Dual autoencoders modeling of electronic health records for adverse drug event preventability prediction

Liao, Wenjun; Derijks, Hieronymus J; Blencke, Audrey A; De Vries, Esther; Van Seyen, Minou; J Van Marum, Robert

Published in:
Intelligence-Based Medicine

DOI:
10.1016/j.ibmed.2022.100077

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

Intelligence-Based Medicine 6 (2022) 100077

Contents lists available at ScienceDirect

Intelligence-Based Medicine

journal homepage: www.sciencedirect.com/journal/intelligence-based-medicine

Dual autoencoders modeling of electronic health records for adverse drug event preventability prediction

Wenjun Liao [a,b], Hieronymus J Derijks [b], Audrey A Blencke [b], Esther de Vries [c,d], Minou van Seyen [b], Robert J van Marum [e,f,*]

[a] The Jheronimus Academy of Data Science, Eindhoven University of Technology, 's-Hertogenbosch, the Netherlands
[b] Department of Pharmacy, Jeroen Bosch Hospital, 's-Hertogenbosch, the Netherlands
[c] Department of Tranzo, Tilburg School of Social and Behavioral Sciences, Tilburg University, Tilburg, the Netherlands
[d] Department of Jeroen Bosch Academy Research, Jeroen Bosch Ziekenhuis, 's-Hertogenbosch, the Netherlands
[e] Department of Elderly Care Medicine, Amsterdam Public Health Research Institute, Amsterdam UMC, Location VUmc, Amsterdam, the Netherlands
[f] Department of Clinical Pharmacology, Jeroen Bosch Hospital, 's-Hertogenbosch, the Netherlands

ARTICLE INFO

Keywords:
Adverse drug event
Electronic health records
Machine learning
Autoencoder
Clinical data science

ABSTRACT

Background: Elderly patients are at increased risk for Adverse Drug Events (ADEs). Proactively screening elderly people visiting the emergency department for the possibility of their hospital admission being drug-related helps to improve patient care as well as prevent potential unnecessary medical costs. Existing routine ADE assessment heavily relies on a rule-based checking process. Recently, machine learning methods have been shown to be effective in automating the detection of ADEs, however, most approaches used only either structured data or free texts for their feature engineering. How to better exploit all available EHRs data for better predictive modeling remains an important question. On the other hand, automated reasoning for the preventability of ADEs is still a nascent line of research.

Methods: Clinical information of 714 elderly ED-visit patients with ADE preventability labels was provided as ground truth data by Jeroen Bosch Ziekenhuis hospital, the Netherlands. Methods were developed to address the challenges of applying feature engineering to heterogeneous EHRs data. A Dual Autoencoders (2AE) model was proposed to solve the problem of imbalance embedded in the existing training data.

Results: Experimental results showed that 2AE can capture the patterns of the minority class without incorporating an extra process for class balancing. 2AE yields adequate performance and outperforms other more mainstream approaches, resulting in an AUPRC score of 0.481.

Conclusions: We have demonstrated how machine learning can be employed to analyze both structured and unstructured data from electronic health records for the purpose of preventable ADE prediction. The developed algorithm 2AE can be used to effectively learn minority group phenotype from imbalanced data.

1. Introduction

An Adverse Drug Event (ADE) is 'any injury due to the use of medication' [1]. Typically, elderly patients are at increased risk for ADEs due to problems such as frailty, multi-morbidity, and polypharmacy [1]. A Dutch study retrospectively evaluating a random sample of 2000 admissions from a total of 155 hospitals by assessing the admission and discharge letters showed that an estimated 10% of unplanned hospital admissions in patients >65 years are thought to be drug-related and half of them are potentially preventable [2]. The costs of potentially preventable hospital admissions related to medication are considerable, with an estimated €5461 per case on average [3]. Based on these figures, the Dutch Guideline "Polyfarmacie bij Ouderen, addendum 2e lijn" (Polypharmacy for elderly admitted to hospital) proposes that all elderly people visiting an Emergency Department (ED) should be screened pro-actively for the possibility of the hospital admission being drug-related. Leveraging the existing data in Electronic Health Records (EHRs) to better capture elderly at risk of ADE in the ED may improve their care [4].

A detailed assessment of the probability of the ED visit being related

* Corresponding author. Department of Elderly Care Medicine, Amsterdam Public Health Research Institute, Amsterdam UMC, location VUmc, Amsterdam, the Netherlands.
E-mail address: r.vanmarum@amsterdamumc.nl (R. J van Marum).

https://doi.org/10.1016/j.ibmed.2022.100077
Received 26 February 2022; Received in revised form 23 August 2022; Accepted 14 September 2022
Available online 20 September 2022

to medication use and subsequent analysis of potential preventability using EHRs data takes a trained clinical pharmacologist an estimated 5–10 min per patient. Since most ED visits and hospital admissions are not caused by a preventable ADE, carrying out this evaluation for all ED visits or hospital admissions will take too much in fact unnecessary staff time, making it not cost-effective. Thus, there is a need for efficient methods to identify preventable ADE cases among elderly ED-visitors and clinically admitted patients.

Existing methods for automating the ADE-preventability detection process heavily rely on rule-based systems such as filtering pre-defined symptoms, abnormal computer-generated life signals, or medication error signals [2,5]. However, these rule-based systems are used only for the ADE detection part whereas upon knowing an ADE case, the preventability assessment is still dependent on expert domain knowledge through manual checks. Adapting the machine learning systems to include an estimation of preventability may provide a solution to this challenge because all labor-intensive tasks could then be handled automatically. Machine learning based predictive modeling using EHRs data is a nascent research area. Whether the *preventability* of ADE can be detected effectively by a machine learning model is a neglected area so far. The majority of interdisciplinary work on Artificial Intelligence (AI) and ADE focused on the detection of the *incidence* of ADE.

