

ՀՀ ԳԱԱ ԻՆՖՈՐՄԱՏԻԿԱՅԻ և ԱԿՏՈՄԱՏԱՑՄԱՆ ՊՐՈԲԼԵՄՆԵՐԻ
ԻՆՍՏԻՏՈՒՏ

Գիշյան Կարեն Միերի

ԴԻՆԱՄԻԿ ՃԱՆԱԶՈՂՈՒԹՅԱՆ ԽՆԴԻՐՆԵՐԻ ՄՈԴԵԼԱՎՈՐՈՒՄ
ԳՐԱՖԱՅԻՆ ԱԼԳՈՐԻԹՄՆԵՐԻ ԵՎ ՄԵՔԵՆԱՅԱԿԱՆ ՈՒՍՈՒՑՄԱՆ
ՄԵԹՈԴՆԵՐԻ ԿԻՐԱՌՄԱՔ

Ե.13.05 «Մաթեմատիկական մոդելավորում, թվային մեթոդներ և
ծրագրերի համալիրներ» մասնագիտությամբ տեխնիկական
գիտությունների թեկնածուի գիտական աստիճանի հայցման
ատենախոսության սեղմագիր

Երևան 2024

INSTITUTE FOR INFORMATICS AND AUTOMATION PROBLEMS OF
NAS RA

Gishyan Karen

MODELING DYNAMIC RECOGNITION PROBLEMS USING GRAPH
ALGORITHMS AND MACHINE LEARNING METHODS

An abstract of dissertation for obtaining a Ph.D. degree in Technical
Sciences on specialty 05.13.05 "Mathematical modeling, numerical methods
and software complexes"

Yerevan 2024

Ատենախոսության թեման հաստատվել է ՀՀ ԳԱԱ Ինֆորմատիկայի և Ավտոմատացման Պրոբլեմների Ինստիտուտում

Գիտական ղեկավար՝ Ֆիզ. մաթ. գիտ. դոկտոր, Հ. Ա. Սահակյան


Պաշտոնական ընդդիմախոսներ՝ տեխ. գիտ. դոկտոր, Հ. Վ. Ասատրյան
Ֆիզ. մաթ. գիտ. թեկնածու, Ս. Խ. Դարբինյան

Առաջատար կազմակերպություն՝ Հայ-Ռուսական Համալսարան

Պաշտպանությունը կայանալու է 2024թ. հունիսի 7-ին ժամը 14:00-ին, ՀՀ ԳԱԱ Ինֆորմատիկայի և ավտոմատացման պրոբլեմների ինստիտուտում գործող 037 «Ինֆորմատիկա և հաշվողական համակարգեր» մասնագիտական խորհրդի նիստում հետևյալ հասցեով՝ Երևան, 0014, Պ. Սևակի 1:

Ատենախոսությանը կարելի է ծանոթանալ ՀՀ ԳԱԱ ԻԱՊԻ գրադարանում:

Սեղմագիրն առաքված է 2024թ.-ի մայիսի 6-ին:

Մասնագիտական խորհրդի գիտական քարտուղար՝  Ֆիզ. մաթ. գիտ. դոկտոր, Մ. Ե. Հարությունյան

The topic of the dissertation was approved at the Institute for Informatics and Automation Problems of NAS RA

Scientific Supervisor: Dr. Phys. Math. Sc., H. A. Sahakyan


Official Opponents: Dr. Tech. Sc., H. V. Astsatryan
Ph.D. Phys. Math. Sc., S. K. Darbinyan

Leading Organization: Russian-Armenian University

The Defense will take place on June 7, 2024, at 14:00, at the Specialized Council 037 "Informatics" at the Institute for Informatics and Automation Problems of NAS RA. Address: Yerevan, 0014, P. Sevak 1.

The Dissertation is available at the library of IIAP NAS RA.

The abstract was delivered on May 6, 2024.

Scientific Secretary of the Specialized Council  Dr. Phys. Math. Sc., M. E. Harutyunyan

Description of the Work

Topic Relevance and Problem Definition: Within the realm of imbalanced and recursive/iterative classification algorithms, there exists a practical need to create novel algorithms that can classify objects into predefined classes through successive transformations. To investigate a dynamic recognition problem using medical datasets, in the thesis, we define a problem called target class classification (TCC), which implies classifying/allocating the agents (objects) into the target-normal class through sequential actions. The work begins with graph theory and discrete optimization, examining models with deterministic transition outcomes, and then we obtain experimental results using sequential modeling procedures such as graph search and reinforcement learning with non-deterministic transitions which are later discussed in detail. We also investigate the binary target/non-target classification case as an extension to the main defined problem. Even though the thesis contains significant medical data processing and modeling components, we understand the challenges and responsibility associated with working with medical datasets, thus we try to put significantly more emphasis on the algorithmic implementations, which can allow using these approaches for other domains as well.

The defined problem of the thesis shares the most similarities with Hierarchical Reinforcement Learning (HRL) research area. HRL provides a divide-and-conquer approach for solving traditional RL problems by abstracting complex problems into smaller sub-problems. HRL tries to achieve compositionality using two main mechanisms: temporal abstractions and state abstractions. Temporal abstraction divides the problem into temporally extended actions (sub-behaviors) consisting of a sequence of primitive actions for solving a more complex problem. The second form of abstraction is the state abstraction. In high-dimensional spaces, it is extremely difficult to learn the best action in every possible state. For this reason, grouped representations of similar states in terms of transition dynamics and reward function are used for building learnable state abstractions¹.

