. In addition to these, there are nurse notes, demographic data as well as administrative (e.g. admissions) and procedural (e.g. caregiver

name) information.

Different forms of the patient data are being used by researchers to achieve different goals. For example, vital signs have been used to predict patient situation (e.g. deterioration) ahead of time that can warn caregivers with plenty of time to react and save the patient's life [1]. Lab test results also have been used to predict patient survival probabilities [2]. However, in this work we address a different problem, namely, recommendation of medication for ICU patients using different clinical data sources. The motivation is that administration of correct and necessary medication for ICU patient is critical. An effective recommender system would be beneficial for caregivers by aiding them in this critical decision making. We have shown that combining clinical data from different sources can improve the effectiveness of the recommendation. We utilize different data sources, including medication and fluid input data, fluid output data, and prescriptions.

There are many challenges involved in building such a system, some of which are mentioned in the literature ([1], [3]-[5]). The first challenge is that of constructing a uniform feature vector. All machine learning tasks require instances to have uniform feature vector. For example, if we consider each patient as an instance, and use his lab test results as features, then we must make sure that for all the patients we use the same set of lab tests as features. However, this may result in many missing values because a particular lab test may not be done for all the patients. In addition, it is also possible that the same lab test has been done more than once (e.g. blood sugar), which brings the problem of multiple values for the same feature. Another problem is that of high dimensionality, which occurs because of many possible data sources and data variations. Finally, class imbalance is also a significant challenge, which is a common phenomenon in medical data.

In this work, we address mainly the first three challenges, i.e., missing values, high dimensionality, and class imbalance. The first two challenges are interrelated, because the missing values are introduced mainly because of high dimensionality, and they bring noise, redundancy, and sparsity in the dataset. Therefore, feature selection is applied to choose the features that contribute most for inferring the target class. We also get rid of features that have too many missing values. We address the class balancing problem by selecting under-sampling of instances from a larger dataset.

Our contributions are as follows. First, we propose a technique for suggesting medications for ICU patients based on different clinical data. In this work, we only use the first 24-hour clinical data for building this recommendation system. We believe this system will be useful for caregivers in taking medication decisions carefully and effectively. Second, we use patient similarity for building the recommender system, which tries to build a network of patient similarity based on the available feature values. Patient similarity has been recently used successfully for cancer prediction and other clinical decision support systems [6], [7]. Finally, we exhibit three different techniques to combine data from multiple sources and the effectiveness of each such technique. These techniques are incremental feature construction, ensemble construction and Affinity Network Fusion (ANF). To the best of our knowledge, this is the first work to propose a clinical decision support specifically for recommending medications using patient similarity.

The rest of the paper is organized as follows. Section 2 discusses the works relevant to our technique. Section 3 describes the proposed approach in detail. Then Section 4 reports the experiment details and analyzes the results. Finally, Section 5 concludes with directions to future research.

2. Related Work

We discuss the related works under three broad categories. The first category deals with different clinical and healthcare support aspects of ICU patients. The other category of related work applies machine learning in general for medical decision support including recommender systems. And the third category

discusses the recent advances in patient similarity network approach for clinical decision support.

First, we discuss the works that specially deal with ICU patients. Mao *et al*. [1], developed a data-mining approach to predict deterioration of patients in the ICU. They used time-series data obtained from different sensors attached to patient's body, such as blood pressure, heart rate, O2 saturation and so on. They preprocessed these time-series data and derived several features. Finally, they applied different feature selection and optimization techniques to build prediction model, which observes good prediction rate. Pirracchio *et al*. [8] proposed a learning algorithm to predict mortality of ICU patients and successfully used in real hospitals. Cismondi *et al*. [9] addressed a different goal involved with ICU patients, namely, how to reduce unnecessary lab testing in the ICU. However, they only focus on gastrointestinal bleeding. In our work, we are targeting all cases in the ICUs.

Purushothan *et al*. [10] discussed a benchmark work on a big ICU database, which propose a deep learning model for survival prediction, ICU stay length prediction and ICD9 prediction task. This work applies extensive data preprocessing and provides comprehensive study on the observed results.

Some other relevant work dealing with ICU patients are as follows. Fialho *et al*. [3] proposed a feature selection technique to find the best features in order to predict ICU readmissions. Cheng *et al*. [5] proposed a clinical decision support system using association rule mining that finds associations among various variables such as patients' conditions, length of ICU stays and so on, and reports interesting findings.

Our proposed work is different from the above in that our goal is to develop a recommender system to complement caregiver decision making in prescribing and administering drugs, whereas the above systems deal with mainly three goals, namely, patient survival prediction, ICU readmission prediction and length of hospital stay prediction. Also, we study a combination of different data sources rather than only vital signs or demographic data.