2. Related works

2.1. Machine learning on ADE detection: structured and unstructured data

Due to the strict permission rules for EHRs access, research opportunities with collaborative use of EHRs are limited [6]. Naturally, literature on AI-based ADE detection is scarce. Most published studies only used partial information extracted from the EHRs database to form the algorithm training features. Some focused on using unstructured data, e.g., clinical notes. Eriksson et al. [7] applied a Named Entity Recognition (NER) tagger on the clinical narrative text to identify ADE phenotypical descriptions that have no requirement for causal drug relationship. A similar approach proposed by Henriksson et al. [8] calculates distributional similarities for medication-symptom pairs based on co-occurrence information in a large clinical corpus. LePendu et al. [9] proposed a novel de-identified patient-feature construction method, where the clinical texts are transformed to a matrix coded on the basis of standard medical terms.

Another mainstream direction is to use structured data such as clinical measurements, lab tests, and medication records. The strategies in utilizing structured EHRs data boil down to incorporating effective ways of extracting tabular format features that represent the longitudinal, time-stamped data without losing the data's temporal and sequential information. Zhao et al. (2015) [10] proposed a single-point representation method to construct features out of various structured datasets. In 2017, they proposed an innovative multivariate time series representation method for extracting features from heterogeneous temporal EHRs data [11]. Cheng et al. [12] represented the medical events as a temporal matrix where time is one dimension and event type the other one.

Few researchers attempted to combine both structured and unstructured EHRs data. How to best utilize all the available information from the EHRs, both structured and unstructured, has still been underexplored. An example is a project carried out across three European countries, where a ranked list of high-priority event types is created using a variety of structured and free texts queries [13]. Based on these, a number of statistical methods were developed for drug safety signal detection, however, the feature extraction mainly focused on the longitudinal records [14]. In this study, we further explore whether it is beneficial to combine various data types from the EHRs for the task of *supervised ADE-preventability prediction*.

2.2. Imbalanced classification problem

Many supervised classification tasks in the healthcare domain face the class imbalance problem. Preventable-ADE cases also constitute only a small fraction of all ED-visits by elderly patients, resulting in an unbalanced labeled training dataset. A mainstream approach to relieving such imbalance problems among classes is resampling, either through oversampling of the minority class or undersampling of the majority class. However, the randomness in the sample selection process may then lead to unstable model performance. Another approach is the use of generative models to synthesize simulated samples of the minority classes. Popular generative models include Generative Adversarial Network (GAN) [15] and Variational Auto-Encoder (VAE) [16], which were applied to solve the class imbalance problem or integrate new samples in several clinical data science and biomedical projects [17–20]. Considering that the differences in feature performance between preventable ADE and non-preventable ADE cases are minor, the synthetic minority samples may partially overlap with the majority class.

In contrast to just seeking ways to balance the classes and train a single model that captures different class patterns in the supervised classification task, some researchers also adopted an approach in which multiple weak learners are trained on different classes. Many of them used autoencoder as the weak learner. Autoencoder, introduced by Hinton et al., in 1986 [21] is the foundational structure of the generative model VAE. It is a type of artificial neural network used to learn a representation for a set of data, which is trained by minimizing the reconstruction errors between the same set of input and output. A single autoencoder based model showed promising performance in prediction tasks using imbalanced clinical data [22]. However, some studies showed that better classification and anomaly detection results are achieved by using an ensemble of autoencoders. In a fraud detection project [23], two autoencoders were trained on the normal and fraud datasets, respectively. During the testing phase, the samples were encoded by both autoencoders where two sets of features were generated, combined, and fed into a neural network. Similarly, a study by Ng et al. [24] constructed two stacked autoencoders with different activation functions to capture different characteristics in the binary classes. Two sets of features were learned by the dual autoencoders and were combined to form the final feature set. The study by Chen et al. [25] went a step further. After obtaining multiple autoencoders trained from different classes, during the testing phase, they calculated the reconstruction errors from each autoencoder instead of combining the feature sets generated by multiple autoencoders. The prediction was made by summarizing these reconstruction errors. Inspired by these previous approaches, we evaluated a similar framework consisting of two autoencoders to handle the class imbalance problem inevitably embedded in the preventable ADE classification task.

3. Methods

In this section, we first introduce the data source and the study population characteristics. We then describe the data types involved in the datasets. Different strategies are proposed to fit the heterogeneous temporal data and unstructured free texts into predictive models. Finally, we derive in detail the proposed Dual Autoencoders (2AE) framework.

3.1. Data source

The study material is provided by the Jeroen Bosch Hospital (JBZ), 's-Hertogenbosch, The Netherlands. Estimates of ADE related hospital admissions in the Netherlands were only based on retrospective analysis of admission and discharge letters, thereby missing a lot of relevant information. Therefore, the clinical pharmacologists at JBZ initiated a project to study if routine assessment by clinical pharmacologists of hospital admissions of elderly persons (≥70 years of age) - in order to

detect potential preventable ADE related admissions - is (cost)-effective. In 2019, a multidisciplinary team, the F ("Farmacologie")-team, was formed to conduct the identification process. The F-team consists of clinical pharmacologists with a mix of professional backgrounds: hospital pharmacist, geriatrician, and ED-physician.

From January to July 2019 the F-team two times a week evaluated all eligible ED-visit patients for medical departments with acute hospital admissions (surgery, urology, internal medicine, gastro-enterology, cardiology, and neurology). Since medication evaluation is standard procedure for geriatricians and additional analysis of medication by the F-team therefore seemed unnecessary, admissions for the geriatric department were not analyzed. The number of patients studied for each department varied and was determined by the F-team. If after a number of assessments, the F-team was convinced that they had a clear view on the prevalence of ADEs for that department, another department was chosen for further evaluation.

For each admission, the F-team checked first if a patient had symptoms that are highly associated with ADEs, such as falls, syncope, fractures, constipation, delirium, and bleeding. Second, the team looked for involvement of the geriatric department. Patients who were also seen by a geriatrician during the ED visit were excluded. Third, for those patients with relevant symptoms, the F-team checked whether a patient's ED-visit could be classified as an ADE case by looking into their EHRs in detail. Fourth, for the identified ADE cases, the F-team made a binary judgment on its preventability (yes or no).

Table 1 shows the characteristics of the study population. We treated the preventable ADE cases (true-case) as the minority class and grouped all the other clusters together to form the majority class. Patient information was extracted from the JBZ EHRs database and transformed to separate datasets in CSV-format based on different fields as shown in Table 2.

