UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Machine Learning techniques applied to identifying clinical factors regarding automated medical prognosis

Pedro Henrique Ferracini de Barros

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

ICMC USP
SÃO CARLOS

Pedro Henrique Ferracini de Barros

Machine Learning techniques applied to identifying clinical factors regarding automated medical prognosis

Master dissertation submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP, in partial fulfillment of the requirements for the degree of the Master Program in Computer Science and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. José Fernando Rodrigues Júnior

USP – São Carlos
December 2022

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

d278m

de Barros, Pedro Henrique Ferracini
Machine Learning techniques applied to identifying clinical factors regarding automated medical prognosis / Pedro Henrique Ferracini de Barros; orientador José Fernando Rodrigues Júnior. -- São Carlos, 2022.
104 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2022.

1. Aprendizado de Máquina. 2. Redes Neurais. 3. Predição de trajetórias clínicas. 4. Explicabilidade de predições clínicas. 5. Prontuários Médicos Eletrônicos. I. Rodrigues Júnior, José Fernando, orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

Pedro Henrique Ferracini de Barros

Técnicas de Aprendizado de Máquina aplicadas à identificação de fatores clínicos ligados ao prognóstico médico automatizado

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. José Fernando Rodrigues Júnior

USP – São Carlos
Dezembro de 2022

ACKNOWLEDGEMENTS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

RESUMO

DE BARROS, P. H. F. **Técnicas de Aprendizado de Máquina aplicadas à identificação de fatores clínicos ligados ao prognóstico médico automatizado**. 2022. 108 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

Algoritmos de Aprendizado de Máquina têm apresentado resultados promissores em diversas áreas do conhecimento, entre elas a medicina preventiva. Ao passo que técnicas de aprendizado profundo têm se mostrado eficazes para o prognóstico médico automatizado, elas carecem de mais transparência e interpretabilidade. Por outro lado, técnicas de agrupamento de dados e árvores de decisão são promissoras para a identificação de fatores de risco, características em comum, e tendências dentre os pacientes de acordo com suas respectivas histórias clínicas, descritas por prontuários médicos eletrônicos (EHRs). Deste modo, esta dissertação de mestrado objetivou a elaboração de um framework de aprendizado de máquina composto de uma rede neural Attentive Encoder-Decoder com o objetivo de predizer os diagnósticos da próxima admissão de pacientes, um algoritmo de agrupamento hierárquico para fenotipar essas predições, e finalmente, uma árvore de decisão objetivando-se a explicabilidade desses fenótipos; cada passo do nosso framework produziu um resultado em particular: a rede Attentive Encoder-Decoder obteve resultados de estado da arte nos datasets MIMIC-III e MIMIC-IV-ED; o algoritmo de agrupamento produziu resultados consistentes de diagnósticos relacionados em um mesmo fenótipo e, também, fenótipos vizinhos demonstraram similaridade de diagnósticos; e finalmente, a árvore de decisão proporcionou a visualização das regras de decisão entre diagnósticos de fenótipos e demonstrou a irrelevância de dados demográficos de pacientes em comparação com seus respectivos diagnósticos na identificação de um fenótipo. Nós resumimos nossas contribuições como: (i) Obtenção de resultados de estado da arte com um modelo versátil baseado em uma arquitetura Attentive Encoder-Decoder, nomeado por nós como *AttentionHCare* (BARROS; RODRIGUES, 2022), (ii) Fornecimento de uma ferramenta de suporte a decisão para especialistas por meio de um modelo explicável, e (iii) Habilidade de identificar padrões (*e.g.* fatores de risco, diagnósticos em comum, bias, etc.) em pacientes com trajetórias clínicas semelhantes por meio de um modelo explicável.

Palavras-chave: Prontuários Médicos Eletrônicos, Predição de trajetórias clínicas, Fenotipação de pacientes, Explicabilidade de predições clínicas, Construção de cohorts de pacientes, Encoder-Decoder, Mecanismos de atenção, Clusterização hierárquica, Árvores de decisão.

ABSTRACT

DE BARROS, P. H. F. **Machine Learning techniques applied to identifying clinical factors regarding automated medical prognosis**. 2022. 108 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

Machine Learning Algorithms have shown promising results in several areas of knowledge, including preventive medicine. While deep learning techniques have been shown to be effective for automated medical prognosis, they lack more transparency and interpretability. On the other hand, data clustering techniques and decision trees are promising for the identification of risk factors, characteristics in common, and trends among patients according to their respective clinical stories, described by their Electronic Health Records (EHRs). In this sense, this MSc thesis aimed to elaborate a machine learning framework composed of an Attentive Encoder-Decoder neural network to predict patients' next admissions' diagnoses, a hierarchical clustering algorithm to phenotype these predictions, and finally, a decision tree aiming the phenotypes' explicability; each step of our framework produced particular results: the Attentive Encoder-Decoder obtained state-of-the-art results over the datasets MIMIC-III and MIMIC-IV-ED; the clustering algorithm produced consistent results of related diagnoses of one same phenotype and, also, neighboring phenotypes demonstrated to have similar diagnoses; and finally, the decision tree provided a visualization of the decision rules between diagnoses of phenotypes and demonstrated the irrelevance of patients' demographic data in comparison to their diagnoses when identifying a phenotype. We summarize our contributions as follows: (i) Achievement of state-of-the-art results with a versatile model based on an Attentive Encoder-Decoder, named by us *AttentionHCare* (BARROS; RODRIGUES, 2022), (ii) Provision of a decision support tool to specialists as an explainable model, and (iii) Ability to identify patterns (*e.g.* risk factors, common diagnoses, bias, etc.) in patients with similar clinical trajectories by using the explainable model.

Keywords: Electronic Health Records, Clinical trajectory prediction, Patient phenotyping, Clinical prediction explicability, Patient cohort construction, Encoder-Decoder, Attention mechanisms, Hierarchical clustering, Decision trees.

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF TABLES

CONTENTS

CHAPTER

1

INTRODUCTION

Although the idea of using information systems with medical records on a daily basis has been in practice since the late 1980s and early 1990s (KIMBLE, 2014), it was not until the last decade that some organizational and governmental initiatives began to incentive the use of Electronic Health Records (EHR) in a major scale; these initiatives aim to harmonize EHR systems, as the case of the European EuroRec Seal initiative (EuroRec, 2009); to provide financial aid for the adoption of these systems, as the North-American HITECH Act (U.S.HHS, 2009); or to regulate and incentive the use in public health systems, as in the case of the Brazillian PLS 474/2008 (PLS 474, 2008). Some common benefits of the EHR adoption are the cost reduction of health care and improvement of quality of care (KIMBLE, 2014; CHAUDHRY *et al.*, 2006), but another, under exploration, is the use of EHRs in predictive analysis of medical conditions in the context of preventive medicine. With the steady increase of available data, commonly in a structured way, diagnosis prediction became a potential field for machine learning exploration.

An EHR is defined as a systematized longitudinal (*i.e.* overtime) collection of electronic health information about individual patients and populations (GUNTER; TERRY, 2005), these collections include information about diagnoses, prescribed medications, clinical procedures, laboratory analysis, etc. EHRs constitute complex documents about the patients' medical stories. It is common for a patient's EHR to have irregular frequencies between admissions, since patients typically visit a hospital only when they feel sick, possibly in an advanced stage of a particular disease.

Recent machine learning research advances, in special deep learning, are demonstrating to be capable to perform prediction of diseases. For example, medical image segmentation (STOLLENGA *et al.*, 2015; ZHOU *et al.*, 2018), content-based image retrieval (ANAVI *et al.*, 2016; LIU; TIZHOOSH; KOFMAN, 2016), breast (SPANHOL *et al.*, 2016; XU *et al.*, 2014) and lung (ARDILA *et al.*, 2019; XU *et al.*, 2019) cancer, among many other applications. Deep learning also drowns attention to exploring longitudinal clinical prediction problems (CHOI *et al.*, 2016a; PHAM *et al.*, 2017; CHE *et al.*, 2018). Such diversity of applications matches the

recent availability of medical data, since deep learning models are very data and computational intensive; these models have shown major benefits in health care such as precision therapies for complex illnesses, reducing medical errors, and improving the enrollment of patients into clinical trials (MILLER; BROWN, 2018).

However, the advance in medical predictions given by deep learning techniques and medical data availability faces a constraint: deep learning models are known to be *"black box models"*, that is, their inner mechanisms are so complex that they are not interpretable (RIBEIRO; SINGH; GUESTRIN, 2018); for differential diagnoses, cause-effect relationships, and other strategies employed for patient care and treatment, interpretation is crucial due to the potential impact in both the diagnosis and therapy/medication clinical workflows (CARUANA *et al.*, 2015). Naturally, this problem has caught the attention of governmental institutions, which have taken initiatives such as the 2018 EU's "Right to Explanation" regulation, and also from part of the machine learning community for the usage of more interpretable methods (RUDIN, 2019) to explain a learned model globally (LAKKARAJU; BACH; LESKOVEC, 2016; LETHAM *et al.*, 2015) (*i.e.* to explain the model as a whole), or locally (BAEHRENS *et al.*, 2010; RIBEIRO; SINGH; GUESTRIN, 2016) (*i.e.* to explain individually selected model predictions).