The other category of related work are all approaches that in general deal with data mining-based solutions for developing clinical decision support systems. Herland *et al*. [11] did a comprehensive survey on this topic, i.e., clinical data mining applications on big data in health informatics. Celi *et al*. [12] applied a statistical approach to predict mortality among patients with acute kidney injury. Cai *et al*. [13] proposed a Bayesian network approach to develop models using EHR for real-time prediction of several targets, including length of hospital stay, mortality, and readmission of hospitalized patients.

Other related applications of machine learning are in the recommender systems. Recommender systems predict the relevance of an item (e.g. medication, drug, therapy, movies, music, research article, social tag, twitter pages) for a give user (e.g patient, user of e-business system) using various machine learning and information retrieval tools [14]. They have been evolved in recent years with real-world applications in on-line shopping, e-commerce, and entertainment areas to solve the problem of information overload [15], [16]. Specifically, the recommender systems are used mainly to suggest an item of interest for a user of information system or Web applications such as Amazon [17], Google [18], and Netflix [19] recommender systems. However, their applications in healthcare are still in its early stages [20].

The main approaches to design recommender systems fall into three categories namely, Content-Based Filtering, Collaborative Filtering, and Hybrid recommender systems [21]. The Content-Based approach is based on collecting and analyzing user's data, activities, and profile to predict the rating of items of interest [22]. Consequently, items that are rated high in the past will be recommended by these systems. The Collaborative filtering [23], which is the most widely used recommender system, first attempt to find other users u_i whose profiles are like that of the user u whom they give a recommendation to. Then, to foresee user u ratings on items of interest, the Collaborative Filtering systems utilize known ratings of these items that are assigned by users u_i. The Hybrid combines both Content-Based Filtering and the Collaborative Filtering techniques [24], [25]. Our work shares some commonalities with the Collaborative Filtering

approach by finding patient similarity based on available clinical feature values. However, it stands out in two ways. Firstly, for similarity computation it combines the data from different sources rather than using only patient profile. Secondly, to the best of our knowledge it is the first to support clinical decision making by recommending medications for ICU patients. Depending on the type of the end user, the proposed clinical recommender system could be used for different purposes. For example, it could be used for clinical decision support, as in our work, if the end users are health professionals. On the other hand, when the end users are patients, it could be utilized to recommend healthy lifestyle that is customized for each patient case [20].

Next, we discuss different types of Patient Similarity Networks and their applications. Patient Similarity Networks (PSNs) are novel paradigm that converts heterogeneous clinical data types into a comprehensive network views which facilitate clinical decision making [6], [26]. Given two or more data types, the PSN builds a network for each data type by adding an edge between each two patients who have pairwise similarity in a given feature. Then it fuses these networks into a single similarity network. PSNs outperform the traditional machine learning because it makes that data more interpretable and surpass the single data type analysis by fusing data from different sources or types to address the challenges of noisy and incomplete data [26], [27].

PSN is an emerging framework and there is a limited prior work that used it for clinical decision support [6], [7], [28]. Pai *et al*. [7] did a survey on different approaches of clustering and classification for clinical decision support using PSN. Wang *et al*. [6] uses PSN as a clustering algorithm to group similar cancer patients into different similarity network for each type of data, then fuses mRNA expression, DNA methylation, and microRNA (miRNA) expression data to generate an overall view of a disease. Li *et al*. [28] identified new subtypes of type 2 diabetes based on EHR and genotype data from 11; 210 patients using PSN. Although we do not use PSN directly, we compare our work with a special implementation of PSN fusion called Affinity network Fusion (AFN) [29], which have been found not so effective for the clinical data types that we use.

3. Proposed Technique

In this section first we give an overview of the proposed technique and then describe it in detail.

3.1. Problem Statement

Given the clinical records of a patient X, and a target medication Y (e.g. Propofol), the problem is to infer whether Y is recommended for X or not. Therefore, it is a two-class classification problem. If the inferred value of $Y = 0$ (i.e., negative class), then the medication is not recommended, and if $Y = 1$ (positive class), then it is recommended. We propose a classification model based on patient similarity in first 24-hour clinical data.

3.2. High Level Description

Fig. 1 shows the high-level architecture of our approach. The ICU patients' data are collected from MIMIC-III database. We use different tables from the database, which includes demographic data, lab test results, fluid intake and output data and medication data. First, we choose two target features for recommendation. Then we choose cohort of patients based on age. This follows selection of features for training. Then class balancing is applied using selective under-sampling of subjects. This follows feature extraction from the processed data. A separate feature vector is constructed per table (e.g. medication data). This is followed by missing value imputation. Then we systematically combine the individual feature vectors and generate classification models using three different techniques, namely, incremental feature construction (IFC), ensemble construction (EC), and affinity network fusion (ANF). Finally, these models are

used to recommend medication of new ICU patients based on his clinical records.