3.2. Feature construction

Feature engineering is the process to construct representation vectors from the raw EHRs data that can be recognized by machine learning algorithms. Each vector represents the information of one patient and is a concatenation of multiple feature vectors constructed from different datasets. During feature construction, only data in the observation window was considered. We set the default observation window at two weeks before the preventable ADE assessment day. This is in alignment with the actual clinical practice, for example, when assessing the complaint-medication associations, domain experts usually only inspect the medications currently in use.

The datasets can be classified into three categories. The first category consists of data sets that are in a structured tabular format, such as the demographics dataset where each patient has only one unique record. These categorical records are therefore one-hot encoded, which results in a vector space where each category is orthogonal and equidistant to the others.

The second category consists of datasets that have a structured longitudinal format, where repeated observations of the same clinical events over some extended time frame are recorded for one patient. Different strategies are made depending on their characteristics. For example, on the one hand, medication history is encoded by a hierarchical classification system, ATC code, thus the medication feature is constructed as the count number aggregation of the times an ATC code was prescribed (see Fig. 1). Clinical measurements, on the other hand, address various types of patient measurements. Each measurement has multiple occurrences with potentially different numerical values. Therefore, clinical measurement features are constructed such as mean, minimum, and maximum of observed measurement values (see Fig. 2). The dataset with laboratory test results shares a similar format with clinical measurements but is slightly different in that the test abnormality levels are described by three ordinal scales (high, normal, low). These ordinal representations were encoded to numbers and a single point feature was extracted by averaging the encoded numerical values. A different observation window of two years was considered for the frequency of specific clinical department visits because certain department visit patterns may indicate the presence of chronic disease.

In this study, the feature engineering on structured longitudinal

Table 1
The characteristics of the study population (Non-1: Patients who do not have ADE-related symptoms; Non-2: Patients who have potentially ADE-related symptoms but are not classified as having an ADE; Non-3: Patients who have non-preventable ADEs; True-case: Patients who have a preventable ADE).

Characteristics	Non-1	Non-2	Non-3	True-case
Population size	390	152	81	91
Age, median (IQR)	79 (75/84)	79.5 (75/85)	82 (78/86)	82 (77/86.5)
Sex, Female (%)	43.9%	52.6%	53.1%	55.0%
BMI, median (IQR)	26.3 (23.7/29.34)	26.3 (23.4/28.4)	23.1 (25.4/29.4)	26.2 (23.2/29.6)
Birth Country, NL (%)	96.4%	94.7%	97.5%	94.5%
Total Current-in-use Unique Medication, median (IQR)	12 (8/17)	12 (7.75/16.25)	11 (8/16.25)	14 (11/18.5)
Smokes	12.1%	7.2%	7.4%	12.1%
Drinks alcohol	48.0%	46.1%	44.4%	44.0%
Drug user	0	0	0	0

Table 2
Datasets extracted from Electronic Health Records.

Fields	Description
Demographics	The basic information of each patient such as sex, age, and birthplace
ED-visit complaints	The ED-visit date and the reason for visiting the ED
Clinical admissions	The clinical admission date and the reason for clinical admission
Clinical measurements	The longitudinal records such as height, weight, heart rate, and blood pressure
Medication history	The medication prescriptions
Medication signals history	The medication signals such as double medication
Anamnesis history	The records of alcohol drinking, smoking, drug use
Allergies history	The records of allergic reactions
Diagnosis history	The doctor's diagnoses in the form of free texts
Decursus history	The free texts such as conclusions written by the doctors
Lab tests history	The longitudinal record of laboratory tests
Vulnerability history	The records of the qualitative vulnerability analysis
Appointments history	The longitudinal record of the patient's appointments

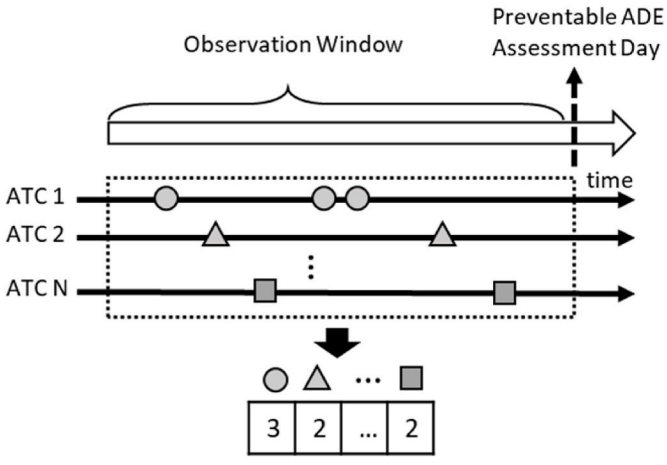

Fig. 1. Illustration of the feature construction process for medication history. A type of ATC prescription is represented as its total number of occurrences during the observation period.

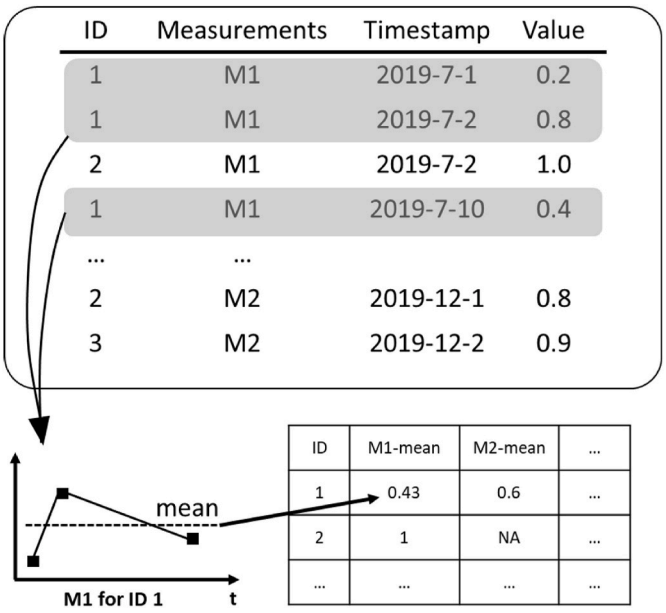

Fig. 2. Illustration of the feature construction from multivariate time series.

format data can be summarized as integrating an event's multiple occurrences into a single-point representation. This approach ignores the intervals between an event's multiple occurrences. Arguably, this study considered a relatively short two-week observation window for most feature extraction processes of the temporal information. Under this scenario, single-point features are sufficient to represent the short-term temporal information.