In this thesis, we put compositionality and stage-based modeling on the basis of our experiments to come up with an algorithmic and software toolset to solve the defined TCC problem. We explore the problem using medical datasets not only from the RL perspective, but also from graph construction and optimal pathfinding, ensemble-based iterative classification, and combinatorial optimization, utilizing computer science and machine learning methodologies, where algorithms are proposed, and validation results are obtained. Problem formulation for target class allocation in a sequential procedure, data division in a stage-based format using tabular and graph structures, algorithmic implementations, and derived results may allow to:

- Use proposed algorithmic approaches to investigate solving target-normal class allocation/classification problems with a stage-based (multi-stage) approach, facilitate research advancements in the medical and other domains.

Currently, there is not much scientific literature that uses a multi-stage approach for finding optimal treatments, especially for graph-based representations, which can be due to:

- The approach is new and has not yet been widely experimented.
- The problem of finding optimal treatment strategies is challenging and can be approached from different aspects.

Overall, we provide a graph-theoretical analysis of the target class classification (TCC) problem and connect these findings to sequential searching and iterative classification with stages, provide a novel link between multilayer graphs and reinforcement learning for solving a TCC task for the medical and other similar domains.

Thesis Aim: The thesis aims can be summarized as follows.

- To come up with a mathematical description and analysis of a sequential target class classification/allocation problem and propose a mathematical model that can solve this task in terms of classification and target class transition logic.
- To explore the obtained mathematical model from the perspective of existing computer science and machine learning research directions, such as graph pathfinding, reinforcement learning, and iterative classification to enhance theoretical results with algorithmic evaluation and come up with a practical toolset for the TCC-type modeling and analysis.
- To propose algorithmic approaches regarding sequential (stage-based) target class classification/allocation.

Research Data and Methodologies: In this thesis, we used graph modeling and search procedures, neural networks, and reinforcement learning-based methods suitable for discrete and continuous learning tasks to come up with algorithms to solve optimization tasks in a sequential manner. In the scope of the thesis, we obtained credentialed access to the MIMIC-III clinical database², which contains health-related information such as demographics, laboratory test results, procedures, medications, and various other medical statistics about forty thousand patients allowing us to conduct numerous analytical studies. We preprocessed this database in a stage-based approach and obtained data in tabular and graph (network) formats so that we could

address the problems defined in the thesis. We also used eight publicly available medical datasets from the UCI Machine Learning repository³ which have smaller dataset sizes and are more suitable for traditional regression and classification problems. We solved procedural and recursive programming tasks in Python, worked with main libraries such as NetworkX, PyTorch, Django, OpenAI Gym, scikit-learn, SciPy, etc.

Thesis Scientific Contribution:

- Mathematical formulation, description, and modeling of a target class allocation task. The simple deterministic case of the problem is analysed and solved using graph-theoretical algorithms.
- An algorithm to solve the TCC task for the non-deterministic case using directed acyclic graphs (DAGs) evaluated on medical datasets. The algorithm constructs a DAG with multiple layers and then searches for the shortest path leading to the vertex corresponding to the target-normal class. Provision of a cost function and an admissible heuristic function for the path to be optimal. Clustering-based evaluation methodology for the searching results and a serverless approach based on backend and cloud techniques for further evaluation.
- An algorithm that trains a Deep Q Networks (DQN) and Value Iteration reinforcement learning algorithms to reach the vertex corresponding to the target-normal class on a multilayer graph representation. It builds on the non-deterministic TCC concept but considers a more complex connectivity logic. As a result, a multilayer graph formation based on medical datasets, connectivity and reinforcement learning state-space construction logic, and reinforcement learning algorithms capable of learning on such representation and performing a sequence of actions at different stages (layers) to reach the target normal class is proposed.
- Ensemble-based iterative neural network architecture comprised of three separate networks capable of classifying target/non-target classes (binary case) as an extension to the TCC problem is proposed, which performs a joint optimization of three networks with medical data.
- An algorithmic procedure aiming to improve gradient descent optimization by providing a cosine similarity-based approach to scale the learning rate during the gradient descent update process in a multiclass logistic regression. The approach is experimented on eight medical datasets.

Practical Importance: Proposed algorithmic approaches allow investigating and solving target-normal class allocation/classification problems with a stage-based (multi-stage) approach and facilitate research advancements in the medical and other domains.

Challenges and Limitations: There are limitations regarding data processing, which could have impacted the models' learning performances. We evaluated learning over three fixed stages and for some experiments having more data and stages could have helped to conduct additional testing and evaluations.

Terminology:

- The terms *vertex*, *state*, *node* are used interchangeably.
- The terms *edges*, *actions*, *treatments* are used interchangeably.
- The terms *object*, *agent*, *patient* are used interchangeably.
- The terms *stage*, *layer* are used interchangeably.

Publications List: There are 5 journal publications, 3 of which in Scopus Indexed journals (1 in Q2, 2 in Q3), and 1 conference proceedings publication.

2 Deterministic Recursion to Target Class Classification

In this section, a mathematical formulation and modeling of the problem of target class classification (TCC) policy from a graph theoretical perspective is given. The formulation includes defining the problem, and the dataset format that forms the state space, providing a TCC model structure in terms of deterministic transitions. Graph structures are obtained and analysed, which can be used as a basis for policy evaluation results for performing recursive classification/target class allocation. These methods and representations of this work directly serve the basis of works described in Sections 3, 4, and 5 to solve medical target class allocation tasks.

Subsection 2.1: The problem and the target class classification (TCC) framework are described, and relevant scientific literature on classification, reinforcement learning is mentioned. The novelty of the approach is first presented.

Subsections 2.2 and 2.3: A detailed description of the TCC framework, including rules about the environment, objects, and transitions is given. A brief description of objects, which are the entities on which the model should act to transfer from one class to the next is described.

Subsections 2.4 and 2.5: Mathematical definition of the features and the learning set is presented. Features are numerical vectors characterizing the objects.

Subsections 2.6, 2.7 and 2.8: Actions and transitions of objects over time are described. The concepts of track and trellis are defined, which is the progression of an object over the time domain after an action is applied to it

at a given state. The transitions of all objects across the time domain is the trellis. Problem tasks are distinguished.

