information

Article

Predicting an Optimal Medication/Prescription Regimen for Patient Discordant Chronic Comorbidities Using Multi-Output Models

Ichchha Pradeep Sharma, Tam V. Nguyen, Shruti Ajay Singh and Tom Ongwere *

Department of Computer Science, University of Dayton, 300 College Park, Dayton, OH 45469, USA; tnguyen1@udayton.edu (T.V.N.); singhs30@udayton.edu (S.A.S.)
* Correspondence: tongwere1@udayton.edu; Tel.: +1-812-361-8621

Abstract: This paper focuses on addressing the complex healthcare needs of patients struggling with discordant chronic comorbidities (DCCs). Managing these patients within the current healthcare system often proves to be a challenging process, characterized by evolving treatment needs necessitating multiple medical appointments and coordination among different clinical specialists. This makes it difficult for both patients and healthcare providers to set and prioritize medications and understand potential drug interactions. The primary motivation of this research is the need to reduce medication conflict and optimize medication regimens for individuals with DCCs. To achieve this, we allowed patients to specify their health conditions and primary and major treatment concerns, for example, costs of medication, interactions with current drugs, and weight gain. Utilizing data gathered from MTurk and Qualtrics, we gained insights into healthcare providers' strategies for making/customizing medication regimens. We constructed a dataset and subsequently deployed machine learning algorithms to predict optimal medication regimens for DCC patients with specific treatment concerns. Following the benchmarking different models, Random forest emerged as the top performer, achieving an accuracy of 0.93. This research contributes significantly to the enhancement of decision-making processes, empowers patients to take a more active role in their healthcare, and promotes more informed and productive discussions between patients and their care teams.

Keywords: complex chronic diseases; discordant chronic conditions; benchmarking; recommender systems; machine learning; decision aids

Citation: Sharma, I.P.; Nguyen, T.V.; Singh, S.A.; Ongwere, T. Predicting an Optimal Medication/Prescription Regimen for Patient Discordant Chronic Comorbidities Using Multi-Output Models. *Information* **2024**, *15*, 31. https://doi.org/10.3390/info15010031

Academic Editor: Giorgio Maria Di Nunzio

Received: 29 November 2023
Accepted: 21 December 2023
Published: 5 January 2024

1. Introduction

In the United States alone, one in four patients have multiple chronic conditions [1]. The increasing number of patients with chronic conditions exerts further pressure on an already strained healthcare system. It is difficult for healthcare providers to thoroughly understand the complex care-needs of these people in the limited time allotted in a medical appointment [2]. This problem is even worse for patients with discordant chronic comorbidities (DCCs), a situation where a patient has two or more conditions with conflicting treatment plans. Patients with DCCs often have to juggle multiple complex treatment plans and interacting diseases/symptoms [3–5]. The discordant chronic comorbidities care (DC^3) model shows how a change in a patient treatment plan can negatively impact symptoms and necessitate revisiting the plan [6]. These interactions make treatment decisions, prioritization, and adherence for DCCs very complex and challenging for patients and their healthcare providers. A treatment plan for DCCs must adapt as the patient's conditions evolves. Machine learning (ML) and artificial intelligence (AI) can support healthcare providers when making these intricate and ever-changing decisions. When deciding on medication, the majority of patients with multiple chronic conditions often are concerned about whether (i) the price of medication is high, (ii) a medication will cause weight gain, (iii) a medication will cause severe side effects, and (iv) a medication will interact/conflict

with their other current medications [6]. Thus, it is essential for researchers looking to support the treatment decision-making process for DCCs to consider these concerns when making treatment recommendations. A plethora of research has been conducted for risk prediction [7], disease diagnosis [8], managing treatment plans [9], and medication recommendation targeting a single chronic disease [10]. We have yet to see the work targeting the prioritization of complex needs experienced by patients with DCCs. Research is needed to explore how to support patients with DCCs in navigating their complex care-needs and prioritizing their treatment plans.

In this study, we trained and tested machine learning models to predict the optimal medication regimens for individuals struggling with DCCs. Our approach leverages data collected from MTurk and Qualtrics, which capture the strategies employed by healthcare providers in prescribing medication regimens that address patients' specific treatment concerns, including cost, weight gain, and drug interactions. We started by distributing a survey through MTurk and Qualtrics, aimed at gaining insights into the decision-making process of healthcare providers when prescribing medications to their patients with complex needs. Using the survey results, we constructed a comprehensive dataset comprising medication-regimen for managing three prevalent DCCs (type 2 diabetes, arthritis, and depression). These combinations of medication were meticulously crafted to accommodate the primary treatment concerns of patients, namely cost, weight, and drug interactions. Subsequently, we employed this dataset to train and test various multi-output models, ultimately arriving at predictions for optimal medication combinations targeting DCCs. This study not only presents the results of our predictive models but also establishes benchmarks to evaluate the performance of each algorithm utilized. By doing so, it makes the following contributions:

1. Enhanced Decision-Making—this research strives to empower healthcare professionals to tailor medication regimens more effectively for patients with DCCs, thereby enabling them to make more informed and precise decisions.
2. Empowering Patients—the study aspires to provide patients with a deeper understanding of alternative medication options, allowing them to actively participate in their healthcare decisions and receive more personalized treatment plans.
3. Facilitating Informed Discussions—patients equipped with knowledge from this study will be better equipped to engage in meaningful discussions with their care teams regarding medication options. This in turn, leads to improved treatment outcomes.

2. Related Works

In this section, we discuss the use of machine learning tools and algorithms to support the management of a single chronic disease. We explore benchmarking suites, decision support systems, and recommender tools.

2.1. Recommender and Decision Support Systems

Clinical decision support systems (CDSS) and recommender systems are designed to assist physicians, nurses, patients, and other professionals in decision-making related to the patient's clinical condition [11]. For example, Lysaght et al. developed and implemented AI-assisted support systems to support healthcare and clinical-practice decision-making processes [12]. They used large datasets from electronic health records (EHRs) and algorithms in CDSS. A CDSS typically employs computerized, predictive analysis algorithms to filter, organize, and search for patterns in big datasets from multiple sources and provide probability analysis upon which healthcare providers can make fast and informed decisions. This research highlighted some of the ethical issues that may arise with the implementation of these systems and the relevant values that decision-makers can draw on in the design and implementation of AI-assisted CDSS into practice [12]. The decision support was further enhanced with tools that could look for the common attributes and the nearest neighbors of these attributes [13]. These tools could predict the most probable future actions of patients and identify the disease (s) a patient would most likely develop in the

near future. With some additional tweaks, these tools could also recommend educational material for such diseases [14]. Many such varieties of these tools were developed to assist patients and healthcare providers.

The majority of patients struggle to find useful resources, strategies, and information when navigating their care and self-management [6,15]. Patients find it hard to identify the most relevant and valuable materials for themselves [15,16]. A system that automatically identifies and recommends appropriate care strategies to patients based on their needs or preferences is needed. There is already research taking that direction, for example, health recommender systems (HRS) are used to provide appropriate educational materials for patients with chronic diseases [17]. Such a recommender system detects the similarities between the patient and text vectors by using keyword extraction. They show how ontology-vector spaces can be used to correlate patient data and educational material. However, such systems do not capture the deep semantic meanings behind sentences or documents.