The third category consists of datasets that contain free texts including clinical admission information, ED-visit complaints, diagnosis, and decursus (the free texts section of the JBZ EHRs system, which contains the conclusion and any other notes written by the doctors). The raw texts went through a processing pipeline before the natural language processing models were built. The processed texts were lowercased and de-accented with grammatical symbols and numbers excluded. First, we trained a customized word embedding from all text corpus using the CBOW Word2Vec algorithm, which is a neural network based technique that learns word associations from a large corpus of text [26]. We used the word embedding to convert each patient's decursus records to a 64-dimensional vector by averaging the word embedding vectors of the words that compose the decursus. The converting process only takes records that falls within the observation window. Second, we trained a bag-of-words (BoW) model from all free texts filtered by the observation window with stop words excluded. The filtered free texts were vectorized to a term frequency vector using the BoW model, where 2194 vocabularies are included after adjusting the maximum and minimum document frequency. Third, after we had asked the domain expert to create a list of keywords that are highly associated with ADE syndromes, we constructed a Boolean vector to represent the existence of such words in the filtered free texts. Forth, a separate BoW vector was created specifically from diagnosis notes with an extended two-year observation window. From the time dimension's perspective, a separate BoW is needed because the overall free texts BoW vector whose observation window is set to two weeks, only captures the most recent information of a patient. In addition, reducing the corpus scope to diagnosis enables the separate BoW vector to incorporate domain-specific vocabularies.

An outlier scheme was designed on the basis of domain knowledge. The numerical values in clinical measurements that were beyond the normal range were set to null values, as described in Ref. [27]. Subsequently, the numerical missing values in the feature data frame were imputed with the mean of each column in which the missing values were located. The categorical missing values were all imputed with the value 0. Finally, we applied a Max-Min scaler to normalize the concatenated representation vector.

3.3. Predictive modeling

2AE consists of two parallel simple autoencoders. In the training phase, one autoencoder is used to estimate the pattern of the preventable ADE group. The other one estimates the pattern from the rest of the samples, which include non-preventable ADE cases and non-ADE cases. The pattern learned by an autoencoder can be regarded as the phenotype of a patient group. Fig. 3 illustrates the process of the testing phase. When a testing sample is fed into the 2AE model, two reconstruction errors can be obtained according to the equation (Eq. (1)),

$$RE_i = Y_i^* - Y_2^2,\ i = 1, 2 \tag{1}$$

where RE_i is the reconstruction error generated by the i^{th} autoencoder and Y_i^* is the reconstruction result according to the input sample representation vector Y. When the second autoencoder is trained from the preventable ADE group, a score in the range from 0 to 1 can be obtained according to the equation (Eq. (2)), which is a normalized exponential function based on the softmax function. The hyperparameter α makes the reconstruction error difference between two autoencoders adjustable, and, is by default 1. When the score is higher than a threshold *th*, we consider the testing sample to be a preventable ADE case.

$$score(i=2) = \frac{1}{1 + e^{\alpha(RE_2 - RE_1)}} \tag{2}$$

4. Experiments

In this section, we assess the effectiveness of the proposed 2AE framework. The architecture of each autoencoder in the 2AE framework is identical and was determined following similar settings from existing work [28]. For a single autoencoder, the input and output dimensions are the feature space. The number of neurons for the encoder layers are 128 and 64 respectively. The decoder layers are symmetric to the encoder layers and all layers are fully connected. A linear activation function is used in the second hidden layer for the decoder, and Relu is used for the remaining hidden layers. The autoencoder model uses Adam optimizer with Mean Squared Error as loss function. Glorot uniform [29] was utilized for network weights initialization and an L1 activity regularizer was applied on encoder's first hidden layer to avoid overfitting.

We included five other commonly used machine learning algorithms for the purpose of comparison: (1) K Nearest Neighbor (KNN) classifier, a lazy-learner algorithm by which all available cases can be stored and new cases are classified based on a similarity measure that performs well on unbalanced data; (2) Multi-Layer Perceptron (MLP) classifier, an

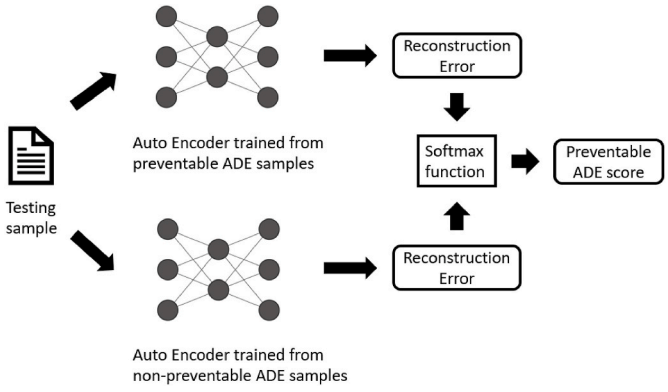

Fig. 3. The process of the testing phase, where a preventable ADE score is generated.

algorithm by which we can model non-linear and complex relationships embedded in the dataset; (3) Random Forest and (4) Adaboost that are based on bagging and boosting approaches during training, respectively; and (5) Single Autoencoder (SE), which is included to compare with the two autoencoders structure.

Experiments in this study were conducted in Python3.7 environment. For feature engineering, the CBOW Word2Vec based word embedding model was implemented using Gensim [30] and the BoW model using Scikit-learn [31]. All comparison algorithms except for the Autoencoder were implemented using Scikit-learn [31]. Keras [32] was adopted to implement the single and dual autoencoder structure. Hyperparameters for the comparison models were chosen via grid search and stratified 5-fold cross-validation, where the distribution of classes and the distribution of types of non-cases are evenly stratified in the train and test data split. The grid search range setting takes reference from Scikit-learn's default setting as well as similar studies [10,27]. Hyperparameters that yield the highest minority class F1 score were chosen. All the comparison models other than autoencoder based methods (SE and 2AE) take the up-sampled training data in which the minority class is randomly up-sampled to match the size of the majority class. The up-sampling was applied to each of the folds in the cross-validation process. Detailed experiment configurations are shown in Table 3.