Accordingly, for medical domains producing only high accuracy predictions is not sufficient. It is also necessary to provide some model explanation for validation. A common belief is that exists a *tradeoff* between model interpretability and accuracy, that is, a more interpretable model will produce less accurate predictions and vice-versa (CARUANA *et al.*, 2015; LIPTON, 2018). For some problems with temporal relationships among features, such as the longitudinal clinical prediction, more interpretable models (such as linear regression or decision trees) would ignore these data relations (CHOI *et al.*, 2016c). Under this context, the explainability of black-box models by interpretation tools becomes an interesting choice instead of using a natively interpretable model due to the potential for better predictions.

1.1 Problem

Diagnosing a disease is a difficult and stressful task physicians have to face in their routines. Irregular time between admissions, overlapping symptoms that can be misleading, and sometimes a long progression of symptoms are some of the characteristics turning this task complex. We believe that clinical trajectory prediction as a support system can facilitate this task; we summarize the clinical trajectory prediction task as follows: "*Given a patient's sequence of admissions, possibly stored as an Electronic Health Record (EHR), predict the most probable diagnoses that shall appear in the next admission of this patient at a given time* $t+1$".

As previously argued, in a medical context, the predictions interpretability is an essential coefficient, so we also state an explicability task as: "*Given a patient next admission prediction obtained by a complex model, provide an interpretable outcome as a support feature to specialists,*

to help them in decision making about the diagnosis".

1.2 Objectives

The focus of this study are two:

1. Propose a model of neural network for the clinical trajectory prediction task, in other words, capable of predicting the diagnoses for a future admission after learning with the sequences of diagnoses of patients' past admissions; and producing state-of-the-art prediction results compared with literature's related works.

2. Perform a method to explain the neural network's predictions firstly by identifying patterns (*i.e.* phenotyping) in patients with similar future admission diagnoses through clustering, later by constructing cohorts with patients' demographic features, and lastly by fitting decision trees (BREIMAN *et al.*, 1984) on these future admission predictions by using the clusters as labels. Thus, composing a framework of clinical trajectory predictions and explicability.

1.3 Rationale

The application of deep learning in trajectory prediction has gained more interest in the last decade (LIPTON *et al.*, 2015) as works showed it to be efficient in modeling this type of data (CHOI *et al.*, 2016a; PHAM *et al.*, 2017; RODRIGUES-JR *et al.*, 2021).

Machine learning models are stochastic, therefore, it is intuitive to imagine that a sequence of machine learning techniques applied to a portion of data will, at each framework step, increase the discrepancy between the predictions and the ground truth. So, firstly, to explain the models' outcome it is fundamental to improve its prediction results by iterating over multiple training rounds. We summarize the rationale of our investigation as follows:

1. The proposing of a model has potential to produce better predictions in the context of computer-aided systems based on EHRs.

2. Due to the subject sensitivity, it is necessary that medical applications to be interpretable or explainable (CARUANA *et al.*, 2015).

3. Post-hoc explainable models are capable to provide a better understanding of *black-box* predictions; however, these models are nondeterministic and do not produce ground-truth explanations (RUDIN, 2019).

4. Hence, proposing a framework capable to produce better predictions and providing their explicability advances the state-of-the-art in the context of both computer-aided clinical prediction based on EHRs and model interpretability.

1.4 Contributions

The contributions of this study are as follows:

- Improve the accuracy of current clinical predictions with a versatile model based on an Attentive Encoder-Decoder, named by us *AttentionHCare*, pushing the state-of-the-art results;
- Provide a decision support tool to specialists as an explainable model;
- Be able to identify patterns (*e.g.* risk factors, common diagnoses, bias, etc.) in patients with similar clinical trajectories by using the explainable model.

This text is organized as follows: in Chapter 2 we describe the theoretical foundation used for this study; in Chapter 3 we present the related work; in Chapter 4 we describe our proposed methodology; in Chapter 5 we present our results, and in Chapter 6 our conclusions.

CHAPTER

2

THEORETICAL FOUNDATION

2.1 Neural networks

Inspired by the human brain, neural networks are powerful machine learning algorithms capable of performing computations such as classification, regression, and pattern recognition. One of the most popular classes of neural networks is the Multi-Layer Perceptrons (MLPs), which are networks composed of multiple layers of perceptrons (ROSENBLATT, 1958), organized as an input and an output layer with one or more hidden layers.

A key element about the popularity of MLPs is the Backpropagation algorithm (RUMELHART; HINTON; WILLIAMS, 1986), which has consolidated as the default learning procedure to train these networks. In general, this algorithm is capable to compute the loss function's gradient with respect to the perceptrons' weights efficiently by applying a derivative chain rule instead of computing the gradient with respect to each weight individually.

2.1.1 Recurrent neural networks

Traditional recurrent neural networks (RUMELHART; HINTON; WILLIAMS, 1985), were initially proposed in the 1980s aiming of modeling time series. Their structure is similar to a MLP neural network; however, recurrent networks are capable of modeling sequences with time delays among their values. This capability is explained by a proposed modification of the Backpropagation algorithm, named Backpropagation Through Time (BPTT) (WILLIAMS; ZIPSER, 1995; WERBOS, 1988) which is capable to compute the gradient of the loss function with respect to the weights at each iteration. Accordingly, unlike the MLPs, recurrent networks are capable of recognizing and learning with the ordinality of values of the input sequences.

Although other basic processing units of recurrent networks have been proposed, such as the Long Short-Term Memory (HOCHREITER; SCHMIDHUBER, 1997) and the Gated Recurrent Unit (CHO *et al.*, 2014), the aim of the recurrence learning initially proposed has been

maintained, which is still considered the state-of-the-art for several tasks including sentiment analysis (GRAY; RADFORD; KINGMA, 2017), time series classification (HORN *et al.*, 2019; KARIM *et al.*, 2019) and stock market forecasting (GHOSH; NEUFELD; SAHOO, 2022).

2.1.2 Vanishing/Exploding gradient problem

Despite being capable of learning with sequential data, the traditional recurrent networks suffer from the named "Vanishing/exploding gradient problem" in long-term dependencies, whenever these networks need to learn connections in long sequences, as noticed by (HOCHREITER, 1991) and (BENGIO; SIMARD; FRASCONI, 1994).

This problem appears because they are deep in time due to their recurrence even when these networks are not "spatially deep", *i.e.*, when they have many layers and neurons. Consequently, in long-term dependencies, gradient propagation through the BPTT algorithm may be affected, since the gradients of the previous time intervals are multiplied at each interval: if the gradients are small, they become even smaller and, if they are large, they become even larger, directly affecting the learning of the network.

One of the proposals to circumvent this problem was formalized by (PASCANU; MIKOLOV; BENGIO, 2013) through a method of clipping the gradients to a threshold in case they exceed the same previously determined threshold. However, despite being a simple and computationally efficient solution, the difficulty of this method consists in finding a threshold value that does not affect the learning of the network. Other proposals involve the use of different processing units, such as the Long Short-Term Memory, which will be shown in the following section.

2.1.3 Long short-term memory (LSTM)

The Long Short-Term Memory networks, or LSTMs, are architectures of recurrent neural networks capable of learning long-term dependencies better than the traditional recurrent neural networks. This occurs because their architecture prevents the vanishing/exploding gradient problem of the traditional RNNs during the gradients propagation. For this purpose, the traditional LSTM unit consists of one memory and three gates with specific objectives, and a capability of controlling how information flows from the inside to outside of the unit at each time interval; this control is regulated by a set of weights and activation equations. In a simplified way, while the backpropagation of the traditional recurrent networks is calculated through a multiplication of hidden states, in the LSTMs it occurs through a sum of terms regarding the differential of the equation that calculates the memory of the unit, and subsequently to the sum of a term concerning the *forget* gate (GERS; SCHMIDHUBER; CUMMINS, 2000).

The mechanics of this unit starts from the initialization of weights of each gate and the memory of the unit. In the literature, the initialization of weights through the Xavier initializer

(GLOROT; BENGIO, 2010) and the memory and biases through zeros are commonly used. Nonetheless, this subject is still in discussion, and there are many debates and suggestions on which is the best way for different applications (ZIMMERMANN; TIETZ; GROTHMANN, 2012).

Subsequently, for each time interval t, the previous unit's hidden state $h(t-1)$ and the input value $x(t)$ are concatenated as $s(t)$ and used in Equation (2.1c) that calculates the forget gate value $f(t)$, in which W_f, U_f and b_f are, respectively, the weights and bias regarding this gate, and σ is the sigmoid function. Similarly, the input gate value $i(t)$ is also calculated, according to Equation (2.1a). Moreover, the matrices $h(t-1)$ and $x(t)$ concatenated as $s(t)$ are used for calculating $z(t)$ through Equation (2.1d); however, the activation function is the hyperbolic tangent (*tanh*) this time. This function concentrates the matrices of weights and biases in an interval $[-1,1]$; therefore, unlike the traditional sigmoidal function, the *tanh* function decreases the propagation of errors while applying the BPTT algorithm (GREFF *et al.*, 2017).