Subsections 2.9, 2.10 and 2.11: The data format is described, a mathematical formulation of the problem of target class classification (TCC) from a graph theoretical perspective is given. The deterministic transition as a tree is presented in Figure 1. Theorems 2.1 and 2.2 are presented as main theoretical outcomes. Conclusion remarks are presented.

Theorem 2.1. If graph G of a TCC model is connected, it is a tree with root vertex v_0 and edges oriented from the terminal/leaf vertices and internal vertices of the tree towards the direction of the vertex v_0 .

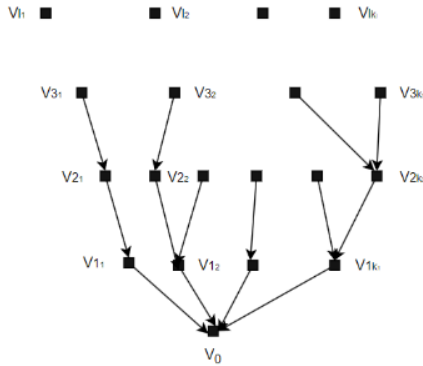


Figure 1: Connected TCC tree structure in case of simple deterministic transitions.

Theorem 2.2. If graph G of a TCC model is disconnected, it consists of one oriented tree, rooted to v_0 , and several other connected components structured as one-cycle oriented cactus graphs. Cactus cycle is oriented, and tree-like structures inside each component have an orientation to the cycle.

Thus, the findings show that if the graph is connected then it is a rooted tree, and the policy is correct/implementable because all the states by transitions will be allocated to the target class. But when the graph is disconnected, then additional components are one-cycle cactus graphs, and states from these components are not transferable by actions to the target class. This means that actions or transitions must be modified in a way to break the cycles and to make transitions from additional components to the basic tree component where the target state belongs.

3 Solving a TCC Problem using Graph-Search Algorithms

This section investigates the non-deterministic case (multiple actions originating from a single vertex) of the TCC problem. Vertices, which denote individual agents having features, are connected to each other based on feature similarity, and the stronger the similarity, the stronger the weight of the connection between two vertices. Features are observed from actual datasets. Such methodology is possible with a DAG data structure, and such structure is constructed layer by layer based on the concept of stage. Each layer consists of vertices (with features) corresponding to the given stage from the actual medical dataset. Unlike section 2, where the path is deterministic, there is a need to search and find the optimal path leading to the target state. For this reason, a heuristic function is also proposed, showing its admissibility to be able to find the shortest path using the A* searching algorithm. In summary, an algorithm is provided that can construct a DAG comprised of multiple (three in the experiment with medical datasets) layers and search for the vertex denoting the target normal class. A second algorithm is also provided, which can construct and search for a target solution using trees and cyclic graphs, but here the construction does not directly use the concept of layers. Evaluation approaches are also presented. The experiment is described using a medical context.

Subsections 3.1 and 3.2: The approach for solving the non-deterministic TCC problem with multiple stages using a graph-search methodology on medical datasets is described. Contributions of the work are presented, and a literature review is conducted.

Subsection 3.3: The data processing is described. Two algorithms are provided, which perform target class allocation on the multi-stage medical datasets by constructing a DAG (Figure 2), a tree, and a cyclic graph. Cost function 1 and heuristic function 2 are provided for performing the search. Root Mean Squared Error (*RMSE*) and Kolmogorov-Smirnov test (*kstest*) methods are chosen to describe the similarity between the features of the given two vertices. The names of the methods match the respective function names from scikit-learn, SciPy libraries.

Definitions 3.1 and 3.2 describe the problem and proposition 3.3 shows that heuristic function 2 is admissible and the path is optimal, which is proved by induction.

Definition 3.1. Suppose G is a DAG representing the multi-stage drug treatment state space. $start$ is the initial state, $T(s_i, a)$ is the transition model that returns state s'_{i+1} belonging to stage $i + 1$ after taking action a at state s

from stage i . d denotes the maximum depth which is equal to the number of treatment stages, and $target$ denotes the end state located at depth d .

Definition 3.2. Let $C(n_{d-k}, target)$ denote the cost of the shortest path from node n at depth $d - k$ to the target, and $h(n_{d-k}, target)$ denote the heuristic cost of n to the target. $C(n_{d-k}, target)$ is given by Equation 1, and $h(n_{d-k}, target)$ is given by Equation 2, where F_n and F_{target} are the feature vectors of node n and the node target. The range of $C(n_{d-k}, target)$ is $R_{C(n_{d-k}, target)}: k \cdot [1, 1024]$, and the range of $h(n_{d-k}, target)$ is $R_{h(n_{d-k}, target)}: k \cdot [0, 1]$.

$$C(n_{d-k}, target) = k \cdot (1 + RMSE(F_n, F_{target})) \quad 1$$

$$h(n_{d-k}, target) = k \cdot (1 - kstest(F_n, F_{target})) \quad 2$$

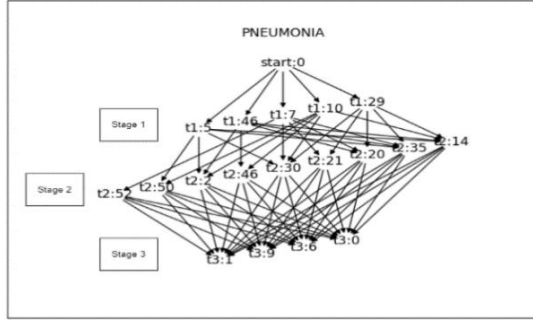


Figure 2: DAG example 1.

Proposition 3.3. Given a DAG G , $h(n_{d-k}, target) \leq C(n_{d-k}, target)$ for any node n k steps away from the maximum depth d .