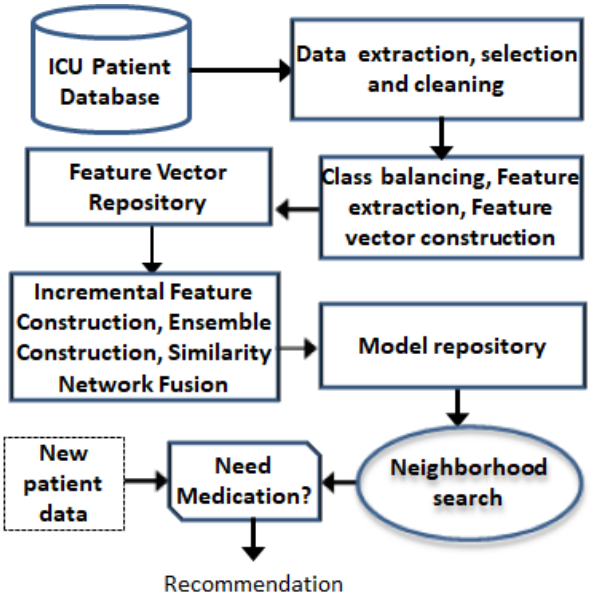

Fig. 1. Architecture of the proposed approach.

3.3. Database

The source of all datasets in this work is MIMIC-III [30]. The database is collected by the Multi-parameter Intelligent Monitoring in Intensive Care (MIMIC) project at the Laboratory of Computational Physiology at MIT, funded by the National Institute of Biomedical Imaging and Bioengineering. The data was collected between 2001 to 2012 and contain more than 50,000 hospital admissions.

The database consists of 26 different tables, each containing different types of information, such as demographic data, hospital admission data, ICU stay data, medication intake, fluid intake, and fluid output, lab test results, chart events, prescription, nurse notes and so on. Most of the tables are event-based, meaning; there is a record for every event. For example, a lab test is an event, which is a record in the database containing the patient ID, timestamp of the event, test ID (links to the type of the test), numeric test value, abnormality flag, unit of measurement and so on. Among these tables, we mainly use the tables containing numeric data, namely, *inputevents* (containing fluid intake data), *outputevents* (containing fluid output data), *chartevents* (containing all charted records), and *prescriptionevents* (containing prescription drugs data) as well as some demographic and admission related information from other tables. These tables contain most of the clinical records.

3.4. Data Selection, Extraction, and Cleaning

Target medications: We choose two targets that are frequently administered medications but used for two different purposes:

- Target 1: Docusate Sodium. This is a frequently administered drug obtained in the *inputevents* table. This medication is mainly used as a laxative.
- Target 2: Propofol. This is a prescription drug obtained in the *presecriptionevents*, which is mainly administered for initiating anesthesia.

Note that alternate medication exists for both docusate sodium and propofol. Therefore, recommendation of these medications is likely to be beneficial for the physician for choosing the right alternative.

Initial dataset selection: From the MIMIC-III database we choose only the patients with age 15+ as

these are the main age groups on which the target medication is applied. This gives us a dataset of about 35,000 patients.

Feature selection: From the initial dataset we retrieve raw data Feature selection: From the initial dataset we retrieve raw data from four different tables, namely, inputevents, outputevents, chartevents, and prescriptionevents. For each of these tables we prepare a separate feature vector for the patients. Then we apply feature selection in several stages. In the first stage, we discard features that contain more than 99% missing value. Still, most of the features remaining features had more than 90% missing value. Then we apply the second stage of selection by choose top 50 features based on their relevancy with the target attributes based on gain ratio criterion.

3.5. Class Balancing, Feature Extraction, and Feature Vector Construction

Class balancing: Based on the target attribute value, we can identify two classes of patients, namely, positive class where the target attribute is present (i.e., the medication was administered), and negative class where the target attribute is absent (i.e., medication was not administered). Then we apply class balancing by selective undersampling to keep the negative: positive class ratio 60: 40. This is explained in the next paragraph.

Second stage data selection: We select two subsets of data from the initial dataset of 35,000 in order to keep the class balance consistent (i.e., 60:40). The first subset consists of 6,000 patients, and the second subset consists of 12,000 patients, both randomly selected from the 35,000 pool. This gives us two datasets (6,000 and 12,000) for each target feature. The reason for choosing two different dataset sizes was to evaluate the effectiveness of the recommender system on varying size of training data.

Feature extraction: For each dataset mentioned above, we extract two kinds of features, namely, binary, and numeric. This is done as follows. First, for each selected feature, we extract only the first 24-hour records from corresponding database tables. When constructing the binary feature vector for a patient, we set 0 as feature value if no record (i.e., measurement) was found for the feature in the first 24 hours of hospital admission for the patient, and set 1 as feature value otherwise (i.e., if there is a record within first 24 hours). When constructing numeric feature vector, we identify a feature value as missing if no record of the feature is found in first 24 hours. Otherwise, if a record is found, we use the numeric value of the feature to be the feature value. If, however, more than one record is found (i.e., more than one measurement was taken in first 24 hours, which is very common for certain features like 'blood sugar test'), then we take the average of all such numeric values as the feature value. Note that this approach gives us four different datasets for each target feature, namely, binary-6000, numeric-6000, binary-12000 and numeric-12000. Also, note that the binary datasets have no missing values, but the numeric datasets have many missing values because it is possible that for many features there is no record in the first 24 hours for a patient. We generate these two different sets of features in order to understand each of their effect in constructing the recommendation system. Note that there are other unstructured data available for the patients (e.g. nurse notes) but we vow to address the issue of using these data for developing recommender systems in future.