Subsequently, the unit of the memory $c(t)$ is updated through Equation (2.1e) with the previously calculated $f(t)$, $c(t-1)$, $i(t)$ and $z(t)$. At last, the output gate $o(t)$ is similarly calculated to the input and forget gates (2.1b), as well as used to filter the unit output $h(t)$, according to Equation (2.1f). In Figure 1, the complete structure of a traditional LSTM unit is presented followed by all of the previously described equations.

$$i(t) = \sigma(W_i s(t) + U_i h(t-1) + b_i) \tag{2.1a}$$
$$o(t) = \sigma(W_o s(t) + U_o h(t-1) + b_o) \tag{2.1b}$$
$$f(t) = \sigma(W_f s(t) + U_f h(t-1) + b_f) \tag{2.1c}$$
$$z(t) = \tanh(W_z s(t) + U_z h(t-1) + b_z) \tag{2.1d}$$
$$c(t) = f(t) \cdot c(t-1) + i(t) \cdot z(t) \tag{2.1e}$$
$$h(t) = o(t) \cdot \tanh(c(t)) \tag{2.1f}$$

Since the consolidation of the unit with the forget gate, topologies with modifications (GERS; SCHMIDHUBER, 2000; KIM; EL-KHAMY; LEE, 2017), inspired architectures (WANG *et al.*, 2016; HU *et al.*, 2017), and ways of organizing (SHI *et al.*, 2015; HUANG; XU; YU, 2015) and stacking LSTMs units (PASCANU *et al.*, 2013) were proposed. Among these architectures used for classification, one commonly found in the literature consists of adding a dense feedforward layer at the end of a recurrent neural network (BOLLMANN; SØGAARD, 2016; SENNHAUSER; BERWICK, 2018; KRATZERT *et al.*, 2019). Accordingly, the LSTM unit outputs processed after the last time interval are passed as input to the densely connected

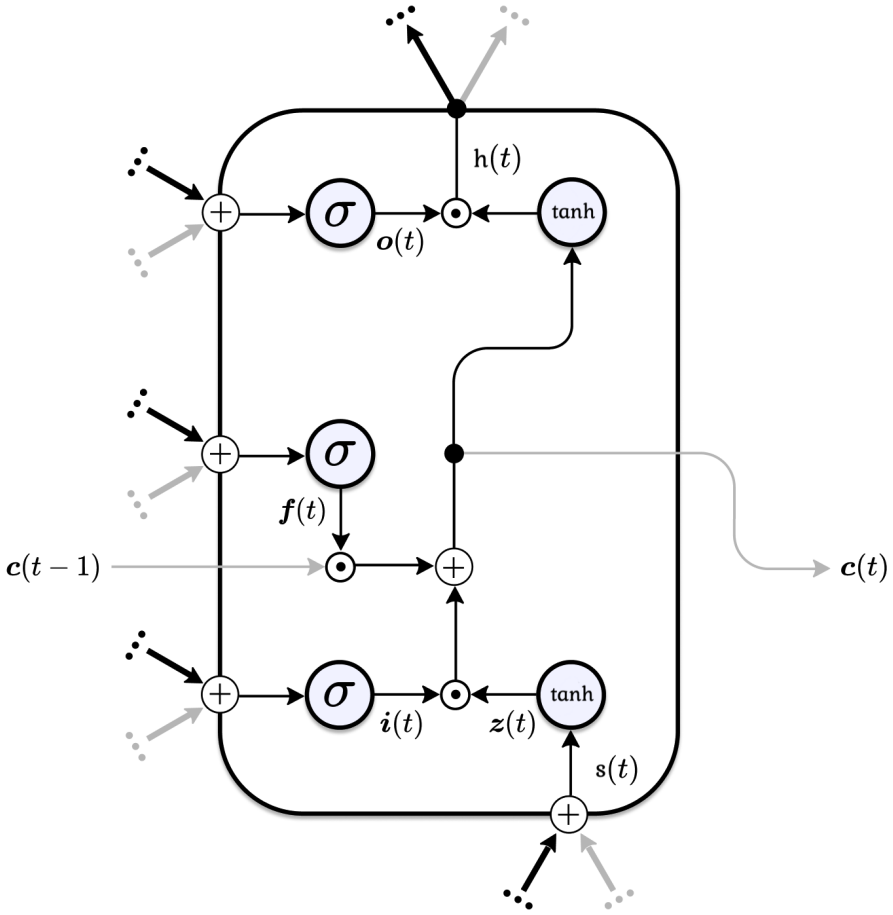

Figure 1 – Schematic of an LSTM unit.

Source: Adapted from Schimidinger (2020).

network to perform classification, as provided in Equation (2.2).

$$y_t = \phi(W_y h(t) + b_y) \tag{2.2}$$

In which the terms W_y and b_y are the weight and the bias, and ϕ is an activation function, commonly $softmax$ for multi-label problems.

One of these proposed architectures was described by (SAK; SENIOR; BEAUFAYS, 2014) and used in the context of speech recognition. In this architecture, the authors chose peepholes (GERS; SCHRAUDOLPH; SCHMIDHUBER, 2002) which is capable to improve predictions over long-term dependencies by using the unit's cell state in the gates equations and suggested a linear projection layer after an LSTM layer to reduce the training parameters and computational complexity of the traditional LSTMs. For this purpose, two changes were suggested in the original equations, as shown in Equations (2.3a) and (2.3b), in which W_r, is a

new trainable weight.

$$r(t) = W_r h(t) \tag{2.3a}$$

$$y_t = \phi(W_y r(t) + b_y) \tag{2.3b}$$

2.1.4 Encoder-Decoder architecture

Encoder-Decoder architectures (CHO *et al.*, 2014; SUTSKEVER; VINYALS; LE, 2014), at their publication period, quickly became the state-of-art in neural machine translation, speech processing, and in a more general way, sequence data processing and prediction. Currently, models based on this architecture compete with models based on *Transformers* (VASWANI *et al.*, 2017) in some benchmarks.

In summary, these models are composed of two recurrent neural networks: one encoding variable-length source sequences into a fixed-dimensional vector representation, which is decoded back to a target variable-length sequence by the decoder network. These networks are trained together as a single one, that is, at each training epoch the encoder network process the source input one-time interval at a time until its end, outputting a hidden state (the sequence summary) at each time interval, according to Equation (2.4) for input sequence x at time interval t.

$$h_t = f(h_{t-1}, x_t) \tag{2.4}$$

These encoder's hidden states, summarized as c in Equation (2.5), are processed by the decoder network, which outputs a h_t hidden state and a y_t target output at each time interval. The major difference between the encoder and decoder networks is that while the encoder works as a standard RNN, the decoder presents some changes on the hidden states equation, which now depends on previous time interval outputs, as shown in Figure 2.

$$h_t = f(h_{t-1}, y_{t-1}, c) \tag{2.5}$$

On the decoder implementation, there are two usual model modifications: the first one consists of using a beam search algorithm instead of a greedy one when selecting the best time interval target output, and the second one consists of using the ground-truth label instead of the previous one generated at each time interval, this technique is called *teacher forcing*.

With the beam search algorithm, the decoder is capable to select some target outputs (the beam size) for each time interval and select the best one based on conditional probability, this method differs from the greedy search algorithm which considers only the best target at each time interval which can lead to sub-optimal solutions. However, the beam search drawback is that the computational cost increases linearly according to beam size (RANZATO *et al.*, 2015).

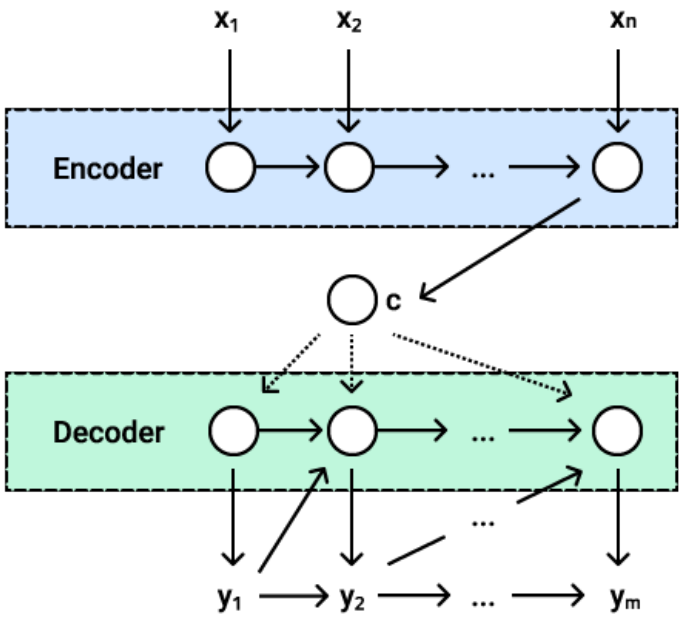

Figure 2 – Encoder-Decoder schematic.

Source: Adapted from Cho *et al.* (2014).

The other approach, known as teacher forcing, requires some training and testing changes. Firstly, at training time, the decoder learns with the ground-truth label at each time interval, instead of a previously generated one. However, at the testing time (or inference time), the process is inverted: the decoder generates target outputs based on the generated previously, this is necessary to avoid ground-truth bias and poor predictions. Teacher forcing can lead to a performance improvement (WU *et al.*, 2018) and help the decoder to learn quickly in the early training epochs (CHIU *et al.*, 2018).

2.1.5 Monotonic Bahdanau attention mechanism

Although the idea of an attention mechanism based on normalized dot-product between neural weights was explored in previous works such as the ones of Schmidhuber and Huber (SCHMIDHUBER; HUBER, 1990) and the most recent of Graves et al. (GRAVES; WAYNE; DANIHELKA, 2014), it was Bahdanau, Cho and Bengio (BAHDANAU; CHO; BENGIO, 2014) and Luong, Pham and Manning works (LUONG; PHAM; MANNING, 2015) that formalized the recent usage of attention mechanisms on recurrent neural networks, more specifically on Encoder-Decoder architectures.