Subsections 3.4, 3.5, and 3.6: The results are presented, clustering, betweenness, and cosine similarity-based evaluations, and a serverless architecture are provided for testing the results of the algorithms on the datasets used in this work, showing that the algorithms can solve a non-deterministic TCC problem on a multi-stage graph representation. Concluding remarks are provided.

4 Solving a TCC Problem using Multilayer Graphs and Reinforcement Learning

This section examines a non-deterministic case of the TCC problem with the use of RL techniques. In addition, we impose an additional complexity element and assume that transitions between the vertices in the same graph layer are also possible. We use the concept of multilayer graphs (Figure 3), where a graph consists of layers, each layer being a graph (cyclic in most cases) itself. A vertex can be connected to vertices both in the same layer and in the successive layer, making the task of finding the correct transition more difficult. The goal is the same as in the previous two sections: solve a TCC problem with the correct transitions leading to a vertex corresponding to a target-normal class located at the last layer of the multilayer graph. The experiment is conducted using medical datasets and, in this context, the vertex denotes the patient features who have been discharged to *Home* location (target class). However, to reach a target class in the last layer, the algorithm should first learn to reach the normal class in each of the previous layers, which we can be regarded as intermediate normal classes. We use model-based Value Iteration and model-free DQN reinforcement learning algorithms to teach the agent to reach the normal class by layer-by-layer navigation. We provide a reinforcement learning setting (state space) using multilayer graphs constructed as a Gym environment and provide special transition and reward logic so that the agent can reach the target class without being stuck in infinite cycles. Comprehensive evaluations obtained from analysing the number of times the target-normal class is reached, steps taken to reach the solution, and collected reward show that we have obtained an environment and learning state space that can efficiently guide the agent (patient) to a target normal class by navigating through sequential stages. The experiment is described using a medical context.

Subsections 4.1 and 4.2: Description of the problem and multilayer state-space representation is provided. Contributions are listed. A comprehensive literature review of reinforcement learning, its applications in healthcare, and multilayer graphs is conducted.

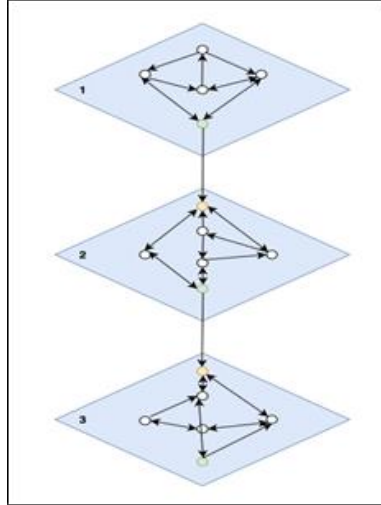


Figure 3: Multilayer graph for performing reinforcement learning.

Subsection 4.3: Data processing to obtain a multilayer graph (Figure 3) over three stages is described and a formal definition is given. The logic of interlayer and intralayer connectivity of the multilayer graph, including the reward formulation, is presented. Model-based learning in the form of the Value Iteration algorithm and model-free learning in the form of the DQN algorithm are detailed. The DQN implementation (Algorithm 1, labeled as Algorithm 3 in the dissertation) denotes the main experiment. Existing algorithms are modified to perform learning on the multilayer graph for solving medical treatment optimization tasks. The RL agents learn to navigate through sequential stages, each stage corresponding to a layer in the multilayer graph to reach to a target class (solution) in the last layer. The last layer corresponds to the last stage in the medical recovery process. Graph statistics for each layer in the multilayer graph are presented.

Subsection 4.4: The algorithmic results through evaluation are described, demonstrating good learning performance for both model-based and model-free scenarios. The results of the small experiment regarding DQN-based binary classification are mentioned.

Subsections 4.5 and 4.6: The main limitation of the work is mentioned and concluding remarks are presented.

Algorithm 1

Algorithm 3: DQN Learning on a multilayer graph

Obtain a dictionary p containing a combination of network parameters from a function which iterates over parameter combinations;

```
for patient_id  $\leftarrow$  1 to  $N_{patients}$  do
  for  $t \leftarrow$  1 to  $N_{stages}$  do
    From  $p$  select the optimizer, learning rate, max timestep, update rate, epsilon greedy;
    Obtain the graph corresponding to timestep  $t$ ;
    Initialize the agent, Q network, target Q network, and empty replay buffer;
    Initialize episode_steps, episode_terminals, episode_rewards empty lists;
    for episode  $\leftarrow$  1 to  $N_{episodes}$  do
      Make reward_sum=0;
      Reset current state to the start state;
      for step  $\leftarrow$  1 to  $N_{max\_timesteps}$  do
        Sample an action from the environment using  $\epsilon$ -greedy method;
        Perform the action on current state, observe next state, collected reward, terminal;
        Add reward to reward_sum;
        Add a list of state, next state, action, reward, terminal to the replay buffer;
        Make next state as current state;
        if terminal is True then
          break
        end
      end
      Append step, terminal, reward_sum to episode_steps, episode_terminals,
      episode_rewards lists respectively;
      Update the Q network using gradient descent, then use Q network parameters for
      updating the target Q network;
    end
    Output a solution using terminal from the last episode;
    Append reward_sum, episode_steps, episode_terminals, episode_rewards to  $p$ ;
  end
end
Obtain a final solution per patient, using solutions from each  $t$  then append to  $p$ ;
end
Evaluate results;
```

5 Solving a Binary Classification Task with Ensemble Iterative Neural Networks

In this section, we shift from graph-based theoretical and experiment evaluations to tabular-based iterative (sequential) classification, where we aim to achieve class separation between target (*Home* class) and non-target classes, unlike the previous findings where we tried to find the optimal path to the single target-normal class.