2.2. Machine Learning Tools and Algorithms for Healthcare

Several studies are currently exploring/implementing machine learning techniques to support the care and wellness of patients. For example, Woldaregay et al. explored state-of-the-art machine-learning strategies and their hybrid systems focusing on blood glucose (BG) anomaly classification and detection [18]. In addition, ML algorithms such as artificial neural networks, support-vector machines, Bayesian networks, decision trees, and back-propagation algorithms, have been applied to create decision-aid systems for supporting healthcare providers and nurses in their decision-making process [19]. Kavakiotis et al. built predictive models using machine learning algorithms and data mining techniques for diabetes prediction [10]. They used the k-means, application of tree algorithms, decision tree algorithms, neural networks, K-means clustering algorithms, and visualization to predict diabetes among patients. Logistic regression gave the highest accuracy of 96%. This approach could also be applied to chronic comorbidities for prediction and recommendations to manage them. Furthermore, Singh et al. created a multi-output career prediction tool that considers the person's background history (i.e., work and education history) [20].

Apart from implementing ML algorithms, there exists a body of research dedicated to performance of ML algorithms. For example, some studies have investigated and created benchmarks of probabilistic matrix factorization, generative adversarial networks (GAN), and attention-based sequence models. These investigations revealed that the factorization method was relatively simple to interpret [21]. Additionally, they highlighted the remarkable enhancements in medical predictive model performance attributed to deep learning approaches [21]. In a separate study, Kumar et al. evaluated anxiety, depression, and stress using machine learning models [22]. They conducted comprehensive benchmarking, using five (n = 5) distinct algorithms, namely decision tree, random forest tree, naive Bayes, SVM, and KNN. For classification, they employed logistic regression, cat boost, naive Bayes, RFT, and SVM. The outcomes indicated superior performance by the decision tree, followed by the random forest and then naive Bayes algorithms. These studies show the potential for recommending both diseases and corresponding medications. However, it is worth noting that while the insights from this research could benefit individuals with DCCs, the majority of machine learning models still struggle to accurately estimate uncertainty and furnish well-calibrated predictions [23,24]. This deficiency can result in overly confident recommendations in scenarios involving dataset shifts or distributional changes [25,26]. To address this, further endeavors that mitigate uncertainty when recommending medication combinations within real-world clinical settings essential.

3. Proposed Framework

This section provides an overview of our study design, including participant selection, data quality and cleansing processes, as well as the attributes employed for predicting optimal medication combinations. It also outlines our approach to iterative testing and multi-model training scenarios.

3.1. Study Design

3.1.1. Participant Selection

Our study employed a mix of data collection methods to recruit highly qualified healthcare providers with self-reported expertise in managing type 2 diabetes and its common discordant chronic comorbidities (DCCs), such as depression, chronic kidney disease, and arthritis. We utilized Amazon Mechanical Turk (MTurk) and collaborated with two healthcare centers. MTurk allowed us to access a diverse group of participants and collect a large number of responses efficiently [27]. Our recruitment criteria were meticulously tailored to the focus of our research, exclusively selecting providers with relevant professional experience in managing conditions that aligned with our criteria. To diversify our participant pool, we also utilized snowball sampling. Healthcare providers who did not treat patients with type 2 diabetes and at least one of these specified DCCs were excluded, ensuring our dataset comprised individuals with pertinent clinical experience in DCC management (see Table 1). The study was reviewed and approved by the University of Dayton Institutional Review Board.

Table 1. The Summary of study participants.

Category	Attribute	Number of Participant
Experience	Knowledgeable	162
	Very knowledgeable	64
Diseases Treated	Type 2 diabetes (T2D)	226
	T2D & one other disease	62
	T2D & two other diseases	109
	T2D & more than two diseases	55
Primary Specialization	Family medicine	112
	Endocrinologist	14
	General internal medicine	63
	General pediatrics	32
	Gynecology	8
	Nephrologist	12
	Other	23

The study collected data from a total of 226 participant responses. All 226 participants self-reported treating patients with type 2 diabetes. Among these, 62 participants reported caring for patients with at least two DCCs (type 2 diabetes and depression or type 2 diabetes and arthritis). Additionally, 109 participants reported treating patients with a combination of three DCCs including type 2 diabetes, depression, and arthritis. Furthermore, 55 participants had experience in attending to patients with type 2 diabetes alongside other health conditions, including heart diseases, cancer, and more.

3.1.2. Participant and Data Screening

We employed several quality assurance measures to ensure data quality and reliability. We utilized the Qualtrics survey platform to construct our survey. We initiated the survey with four unpaid screening questions to filter out participants who did not meet our predefined selection criteria. This strategic placement of screening questions at the beginning of the survey minimized time commitment for disqualified participants, thus optimizing data collection efficiency. Additionally, we provided clear instructions and detailed explanations in each survey section and/or subsection of our survey to reduce potential sources of ambiguity or misunderstanding among respondents. Finally, we validated the responses gathered by enlisting five clinical experts (from the local hospital) who had also participated in the Qualtrics survey and were familiar with our research. Each expert played a substantial role in assessing and confirming the professionalism of the participant (N = 20) responses from MTurk that were randomly assigned to them. The clinical experts' insights were instrumental in pinpointing and flagging participants whose responses deviated from the survey questions or contained nonsensical responses. They started by identifying instances where responses appeared unrelated to the provided survey questions and there

was nonsensical or incoherent language present. Furthermore, we asked them to look for indications suggesting a participant may have misinterpreted the survey question. All responses from any participants flagged during this validation process because of potential relevance, coherence, or comprehension concerns were subsequently excluded from our study, contributing to the data quality and accuracy obtained and confirming its reliability for our research.

The following are questions we used to guide the validation process:

1. Are there instances in the responses that appeared unrelated to the provided question?
2. Do the responses contain nonsensical phrases or incoherent language?
3. Is there any indication from the responses that suggests participants might have encountered difficulties in comprehending the questions?

3.1.3. Participant Data Quality and Cleaning

As mentioned above, clinical experts were recruited via targeted qualifications and screening on Amazon Mechanical Turk (MTurk) to verify their domain expertise. A comprehensive multi-step selection process was utilized to identify healthcare providers with expertise in treating patients with type 2 diabetes and sound understanding of complex medication needs. The process commenced with a screening survey, followed by detailed descriptions for each survey section, outlining the specific objectives and expectations for participants. After completion, participants' responses were verified by a member of the research team and validated by clinical experts before the data was collected for analysis. This strict recruitment and survey process ensured the engagement of expert participants possessing the required knowledge and skills, thereby enhancing the quality and precision of the collected data. Following data collection, a rigorous data cleaning process was initiated, aimed at enhancing the quality and reliability of our dataset. Initially, we conducted a thorough examination to identify and rectify missing data, eliminating any incomplete responses from our dataset. Subsequently, a comprehensive data quality assessment was carried out to identify and address outliers, inconsistencies, and errors. This meticulous data cleaning procedure was instrumental in ensuring the validity and reliability of our dataset, which ultimately consisted of 3931 rows (see Table 2). We used the cleaned dataset to train and test a carefully selected set of algorithms and models. Each model was treated as a single-output classification problem, producing a singular medication node as the output, which was a concatenated string of medications for the specific diseases under consideration. To facilitate this approach, our data format incorporated binary inputs to represent the presence or absence of particular features in each sample. This binary input format was utilized to focus on distinct aspects of the problem and enabled the development of classification models utilizing machine-learning techniques with an emphasis on average accuracy. Additionally, an alternative strategy involved the representation of our data with multiple output values corresponding to medications for various diseases. This format integrated binary inputs for diseases, along with text-based features and factors, offering a comprehensive view of the dataset.

We then used a correlation matrix to show how features in our dataset relate to one another or to the goal feature/attributes. The heatmap is a graphical representation of the correlation matrix such that the variables are displayed on both the x-axis and y-axis, with the correlation coefficients represented by color gradients in the cells [28].