Standard Encoder-Decoder architectures have a bottleneck between the encoder and

decoder steps, since the encoder network needs to summarize the whole input sequence before passing this representation to the decoder network. The quality of this representation normally degrades as the sentences become longer due to the representation fixed-size, which implies in worst decoding results. The goal of attention mechanisms is to prevent this bottleneck by employing a neural network, usually between the encoder and the decoder, to calculate weights given the hidden states outputted by the encoder at each sequence step. The attention mechanism weights are trained to represent the importance between parts of the sequence, which contributes to the context vector given to the decoder network after the whole sequence encoding.

Although new attention mechanisms and forms to employ them on sequence models have been proposed, the traditional form of attention is known as the Bahdanau Attention, this type of mechanism, represented in Figure 3, employs a neural network between encoding and decoding steps which aims to learn the alignments between the inputs and outputs and to output a soft score, *i.e.*, a $[0, 1]$ significance value between between the input and output parts-of-sequence.

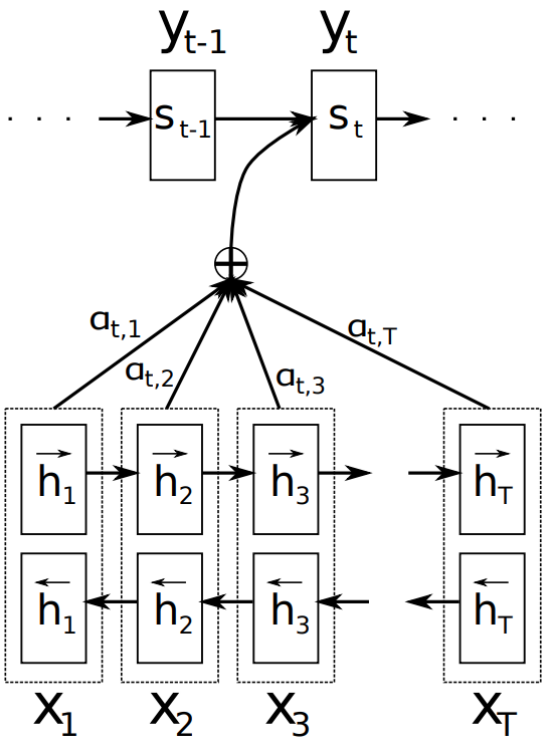

Figure 3 – Bahdanau Attention on Encoder-Decoder Architecture.

Source: Bahdanau, Cho and Bengio (2014).

To improve the decoding results, the Bahdanau Attention works with a set of equations, as described in Equations (2.6). Where h_j is the actual encoder hidden state; s_{i-1} the decoder previous hidden state; a an nonlinear function; $\alpha_{i,j}$ the attention weights for index i and j of states; v, W and V a learnable vector and matrices of weights; and c_i the context vector for index

i used on decoder's predictions.

$$a(s_{i-1}, h_j) = v^\top \tanh(W s_{i-1} + V h_j + b) \tag{2.6a}$$

$$e_{i,j} = a(s_{i-1}, h_j) \tag{2.6b}$$

$$\alpha_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k=1}^{T} \exp(e_{i,k})} \tag{2.6c}$$

$$c_i = \sum_{j=1}^{T} \alpha_{i,j}\, h_j \tag{2.6d}$$

Although being capable to improve prediction results, the standard Bahdanau Attention presents some algorithmic issues such as a quadratic time complexity to compute $\alpha_{i,j}$ for $j \in 1, ..., T$ for each output step i and it can not be employed in an online manner, that is, to compute the context vector without the whole sentence as input.

To overcome these issues, Raffel et al. (RAFFEL *et al.*, 2017) proposed modifications on Bahdanau Attention to reach linear time complexity and capacity to work on online setups, as presented in Equations (2.7). However, this mechanism assumes that the input and output are monotonic, that is, a given part of the input sequence will always come before the corresponding part of the output sequence.

$$a(s_{i-1}, h_j) = g \frac{v^\top}{\|v\|} \tanh(W_{s_{i-1}} + V_{h_j} + b) + r \tag{2.7a}$$

$$e_{i,j} = a(s_{i-1}, h_j) \tag{2.7b}$$

$$p_{i,j} = \sigma(e_{i,j}) \tag{2.7c}$$

$$q_{i_j} = (1 - p_{i,j-1})\, q_{i_j - 1} + \alpha_{i-1,j} \tag{2.7d}$$

$$\alpha_{i,j} = p_{i,j}\, q_{i_j} \tag{2.7e}$$

In order to reduce the time complexity, Local Monotonic Bahdanau Attention modify the quadratic Equation (2.6c) to a sigmoid activation and a distribution sampling on Equations (2.7c) and (2.7d) respectively. Another modification is the nonlinear function (2.7a): a scalar variable r, which according the authors allows the model to learn the appropriate offset for the pre-sigmoid activations, is added; and the term $v^\top$ is changed to $g\frac{v^\top}{\|v\|}$, which according the authors reduces the sensitive to the scale of the terms $e_{i,j}$.

2.2 Unsupervised clustering

While classification tasks are performed in a supervised manner and aim to predict categories of unknown data, clustering tasks are generally unsupervised and have the purpose of

describing and linking data through similarities or differences (ROKACH, 2010). Depending on prior knowledge of a problem or assumptions about the domain, clustering tasks can be formalized in different ways (GRIRA; CRUCIANU; BOUJEMAA, 2005).

According to the taxonomy proposed by (JAIN; MURTY; FLYNN, 1999) and modified by (ARAUJO, 2015), as shown in Figure 4, clustering techniques can be divided into two broad categories: Partitional Clustering techniques and Hierarchical Clustering techniques. The main difference between them is how the clusters are linked. In hierarchical clustering, data partitioning is recursively nested and produces a hierarchy of tree-shaped partitions, whereas in partitional clustering does not exist a nesting and all of the clusters are independent partitions (STEINBACH; ERTÖZ; KUMAR, 2004). Some examples of hierarchical clustering algorithms are the agglomerative and divisive algorithms, while some popular partitional clustering algorithms are K-Means (MACQUEEN, 1967), DBSCAN (ESTER *et al.*, 1996) and CLIQUE (AGRAWAL *et al.*, 1998).

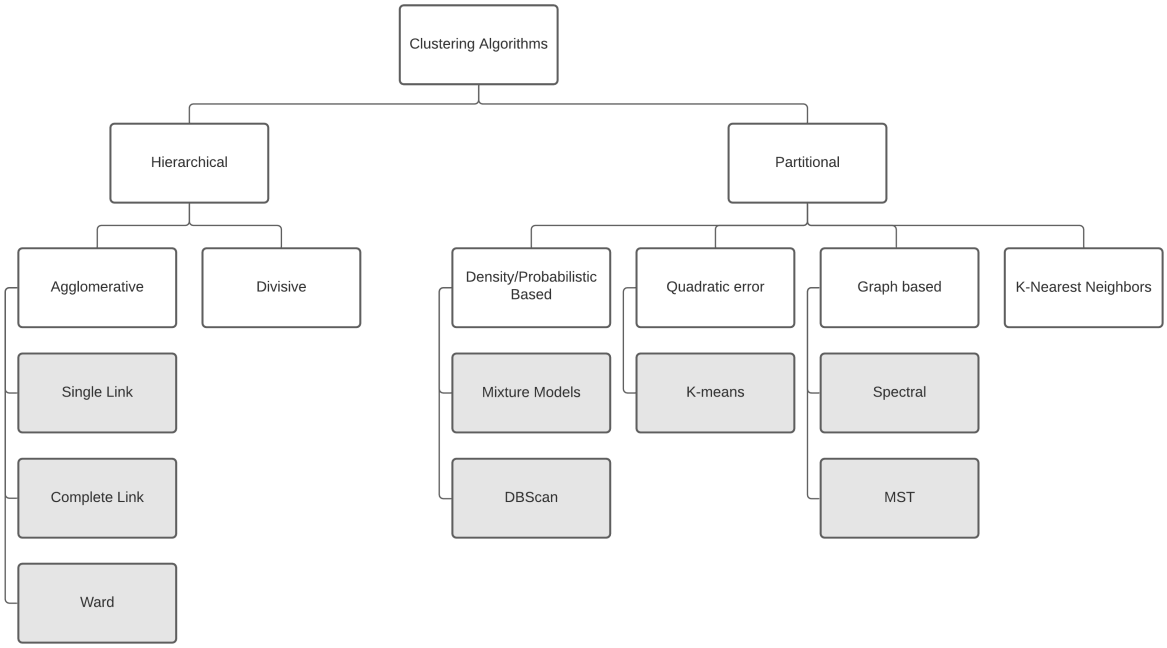

Figure 4 – Taxonomy of clustering algorithm.

Source: Adapted from Jain, Murty and Flynn (1999), Araujo (2015).

2.2.1 Agglomerative hierarchical clustering

Unlike some partitional clustering algorithms, such as the K-Means, the hierarchical clustering algorithms do not require a specific hyperparameter for the number of clusters to be produced and they are not dependent on initialization values. These differences, associated with a dendrogram resulting from the algorithm, become attractive in problem-solving when the number of clusters on data is unknown, for example.

The hierarchical clustering algorithms can be divided into two categories: agglomerative and divisive. Their main difference is that the agglomerative algorithms have a bottom-up method while the divisive algorithms are top-down. In other words, in agglomerative algorithms, all data of a set initially have their cluster that is merged with the others at each iteration until resulting in a single cluster. On the other hand, in divisive algorithms, all data initially belongs to a single cluster that is divided at each iteration until resulting in clusters with a single data sample (ROKACH, 2010).