Feature vector construction: For each of the datasets (e.g. binary-6000) we construct feature vector for each of the tables mentioned above (e.g. outputevents). This way we come up with three different feature vectors, which will be called as follows: OUT (corresponds to outputevents), IN (corresponds to inputevents), and PC (corresponds to prescriptions and chartevents combined). Missing value imputation and data normalization: For each numeric feature vector we apply standard imputation technique, for missing values, i.e., replace the missing value with the mean of the feature values. After this, we normalize all the features to obtain scaled value between 0 and 1.

3.6. Recommender Model Construction

Now we discuss our model construction techniques.

3.6.1. Notations and definitions

We denote $\mathcal{D}(X_i, X_j)$ to be the Euclidean distance between patient vectors X_i and X_j.

DEFINITION 1 (PATIENT SIMILARITY $\mathcal{S}(X_i, X_j)$): Patient similarity is based on Euclidean distance between two patient vectors, X_i and X_j, and is calculated as follows:

$$\mathcal{S}(X_i, X_j) = exp[-\mu \mathcal{D}^2(X_i, X_j)] \tag{1}$$

where μ is an adjustable parameter.

DEFINITION 2 (k-NEAREST NEIGHBORHOOD $kNN(X,T)$): For each test instance X, the k-nearest neighborhood of X in training data T is the set of k training instances $x_i \in T$ that have highest similarity with X. In other words, $kNN(X) = \{x_1, ..., x_k\}$ where x_i belongs to training data and $\forall x_i \in kNN(X); \forall x_j \notin kNN(X)$

$$\mathcal{S}(X, x_i) > \mathcal{S}(X, x_j) \tag{2}$$

where k is an adjustable parameter.

3.6.2. Incremental feature construction (IFC)

Incremental feature construction is done by combining two or more sets of feature vectors.

Let $F = \{f_1, ..., f_m\}$ be the one set of features and $G = \{g_1, ..., g_n\}$ be another set of features. Therefore, the incremental construction $IFC(F,G)$ is a merger of these two feature sets, i.e.,

$$IFC(F,G) = F \cup G = \{f_1, ..., f_m, g_1, ..., g_n\} \tag{3}$$

The resultant feature vector contains all the features in both sets. In our experiments we apply IFC with OUT, IN, and PC.

3.6.3. Ensemble construction (EC)

Ensemble construction is done on two or more sets of feature vectors by constructing one similarity model per feature vector and then applying weighted majority voting on all similarity models.

Let $F = \{f_1, ..., f_m\}$ be the one set of features and

$G = \{g_1, ..., g_n\}$ be another set of features. Then $EC(F,G)$ will first compute nearest neighborhood on F and then on G, and then apply majority voting on the constructed neighborhoods. This is done as follows. Let X be the test instance, and $T = \{x_1, ..., x_N\}$ be the training data. Also let $U = kNN(X_F, T)$ be the k-nearest neighbors of X in T using feature vector F and $V = kNN(X_G, T)$ be the k-nearest neighbors of X in T using feature vector G Also, assume that the target feature value for training instance x_i is y_i. Therefore, the majority voting is done as follows:

$$EC(F,G) = \sum_{x_i \in (U \cup V)} y_i \mathcal{S}(X, x_i) / \sum_{x_i \in (U \cup V)} \mathcal{S}(X, x_i) \tag{4}$$

3.6.4. Affinity network fusion (ANF)

We use the ANF technique proposed in [29] to combine two more sets of features. This is done as follows.

Let $F = \{f_1, ..., f_m\}$ be the one set of features and $G = \{g_1, ..., g_n\}$ be another set of features. Then $ANF(F, G)$ works as follows. First an affinity (i.e., similarity) matrix is generated for each of features, using both training and test instances (after target feature removed). Let $AM(F)$, $AM(G)$ be the affinity matrix for feature vectors F, and G, respectively. Each such matrix is a square matrix, where each row i corresponds to an instance x_i and each column j corresponds to an instance x_j, and the cell value at (i, j) denotes similarity between x_i and x_j calculated using the given feature vector. After this, these two matrices are merged using a fusion technique that considers similarity values in both matrices. The fused matrix is identified as $ANF(F, G)$. For a test instance X, the $kNN(X, T)$ can then be computed by using the similarity values found at the row corresponding to X, and columns corresponding to T.

4. Experiments

We now describe the experiments and analyze the results.