Evaluation was done through stratified 5-fold cross-validation. The model performance evaluation metrics include recall, precision, and F1 score. These metrics were measured at the default decision threshold of 0.5. Besides, we included the area under the precision and recall curve (AUPRC), which is not dependent on a chosen decision threshold and can provide the performance summary of a classifier. AUPRC is recommended in the case of class imbalance and when the minority class is of more interest [33,34].

The experimental results are summarized in Table 4. Our proposed method, 2AE, yields the highest F1 score for the minority class, the preventable ADE cases, while keeping a relatively high AUPRC score in comparison with the other five models. Fig. 4 illustrates the model performance under different thresholds. It shows that 2AE has the potential to reach a high minority class precision by increasing the threshold while keeping a relatively high minority class F1 score. Meanwhile, we examined the impact of the hyperparameter α. Fig. 5 shows the F1 score for the minority class using different thresholds with different choices of α, respectively. When the threshold is set to 0.5, the hyperparameter α does not impose power on the model performance. When a different threshold is chosen, the best evaluation metrics score can be achieved by tuning the hyperparameter α. As α increases, the minority class F1 score increases and finally reaches a stable state.

We also evaluated the performance of the other models under different thresholds. None could outperform the 2AE on the best achieved minority class F1 score under our experimental setting. The advantage in 2AE's best-achieved minority class F1 score was at significance 5% greater than Random Forest's best achieved result, which is the second highest among the best achieved results of other comparison models.

Table 3
The algorithms used for comparison, and their configurations.

Classifier	Description	Configuration
KNN	K nearest neighbors	K = 9
RF	Random Forest	Tree quantity = 100
AdaBoost	Adaptive Boosting	Boosting is terminated at a maximum of 50 estimators
MLP	Multi-layer Perceptron classifier	Optimizer: 'Adam', hidden layer: 100, L2 penalty parameter = 0.0001
SE	A single autoencoder structure	Error threshold = 0.5, epoch = 50
2AE	The proposed method	Error threshold = 0.5, epoch = 50

Table 4
Predictive performance of the 2AE and comparison models. True-case stands for the samples which are labeled as preventable ADE cases and non-case stands for all the other samples.

Model	True-case			Non-case		AUPRC
	Recall	Precision	F1	Recall	Precision	
KNN	0.354	0.316	0.331	0.894	0.905	0.362
RF	0.321	0.576	0.391	0.958	0.906	0.418
AdaBoost	0.408	0.368	0.380	0.896	0.912	0.365
MLP	0.210	0.663	0.300	0.974	0.894	0.406
SE	0.143	0.251	0.174	0.934	0.882	0.250
2AE	0.486	0.559	0.501	0.939	0.926	0.481

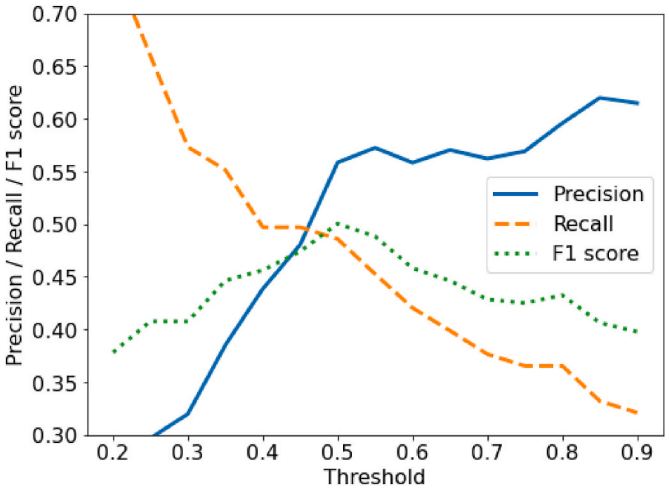

Fig. 4. The predictive performance of the 2AE model with respect to different thresholds set for the preventable ADE score. Precision, recall, and F1 score for the minority class, the preventable ADE cases, are shown.

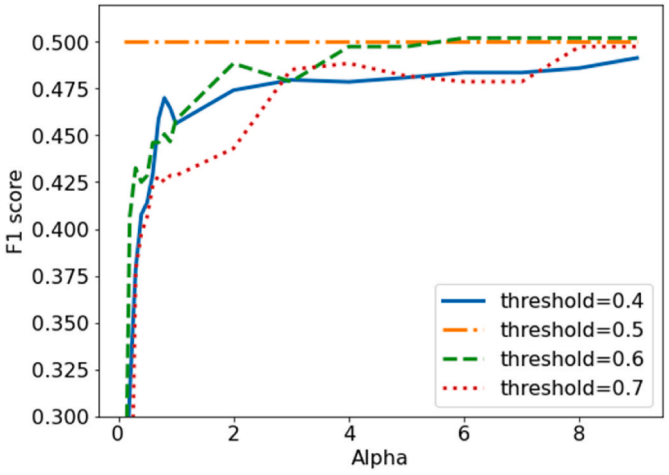

Fig. 5. The F1 score for preventable ADE prediction using different thresholds with respect to different choices of alpha.

5. Discussion

In this work, we analyzed the dual autoencoder framework to detect preventable ADE cases under an unbalanced class situation. To our knowledge, this is the first report of machine learning comprising both structured and unstructured data for the detection of preventable-ADE. The feature engineering module on the unstructured free texts in this project was constructed for a Dutch language context. The high

similarity among western Germanic languages means that the same methodologies used in this project can probably be transferred to other related languages.