Learning is again based on stage-by-stage progression using the medical MIMIC-III clinical database. We preprocess the data based on sequential logic and obtain three stages, where each stage includes clinical observations of patients from their recovery process. Although the learning space is not graph-based, the data processing follows the data, feature, and component descriptions presented in Section 2. We obtain an ensemble-based iterative neural network architecture, which uses features and medications at each stage to correctly identify the medications for each successive treatment stage

and in the end perform binary classification to predict whether the patient (agent) will be discharged to *Home* (target-normal class) or not. We perform a joint optimization of the parameters of three networks using an MSE loss function. Synthetic data is generated based on the preprocessed medical data to increase the dataset size, and the results show that based on the initial features and medications, it has been possible to identify actions (treatments) and classify target/non-target classes with high accuracy treatments for each learning stage.

Subsections 5.1, 5.2 and 5.3: The iterative classification problem, proposed feedforward neural network-based iterative ensemble model, and contributions are listed. A literature review is conducted, and data processing is presented in detail.

Subsection 5.4: The experimental approach and the iterative ensemble-based neural network architecture (Figure 4) are described. The approach uses a treatment prediction network (*Network 1*) and a feature prediction network (*Network 2*) during a given stage to correctly identify the treatments for each successive stage, and in the end perform binary classification (*Network 3* for output) to predict whether the agent will be discharged to Home (positive label) or not. In Equation 3, $\theta_1, \theta_2, \theta_3$ are the parameters of individual MSE loss functions L_1, L_2 and L_3 (treatment prediction loss for two stages and output prediction loss) and n is the number of training samples. A joint optimization of these networks is performed for each epoch.

$$L = \frac{1}{n} (L_1(\theta_1) + L_2(\theta_2) + L_3(\theta_3)) \quad 3$$

Subsection 5.5: This subsection lists the model's training and evaluation results through visualization and F1 score. Good learning performance is observed, but there is also room for improvement.

Subsections 5.6 and 5.7: Limitations and concluding remarks are provided.

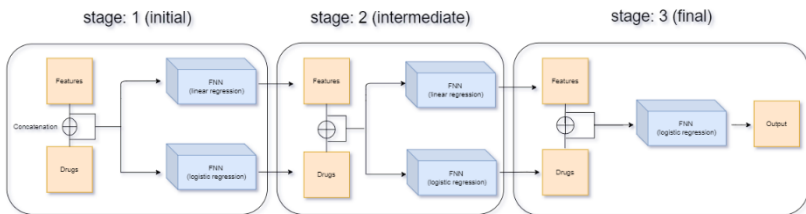


Figure 4: Multi-stage classification architecture.

6 Solving a Binary Classification Task using Cartesian Product Action-Combinations and Neural Networks

This section is dedicated to supplemental findings that, while not directly central to the TCC problem, provide additional context, experiments, and understanding. Similar to the previous section, in this section we use stage-based medical tabular data to correctly classify target/non-target classes. Here we try to optimize the medical treatments at the stage, analogous to the task of finding the best action at each stage. For each agent (patient), we try to identify potential best actions (treatment strategies) at each stage using distance *MSE*, *MAE*, clustering *KMeans*, *Mini Batch KMeans*, and distribution-based Kolmogorov-Smirnov methods, obtain a Cartesian product set and vector concatenations of all the triplets in the set. The names of the methods match the respective function names from scikit-learn, SciPy libraries. The result is a dataset denoting different possible actions, then we perform binary classification using a feedforward neural networks-based (FNN) model to check the efficiency of the approach as an extension to the target class classification task, which implies classifying whether the agent will be discharged to *Home* or not. The experiment is described using a medical context.

Subsections 6.1 and 6.2: The problem is described, and the contributions are listed. A literature review is conducted.

Subsection 6.3: This subsection first describes the data and data preprocessing methodology. For each agent, potential best treatments for each stage are identified using distance *MSE*, *MAE*, clustering *KMeans*, *Mini Batch KMeans*, and distribution-based *Kolmogorov-Smirnov* methods (Equation 4), obtain vector concatenations of Cartesian product treatment combinations (Equation 5) over three stages as a treatment dataset and test them on a classification task.

$$X_i = Top_5\{f(x, r): x \in D_i\} \quad 4$$

In Equation 4, D_i denotes a dataset from one of the three stages, r is a given row, f is a given function (method) denoting similarity, and Top_5 is an operator that finds the top five arguments maximizing the similarity function f . X_i is the set of five instances.

$$C = (x_{1i}, x_{2j}, x_{3k}): x_{1i} \in X_1, x_{2j} \in X_2, x_{3k} \in X_3 \quad 5$$

In Equation 5, (x_{1i}, x_{2j}, x_{3k}) is the set of all tuples such that x_{1i} is a vector in X_1 , x_{2j} is a vector in X_2 , and x_{3k} is a vector in X_3 . C denotes the Cartesian product.

Subsections 6.4 and 6.5: The results show that clustering and especially distribution-based similarity metrics can result in the identification of optimal actions and dataset formation leading to a classification of the target/non-target classes with high accuracy. The training performance from a five-fold cross-validation of the distribution-based method is shown in Figure 5.

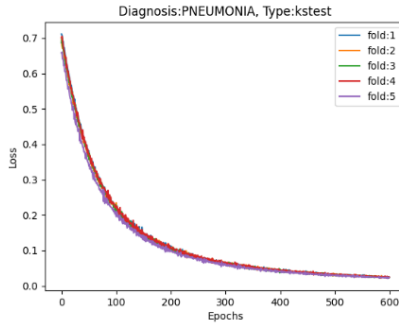


Figure 5: Kolmogorov-Smirnov test (kstest) results.

Subsections 6.6 and 6.7: Limitations and summary results are provided.

7 Stochastic Gradient Descent Update Rule using Cosine Similarity

In Sections 4, 5, and 6 it was observed that neural networks struggle to learn well when stochastic gradient descent (SGD) is chosen as the optimization algorithm. This work tries to address this problem by providing cosine similarity-based scaling of the learning rate during gradient descent update in the multiclass logistic regression algorithm. Here we evaluate the algorithm not on the preprocessed stage-based datasets as it was done in Sections 3, 4, 5, and 6, but instead on benchmark medical datasets suited for classification tasks from the UCI Machine Learning repository. The results in terms of the F1 score show that this approach can potentially improve the classification performance.