In both methodologies, a hierarchical clustering algorithm produces a dendrogram, as exemplified in Figure 5. This tree-shaped dendrogram represents each of the clusters in different granularities and the connection between them. Consequently, it is noticeable which clusters are the closest and what are their clustering structure (BERKHIN, 2006; MURTAGH; CONTRERAS, 2012). The dendrogram can also be visually analyzed to establish the best linkage criteria and which is the most feasible height to cut. After the cut, each disconnected tree below the cut line becomes an independent cluster. One of these analyses involves visualizing the size of the vertical edges of a cluster and their sub-clusters, in which the Y-axis of the dendrogram represents the distance determined by the linkage criteria. An indication of a clustering relatable to the data structure is the growth of edges according to the combination of clusters, that is, the vertical edge of a parent cluster is greater than that of its child clusters (METZ, 2006). Moreover, the size of this edge may indicate an interesting cut point because, as they show the distance between the clusters, it is more interesting to apply the cut to the larger edges of the dendrogram.

Another dendrogram visual analysis starts from the horizontal edges, in which the X-axis represents the dataset samples; therefore, the larger the horizontal edge that represents the linkage between two clusters, the greater the dissimilarity between them (RICHETTE *et al.*, 2015). In conclusion, the presence of larger horizontal edges, which represent the combination of two clusters, indicates that the hierarchical algorithm created clusters most relatable to the data structure.

As previously mentioned, in agglomerative hierarchical clustering, each object is initially assigned to a cluster containing only itself, and at each iteration of the algorithm, new objects are clustered together by joining smaller clusters. For this purpose, there must have some linkage criteria between the objects that will be detailed in the following subsection. In general, the agglomerative algorithm is described in Algorithm 1, in which C_a and C_b represent arbitrary clusters from the dataset with the cardinalities N_a and N_b, and the similarity matrix M_{sim} is determined according to the linkage criteria used.

For the calculation of the similarity matrix M_{sim}, it is possible to apply many different types of distance measures, including the Euclidean distance, Minkowski distance, Manhattan distance, Cosine similarity, or Jaccard similarity. However, some linkage criteria are capable of working with only specific distance measurements, such as the Ward method (WARD, 1963) that can only be operated using the Euclidean distance. This may be a limiting factor while choosing

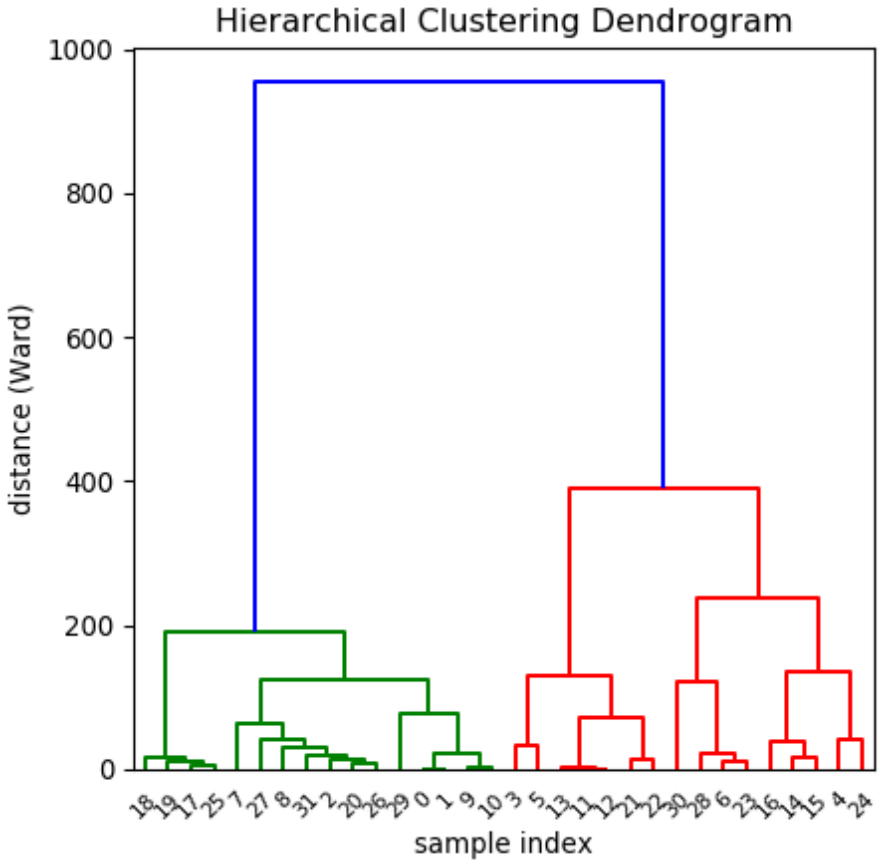

Figure 5 – Example of a dendrogram.

Source: The Python Graph Gallery (2017).

Algorithm 1 – Agglomerative Hierarchical Clustering (Adapted from (REDDY; VINZAMURI, 2013)).

1: **procedure** AGG_HIERARCHICAL_CLUST
2: Calculate the similarity matrix M_{sim} between all the objects
3: **while** There is not only one cluster **do**
4: Merge the clusters as $C_{a\cup b} = C_a \cup C_b$ according to the linkage criteria
5: Attribute the cardinality of the new cluster $C_{a\cup b}$ as $N_{a\cup b} = N_a + N_b$
6: Calculate the distance between $C_{a\cup b}$ and the other clusters
7: Append the distance calculated in line 6 to the similarity matrix M_{sim}
8: **end while**
9: **end procedure**

a method, especially in cases of high dimensionalities of data, in which the Euclidean distance, for example, may not be the best choice (AGGARWAL; HINNEBURG; KEIM, 2001).

As well as the similarity measures, these criteria are essential to set clusters through an algorithm and they are sensitive to how data are displaced in space; Figure 6 represents the clusters set by some linkage criterion in different forms of data in a two-dimensional space. It is important to emphasize that, as the analysis of the dendrogram, the arrangement of the visualized

data is also a significant factor to establish which linkage criteria to use.

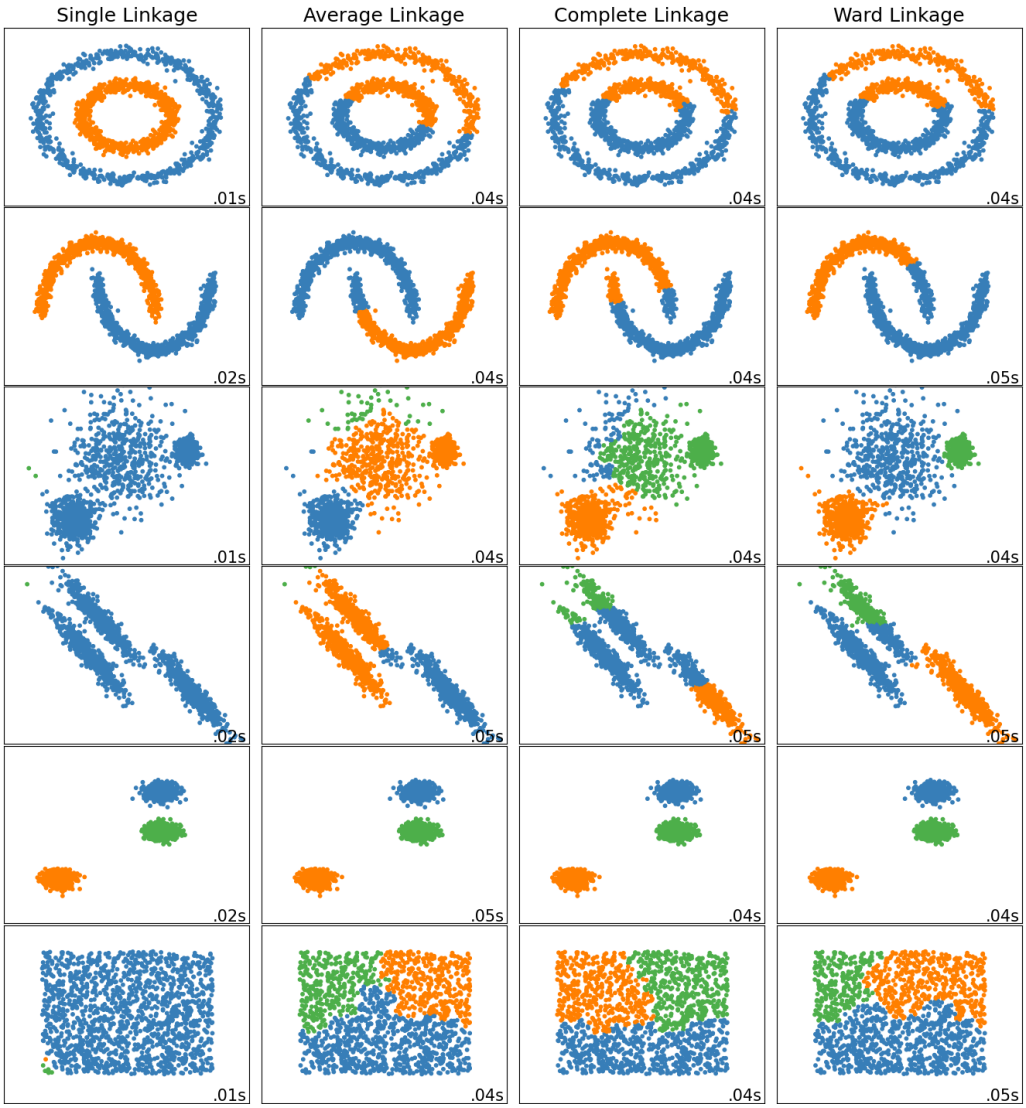

Figure 6 – Examples of data linkage criteria with different shapes.