This study included AUPRC and F1 score to evaluate the model's performance under the imbalanced classification scenario. The expected value for random guessing of the AUPRC is close to 0 when prevalence is low [33–36]. For the data provided, the fraction of positives (preventable ADE cases) is 12.7%; therefore, the baseline AUPRC score is 0.127. The best configuration of 2AE yields an AUPRC score of 0.481 and a minority class F1 score of 0.501. The performance can be considered adequate, which is consistent with conclusions in similar recent researches. In Ref. [37], the best achieved AUPRC reported on the Cardiovascular Event prediction is 0.285 when training model with longitudinal EHRs (prevalence of 8.97%), and 0.427 when training model with sub-group that has genetic data (prevalence of 24.13%). In Ref. [38], the best achieved AUPRC reported on the Sepsis from onset time prediction (prevalence of 2.44%) is 0.43. In Ref. [39], the best-achieved F1 score reported on the readmission prediction (imbalance ratio of 0.28) is 0.51, and in Ref. [40], 0.5 for the breast cancer distant recurrence prediction (prevalence of 6.92%). These are all considered satisfactory predictive performances by the authors. Our experiment results show that 2AE can capture the patterns of the minority class without incorporating an extra process for class balancing. 2AE itself consists of two simple multi-layer perception structures as opposed to more sophisticated neural network architectures. All this renders our proposed model easy to implement and time-efficient to train.

It is important to realize that a machine learning model, though effective, can never replace the 'human-in-the-loop'. Within the clinical context, the predicted cases should always be checked by doctors and pharmacists. Therefore, from a pragmatic model deployment's perspective, the precision, which represents the percentage of true positive cases among all predicted true cases, is the parameter of paramount importance for successful implementation in everyday practice considering the limited checking capacity of the evaluators. The higher the precision, the less time will be wasted on checking false positive cases. The experimental results show that by tuning the threshold and hyperparameter, 2AE can achieve a precision higher than 60% while keeping the minority class F1 score high.

5.1. Bias cancellation

During the ADE preventability assessment, the F-team would note down remarks in the decursus regarding the ADE preventability situation. Thus, the extracted raw training data inevitably contain texts that make the machine learning model biased towards the cases that contain texts written by the F-team. In order to cancel this bias, we asked the domain experts to create an 'F-team words' list, which contained the vocabularies that are only used by the F-team for ADE status description. These vocabularies were now eliminated during BoW and word2vec feature construction. However, if we would have the chance to test a new group of patients in the future, we would include these 'F-team words' in the word2vec features. The rationale behind this is that for such new cases, the F-team has not yet been involved. When analyzing a cluster of free texts from such new cases, if the averaged free texts vector shows similarity with the word embedding vector of one of the 'F-team words', it suggests that the new cases might share the same traits as the cases that have been labeled as preventable ADE in the current study.

5.2. Pitfalls and limitations

One pitfall in this study derives from the accuracy of the labeling of the training data. As shown in Table 1, besides the minority class (preventable ADE cases), there are three other clusters of cases. These clusters resulted from the logistics of the domain expert's labeling process. While assessing the ADE preventability, the domain experts first excluded those who didn't have any predefined ADE-related symptoms. For the rest of the patients, the ADE-medication association was checked. If no links were found, a patient would not be considered as an ADE case. There is a possibility that some ADE cases show non-specific complaints or showed obvious symptoms but had a weak link to the suspicious medications [41]. If such ADE cases were indeed preventable, then they formed falsely labeled cases. If the labeling process could have been re-designed, a thorough check on the ADE-preventability would have been applied to every patient.

Another issue involving labeling accuracy was the consensus rate. For some not so clear-cut cases, different domain experts may hold different opinions regarding their ADE preventability. The labeling process had been designed in such a way that every case was checked multiple times by different persons. With an uncertain consensus, it is inevitable that the precision in the definition of preventable ADE is affected, thus leading to the inaccuracy of data labels.

The framework designed in this study provides a score-based mechanism to assist doctors in prioritizing which patient should go through the ADE-preventability assessment. A limitation is that the explainability of the probability score generation process is not covered in our design. As opposed to some other models that incorporate the feature importance study such as tree-based models, 2AE is neural network based, thus its computational reasoning is essentially a black box for end-users. Nevertheless, there are multiple ways to overcome this obstruction. A prominent technique is Permutation Feature Importance (PFI) [42], a method that can be applied to any machine learning model to calculate the model's prediction error variation after permuting values of a single feature while keeping the other feature values unchanged. This method interprets that a feature is important if shuffling its values significantly increases the model error. In this study, the PFI computation for the proposed 2AE model was realized using a Python package ELI5 [43] and is presented in Appendix A. When model explainability and interpretability are top concerns, Random Forest, which provides straightforward Gini-based importance measures [44], may serve as an alternative. Our experiments reveal that with a carefully selected threshold, Random Forest performs almost on par with the 2AE model and with considerable distance to the other tested comparison algorithms.

This study also has some limitations in the feature engineering process. One exists in that the natural language processing techniques used in this study could not accurately identify the negation status of clinical concepts in sentences. In particular, we only considered unigrams when building the BoW model in order to restrict the feature size. Incorporating bigrams or trigrams may potentially capture the negation features. Another point worth deliberating is the processing of structured longitudinal format data. Features such as the total number of occurrences of a medication and the mean value of a clinical measurement's time series readings do not reflect the temporal information. There are various algorithms to extract features from time series without losing the underlying temporal order, such as Symbolic Aggregate approXimation (SAX) [45]. Although we argue that the single-point features are sufficient to represent the short-term temporal information, the limited observation window could in itself be a potential limitation. Therefore, data mining on the long period history might reveal some unexpected hidden insights that help to improve the machine learning model performance.

6. Conclusion

We have demonstrated how machine learning can be employed to analyze both structured and unstructured data from the EHRs for the purpose of preventable ADE prediction. In this study, we proposed a dual-autoencoder algorithm that learns patterns from differently labeled groups. The empirical experimental results show that our approach yields good performance and outperforms other more mainstream approaches. The proposed method can be further adapted to detect ADE

incidence only. The proposed 2AE model can be configured using the threshold and hyperparameters according to the actual needs, such as to give more attention to the precision level to accommodate practical use by an evaluator working under time constraints.