3.1.4. Feature Selection

Major patient medication concerns, including effects on sex drive, weight gain, medication cost, easy-to-stop medication, and interactions with existing medications, are well founded and supported by Mayo Clinic research. Mayo Clinic's extensive work on decision aids highlights how these concerns play significant roles in healthcare decision-making and patient outcomes. Our dataset captures weight gain, cost, and drug interactions because these three factors directly impact treatment adherence, access to medication, and treatment outcome.

Table 2. Data with single outputs.

Diabetes	Arthritis	Depression	Cost	Weight	Interaction	Medicine
1	0	0	1	1	0	Metformin
1	0	0	1	1	0	Victoza
1	0	0	1	1	0	Metformin
1	0	0	1	1	0	Victoza
1	0	0	1	1	0	Victoza
1	0	0	1	1	0	Metformin
1	0	0	1	0	1	Victoza
1	0	0	1	0	1	Metformin
1	0	0	1	0	1	Victoza
1	0	0	1	0	1	Metformin
1	0	0	1	0	1	Metformin
1	0	0	1	0	1	Metformin

During data cleaning, each variable forms a column, and each observation forms a row. One cell data creates one row in the cleaned data. The processed data are free of irregular and non-selected values. The medicines are mapped to combination of DCC diseases and patients' major concerns (a diabetic patient who cares about the cost of medication and weigh gain).

3.2. Exploring the Potential Machine Learning Algorithms

We selected random forest, k-nearest neighbors (KNN), AdaBoost, and XGBoost as the machine learning models for predicting medication recommendations for patients with complex chronic conditions, primarily because they are popular machine learning algorithms and commonly used algorithms in the healthcare context [29,30].

3.2.1. Technical Backgrounds of Machine Learning Algorithms Adapted in This Study

- Random Forest: Random forest is a widely used ensemble learning algorithm known for its ability to create robust and accurate models. Operating as an ensemble of decision trees, each trained on a random subset of data and features, it combines their predictions for enhanced accuracy and robustness [31]. Random forest has been used in medical recommender systems for disease diagnosis, medication recommendations [32], treatment planning [33], patient outcome prediction [34], and drug–drug interaction prediction [32,35,36]. Its capability to handle heterogeneous medical data and generate interpretable results makes it a valuable tool for developing effective medical recommender systems.
- K-Nearest Neighbors (KNN) K-nearest neighbors (KNN) is a supervised machine learning algorithm used for classification and regression tasks [37]. It operates by assessing the similarity of data points in a multi-dimensional space and making predictions based on the majority class of its k-nearest neighbors. KNN has been widely applied for type 2 diabetes diagnosis [38], medication recommendations [32], personalized treatment planning [33], and drug interaction detection [32,35,36]. Such applications use similarities in patient data to enhance healthcare decision-making and outcomes. Thus, proper selection of distance metrics, as well as addressing issues such as feature selection and missing data, are crucial when using such applications.
- AdaBoost: AdaBoost, short for adaptive boosting, is an ensemble learning method developed to enhance the performance of weak learners [39]. It operates by assigning weights to training instances, giving more weight to misclassified examples in each iteration. Sequential iterations build an ensemble of weak classifiers, with each subsequent classifier correcting the errors of its predecessors. Classifier weighting is based on accuracy, and a final ensemble classifier is formed by combining the weighted base classifiers [39]. AdaBoost has been used in medication and treatment recommendation systems [40] to provide personalized health advice and brain MR image classification and analysis [41,42].
- XGBoost: XGBoost, or extreme gradient boosting, is a highly efficient and versatile machine learning algorithm known for its proficiency in structured and tabular data

tasks [43]. It operates on the gradient boosting framework, sequentially building an ensemble of decision trees to correct errors, and includes L1 and L2 regularization terms to prevent overfitting. XGBoost has been applied in type 2 diabetes prediction and diagnosis [44], personalized treatment recommendation, drug interaction prediction [45], and health behavior recommendations [44]. These applications use XGBoost's structured data handling capabilities and efficiency for improved clinical decision-making and patient outcomes.

3.2.2. Predicting Optimal Medication Combination

We utilized the created dataset to predict medication combinations for patients dealing with DCCs, each with specific health concerns. For instance, a patient with DCCs who is primarily concerned about managing their weight may initially receive a medication recommendation that includes Victoza and Hydroxychloroquine (the attributes we aim to predict). However, if the same patient later expresses concerns about the cost of their treatment, the recommended medication combination shifts to Empagliflozin and Aspirin. This exemplifies how the medication combination adapts in response to the incorporation of additional factors. See Figure 1 for additional factors.

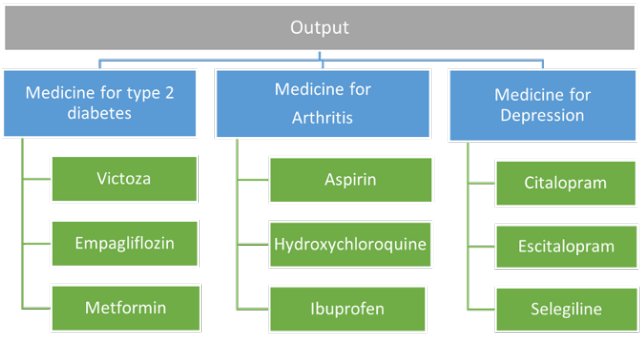

Figure 1. Classes for the multi-output model. This figure illustrates how medication recommendations may evolve for a patient with multiple concurrent conditions (DCCs) based on their changing concerns. A patient whose primary concern is weight management is prescribed to take Victoza and Hydroxychloroquine. However, if they add cost to list of their concerns, their prescribed medication combination shifts to Empagliflozin and Aspirin.

In the subsequent subsections, we outline our methodology for conducting comprehensive testing through multiple test cases (iterations). Our predictions provide approximate results based on the selected factors/concerns and combinations of diseases. See Figure 2. During the testing phase, we present a comparative analysis of predictions generated by various trained models. Furthermore, we investigate the medication recommendations provided by these different models, shedding light on the effectiveness of each approach.

3.3. Iteration 1: Base Classification Models

We adopted an iterative approach (three iterations) to train and test a selected number of classifiers. This approach allowed us to improve the performance by changing the models' nature and parameters, introducing wrapper classes, and changing the data format. In our first iteration, we focused on developing a standard model using a single output machine learning approach. This approach is specifically designed to recommend or categorize a single target variable or output. For the implementation of our base model, we utilized the widely acclaimed scikit-learn library [20,46]. In this segment, we will unveil the data format, classifiers, and parameters used for the benchmarking suite.

3.3.1. Data Format

During the first iteration phase, we represented data to have multiple features and a single output. The target values were set to be a concatenated value of all the selected

medication. See Table 3. We used these concatenated data to train our models. Since the data had multiple data types, we identified the need to implement encoders to our dataset. By using various encoding methods, we transformed our dataset from categorical or text data into numerical representations that can be processed by machine learning algorithms.