4.1. Datasets and Preprocessing

The source of all datasets in this experiment is MIMIC-III [30], and the preprocessing is done according to description in Section 3. We would like to recall to the reader about the notations that we will be using for the processed datasets.

Docuset: The first target feature (medication) Docuset Sodium (Laxative). **Propofol:** The second target feature (medication) Propofol (Anesthetic). **OUT:** Feature vector constructed using outputevents table.

IN: Feature vector constructed using inputevents table. **PC:** Feature vector constructed using prescriptions and chartevents table. **binary-6000:** The dataset (containing one of the above feature vectors) obtained with selected 6000 patients, where feature values are binary. **numeric-6000:** The dataset (containing one of the above feature vectors) obtained with selected 6000 patients, where feature values are numeric. **binary-1200:** The dataset (containing one of the above feature vectors) obtained with selected 12000 patients, where feature values are binary. **numeric-12000:** The dataset (containing one of the above feature vectors) obtained with selected 12000 patients, where feature values are numeric. Each dataset is evaluated using three-fold cross validation.

4.2. Competing Approaches

IFC: The incremental feature construction approach discussed in Section 3.

EC: The ensemble construction approach discussed in Section 3.

ANF: The affinity network fusion approach discussed in Section 3.

VC: This is the vertical clustering approach proposed in [2]. In this approach, we use the combined feature vector (IN+OUT+PC), and then apply vertical clustering to create two clusters. Each such cluster will have a subset of features from the combined feature vector. Then we use two variations of evaluation. VC+EC uses Ensemble construction approach using the two clusters to evaluate the prediction performance, and VC+ANF uses ANF of the two clusters to evaluate prediction performance.

4.3. Parameters and Other Setup

Parameter μ: this parameter is used for similarity computation. We use μ = 1 in all experiments. Parameter k: this is the k for k NN computation, and we use k =3 in all experiments. Both parameters were tuned to obtain good performance. For computing ANF, we use the tool by authors [29].

4.4. Evaluation

Evaluation metric: We use Area Under the ROC curve or AUROC in all experiments.

Results in small datasets (numeric-6000 and binary-6000): Fig. 2 shows the performance of each competing approach in the numeric-6000 dataset for both target features. The best performance is obtained by IFC (IN, OUT, PS), which is the combination of all feature vectors. For Docuset this value is 0.64, and for Propofol it is 0.73. The 2nd best performance is observed for the EC (IN, OUT), being 0.63 and 0.71 for Docuset and Propofol, respectively. We observe the benefit of incrementally adding feature vectors from these graphs. For example, when using only OUT features, the Propofol AUROC is found to be 0.5. However, by adding IN, and then again adding PC, we observe AUROC improved to 0.7, and 0.73, respectively. This same trend is also observed for Docusate. ANF performs relatively poorly on both target features.

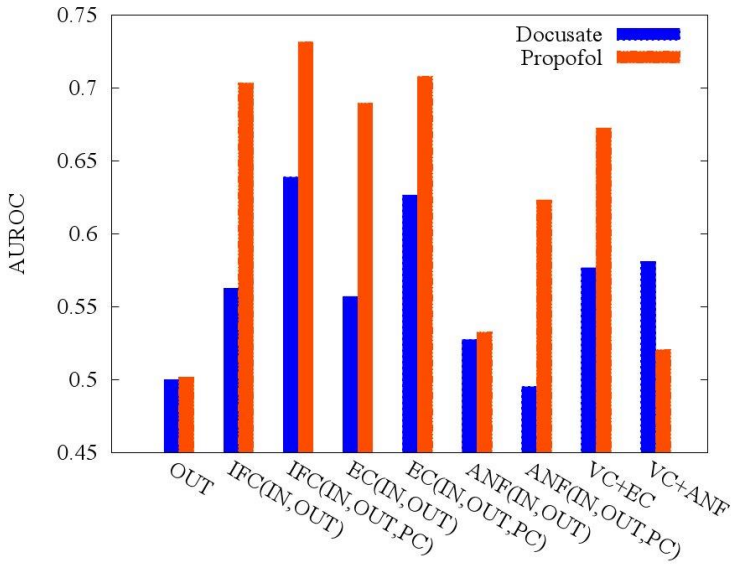

Fig. 2. AUROC on the numeric-6000 dataset.