In our future work, we plan to validate the model in a real-world clinical context at the JBZ hospital, and potentially integrate this framework into a clinical AI assistant. With such a tool, the doctors and pharmacists would be supported to detect preventable ADE or general ADE cases upon a patient's ED-visit more efficiently, where preventable ADE or general ADE probability score generated by the AI assistant can be a reference for the doctors and pharmacists to decide the patient checking priority. We plan to construct a feedback loop where the doctors and pharmacists can assess the predicted cases and feed back their manual assessment to the model enabling continuous learning, resulting in a model representing up to date almost real-time clinical knowledge. Parallelly, to enrich the training data, 2AE model can be applied to historical patient data or data from other hospitals to quickly narrow down the checking scope, where additional preventable ADE cases can be built up with the help of domain experts' judgment.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

Figure A1 presents the prediction performance of the comparison models with respect to different thresholds set for the preventable ADE score. The missing values in the images imply that they are null values in the results.

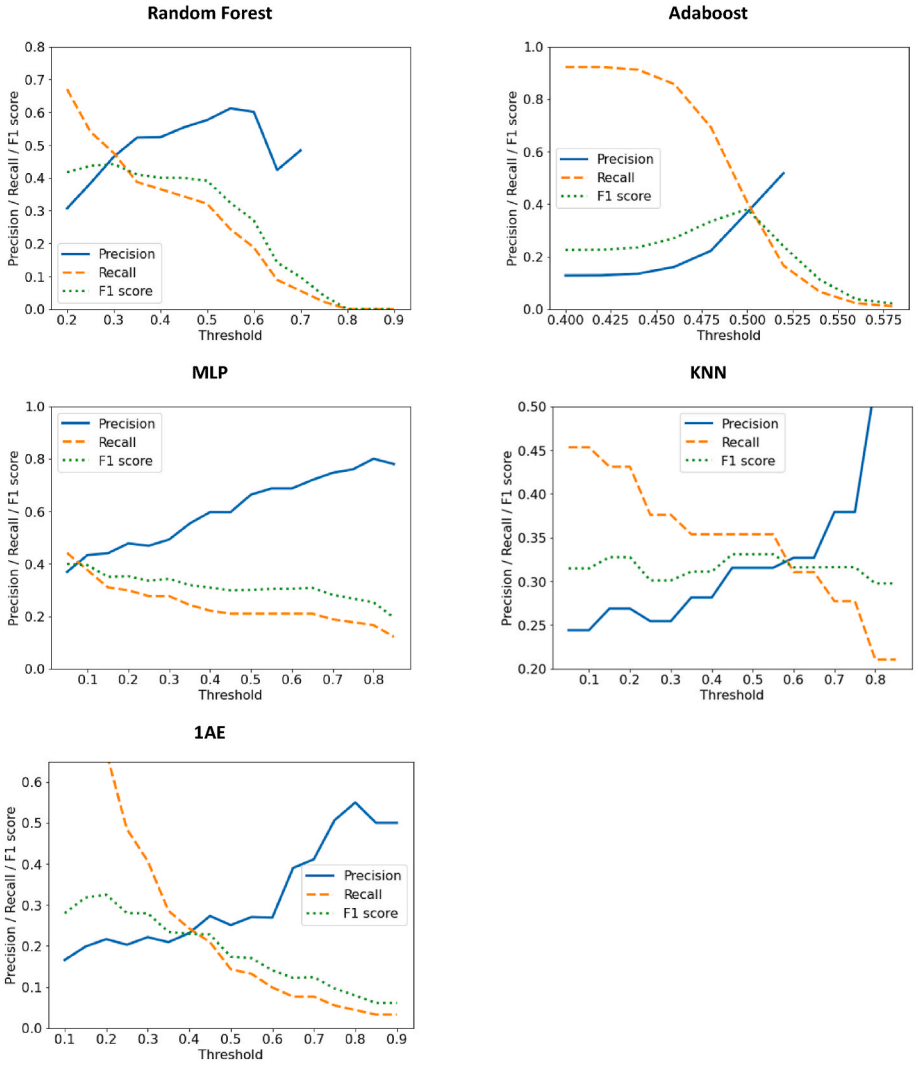

Fig. A.1. The prediction performance of the other five comparison models with respect to different thresholds set for the preventable ADE score

As shown in Table A1, among the top 10 important features in the 2AE model, the majority are BoW features. The second most important feature group is current-in-use medication records, whose feature names have the prefix "CS" in them.

Table A.1
The top 10 important features in the 2AE model computed by Python package ELI5. The importance score is defined as the decrease in minority class F1 score when a feature's values are shuffled.

Feature	Importance Score
bow_centrum	0.027473
bow_traumatische	0.014245
bow_besproken	0.014245
CS000034	0.014245
CS000032	0.014245
bow_mgl	0.014245
CS000096	0.014245
bow_presenteert	0.014245
bow_farmacologie	0.014245
J01DC02	0.014245

References

[1] Klopotowska JE, et al. The effect of an active on-ward participation of hospital pharmacists in Internal Medicine teams on preventable Adverse Drug Events in elderly inpatients: protocol of the WINGS study (Ward-oriented pharmacy in newly admitted geriatric seniors). BMC Health Serv Res 2011;11(1):124.

[2] Leendertse AJ, Egberts ACG, Stoker LJ, van den Bemt PMLA. Frequency of and risk factors for preventable medication-related hospital admissions in The Netherlands. Arch Intern Med 2008;168(17):1890–6.

[3] Leendertse AJ, Van Den Bemt PMLA, Poolman JB, Stoker LJ, Egberts ACG, Postma MJ. Preventable hospital admissions related to medication (HARM): cost analysis of the HARM study. Value Health 2011;14(1):34–40.

[4] Ouchi K, Lindvall C, Chai PR, Boyer EW. Machine learning to predict, detect, and intervene older adults vulnerable for adverse drug events in the emergency department. J Med Toxicol 2018;14(3):248–52.

[5] Gurwitz JH, et al. Incidence and preventability of adverse drug events among older persons in the ambulatory setting. JAMA 2003;289(9):1107–16.

[6] Choi E, Biswal S, Malin B, Duke J, Stewart WF, Sun J. Generating multi-label discrete patient records using generative adversarial networks. In: Machine learning for healthcare conference; 2017. p. 286–305.