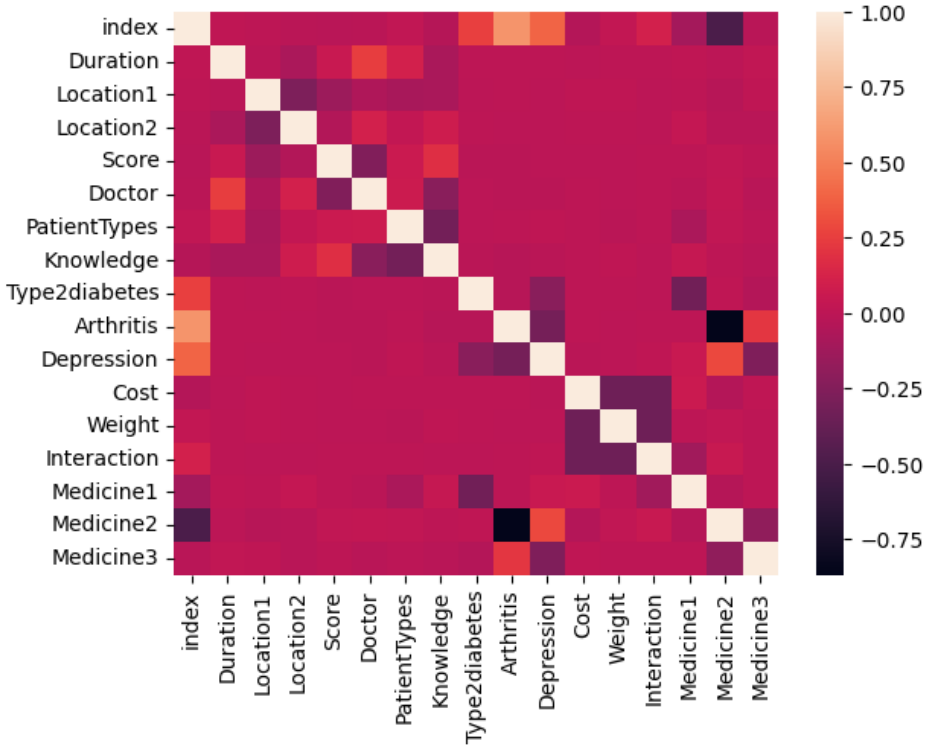

Figure 2. Correlation matrix showing how features related to one another. The intensity of the color represents the strength of the correlation between the variables, with darker colors indicating a stronger correlation and lighter colors indicating a weaker correlation. See figure heatmap, Medicine 2 has a high correlation with Arthritis, and Interaction has a high correlation with Weight.

Table 3. The data population from survey response.

Diabetes	Arthritis	Depression	Cost	Weight	Interaction	Medicine
1	0	0	1	1	0	Metformin
1	0	0	1	1	0	Victoza
1	0	0	1	1	0	Metformin
1	0	0	1	1	0	Victoza
1	0	0	1	1	0	Victoza
1	0	0	1	1	0	Metformin
-	-	-	-	-	-	-
-	-	-	-	-	-	-
1	1	1	1	1	1	V + H + Es
1	1	1	1	1	1	M + H + Es
1	1	1	1	1	1	Em + I + S
1	1	1	1	1	1	Em + H + Es
1	1	1	1	1	1	M + I + Es

In this table, we show a sample of classes for the multi-output model, each representing a distinct combination of medications. These classes include V + H + Es (Victoza, Hydroxychloroquine, and Escitalopram); M + H + Es (Metformin, Hydroxychloroquine, and Escitalopram); Em + I + S (Empagliflozin, Ibuprofen, and Selegiline); Em + H + Es (Empagliflozin, Hydroxychloroquine, and Escitalopram); M + I + Es (Metformin, Ibuprofen, and Escitalopram). These classes serve as a categorization framework for the medication combinations used in the study.

3.3.2. Algorithms

We evaluated classification algorithms with random forest, KNN, AdaBoost, and XGBoost for a single output machine learning model. Let us dive into the classifiers and the settings that influenced the base model.

- Random Forest: We utilized a random forest classifier and implemented it as our classification model. We instantiated the classifier with 100 decision trees and a maximum depth of 12. We set the random state to 2 for consistent results during model training and then fitted the model. After training the model, we assessed its accuracy on the training set using the 'score' method. The accuracy score was calculated by comparing the recommendations made by the trained model to the true labels in the training set. In Figure 3, we can see how tree construction starts from the root node, and at each step, the algorithm selects the best split point based on certain criteria, usually maximizing information gain. The obtained results, in terms of accuracy, were not high.

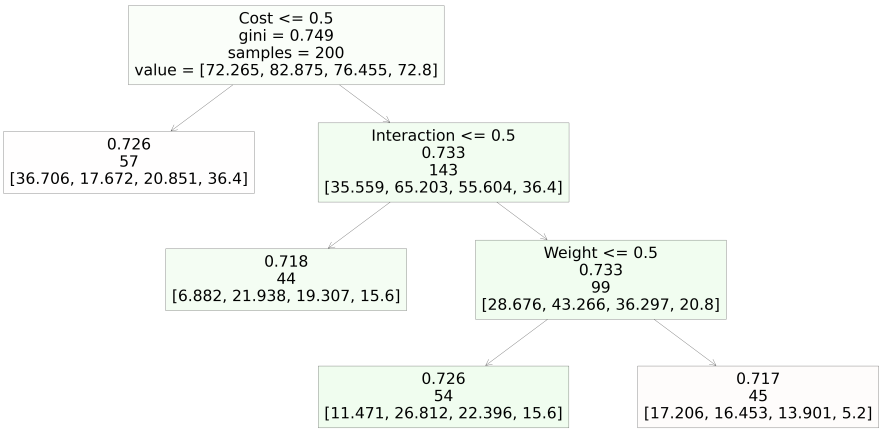

Figure 3. Random forest decision tree. The first decision point "Cost <= 0.5" exhibits a Gini impurity of 0.749 and consists of 200 samples with values [72.265, 82.875, 76.455, 72.8]. The second layer, the left subtree, has a threshold of "0.72", 57 samples, and values [36.706, 17.672, 20.851, 36.4], while the right subtree is determined with "Interaction <= 0.5," and has a Gini impurity of 0.733 and encompasses 143 samples with values [35.559, 65.203, 55.604, 36.4]. Further, the third layer shows the right subtree, evaluated by "Weight <= 0.5," with a Gini impurity of 0.733 and 99 samples returning values [28.676, 43.266, 36.297, 20.8].

- K-Nearest Neighbors Classifier: KNN classifier is a non-parametric and instance-based learning method that makes recommendations based on the similarity of the input data to its neighboring data points [47]. We configured the model with the following parameters: 'algorithm' set to 'kd_tree', 'leaf_size' set to 10, 'metric' set to Euclidean, 'n_jobs' set to 10, 'n_neighbors' set to 4, 'p' set to 3, and 'weights' set to 'uniform'. We performed the tuning by implementing both Euclidean and Minkowski distance. We achieved the best results using the Euclidean distance paired with uniform weights, 3 neighbors.
- Extreme Gradient Boosting (XGBoost) Classifier: The XGBoost classifier is an implementation of the gradient boosting algorithm, which is known for its effectiveness in various machine learning tasks [48]. We instantiated with specific hyperparameters including 200 estimators, a maximum depth of 4, a learning rate of 0.1, subsampling and feature subsampling rates of 0.2, regularization parameters of 0.01 for both L1 and L2 regularization, a gamma value of 0.05, and a random state of 32. Finally, we used the trained classifier to predict the labels for the testing data, and the predicted labels are stored in the 'preds' variable for further analysis or evaluation.

- Adaptive Boost (AdaBoost) Classifier: To instantiate the AdaBoost classifier, we configured the following parameters: 'n_estimators' set to 128, 'learning_rate' set to 0.001, and 'random_state' set to 42. These parameters control the number of weak classifiers to combine, the learning rate of the model, and the random seed for reproducibility. The use of a single output classifier in the initial iteration of the project had its limitations. While this approach is designed to predict or categorize a single target variable, it may not be the optimum solution for problems involving multiple output variables or complex dependencies among the variables. Additionally, the single output classifier may have limited complexity, hindering its ability to capture intricate patterns in the data. The choice of a single output classifier also restricted the modeling of the concatenated target values representing multiple medications. These limitations highlight the need to explore alternative approaches that can better handle multiple output variables and capture the complexity of the problem at hand.