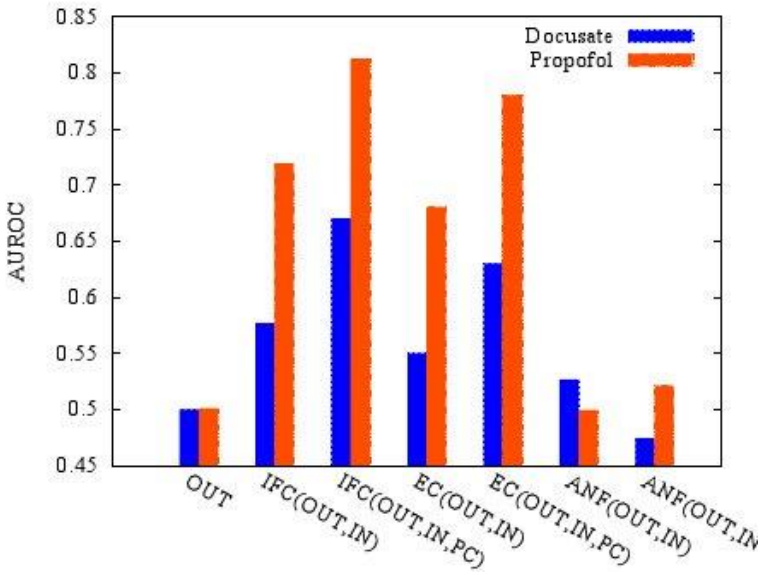

Fig. 3. AUROC on the binary-6000 dataset.

Fig. 3 shows the performance of each competing approach in the binary-6000 dataset for both target

features. Here we also observe the same phenomenon, i.e., IFC (OUT, IN, PC) achieves the highest AUROC for both target features, followed by EC (OUT, IN, PC). Also, the gradual improvement as a result of combining more features is evident here too. Another important observation is that binary features perform better than numeric features. For example, the best Propofol AUROC obtained is for binary features is 0.81, whereas that of numeric features is 0.73.

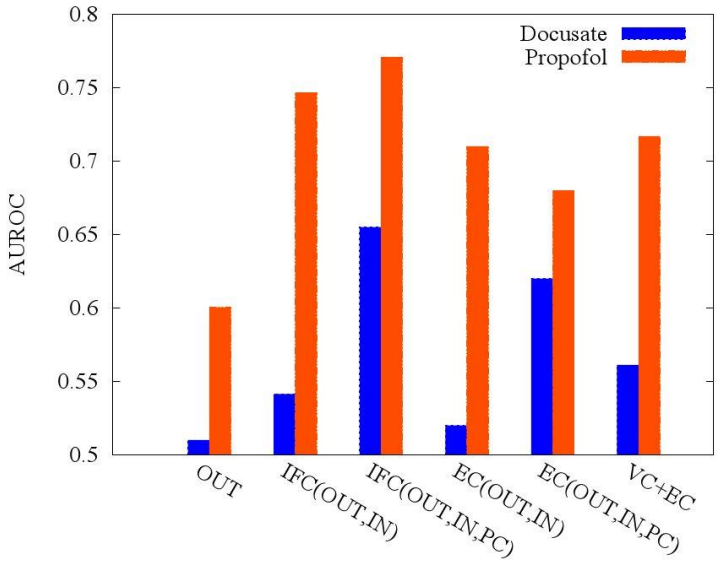

Fig. 4. AUROC on the numeric-12000 dataset.

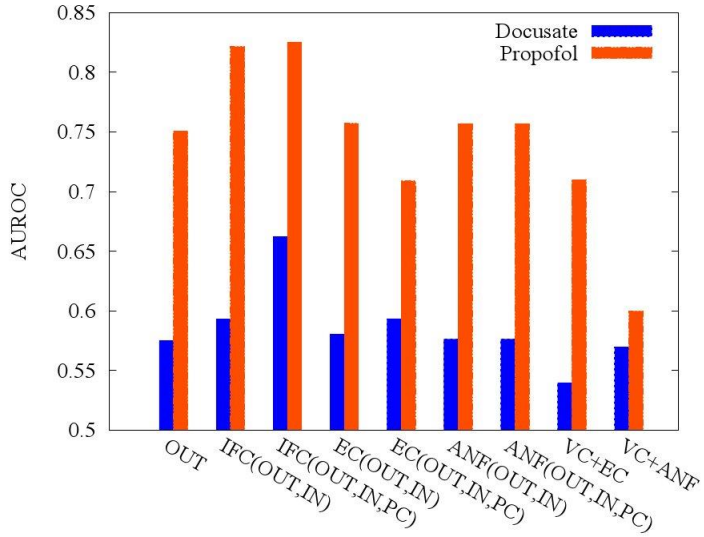

Fig. 5. AUROC on the binary-12000 dataset.

Results in larger datasets (numeric-12000 and binary-12000):

Fig. 4 and Fig. 5 show the performance of each competing approach in the 12000 datasets for both target features. The results follow similar trends as the 6000 datasets. For example, the best performance is again obtained by IFC (IN, OUT, PS), gradual improvement is observed with combination of more features, and better AUROC value for the binary dataset than the numeric.

Results summary and discussion: Table 1 shows the summary of above results.