[7] Eriksson R, Jensen PB, Frankild S, Jensen LJ, Brunak S. Dictionary construction and identification of possible adverse drug events in Danish clinical narrative text. J Am Med Inf Assoc May 2013;20(5):947–53. https://doi.org/10.1136/amiajnl-2013-001708.

[8] Henriksson A, Kvist M, Hassel M, Dalianis H. Exploration of adverse drug reactions in semantic vector space models of clinical text. 2012.

[9] LePendu P, et al. Pharmacovigilance using clinical notes. Clin Pharmacol Therapeut 2013;93(6):547–55.

[10] Zhao J, Henriksson A, Asker L, Boström H. Predictive modeling of structured electronic health records for adverse drug event detection. BMC Med Inf Decis Making 2015;15(4):S1. https://doi.org/10.1186/1472-6947-15-S4-S1.

[11] Zhao J, Papapetrou P, Asker L, Boström H. Learning from heterogeneous temporal data in electronic health records. J Biomed Inf 2017;65:105–19.

[12] Cheng Y, Wang F, Zhang P, Hu J. Risk prediction with electronic health records: a deep learning approach. In: Proceedings of the 2016. SIAM International Conference on Data Mining; 2016. p. 432–40.

[13] Trifirò G, et al. Data mining on electronic health record databases for signal detection in pharmacovigilance: which events to monitor? Pharmacoepidemiol Drug Saf 2009;18(12):1176–84.

[14] Schuemie MJ, et al. Using electronic health care records for drug safety signal detection: a comparative evaluation of statistical methods. Med Care 2012;50(10):890–7.

[15] Goodfellow I, et al. Generative adversarial nets. In: Advances in neural information processing systems; 2014. p. 2672–80.

[16] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[17] Yang Y, et al. GAN-based semi-supervised learning approach for clinical decision support in health-IoT platform. IEEE Access 2019;7:8048–57.

[18] Zhang L, Yang H, Jiang Z. Imbalanced biomedical data classification using self-adaptive multilayer ELM combined with dynamic GAN. Biomed Eng Online 2018;17(1):181.

[19] Lee D, et al. Generating sequential electronic health records using dual adversarial autoencoder. J Am Med Inf Assoc 2020;27(9):1411–9.

[20] Simidjievski N, et al. Variational autoencoders for cancer data integration: design principles and computational practice. Front Genet 2019;10:1205.

[21] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error propagation. California Univ San Diego La Jolla Inst for Cognitive Science; 1985.

[22] Alhassan Z, Budgen D, Alshammari R, Daghstani T, McGough AS, Al Moubayed N. Stacked denoising autoencoders for mortality risk prediction using imbalanced clinical data. 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA) 2018:541–6.

[23] Wu E, Cui H, Welsch RE. Dual autoencoders generative adversarial network for imbalanced classification problem. IEEE Access 2020;8:91265–75.

[24] Ng WWY, Zeng G, Zhang J, Yeung DS, Pedrycz W. Dual autoencoders features for imbalance classification problem. Pattern Recogn 2016;60:875–89.

[25] Chen Z, Tian Y, Zeng W, Huang T. Detecting abnormal behaviors in surveillance videos based on fuzzy clustering and multiple auto-encoders. 2015 IEEE International Conference on Multimedia and Expo (ICME) 2015:1–6.

[26] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. 2013. *arXiv preprint arXiv:1301.3781*.

[27] Xie F, Chakraborty B, Ong MEH, Goldstein BA, Liu N. Autoscore: a machine learning–based automatic clinical score generator and its application to mortality prediction using electronic health records. JMIR Med Inform 2020;8(10):e21798.

[28] Chollet F. Building autoencoders in keras. The Keras Blog 2016;14.

[29] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics; 2010. p. 249–56.

[30] Řehůřek R, Sojka P. Gensim—statistical semantics in python. Retrieved from genism. org; 2011.

[31] Pedregosa F, et al. Scikit-learn: machine learning in Python. J Mach Learn Res 2011;12:2825–30.

[32] Chollet François, GitHub - keras-team/keras. Deep learning for humans. https://github.com/keras-team/keras.

[33] Ozenne B, Subtil F, Maucort-Boulch D. The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases. J Clin Epidemiol 2015;68(8):855–9.

[34] Goadrich M, Oliphant L, Shavlik J. Gleaner: creating ensembles of first-order clauses to improve recall-precision curves. Mach Learn 2006;64(1):231–61.

[35] Sabanayagam C, et al. A deep learning algorithm to detect chronic kidney disease from retinal photographs in community-based populations. The Lancet Digital Health 2020;2(6):e295–302.

[36] Saito T, Rehmsmeier M. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS One 2015;10(3):e0118432.

[37] Zhao J, et al. Learning from longitudinal data in electronic health record and genetic data to improve cardiovascular event prediction. Sci Rep 2019;9(1):1–10.

[38] Lauritsen SM, et al. Explainable artificial intelligence model to predict acute critical illness from electronic health records. Nat Commun 2020;11(1):1–11.

[39] Ashfaq A, Sant'Anna A, Lingman M, Nowaczyk S. Readmission prediction using deep learning on electronic health records. J Biomed Inf 2019;97:103256.

[40] Wang H, Li Y, Khan SA, Luo Y. Prediction of breast cancer distant recurrence using natural language processing and knowledge-guided convolutional neural network. Artif Intell Med 2020;110:101977.

[41] Nickel CH, et al. Drug-related emergency department visits by elderly patients presenting with non-specific complaints. Scand J Trauma Resuscitation Emerg Med 2013;21(1):15.

[42] Fisher A, Rudin C, Dominici F. All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously. J Mach Learn Res 2019;20(177):1–81.

[43] TeamHG-Memex. TeamHG-Memex/Eli5. https://github.com/TeamHG-Memex/eli5.

[44] Qi Y. Random forest for bioinformatics. In: Ensemble machine learning. Springer; 2012. p. 307–23.

[45] Lin J, Keogh E, Lonardi S, Chiu B. A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery; 2003. p. 2–11.