3.4. Iteration 2: Multi-Class Multi-Output Classification Models

With the need to improve the performance of the base model, we proceeded to the next iteration, where we expanded our approach to handle multiple classes and outputs simultaneously. It is an extension of the traditional multi-class classification problem, where the goal is to assign an input instance to one of several predefined classes. This advancement allowed us to tackle more complex classification tasks and provide a comprehensive analysis of the data. Table 4 shows the classification problems and the number of targets that they can achieve.

Table 4. The number and dimensions of targets according to classification type.

Classification	Number of Targets	Targets Cardinality
Multiclass	1	>2
Multi-label	>1	2 (0 or 1)
Multiclass and multi-output	>1	>2

3.4.1. Data Format

In order to capture complexities within our dataset, we introduced a pivotal enhancement to the data format. This enhancement and transformation differentiated our dataset from the previous approaches that featured only a singular output variable. We included the incorporation of manifold output variables aligned with distinct disease criteria. Additionally, our novel data structure consisted of a multi-class configuration wherein individual classes corresponded to specific disease categories. As such, the multi-class scheme was now accompanied by a plurality of associated outputs. Table 5 shows the visual exposition of the output categorizations. Through this strategic modification, the model gained the capacity to concurrently address multiple target variables. Consequently, the analytical and predictive capabilities pertaining to disease criteria attained a heightened level of comprehensiveness.

3.4.2. Algorithms

In the second iteration, we performed an evaluation of classification using the same set of algorithms as in the first iteration. This allowed us to compare and assess the performance of these algorithms on the given task. By applying these algorithms to the data and analyzing their results, we aimed to gain insights into their effectiveness in capturing the patterns and making accurate predictions.

- Random Forest with Multi-Output Classifier: To enable multi-output functionality, the random forest classifier is wrapped inside a multi-output classifier. This wrapper allows the random forest classifier to handle multiple target variables simultaneously. We updated the classifier to have 100 decision trees and a maximum depth of 12. We set the random state to 2 for reproducibility. We updated the n_jobs parameter to 2,

indicating the number of parallel jobs to use for model training. As a result of these modifications, we observed an improvement in the accuracy score. The accuracy score provides a measure of the model's performance, and the increase in accuracy suggests that the multi-output random forest classifier can better capture the underlying patterns and make more accurate predictions on the given task.

- K-Nearest Neighbor with Multi-Output Classifier: The KNN algorithm works by finding the K-nearest neighbors to a given input data point in the feature space and then averaging their corresponding target variable values to make a prediction. We instantiated a K-nearest neighbors regressor (KNN) with the following configuration: algorithm = 'kd_tree', leaf_size = 20, metric = 'minkowski', n_jobs = 15, n_neighbors = 2, and weights = 'uniform'. Then, we wrapped this regressor inside a multi-output classifier, allowing it to handle multiple target variables simultaneously. To assess the accuracy of the regressor on the training set, we used the 'score' method, which calculates the coefficient of determination (R-squared score). The R-squared score represents the proportion of variance in the target variables that can be explained by the regressor.
- Extreme Gradient Boosting (XGBoost) with Multi-Output Classifier: XGBoost (extreme gradient boosting) is an advanced machine learning algorithm that belongs to the gradient boosting family of models. We instantiated an XGBoost classifier (XGBClassifier) with the provided hyperparameters. We set the number of estimators to 200, the maximum depth of each tree to 2, the learning rate to 0.1, the subsample ratio to 0.5, the column subsampling ratio to 0.5, the L1 regularization term to 0.01, the L2 regularization term to 0.01, the minimum loss reduction required for a split (gamma) to 0.05, and the random state to 32 for reproducibility. To enable multi-output functionality, we have wrapped the XGBoost classifier inside a multi-output classifier. This approach leveraged the boosting technique and ensemble learning, enabling us to make accurate predictions for multiple target variables.
- Adaptive Boosting (AdaBoost) with Multi-Output Classifier: AdaBoostClassifier is a powerful algorithm that can improve the performance of classification models by combining multiple weak classifiers into a strong ensemble classifier. We instantiated an AdaBoost classifier with specific hyperparameters, including the number of estimators set to 64, the learning rate set to 0.01, and the random state set to 128. To enable multi-output functionality, we wrap the AdaBoostClassifier inside a multi-output classifier.

Table 5. Data with multiple outputs.

Diabetes	Arthritis	Depression	Cost	Weight	Interaction	Med1	Med2	Med3
1	0	0	1	1	0	Metformin	None	None
1	0	0	1	1	0	Victoza	None	None
1	0	0	1	1	0	Metformin	None	None
1	0	0	1	1	0	Victoza	None	None
1	0	0	1	1	0	Victoza	None	None
1	0	0	1	1	0	Metformin	None	None
1	1	1	1	1	1	V	H	Es
1	1	1	1	1	1	M	H	Es
1	1	1	1	1	1	Em	I	S
1	1	1	1	1	1	Em	H	Es
1	1	1	1	1	1	M	I	Es

Although the accuracy of the multi-class classifying algorithms showed improvement compared to the previous iterations, it is worth noting that the obtained accuracy scores were not the highest possible. This indicates that there is still room for further optimization and improvement in the performance of the classification models. Continuous evaluation, experimentation, and refinement of the algorithms can lead to achieving higher accuracy and better overall performance.

3.5. Iteration 3: Classification Model Optimization

Building upon the foundation laid in Iteration 2, in Iteration 3, we further optimize our models by tuning parameters and making additional changes. Parameter tuning plays a crucial role in improving the performance of machine learning models, allowing us to find the optimal configuration that maximizes accuracy and minimizes errors. The parameter tuning process involves testing different combinations of parameter values to determine which combination yields the best performance. We used a technique called grid search, which involves testing all possible combinations of parameter values within a defined range. The combination tests included changing the depth of the tree, the random state, estimators, and even gamma values. The combination that yielded the best performance on our evaluation metrics was selected as the optimal parameter values for the model.

4. Results and Discussion

In our work, we aimed to design a tool powered by a machine learning model to predict optimal medication for patients with discordant chronic comorbidities, specifically type 2 diabetes, arthritis, and depression. To achieve our goals, we implemented several machine learning algorithms, including random forest, K-nearest neighbors (KNN), AdaBoost, and XGBoost. Table 6 reports results for iterations 1, 2 and 3 and we discus them favor in this section.

Table 6. Priority of medications/treatments for each concern.

Algorithm/Model	Iteration 1	Iteration 2	Iteration 3
Random Forest	65.5%	83.5%	93.3%
KNN	60.2%	67.4%	78.5%
AdaBoost	35.8%	36.4%	67.3%
XGBoost	53.8%	62.4%	76.4%

The results from iterations 1, 2, and 3 offer insights into our prediction and performance evaluation based on accuracy. Accuracy serves as a transparent and easily interpretable metric, assessing the model's proficiency in correctly classifying instances by determining the ratio of accurate predictions to the total predictions made. Our next step involves the inclusion of additional metrics, such as recall, precision, F1 score, and AUC-ROC, to provide a comprehensive evaluation of our model's predictive capabilities.

4.1. Analysis of Iterations 1, 2, and 3

4.1.1. Iteration 1

We started out by training all four multi-output algorithms using default hyperparameters to establish a baseline performance for single output. The random forest algorithm for a single output showed promising results, achieving an accuracy of 65.5%. However, KNN, AdaBoost, and XGBoost achieved lower accuracies of 60.2%, 35.8%, and 53.1% respectively. These standard ML models, which were primarily designed for simpler tasks, struggled to effectively handle the intricacies of the dataset, resulting in lower accuracies and time-consuming training procedures. To overcome these challenges, we embarked on a thorough study of alternative approaches and methodologies. After carefully exploring diverse options, we made the decision to transition towards utilizing a multi-output classifier. This choice offered a promising solution, as it specifically caters to the challenges presented by datasets with multiple outputs.