Table 1. Summary Result on All Datasets (Best Two Competitors)

Competitor	Target	numeric6000	binary6000	numeric12000	binary12000	Overall
IFC (OUT, IN, PC)	Docuset	0.64	0.67	0.66	0.66	0.66
	Propofol	0.73	0.81	0.77	0.83	0.79
EC (OUT, IN, PC)	Docuset	0.63	0.63	0.62	0.59	0.62
	Propofol	0.71	0.78	0.68	0.71	0.72

It is evident from the results that binary features achieve better results than numeric. This is because binary features do not contain any missing values, but numeric features contain substantial amount of missing values. Although imputation is done, it may not work well when too many feature values are missing (which is the case for many features and instances). Another observation is that Propofol prediction achieves better performance than Docusate. This happens because Docusate is a relatively more common medication that is used as laxatives. This may be used for a wide variety of patients. On the other hand, Propofol is only used during general anesthesia, meaning, its application is on a more constrained and less versatile pool of patients. This causes the models to learn better and make better predictions. Also, we have seen that ANF did not perform well. One possible reason is that ANF has many parameters, which was not tuned but could be tuned to get a better result. Finally, EC approach performs relatively worse than IFC because individual feature sets may not be big enough to construct a strong ensemble. However, for a larger feature vector, EC is expected to have better performance.

5. Conclusion

We have demonstrated a technique for utilizing combination of various clinical data sources for medication recommendation of ICU patient using patient similarity in the clinical data. The proposed technique uses only first 24 hours clinical data of patients to find the most similar patients, and recommends medication based on the similar patient's medication record. We have shown different ways to combine different clinical data sources and examined their relative performance. We also discussed in detail the data preprocessing step, which also requires lots of effort and poses many challenges. We believe that the proposed work will be a valuable contribution to healthcare decision support, especially for clinical recommender systems where data exhibits similar characteristics. In future, we would like to emphasize more on other issues such as missing data handling, improving similarity network construction and handling of multiple values for the same feature. Furthermore, we would like to increase the effectiveness of the prediction performance by combining other data sources such as nurse notes.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

All authors had conducted the research and approved the final version.

Acknowledgment

This work was supported in part by ADEK Award for Research Excellence Award No: AARE17-182.

References

[1] Mao, Y., Chen, W., Chen, Y., Lu, C., Kollef, M., & Bailey, T. An integrated data mining approach to real-time clinical monitoring and deterioration warning. *Proceedings of SIGKDD '12* (pp. 1140–1148).

[2] Masud, M. M., & Cheratta, M. (2018). A framework for utilizing lab test results for clinical prediction of

ICU patients. *Proceedings of Workshops at the 32nd AAAI Conference on Artificial Intelligence* (pp. 479–486).

[3] Fialho, A., Cismondi, F., Vieira, S., Reti, S., Sousa, J., & Finkelstein, S. (2012). Data mining using clinical physiology at discharge to predict ICU readmissions. *Exp. Sys. With App*., *39(18)*, 13158–65.

[4] Baumgartner, B., Rödel, K., & Knoll, A. (2012). A data mining approach to reduce the false alarm rate of patient monitors. *IEEE Eng Med Biol Soc*., 5935–8.

[5] Cheng, C. W., Chanani, N., Venugopalan, J., Maher, K., & Wang, M. D. (2013). Icuarm-an ICU clinical decision support system using association rule mining. *IEEE J. of Translational Engg in Health and Medicine*, *1*, 4400110–4400110.

[6] Wang, B., Mezlini, A. M., Demir, F., Fiume, M., Tu, Z., Brudno, M., Haibe-Kains, B., & Goldenberg, A. (2014). Similarity network fusion for aggregating data types on a genomic scale. *Nature methods*, *11(3)*, 333.

[7] Pai, S., Hui, S., Isserlin, R., Shah, M. A., Kaka, H., & Bader. G. D. (2018). netdx: Interpretable patient classification using integrated patient similarity networks. *bioRxiv*.

[8] Pirracchio, R., Petersen, M. L., Carone, M., Rigon, M. R., Chevret, S., & Laan, M. J. (2015). Mortality prediction in intensive care units with the Super ICU Learner Algorithm (SICULA): A population-based study. *Lancet Respir Med*, *3(1)*, 42–52.

[9] Cismondi, F., *et al*. (2013). Reducing unnecessary lab testing in the ICU with artificial intelligence. *IntL. J. of Medical Informatics*, *82(5)*, 345-358.

[10] Purushotham, S., *et al*. (2017). Benchmark of deep learning models on large healthcare MIMIC datasets. *CoRR.*

[11] Herland, M., Khoshgoftaar, T. M., & Wald. R. (2013). Survey of clinical data mining applications on big data in health informatics. *Proceedings of ICMLA '13* (vol. 02, pp. 465–472).

[12] Celi, L. A., *et al*. (2011). A clinical database-driven approach to decision support: Predicting mortality among patients with acute kidney injury. *J Healthc Eng*, *2(1)*, 97–110.

[13] Cai, X., *et al*. (2016). Real-time prediction of mortality, readmission, and length of stay using electronic health record data. *Journal of the American Medical Informatics Association*, *23(3)*, 553–561.