Subsections 7.1, 7.2 and 7.3: The problem, methodology, and contributions are listed. A concise literature review on the gradient descent-based optimization methods is provided and the datasets used for the experiments obtained from the UCI machine learning repository are mentioned.

Subsection 7.4: Cosine similarity between each training record and the average vector of the training batch is calculated and then the learning rate is scaled which updates the gradient vector (Equation 6). The rationale is that the smaller the directional similarity between an instance and the batch

average, the smaller the update, and vice versa. This is done to prevent dissimilar records from having a big impact on the gradient update. This approach is evaluated for standard SGD (Algorithm 2, labeled as Algorithm 5 in the dissertation), SGD with Momentum, and SGD with Nesterov acceleration for different epochs.

$$\text{learning_rate} = \text{cosine_similarity}(\bar{\mathbf{X}}, \mathbf{x}) \times \text{learning_rate} \quad 6$$

Algorithm 2

Algorithm 5: SGD scaling with Cosine Similarity

Data: X, y, learning_rate, N_epochs, scale
Initialization;
 $\bar{\mathbf{X}}$: vector of size $1 \times n$, where n is the number of columns in X;
model.classes: distinct classes in y;
model.loss: empty list;
model.weights: zero matrix of size $m \times n$, where m is the number of model classes, n is the number of columns in X;
model.bias: zero matrix of size $1 \times n$, where n is the number of model classes;
model.predict(): performs linear regression followed by softmax activation;
model.cross_entropy(): calculates cross entropy loss;
model.get_gradients(): calculates gradients of weights and bias;

```

for n ← 1 to N_epochs do
  for i ← 0 to N_data do
    X_batch=X[i,:];
    y_batch=y[i,:];
    y_pred=model.predict(X_batch);
    loss=model.cross_entropy(y_batch,y_pred);
    model.loss.append(loss);
    d_weight,d_bias=model.get_gradients(y_batch,y_pred,X_batch);
    *if scale=True then
      cosine_similarity=cosine_similarity( $\bar{\mathbf{X}}$ , X_batch) ;
      if cosine_similarity ≤ 0 then
        | continue
      end
      learning_rate=cosine_similarity × learning_rate
    end
    model.weights-=learning_rate × d_weight; model.bias-=learning_rate × d_bias;
  end
end

```

* sign in the *if statement* of Algorithm 2 denotes the scaling, the rest is the standard implementation of the multiclass logistic regression algorithm. \mathbf{x} in Equation 6 is equivalent to X_batch in Algorithm 2.

Subsections 7.5 and 7.6: The results in terms of the F1 score show that this approach can improve the models' classification performance. Concluding remarks are presented.

8 Algorithmic Complexity Analyses

Complexity analyses of the algorithms are provided. All the algorithms have higher-than-quadratic complexity.

Summary: Figure 6 presents a high-level visual summary of the dissertation. Section 7 experiment, which supplements other findings but is not based on a multi-stage approach is not listed in the figure.

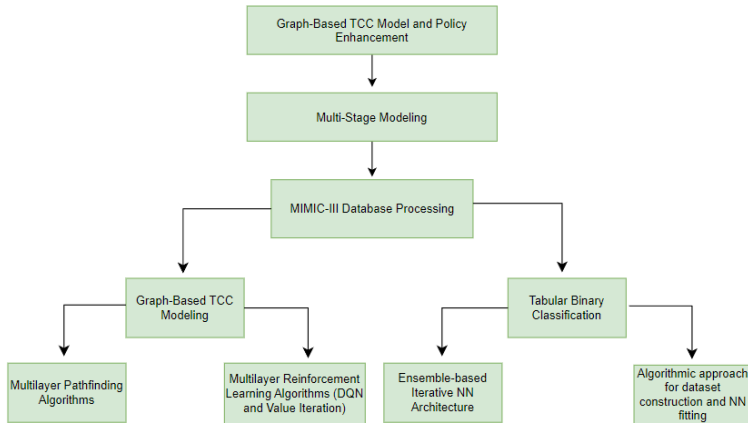


Figure 6: Visual summary.

Conferences and Publications:

- L. Aslanyan, K. Gishyan, and H. Sahakyan, “TARGET CLASS CLASSIFICATION RECURSION PRELIMINARIES,” in *13th conference for data analysis methods for software systems*, 2022, pp. 6–7. Available: https://www.mii.lt/damss/files/damss_2022.pdf
- K. Gishyan, “Drug-treatment generation combinatorial algorithm based on machine learning and statistical methodologies,” *Open Journal of Applied Sciences*, vol. 13, no. 4, pp. 548–561, 2023. <https://doi.org/10.4236/ojapps.2023.134044>
- K. Gishyan, H. Sahakyan, and L. Aslanyan, “Time-stage driven pathfinding framework for optimized medical treatments,” *Cogent Engineering*, vol. 10, no. 1, p. 2249258, 2023, doi: [10.1080/23311916.2023.2249258](https://doi.org/10.1080/23311916.2023.2249258).
- K. M. Gishyan, “Multi-stage classification scheme to optimize medical treatments,” *Mathematical Problems of Computer Science*, vol. 60, pp. 40–51, 2023. <https://doi.org/10.51408/1963-0108>
- L. Aslanyan, K. Gishyan, and H. Sahakyan, “Target class classification recursion preliminaries,” *Balt. J. Mod. Comput.*, vol. 11, no. 3, 2023, doi: [10.22364/BJMC.2023.11.3.03](https://doi.org/10.22364/BJMC.2023.11.3.03).
- L. Aslanyan, K. Gishyan, and H. Sahakyan, “Deterministic recursion to target class classification,” *Pattern Recognition and Image Analysis*, vol. 33, no. 3, pp. 584–598, 2023. <https://doi.org/10.1134/S1054661823030033>

Under Journal Review

K. Gishyan. Combining Reinforcement Learning and Multilayer Graphs for Generating Optimal Lines of Therapy. [Manuscript submitted for publication], n.d.