Source: Scikit-learn Examples (2020).

The following are some of the main linkage criteria commonly described in the literature:

- *Single link*: the single link criteria (MCQUITTY, 1957; SOKAL, 1966) is one of the oldest and most popular linkage criteria. This method attaches more importance to local similarities and closer samples. Considering the samples of a cluster C_1 and other clusters' samples $C_2, C_3, ..., C_n$, the single link will merge the C_1 with the cluster that has the closest data of any sample of C_1; consequently, these criteria are also known as the "nearest neighbor method". This methodology is capable of clustering data arranged in non-elliptical and elongated forms; however, it is sensitive to noises and outliers (REDDY; VINZAMURI, 2013).

- *Complete link*: along with the single link, the complete link is also one of the most traditional linkage criteria. Nonetheless, they have different methodologies: while the single link is known as the "nearest neighbor method", the complete link is known as the "furthest neighbor method". As a result, given a cluster C_1 among the clusters $C_1, C_2, C_3, ..., C_n$ of the dataset, this criteria will merge C_1 with the cluster that has the furthest sample of any C_1 samples. This method aims to form clusters without considering only the structure of local clusters; however, it is also sensitive to noises and outliers as the single link (REDDY; VINZAMURI, 2013).

- *Average link*: while the previous criteria considered the distance between any objects of a cluster with objects from the other clusters, the average link (SOKAL, 1958) (also known as UPGMA - Unweighted Pair Group Method with Arithmetic Mean) obtains an average between the samples distances of a cluster with the samples of the other clusters. Equation (2.8) shows the calculation of criteria A between the clusters C_a and C_b for the samples i and j, respectively.

$$A(C_a, C_b) = \sum_{i,j} \frac{dist(C_a[i], C_b[j])}{|C_a| * |C_b|} \tag{2.8}$$

  As well as the complete link, the average link also considers more global structures, but it is less sensitive to noises; however, this criteria can cause elongated clusters to divide, and to merge portions of elongated clusters to neighbor clusters (ROKACH, 2010).

- *Weighted link*: the weighted link (SOKAL, 1958) is a weighted version of the average link, also known as WPGMA - Weighted Pair Group Method with Arithmetic Mean. The differential of this method is to consider the cardinalities of the last merged clusters when merging them to a new cluster; therefore, this method is computationally lighter than the non-weighted version. In Equation (2.9), C_a and C_b represent the last merged clusters and C_k is the new one to be merged.

$$W(C_{a \cup b}, C_k) = \frac{dist_{C_a, C_k} + dist_{C_b, C_k}}{2} \tag{2.9}$$

- *Centroid*: the centroid method (SOKAL, 1966), also known as UPGMC - Unweighted Pair Group Method with Centroid, uses the clusters' centroids as a metric for deciding whether or not to merge two clusters. This calculation is shown in Equation (2.10), in which C_a and C_b are two clusters with their respective centroids c_a and c_b. Since this method only uses centroids for calculation, it is more scalable and requires less computing time.

$$Ce(C_a, C_b) = \|c_a - c_b\|_2 \tag{2.10}$$

- *Median method*: the median method is a weighted modification of the centroid method in the same way that the weighted is a modification of the average method. It is also

known as WPGMC - Weighted Pair Group Method with Centroid, but despite being called the "median method", its implementation is not very relatable to the average calculation between the clusters. Equation (2.11) shows how the calculation would have been performed for the clusters $C_{a\cup b}$ and C_k with centroids c_a, c_b and c_k.

$$Ce(C_{a\cup b}, C_k) = \frac{\|c_a - c_k\|_2 + \|c_b - c_k\|_2}{2} \tag{2.11}$$

- *Ward's method*: at last, Ward's method (WARD, 1963) has a more complex formula than the previous criteria. One of the strengths of this method is the development of compact and uniformly-sized clusters (SZMRECSANYI, 2012). Its formula assumes that choosing the pair of clusters to be merged at each step of the algorithm is based on the ideal value of an objective function, such as the error sum of squares. Equation (2.12) shows the calculation of the distance associated with this method for the clusters $C_j = C_a \cup C_b$ and C_k.

$$Ward(C_j, C_k) = \sqrt{\frac{(|k| + |a|)dist(k,a)^2 + (|k| + |b|)dist(k,b)^2 - |k|dist(a,b)^2}{|j| + |k|}} \tag{2.12}$$

Connection constraints

Besides linkage criteria, in the agglomerative hierarchical clustering methods, it is also possible to control and restrain the formation of clusters with connection matrices. These matrices act like connection constraints between data points or clusters by describing which points can be connected with which by the chosen linkage criteria. In some cases, the usage of these constraints is capable of creating clusters with better shape, that is, clusters that are visually better or with better evaluation scores.

In practice, these connection matrices can be any graph that describes the connection between each data point, but, one popular way to generate these matrices is to create a graph from the execution of a K-Nearest Neighbors (KNN) algorithm. This lazy learner algorithm, when used unsupervised, works by connecting the K nearest points for each data point projected in an Euclidean space, according to their distances. In Figure 7 are presented to the same data points the usage or not of connection constraints generated by KNN with K equal to 20.

2.2.2 Methods to estimate the optimal number of clusters

Some clustering algorithms, such as the K-means and the Gaussian Mixture Model, require an explicit hyperparameter stating the number of clusters that will be returned after clustering. Other algorithms, such as the agglomerative hierarchical clustering, do not demand the number of clusters; however, in some cases is desirable to known what is the best dendrogram cut and, consequently, the best number of resulting clusters.

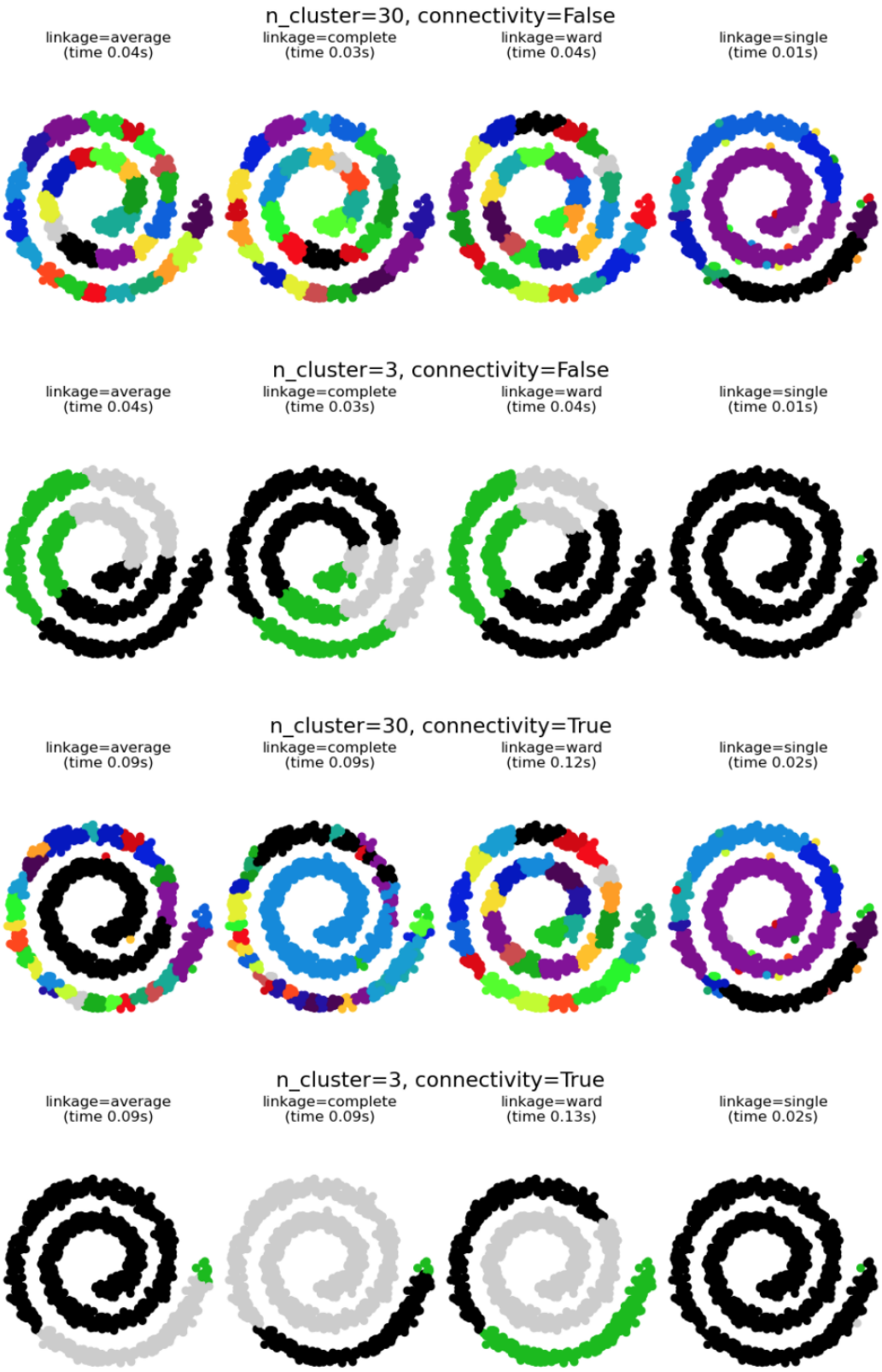

Figure 7 – Examples of clustering results with and without connection constraints generated by KNN (K=20).