[14] Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, *40(3)*, 56–58.

[15] López-Nores, M., *et al*. (2012). Property-based collaborative filtering for health-aware recommender systems. *Expert Systems with Applications*, *39(8)*, 7451–7457.

[16] Zhang, S., Yao, L., & Sun, A. (2017). Deep learning based recommender system: A survey and new perspectives. *arXiv preprint arXiv*:1707.07435.

[17] Linden, G., Smith, B., & York, J. (2003). Amazon.Com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, *(1)*, 76–80.

[18] Cheng, H.-T., *et al*. (2016). Wide & deep learning for recommender systems. *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems* (pp. 7–10).

[19] Gomez-Uribe, C. A., & Hunt, N. (2016). The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS)*, *6(4)*, 13.

[20] Hors-Fraile, S., *et al*. (2018). Analyzing recommender systems for health promotion using a multidisciplinary taxonomy: A scoping review. *International Journal of Medical Informatics*, *114*, 143–155.

[21] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Tran.s on Knowledge & Data Engineering*, *(6)*, 734–749.

[22] Bridge, D., *et al*. (2005). Case-based recommender systems. *The Knowledge Engineering Review*, *20(3)*, 315–320.

[23] Pazzani, M. J. (1999). A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, *13(5-6)*, 393–408.
[24] Lopez-Nores, M., *et al*. (2010). Electronic health records as a new dimension of user profiles in recommender systems. *Global Telemedicine and eHealth Updates*, *3*, 137–140.
[25] López-Nores, M., *et al*. (2010). Mispot: Dynamic product placement for digital tv through mpeg-4 processing and semantic reasoning. *Knowledge and Information Sys.*, *22(1)*, 101–128.
[26] Pai, S., & Bader, G. D. (2018). Patient similarity networks for precision medicine. *J. of molecular Biology.*
[27] Gligorijevi´c, V., Barot, M., & Bonneau, R. (2018). Deepnf: Deep network fusion for protein function prediction. *Bioinformatics*.
[28] Li, L., *et al*. (2015). Identification of type 2 diabetes subgroups through topological analysis of patient similarity. *Science Translational Medicine*, *7(311)*, 311ra174–311ra174.
[29] Ma, T., & Zhang, A. (2017). Integrate multi-omic data using affinity network fusion (anf) for cancer patient clustering. *Proceedings of 2017 IEEE Intl. Conf. on Bioinformatics and Biomedicine (BIBM)* (pp. 7–10). IEEE.
[30] Physionet-MIMICIII. https://www.physionet.org/physiobank/database/mimic3cdb/.

Mohammad Mehedy Masud is currently an associate professor at College of Information Technology (CIT) at United Arab Emirates University (UAEU). Prior to joining UAEU in January 2012, Dr. Masud worked at the University of Texas at Dallas as a research associate for 2 years. He received his Ph.D. degree in Computer Science from University of Texas at Dallas, USA, in December 2009. Dr. Masud's research interests include big data mining, data stream mining, machine learning, healthcare data analytics and e-health. His research also contributes to cyber security (network security, intrusion detection, and malware detection) using machine learning and data mining. He has published more than 50 research articles in high impact factor journals including IEEE Transactions on Knowledge and Data Engineering (TKDE), Journal of Knowledge and Information Systems (KAIS), and top tier conferences including IEEE International Conference on Data Mining (ICDM). His is the lead author of the book "Data mining tools for malware detection" and the principal inventor of a US Patent. He is the principal investigator of several prestigious research grants funded by government and private funding organizations. More details can be found here: faculty.uaeu.ac.ae/m_masud/

Muhsin Cheratta is a research assistant at the College of IT, UAEU. Since joining UAEU in 2017, Muhsin has been working on various research & development areas, including clinical big data processing and analytics, developing different Machine learning algorithms including deep learning, as well as setting up and working on cloud computing environment. He gained extensive development skills in different languages including Java, Python, C++ and development platforms including Netbeans, Eclipse, Pycharm, Android studio, and Tizen development environments.

Kadhim Hayawi received his PhD in electrical and computer engineering from University of Waterloo, Canada in 2018, the M.Sc. in computer science from Dalhousie University, Canada in 2004, the M.Sc. in electronics and telecommunication engineering and the B.Sc. in electrical engineering from University of Baghdad, Iraq in 1996 and 1991 respectively. He has published books, and papers in international IEEE and ACM venues, and served on the organizing and technical program committees of several international conferences. Kadhim has worked in the Information security, Machine learning, Networking, and Telecommunication fields for many years, with jobs in engineering design, technical consulting, teaching, and course development. He holds the 4013 Information Assurance Recognition by the US Government National Security Agency (NSA) and the Committee on National Security Systems (CNSS), and the 4011 Information Systems Security Recognition by the US Government National Security Agency (NSA), CNSS and National Training Standard for Information Systems Security (INFOSEC) Professionals (NSTISSI).