4.1.2. Iteration 2

By adopting a multi-output classification, we were able to leverage its inherent capability to handle multiple outputs simultaneously. The multi-output regression allowed us to model the dependencies between the 11 inputs and 3 outputs more accurately, yielding improved predictions and reducing the code complexity. The standard models when wrapped in the multi-output classifier resulted in better performance. The random forest multi output classifier model yielded the highest accuracy of 83.5%, followed by XGBoost, KNN, and AdaBoost with the accuracies 67.4%, 62.3%, and 36.4%, respectively.

4.1.3. Iteration 3

In the final iteration, we adopted grid search techniques to fine-tune the parameters of our multiclass classification approach. This approach allowed us to define an array of parameter values and search through all possible combinations to determine the optimal parameter configuration. Following this comprehensive process, we successfully identified the best parameter set to train our classifiers and assess their performance. The resulting accuracy scores for our classification algorithms were as follows: random forest achieved 93.3%, KNN demonstrated 78.5%, AdaBoost scored 67.3%, and XGBoost reached 76.4%. These results reflect the effectiveness of our approach in solving the multiclass classification problem. These performance variations can be attributed to several factors. The first factor includes the inherent complexity of each algorithm and how they interact with the specific dataset. Random forest excelled in capturing complex data relationships, adapting well to multi-class problems and robustly handling noisy data. KNN's simplicity may have limited its performance, particularly with high-dimensional data and complex decision boundaries. We discuss the performance of each of these algorithms below.

4.2. Assessing Algorithms' Performance

In this study, we conducted a comprehensive assessment of four machine learning algorithms in a multi-class classification model (Iteration 3), focusing on their accuracy scores as performance metrics among these algorithms. Random forest emerged as the top-performing algorithm, achieving an accuracy of 93.3%. Its strength lies in its ability to effectively classify instances across multiple classes by combining multiple decision trees, and it exhibits robustness against overfitting, rendering it a dependable choice for a wide range of applications. K-nearest neighbors (KNN) performed decently, with an accuracy score of 78.5%, although its sensitivity to parameters such as 'k' and distance metrics suggests potential for improvement through fine-tuning. AdaBoost, with an accuracy of 67.3%, showcased its characteristic power in boosting weaker learners. However, AdaBoost may benefit from a closer look at base classifiers and hyperparameters, which indicates the need for further exploration and parameter adjustments. XGBoost recorded an accuracy of 76.4%, which demonstrates good efficiency in structured tabular data tasks, albeit slightly trailing behind random forest and KNN. The scope for performance enhancement in XGBoost was associated with hyperparameter settings and feature engineering.

In summary, random forest excelled with the highest accuracy, while KNN and XGBoost also performed well, warranting potential improvements through hyperparameter adjustments. AdaBoost, while a powerful algorithm, has room for improvement in this specific context. It is important for future studies looking at similar problems to consider not only the accuracy but also other metrics such as precision, recall, and F1-score, AUC-ROC, and to conduct further analysis to understand the specific strengths and weaknesses of each algorithm for a given dataset.

4.3. Benchmarking Suite

Our results indicate that the random forest algorithm is the most suitable for recommending medications for patients with type 2 diabetes, arthritis, and depression (See Figure 4). By achieving an optimal accuracy of 93.3%, it outperformed the KNN, AdaBoost, and XGBoost algorithms. The decision to utilize random forest in addressing the multiclass, multi-output problem was motivated by its remarkable capabilities. By employing an ensemble of decision trees, random forest efficiently tackles the issue of overfitting and skillfully captures intricate relationships present in the data. Its robustness in handling noisy data points further enhanced the model's reliability. Random forest offers a notable advantage in dealing with non-linear relationships, which proves to be highly advantageous, particularly considering the complex nature of the underlying data distribution. Furthermore, the algorithm's minimal data preprocessing requirements simplify the experimentation process, saving significant time and resources. In a research study conducted by Singh et al., they found that random forest performed exceptionally well for their specific

needs, which involved multiple classes and multiple outputs [20]. Similarly, our own results align with theirs, confirming that random forest is also the most effective choice for handling our multiclass, multi-output requirements.

Nevertheless, it important to note that performance of any machine learning algorithm is tied to the unique attributes of the dataset and the complexities inherent to the problem domain. Consequently, we conducted an exhaustive examination of multiple algorithms and carried out comprehensive assessments to validate that random forest was the preferred selection. This determination was based on its inherent strengths and compatibility with the specific challenges presented by our multifaceted, multiclass, and multi-output task within the domain. Further, we show how an interactive interface can be valuable in addition to machine learning tools looking to enable and empower healthcare professionals and patients with (DCCs) to choose the diseases and specify factors that are important for them when prioritizing the DCCs' treatment regimen. However, it is important to recognize that the performance of machine learning algorithms is intricately linked to the distinct characteristics of the dataset and the inherent complexities of the problem domain. As a result, we conducted an evaluation of multiple algorithms, performing thorough assessments to affirm that random forest emerged as the best performing algorithm. This determination was based on its inherent strengths and compatibility with the specific challenges presented by our multifaceted, multiclass, and multi-output task within the domain.

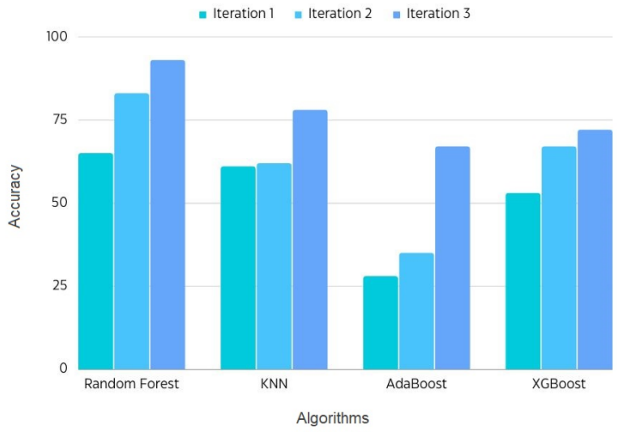

Figure 4. Algorithm performances and their accuracy.

As a part of the testing phase, we compared the results of various trained models. With the help of this tool, we can explore the medication suggestions made by different models. The improved accuracy of the random forest model can be attributed to its ability to handle complex interactions among features and effectively deal with noisy and correlated data. The ensemble nature of random forest, which combines multiple decision trees, enables robust recommendations and reduces the risk of overfitting.

In this work, we trained and tested a select set of models to recommend optimal combination medication for patients with DCCs. We then compared how each of these algorithms performed. Below, we discuss the two best performing classifiers (random forest and XGBoost). The dataset used for this study consisted of various features related to patients' prescriptions and concerns for multiple DCCs. The random forest classifier, known for its ensemble nature and use of decision trees, demonstrated superior performance in comparison to the XGBoost classifier. Random forests exhibited higher accuracy metrics than XGBoost when suggesting optimal medication recommendation. The random forest algorithm has an ability to handle complex interactions among features (concerns) and reduce overfitting. On the other hand, while the XGBoost classifier also exhibited competitive performance, it fell slightly short in terms of overall accuracy and robustness. Despite its gradient boosting technique and advanced regularization methods, the XGBoost model faced challenges in capturing certain intricate patterns present in the medication

recommendation dataset we created. In addition, our results show that the random forest algorithm demonstrated greater resilience, making it our favorable choice when dealing with heterogeneous and multidimensional medical data. Based on this specific medication recommendation task, the random forest classifier outperformed the XGBoost classifier in terms of accuracy. Its stronger performance was attributed to its ability to handle complex relationships, mitigate overfitting, and handle missing data more effectively.