Source: Scikit-learn Examples (2021).

Accordingly, when it is unknown how many clusters should be obtained, heuristics can be used because they are capable of aiding in estimating the optimal or sub-optimal number of clusters. It is worth mentioning that, even with the use of these heuristics, choosing the number of clusters also depends on other factors: the problem to be treated, previous knowledge about data, and data visualization that can support this decision. Next, some heuristics are discussed.

Another option is to apply more than one heuristics to the same dataset with the same clustering algorithm because, in case of obtaining a great discrepancy between the optimal number of clusters between the different methods, this can be an indicator that the clustering algorithm is not the most suitable for clustering this dataset (TAN *et al.*, 2018).

2.2.2.1 Silhouette score

The Silhouette method (ROUSSEEUW, 1987) is a traditional alternative to evaluate the number of clusters to be selected. This method assigns a score from -1 to 1 for how similar the objects are to their clusters regarding the other clusters, in which higher score values mean that, on average, objects have been assigned correctly to clusters.

Mathematically, in a dataset with n objects, and considering an object i assigned to a cluster C_i among k possible clusters, $i \in C_i$. The average distance between i and the other objects of its same cluster is defined in Equation (2.13), in which $d(i,j)$ is the distance between the objects i, j and the average dissimilarity, as described in Equation (2.14)

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, j \neq i} d(i,j) \tag{2.13}$$

$$b(i) = min \frac{1}{|C_k|} \sum_{j \in C_k} d(i,j) \tag{2.14}$$

At last, the silhouette coefficient of the object and the average silhouette coefficient, or score, of the performed clustering are obtained through Equations (2.15) and (2.16), respectively.

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}}, se|C_i| > 1 \tag{2.15}$$

$$S = \frac{\sum_1^n s(i)}{n} \tag{2.16}$$

As a result, with the coefficients and score, a quantitative way is sought to determine the quality of the performed clustering, as well as the most optimal number of clusters, according to this metric.

2.2.2.2 Davies-Bouldin's index

Davies-Bouldin index (DAVIES; BOULDIN, 1979) is another heuristic for quantifying the quality of a clustering task, including the number of clusters. It aims to measure the proportion between an intra-cluster dispersion and an inter-cluster distance, that is, this method defines a clustering task as successful if the sets of the produced clusters are compact and well-separated (BOLSHAKOVA; AZUAJE, 2003). For this purpose, two equations are initially determined: the first quantifies the intra-cluster dispersion of each sample regarding its cluster centroid, and the second calculates the distance between two clusters' centroids. These equations are detailed below, as well as the final calculation of the index.

As previously mentioned, intra-cluster dispersion aims at quantifying how compact is a cluster of samples. Given a cluster C_i with T_i samples and X_j the feature vector of a point belonging to C_i. The dispersion S_i is defined as shown in Equation (2.17), in which A_i refers to the centroid point of the cluster C_i and p is an arbitrary value, being usually defined as 2 in order to calculate the Euclidean distance between X_j and A_i.

$$S_i = \left(\frac{1}{T_i} \sum_{j=1}^{T_i} |X_j - A_i|^p \right)^{\frac{1}{p}} \tag{2.17}$$

On the other hand, the inter-cluster distance aims at quantifying how separated two clusters C_i and C_j are among N clusters produced by the employed algorithm. Therefore, in the equation, the distance between the centroids of each clustering is calculated for each feature a_{ki}, in which k refers to the k-th feature of the cluster's centroid i, according to Equation (2.18).

$$M_{i,j} = \left\| A_i - A_j \right\|_p = \left(\sum_{k=1}^{n} |a_{k,i} - a_{k,j}|^p \right)^{\frac{1}{p}} \tag{2.18}$$

As a result, the Davies-Bouldin DB index can be calculated through Equation (2.19).

$$DB = \frac{1}{N} \sum_{\substack{i,j=1 \\ i \neq j}}^{N} max \, \frac{S_i + S_j}{M_{i,j}} \tag{2.19}$$

It is important to emphasize that the closer to 0 is DB, the better the quality of a clustering task and, consequently, the number of clusters since the ratio calculated in Equation (2.19) will have smaller values when the clusters S_i and S_j are less dispersive and the distance $M_{i,j}$ between them is greater.

2.3 Decision trees

The Decision Tree, more specifically, a Classification and Regression Trees (or CART) was firstly introduced in (BREIMAN *et al.*, 1984). This model aims to build a tree with decision rules as branches and classes as leaves, allowing it to walk on a decision path from the tree's root to the outputted classification, as represented in Figure 8.

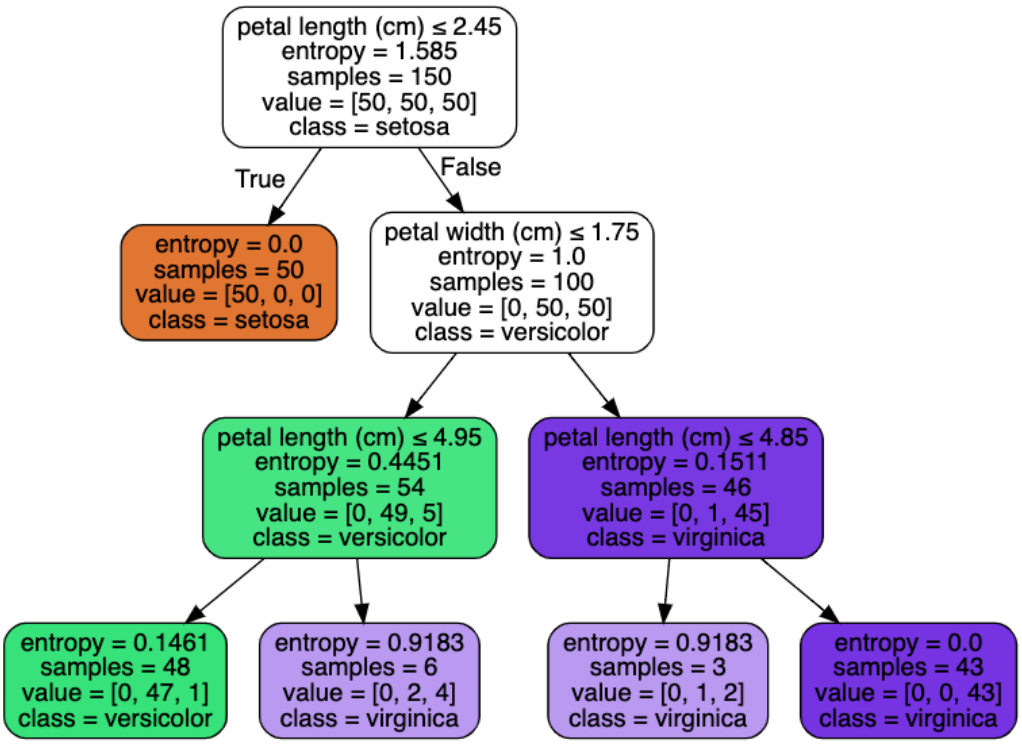

Figure 8 – A decision tree, fitted on Iris dataset (FISHER, 1936), visualization using Graphviz (ELLSON *et al.*, 2002).

Source: TinaGongting (2019).

A decision tree building with CART algorithm works in a top-down recursive manner with a splitting criteria based on the outcome of a discrete function of the input attributes, known as the Gini impurity measure. Equation (2.20) shows how to calculate it for each sample belonging to label i with the probability to be randomly picked $p(i)$ in a set of C total classes.

$$G = \sum_{i=1}^{C} p(i) * (1 - p(i)) \tag{2.20}$$

The algorithm proceeds as follows: starting with a single leaf (the root), containing all samples of the dataset assigned as a label according to the dataset label majority, a series of iterations splitting the data and selecting the features is performed. Among all possible splits, with their respective Gini impurity measure, the selected split is the one that maximizes the measure. The same process occurs from the newly selected splitted node until no split gains

the sufficient splitting measure or a stopping criterion is satisfied (ROKACH; MAIMON, 2005; SHALEV-SHWARTZ; BEN-DAVID, 2014).

Decision trees are known for being easily interpretable, that is, for having classification rules of easy visualization and interpretation. For that reason, can be found in the literature being used both to add interpretability in *black box* models (LAKKARAJU *et al.*, 2017; KRISHNAN; SIVAKUMAR; BHATTACHARYA, 1999; RAI, 2019) and clustering tasks (QUIROS, 2017; APILETTI *et al.*, 2016; MORICHETTA; CASAS; MELLIA, 2019; HANCOCK; COOMANS; EVERINGHAM, 2003), which is one of the objectives of this dissertation.

Also, it is important to emphasize that the decision trees are not deterministic; therefore, when they are trained with the clusters resulting labels, the generated tree and the rules for data separability, according to features, will not completely reflect on the result of the clustering algorithm. Nonetheless, measures used for evaluating supervised models, such as accuracy, precision, recall, etc. may indicate how much the structure of the trained decision tree is similar to the clustering results (MORICHETTA; CASAS; MELLIA, 2019; KRISHNAN; SIVAKUMAR; BHATTACHARYA, 1999) and are capable to indicate the quality of model prediction.