Դինամիկ ճանաչողության խնդիրների մոդելավորում գրաֆային ալգորիթմների և մեքենայական ուսուցման մեթոդների կիրառմամբ

Ամփոփագիր

Ռեկուրսիվ/իտերատիվ դասակարգման ալգորիթմների տիրույթում առկա է նոր ալգորիթմներ ստեղծելու գործնական անհրաժեշտություն, որոնք կարող են առարկաները դասակարգել նախապես սահմանված դասերի՝ հաջորդական վերափոխումների միջոցով: Բժշկական տվյալների կիրառմամբ դինամիկ ճանաչողության խնդիրը հետազոտելու նպատակով ատենախոսության մեջ սահմանվում է թիրախային դասի դասակարգման (TCC) խնդիր, ինչը ենթադրում է առարկաների դասակարգում/հատկացում թիրախային-նորմալ դասին՝ հաջորդական գործողությունների միջոցով: Աշխատանքը սկսվում է գրաֆների տեսությունից և դիսկրետ օպտիմալացումից, այնուհետև ստացվում են փորձարարական արդյունքներ՝ օգտագործելով հաջորդական (փուլային) մոդելավորման ընթացակարգեր, ինչպիսիք են փնտրումը գրաֆում և ամրապնդման ուսուցումը: Չնայած ատենախոսությունը պարունակում է բժշկական տվյալների մշակման և մոդելավորման կարևոր բաղադրիչներ, էական ուշադրությունը դարձվում է ալգորիթմական իրագործումների վրա, որոնք թույլ կտան մոտեցումները կիրառել նաև այլ տիրույթների համար: Սահմանված խնդիրն ունի նմանություն հիերարխիկ ամրապնդման ուսուցման հետ (HRL), ինչն առաջարկում է բաժանիր և տիրիր մոտեցում՝ բարդ խնդիրները վերածելով ավելի փոքր ենթախնդիրների: HRL-ը փորձում է հասնել բաղադրելիության օգտագործելով երկու հիմնական մեխանիզմներ՝ ժամանակային և վիճակային աբստրակցիաներ: Ատենախոսության հիմքում դրվում է բաղադրելիությունը և փուլային եղանակով կատարված մոդելավորումը՝ ստանալու TCC խնդրի լուծման ալգորիթմական և ծրագրային գործիքակազմ: Օգտագործելով բժշկական տվյալներ, խնդիրը ուսումնասիրվում է ոչ միայն ամրապնդման ուսուցման, այլ նաև գրաֆների կառուցման և գրաֆում օպտիմալ ճանապարհի փնտրման, մոդելների միավորման վրա հիմնված իտերատիվ դասակարգման, կոմբինատոր օպտիմալացման տեսանկյուններից՝

օգտագործելով համակարգչային գիտության և մեքենայական ուսուցման մեթոդներ, որտեղ առաջարկվում են ալգորիթմներ՝ տալով գնահատումներ և ստանալով կիրառական արդյունքներ:

Ատենախոսության նպատակը

- Ստանալ հաջորդական եղանակով թիրախային դասի դասակարգման (TCC) խնդրի մաթեմատիկական նկարագիրը և վերլուծությունը և առաջարկել մաթեմատիկական մոդել, որը կարող է խնդիրը լուծել դասակարգման և թիրախային դասի անցման տրամաբանության տերմիններով:
- Ստացված մաթեմատիկական մոդելը ուսումնասիրել համակարգչային գիտության և մեքենայական ուսուցման առկա հետազոտական ուղղությունների միջոցով, ինչպիսիք են գրաֆում օպտիմալ ճանապարհի փնտրումը, ամրապնդման ուսուցումը և իտերատիվ դասակարգումը՝ տեսական արդյունքները լրացնելով ալգորիթմական գնահատումներով և առաջարկել TCC-մոդելավորման և վերլուծության գործիքակազմ:
- Առաջարկել հաջորդական (փուլային) թիրախային դասի դասակարգման/հատկացման ալգորիթմական մոտեցումներ:

Ատենախոսության գիտական նորույթը

- Թիրախային դասի դասակարգման/հատկացման խնդրի մաթեմատիկական ձևակերպում, նկարագիր և մոդելավորում: Պարզ դետերմինիստիկ դեպքի վերլուծություն և լուծում՝ գրաֆ-տեսական մեթոդներով:
- Թիրախային դասի դասակարգման խնդրի ոչ-դետերմինիստիկ դեպքի լուծման ալգորիթմ՝ օգտագործելով կողմնորոշված ացիկլիկ գրաֆներ, գնահատված բժշկական տվյալների վրա: Ալգորիթմը կառուցում է մի քանի շերտից բաղկացած կողմնորոշված ացիկլիկ գրաֆ, այնուհետև փնտրում է թիրախ-նորմալ դասին համապատասխանող գագաթ տանող կարճագույն ճանապարհը: Սահմանվում է արժեքի ֆունկցիա և ընդունելի էվրիստիկ ֆունկցիա ճանապարհի օպտիմալություն ստանալու համար: Տրվում է փնտրման արդյունքների կլաստերային գնահատման մեթոդ և serverless մոտեցում հիմնված backend և ամպային հաշվման տեխնոլոգիաների վրա՝ հավելյալ գնահատումների համար:
- Ալգորիթմ, որը ուսուցանում է Deep Q Networks (DQN) և Value Iteration ամրապնդման ուսուցման ալգորիթմները՝ բազմաշերտ գրաֆներում հասնելու թիրախ-նորմալ դասին համապատասխանող գագաթին՝ թիրախային դասի դասակարգման խնդրի ոչ-