5. Conclusions

In this paper, we address the problem of how patients with DCCs often experience multiple obstacles when prioritizing treatment plans and prescriptions. Treatment suggestions and medication interactions can be provided by a variety of machine learning (ML) or deep learning (DL) techniques by resolving the problem of conflicting drugs. We collected the data for the dataset from the survey responses of the patients. The dataset consisted of a single disease (type 2 diabetes) with three treatments, depending on the replies of the individuals. There are 74 samples in this dataset, with 6 independent factors (sleep, cost, weight, symptoms, addictiveness, and impacts on sexual performance) and 1 dependent variable (class) (that is the medication). Regarding the evaluation, we investigated different machine learning algorithms such as random forest, KNN, AdaBoost, and XGBoost in the above dataset. Following the benchmarking, random forest achieved the highest performance with an accuracy of 93%. Our exploration centered on formulating a machine learning framework tailored to recommend suitable medications for discordant chronic comorbidities (DCCs)—encompassing type 2 diabetes, arthritis, and depression. We took into account a multitude of factors influencing patient treatment, striving to address the intricacies and challenges tied to managing patients with multiple chronic ailments. The study underscores the potential of AI in aiding healthcare professionals and patients in making well-informed decisions about treatments. As we look ahead, there is room for enhancements and further research to refine the model's accuracy and extend its utility within analogous contexts.

Author Contributions: Conceptualization, T.O. and I.P.S.; methodology, T.O. and T.V.N.; formal analysis, T.O., I.P.S., S.A.S. and T.V.N.; resources, T.O., I.P.S. and T.V.N.; writing—original draft preparation, T.O., I.P.S. and T.V.N.; writing—review, T.O. and T.V.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data available in a publicly accessible repository that does not issue DOIs. This data can be found here: (https://github.com/sharmaichchhaa/medicine-prediction-Machine-Learning).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zheng, D.D.; Loewenstein, D.A.; Christ, S.L.; Feaster, D.J.; Lam, B.L.; McCollister, K.E.; Curiel-Cid, R.E.; Lee, D.J. Multimorbidity patterns and their relationship to mortality in the US older adult population. *PLoS ONE* **2021**, *16*, e0245053. [CrossRef] [PubMed]
2. Lu, H.; Uddin, S. A disease network-based recommender system framework for predictive risk modelling of chronic diseases and their comorbidities. *Appl. Intell.* **2022**, *52*, 10330–10340. [CrossRef]
3. Ongwere, T.; Cantor, G.; Martin, S.R.; Shih, P.C.; Clawson, J.; Connelly, K. Design hotspots for care of discordant chronic comorbidities: Patients' perspectives. In Proceedings of the 10th Nordic Conference on Human-Computer Interaction, Oslo, Norway, 29 September–3 October 2018; pp. 571–583.
4. Ongwere, T.; Cantor, G.S.; Clawson, J.; Shih, P.C.; Connelly, K. Design and care for discordant chronic comorbidities: A comparison of healthcare providers' perspectives. In Proceedings of the 14th EAI International Conference on Pervasive Computing Technologies for Healthcare, online, 6–8 October 2020; pp. 133–145.
5. Ongwere, T.; Stolterman, E.; Shih, P.C.; James, C.; Connelly, K. Translating a DC 3 Model into a Conceptual Tool (DCCs Ecosystem): A Case Study with a Design Team. In Proceedings of the International Conference on Pervasive Computing Technologies for Healthcare, Virtual, 6–8 December 2021; Springer International Publishing: Cham, Switzerland; pp. 381–397.

6. Ongwere, T.; Stolterman, E.; Shih, P.C.; James, C.; Connelly, K. Design for Discordant Chronic Comorbidities (DCCs): A DC^3 Model. In Proceedings of the International Conference on Pervasive Computing Technologies for Healthcare, Virtual Event, 6–8 December 2021; Springer International Publishing: Cham, Switzerland, 2021; pp. 346–362.
7. Kilic, A.; Dochtermann, D.; Padman, R.; Miller, J.K.; Dubrawski, A. Using machine learning to improve risk prediction in durable left ventricular assist devices. *PLoS ONE* **2021**, *16*, e0247866. [CrossRef]
8. Nuryani, N.; Ling, S.S.H.; Nguyen, H.T. Electrocardiographic signals and swarm-based support vector machine for hypoglycemia detection. *Ann. Biomed. Eng.* **2012**, *40*, 934–945. [CrossRef]
9. Mitchell, E.G.; Heitkemper, E.M.; Burgermaster, M.; Levine, M.E.; Miao, Y.; Hwang, M.L.; Desai, P.M.; Cassells, A.; Tobin, J.N.; Tabak, E.G.; et al. From reflection to action: Combining machine learning with expert knowledge for nutrition goal recommendations. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, Yokohama, Japan, 8–13 May 2021; pp. 1–17.
10. Kavakiotis, I.; Tsave, O.; Salifoglou, A.; Maglaveras, N.; Vlahavas, I.; Chouvarda, I. Machine learning and data mining methods in diabetes research. *Comput. Struct. Biotechnol. J.* **2017**, *15*, 104–116. [CrossRef] [PubMed]
11. Shah, A.M.; Muhammad, W.; Lee, K.; Naqvi, R.A. Examining different factors in web-based patients' decision-making process: Systematic review on digital platforms for clinical decision support system. *Int. J. Environ. Res. Public Health* **2021**, *18*, 11226. [CrossRef]
12. Lysaght, T.; Lim, H.Y.; Xafis, V.; Ngiam, K.Y. AI-assisted decision-making in healthcare. *Asian Bioeth. Rev.* **2019**, *11*, 299–314. [CrossRef]
13. Hu, Y.-J.; Ku, T.; Jan, R.; Wang, K.; Tseng, Y.; Yang, S. Decision tree-based learning to predict patient controlled analgesia consumption and readjustment. *BMC Med. Inform. Decis. Mak.* **2012**, *12*, 131. [CrossRef]
14. Rush, K.L.; Hatt, L.; Janke, R.; Burton, L.; Ferrier, M.; Tetrault, M. The efficacy of telehealth delivered educational approaches for patients with chronic diseases: A systematic review. *Patient Educ. Couns.* **2018**, *101*, 1310–1321. [CrossRef]
15. Milette, K.; Thombs, B.D.; Maiorino, K.; Nielson, W.R.; Körner, A.; Peláez, S. Challenges and strategies for coping with scleroderma: Implications for a scleroderma-specific self-management program. *Disabil. Rehabil.* **2019**, *41*, 2506–2515. [CrossRef]
16. Palmer, R.; Enderby, P.; Paterson, G. Using computers to enable self-management of aphasia therapy exercises for word finding: The patient and carer perspective. *Int. J. Lang. Commun. Disord.* **2013**, *48*, 508–521. [CrossRef] [PubMed]
17. Wang, Z.; Huang, H.; Cui, L.; Chen, J.; An, J.; Duan, H.; Ge, H.; Deng, N. Using natural language processing techniques to provide personalized educational materials for chronic disease patients in China: Development and assessment of a knowledge-based health recommender system. *JMIR Med. Inform.* **2020**, *8*, e17642. [CrossRef] [PubMed]
18. Woldaregay, A.Z.; Årsand, E.; Botsis, T.; Albers, D.; Mamykina, L.; Hartvigsen, G. Data-driven blood glucose pattern classification and anomalies detection: Machine-learning applications in type 1 diabetes. *J. Med. Internet Res.* **2019**, *21*, e11030. [CrossRef]
19. Bzdok, D.; Krzywinski, M.; Altman, N. Machine learning: Supervised methods. *Nat. Methods* **2018**, *15*, 5. [CrossRef] [PubMed]
20. Singh, S.; Gupta, A.; Baraheem, S.S.; Nguyen, T.V. Multi-Output Career Prediction: Dataset, Method, and Benchmark Suite. In Proceedings of the 2023 57th Annual Conference on Information Sciences and Systems (CISS), Baltimore, MD, USA, 22–24 March 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6.
21. Liu, K.; Dou, Y.; Zhao, Y.; Ding, X.; Hu, X.; Zhang, R.; Ding, K.; Chen, C.; Peng, H.; Shu, K.; et al. Bond: Benchmarking unsupervised outlier node detection on static attributed graphs. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 27021–27035.
22. Kumar, P.; Garg, S.; Garg, A. Assessment of anxiety, depression and stress using machine learning models. *Procedia Comput. Sci.* **2020**, *171*, 1989–1998. [CrossRef]
23. Seckler, H.; Szwabiński, J.; Metzler, R. Machine-Learning Solutions for the Analysis of Single-Particle Diffusion Trajectories. *J. Phys. Chem. Lett.* **2023**, *14*, 7910–7923. [CrossRef]
24. Zand, J. Multimodal Probabilistic Reasoning for Prediction and Coordination Problems in Machine Learning. Ph.D. Thesis, University of Oxford, Oxford, UK, 2021.
25. Challen, R.; Denny, J.; Pitt, M.; Gompels, L.; Edwards, T.; Tsaneva-Atanasova, K. Artificial intelligence, bias and clinical safety. *BMJ Qual. Saf.* **2019**, *28*, 231–237. [CrossRef]
26. Angelopoulos, A.N.; Bates, S. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv* **2021**, arXiv:2107.07511.
27. Hunt, N.C.; Scheetz, A.M. Using MTurk to distribute a survey or experiment: Methodological considerations. *J. Inf. Syst.* **2019**, *33*, 43–65. [CrossRef]
28. Ivanisevic, J.; Benton, H.P.; Rinehart, D.; Epstein, A.; Kurczy, M.E.; Boska, M.D.; Gendelman, H.E.; Siuzdak, G. An interactive cluster heat map to visualize and explore multidimensional metabolomic data. *Metabolomics* **2015**, *11*, 1029–1034. [CrossRef]
29. Ghosh, M.; Raihan, M.M.S.; Raihan, M.; Akter, L.; Bairagi, A.K.; Alshamrani, S.S.; Masud, M. A Comparative Analysis of Machine Learning Algorithms to Predict Liver Disease. *Intell. Autom. Soft Comput.* **2021**, *30*, 917–928. [CrossRef]
30. Prastyo, P.H.; Paramartha, I.G.Y.; Pakpahan, M.S.M.; Ardiyanto, I. Predicting breast cancer: A comparative analysis of machine learning algorithms. *Proceeding Int. Conf. Sci. Eng.* **2020**, *3*, 455–459. [CrossRef]
31. Parmar, A.; Katariya, R.; Patel, V. A review on random forest: An ensemble classifier. In Proceedings of the International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018, Coimbatore, India, 7–8 August 2018; Springer International Publishing: Cham, Switzerland, 2019; pp. 758–763.

32. Mathew, R.B.; Varghese, S.; Joy, S.E.; Alex, S.S. Chatbot for disease prediction and treatment recommendation using machine learning. In Proceedings of the 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 23–25 April 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 851–856.
33. Assegie, T.A.; Suresh, T.; Purushothaman, R.; Ganesan, S.; Kumar, N.K. Early Prediction of Gestational Diabetes with Parameter-Tuned K-Nearest Neighbor Classifier. *J. Robot. Control (JRC)* **2023**, *4*, 452–457.
34. Pettit, R.W.; Fullem, R.; Cheng, C.; Amos, C.I. Artificial intelligence, machine learning, and deep learning for clinical outcome prediction. *Emerg. Top. Life Sci.* **2021**, *5*, 729–745. [CrossRef] [PubMed]
35. Vo, T.H.; Nguyen, N.T.K.; Le, N.Q.K. Improved prediction of drug-drug interactions using ensemble deep neural networks. *Med. Drug Discov.* **2023**, *17*, 100149. [CrossRef]
36. Guo, C.; Zhang, J.; Liu, Y.; Xie, Y.; Han, Z.; Yu, J. Recursion enhanced random forest with an improved linear model (RERF-ILM) for heart disease detection on the internet of medical things platform. *IEEE Access* **2020**, *8*, 59247–59256. [CrossRef]
37. Siddalingappa, R.; Kanagaraj, S. K-nearest-neighbor algorithm to predict the survival time and classification of various stages of oral cancer: A machine learning approach. *F1000Research* **2022**, *11*, 70. [CrossRef]
38. Salem, H.; Shams, M.Y.; Elzeki, O.M.; Abd Elfattah, M.; Al-Amri, J.F.; Elnazer, S. Fine-tuning fuzzy KNN classifier based on uncertainty membership for the medical diagnosis of diabetes. *Appl. Sci.* **2022**, *12*, 950. [CrossRef]
39. Walker, K.W.; Jiang, Z. Application of adaptive boosting (AdaBoost) in demand-driven acquisition (DDA) prediction: A machine-learning approach. *J. Acad. Librariansh.* **2019**, *45*, 203–212. [CrossRef]
40. Hatwell, J.; Gaber, M.M.; Atif Azad, R.M. Ada-WHIPS: Explaining AdaBoost classification with applications in the health sciences. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 250. [CrossRef] [PubMed]
41. Nayak, D.R.; Dash, R.; Majhi, B. Brain MR image classification using two-dimensional discrete wavelet transform and AdaBoost with random forests. *Neurocomputing* **2016**, *177*, 188–197. [CrossRef]
42. Islam, M.S.; Hasan, M.M.; Wang, X.; Germack, H.D.; Noor-E-Alam, M. A systematic review on healthcare analytics: Application and theoretical perspective of data mining. *Healthcare* **2018**, *6*, 54. [CrossRef]
43. Jiang, D.; Wu, Z.; Hsieh, C.Y.; Chen, G.; Liao, B.; Wang, Z.; Shen, C.; Cao, D.; Wu, J.; Hou, T. Could graph neural networks learn better molecular representation for drug discovery? A comparison study of descriptor-based and graph-based models. *J. Cheminformatics* **2021**, *13*, 12. [CrossRef] [PubMed]
44. Liu, Q.; Zhang, M.; He, Y.; Zhang, L.; Zou, J.; Yan, Y.; Guo, Y. Predicting the risk of incident type 2 diabetes mellitus in chinese elderly using machine learning techniques. *J. Pers. Med.* **2022**, *12*, 905. [CrossRef]
45. Shtar, G.; Rokach, L.; Shapira, B. Detecting drug-drug interactions using artificial neural networks and classic graph similarity measures. *PLoS ONE* **2019**, *14*, e0219796. [CrossRef]
46. Kramer, O.; Kramer, O. Scikit-learn. In *Machine Learning for Evolution Strategies*; Springer: Cham, Switzerland, 2016; pp. 45–53.
47. Adeniyi, D.A.; Wei, Z.; Yongquan, Y. Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method. *Appl. Comput. Informatics* **2016**, *12*, 90–108. [CrossRef]
48. González, S.; García, S.; Ser, J.D.; Rokach, L.; Herrera, F. A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities. *Inf. Fusion* **2020**, *64*, 205–237. [CrossRef]