դետերմինիստիկ դեպք լուծելու նպատակով: Արդյունքում, կառուցվում է բժշկական տվյալների վրա հիմնված բազմաշերտ գրաֆ, տրվում է կապակցվածության և ամրապնդման ուսուցման միջավայրի կառուցման տրամաբանությունը, ինչպես նաև, ալգորիթմների ուսուցանում բազմաշերտ գրաֆներում, ինչը թույլատրում է հասնել թիրախ-նորմալ դասին ամեն շերտում (փուլում) կատարելով հաջորդական գործողություններ:

- Մոդելների միավորման (model ensemble) վրա հիմնված խտրատիվ նեյրոնային ցանցի ճարտարապետություն՝ կազմված երեք առանձին նեյրոնային ցանցերից, որոնք կարողանում են լուծել թիրախային/ոչ թիրախային դասերի դասակարգման խնդիր (երկուական դեպք), որպես TCC խնդրի ընդլայնում՝ կատարելով երեք ցանցերի միացյալ օպտիմալացում օգտագործելով բժշկական տվյալներ:
- Ալգորիթմական ընթացակարգ, որը կոսինուսի նմանության վրա հիմնված մոտեցման միջոցով փորձում է բարելավել գրադիենտի անկման ուսուցանումը բազմադաս լոգիստիկ ռեգրեսիայում; այն փորձարկվել է ութ տարբեր բժշկական տվյալների հավաքածուների վրա:

Моделирование задач динамического распознавания с использованием графовых алгоритмов и методов машинного обучения

Резюме

В сфере рекурсивно-итерационных алгоритмов классификации возникает практическая необходимость в разработке новых методов, способных последовательно классифицировать объекты к заданным классам. В рамках данной работы исследуется динамическое распознавание объектов с интерпретацией на медицинских наборах данных, сосредотачиваясь на проблеме классификации целевых классов (ТСС). Это предполагает классификацию/распределение агентов (объектов) в класс "целевой-нормальный" через последовательные шаги. Работа начинается с теории графов и дискретной оптимизации, затем мы получаем экспериментальные результаты с помощью процедур последовательного моделирования, таких как поиск графов и обучение с подкреплением. Несмотря на то, что диссертация содержит значительные компоненты обработки и моделирования медицинских данных, мы уделяем значительно больше внимания алгоритмическим реализациям, что может позволить использовать эти подходы и для других областей.

Данная задача имеет наибольшее сходство с областью исследований иерархического обучения с подкреплением (HRL), которая обеспечивает подход «разделяй и властвуй», абстрагируя сложные проблемы на более мелкие подзадачи. HRL пытается достичь композиционности, используя два основных механизма: временные абстракции и абстракции состояний. В диссертации мы ставим композиционность и стадийное моделирование на основе наших экспериментов, чтобы разработать алгоритмический и программный инструментарий для решения поставленной задачи ТСС. Используя медицинские наборы данных, проблема исследуется не только с точки зрения обучения с подкреплением, но и с точки зрения построения графов и поиска оптимального пути в графе, ансамблевой итерационной классификации и комбинаторной оптимизации, используя методологии информатики и машинного обучения, где предлагаются алгоритмы, дающие валидации и получающие прикладные результаты.

Цель диссертации

- Сформулировать математическое описание и анализ последовательной классификации/распределения целевых классов и предложить математическую модель, способную решить эту задачу с точки зрения классификации и логики перехода в целевой класс.
- Исследовать полученную математическую модель с существующими направлениями исследований в области информатики и машинного обучения, такими как поиск путей в графах, обучение с подкреплением и итеративная классификация, для дополнения теоретических результатов алгоритмической оценкой и создания практического инструментария для моделирования и анализа ТСС.
- Предложить алгоритмические подходы к последовательной (поэтапной) классификации/распределению целевых классов.

Научный вклад в диссертацию

- Математическое формулирование, описание и моделирование задачи распределения целевых классов. Простейший детерминированный случай задачи анализируется и решается с использованием теоретико-графовых алгоритмических подходов.
- Алгоритм решения задачи ТСС для недетерминированного случая с использованием направленных ациклических графов (DAG), оцениваемых на медицинских наборах данных. Алгоритм формирует группу доступа к базам данных с несколькими слоями,

а затем ищет кратчайший путь к вершине, соответствующей целевому нормальному классу. Обеспечивается функция стоимости и допустимая эвристическая функция для оптимального пути. Методика оценки результатов поиска на основе кластеризации и бессерверный подход, основанный на бэкэнд и облачных методах для дальнейшей оценки.

- Алгоритм, обучающий алгоритмы обучения с подкреплением Deep Q Networks (DQN) и Value Iteration для достижения вершины, соответствующей целевому нормальному классу в многослойном графовом представлении для решения недетерминированного случая задачи классификации целевого класса. В результате создается многослойный граф на примере медицинских наборов данных, формирование пространства состояний с использованием связности и обучения с подкреплением, а также алгоритмов обучения с подкреплением, способных обучаться на таком представлении и выполнять последовательность действий на разных этапах (слоях) для достижения целевого нормального класса.
- Архитектура итерационной нейронной сети на основе ансамбля, состоящая из трех отдельных сетей, способных классифицировать целевые/нецелевые классы (бинарный случай) в качестве расширения задачи ТСС, которая выполняет совместную оптимизацию трех сетей с медицинскими данными.
- Алгоритмическая процедура, направленная на улучшение оптимизации градиентного спуска путем предоставления подхода, основанного на косинусном подобию, для масштабирования скорости обучения в процессе обновления градиентного спуска в многоклассовой логистической регрессии. Подход протестирован на восьми медицинских наборах данных.