2.4 Evaluation metrics

Evaluation metrics represent a quantitative way of determining the classification performance of supervised learning algorithms and comparing them with other models. Therefore, these metrics use counts of samples classified as true positives (considering a sample of the class of positives; the algorithm classified it as belonging to the class of positives), false positives (considering a sample of the class of negatives; the algorithm classified it as belonging to the class of positives), true negatives (considering a sample of the class of negatives; the algorithm classified it as belonging to the class of negatives), and false negatives (considering a sample of the class of positives; the algorithm classified it as belonging to the class of negatives).

In this dissertation, we evaluate our results of patient clinical trajectories prediction with metrics that are frequently used to evaluate recommendation systems due to the similarity of the nature of the problem. We also employ commonly used evaluation metrics for classification problems in order to evaluate decision tree learned model. These metrics are described below.

- Precision: indicates the proportion of samples classified as positive regarding all the actual positives. High precision shows a low rate of false positives.

$$Precision = \frac{True\ Positives}{True\ Positives\ +\ False\ Positives} \tag{2.21}$$

- Precision@k: similar to precision, precision@k shows the proportion of recommended results, that is, the positive predictions among the k relevant values, namely the actual

positive classes.

$$Precision@k = \frac{True\ Positives\ @k}{True\ Positives\ @k\ +\ False\ Positives\ @k} \tag{2.22}$$

- Recall (or Sensitivity): is a metric that evaluates the proportion of samples correctly classified as positives. It is a more conservative measure frequently used when it is preferable to avoid false-negative predictions.

$$Recall = \frac{True\ Positives}{True\ Positives\ +\ False\ Negatives} \tag{2.23}$$

- Recall@k (or Sensitivity@k): similar to recall, recall@k specifies a proportion of k relevant values, that is, the actual positive classes among recommended results, namely the positive predictions.

$$Recall@k = \frac{True\ Positives\ @k}{True\ Positives\ @k\ +\ False\ Negatives\ @k} \tag{2.24}$$

- Specificity: while recall evaluates the proportion of samples correctly classified as positives, i.e. the true positive rate, the specificity measures the proportion of samples correctly classified as negatives, *i.e.* the true negative rate

$$Specificity = \frac{True\ Negatives}{True\ Positives\ +\ False\ Negatives} \tag{2.25}$$

- Specificity@k: is analog of the recall@k, however, measures the actual negative classes among the recommended results.

$$Specificity@k = \frac{True\ Negatives\ @k}{True\ Positives\ @k\ +\ False\ Negatives\ @k} \tag{2.26}$$

- AUC (Area Under the Curve) is a metric extracted from the ROC (Receiver Operating Characteristic) curve obtained from the rates of true positives with false positives. A common interpretation of the AUC metric concerns the expected proportion of positives to be classified before classifying an uniformly-sorted random negative. This metric is frequently used to compare classification models, and models with higher AUC are considered better.

- F1-Score: is a metric that evaluates the balance between precision and recall, and the advantage of its usage is having more robustness to the imbalance of classes than accuracy.

$$F1 - Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{2.27}$$

- Accuracy: accuracy is the most intuitive metric. Its purpose is to evaluate within the predictions which ones were correct, that is if the prediction and label were the same. However, accuracy is a metric sensible to the imbalance of classes.

$$Accuracy = \frac{True\ Positives\ +\ True\ Negatives}{True\ Positives\ +\ False\ Negatives\ +\ True\ Negatives\ +\ False\ Positives} \tag{2.28}$$

2.5 International Classification of Diseases (ICD) codes

For more than a century, the International Classification of Diseases (ICD) (World Health Organization, 2022) is an international standard, maintained by the World Health Organization (WHO), for organizing clinical terms such as diseases, symptoms, abnormal findings, and other elements of patient's diagnoses in a way that is universally accepted by those in the medical and insurance fields (Trisha Torrey, 2022).

Currently in the 11th revision (*ICD-11*), this standard is periodically revised in order to include new disorders or to reclassify old ones. However, it maintains its structure of grouping ranges of conditions' codes in a hierarchical way (from the more general to the more specific), as the ICD-9's intestinal infectious diseases exemplified bellow.

- 001 - Cholera disease
- 002 - Typhoid and paratyphoid fevers
  - 002.0 - Typhoid fever
  - 002.1 - Paratyphoid fever A
  - 002.2 - Paratyphoid fever B
  - 002.3 - Paratyphoid fever C
  - 002.9 - Paratyphoid fever unspecified
- 003 - Other Salmonella infections
  - 003.0 - Salmonella gastroenteritis
- 004 - Shigellosis
  - 004.9 - Shigellosis, unspec.

The major adoption of this standard by hospitals and healthcare systems worldwide provided benefits for patients' data exchange between systems, and also statistical analysis about patients' diseases progression and cause of mortality. For this reason, the ICD is the main standardization of classification we used in this work.

2.6 Considerations about the theoretical foundation with proposed work

Our proposed work comprehends a variety of machine learning methods and techniques. Since our objectives are to predict the diagnosis of patients' sequence of admissions and to give interpretability to these predictions through cluster phenotyping, we firstly use recurrent neural networks to sequence learning and next admission prediction, and lastly agglomerative hierarchical clustering and decision trees to perform cohort clustering and predictions interpretability.

CHAPTER

3

RELATED WORK

This chapter presents the works related to the framework proposed in this project. These works are divided into three groups: “Prediction of clinical trajectories” which refers to works on the longitudinal prediction of clinical diagnoses using electronic medical records as input and through the use of different statistical models and machine learning. “Clinical phenotyping”, which comprises the identification of cohorts with certain diseases or medical conditions taking into account the diagnoses, treatments, demographic data, and risk factors of the respective patients. And, lastly, “Explainability of clinical predictions” which refers to works that seek to complement predictions with explainability, given the need for experts to evaluate the models.

Some of the selected works have intersections between the groups, such as works that perform phenotyping and propose their explainability. The studies are initially described within the group considered most relevant, and at the end of the chapter, they are compared with the proposed methodology.

3.1 Clinical trajectories prediction

The problem of clinical trajectories prediction (also called longitudinal trajectories prediction) is understood as: given t sequences of patient’s admissions, each one with their respective diagnoses, the objective is to predict the most probable diagnoses on admission $t+n$, in which commonly $n = 1$. We detail the problem description in Section 4.2.1. Thus, the related works described below focus on predicting future diagnoses in clinical trajectories based on electronic medical records.

The first work to explore the use of LSTMs to predict clinical trajectories was the one presented in the work of Lipton et al. (LIPTON *et al.*, 2015). In this work, the authors proposed a model of two LSTM layers followed by a densely connected layer and an element-wise sigmoid activation layer to perform a multi-label prediction. Two LSTM architectures were tested: in the

first one, the loss function was calculated after the recurrence of the entire sequence of inputs; in the second one, at each recurrence step the target tensors were presented (this technique is called target replication), then the final loss was calculated taking into account the average of all intermediate losses. The experiments were performed using the Children's Hospital LA dataset to predict the 128 most frequent diagnoses among the 429 present in the dataset. In the end, the results were compared with an MLP network with manually constructed features and the authors' proposal showed better results of Area Under Curve - AUC, F1-score, and Precision@10.

A more specific proposal was presented in the paper of Pham et al. (PHAM *et al.*, 2017) in which the authors proposed a modified LSTM with an architecture to predict diagnoses in a cohort of diabetes and mental illnesses, however, the authors stated that their architecture is capable to be generalized to more general predictions. The proposed modification, called Care-LSTM, or C-LSTM, made it possible two irregular modelings of time: the first one was a decay function that took into account the time between admissions whose result was multiplied by the forget gate, and the second one was a time between admissions parameterized function to capture chronic conditions whose result was multiplied by one of the forget gate weights. Initially, the tensor input consisted of four dimensions: a set of admissions codes, a set of intervention codes, the type of admissions, and the time interval between the current and the previous admission. The input sequence passed through an embedding layer that produced dense tensors that were given as input to a C-LSTM layer. Lastly, three different pooling layers for C-LSTM outputs were proposed: max, normalized sum, and min. After the pooling layer, the results were classified by a densely connected layer with softmax activation. The results were compared with pure LSTMs and with models based on Markov chains and DeepCare, the architecture with mean pooling demonstrated better results.

In the paper of Rajkomar et al. (RAJKOMAR *et al.*, 2018), the authors ensembled three different neural network models: a weighted recurrent neural network model, a feedforward network with time-aware attention, and an embedded time-series model in the tasks of trajectory prediction, mortality prediction, unexpected readmission and length of stay using private datasets provided by the University of California, San Francisco (UCSF) and the University of Chicago Medicine (UCM) and the public dataset Medical Information Mart for Intensive Care III (MIMIC-III) (JOHNSON *et al.*, 2016). The models were employed to predict 228 distinct diagnosis codes per patient admission in EHR records based on a data structure called Fast Healthcare Interoperability Resources (FHIR) (MANDEL *et al.*, 2016). In comparison with baselines which consisted of statistics and logistic models, the authors reported better results for the proposed ensemble model.

In the work of Che et al. (CHE *et al.*, 2018), a modification of the recurrent neural network GRU, named GRU-D, was proposed. This new unit aimed to provide the possibility of modeling the representation of missing patterns through masking and irregular time intervals. For this purpose, a decay function with trainable parameters was proposed for input gate and

hidden states, and, despite having the same formulation, each of these functions had different trainable weights. At each recurrence, these decay tensors were combined with the input gate and hidden state, respectively. For the tests, synthetic and real datasets were used: among the real ones, the PhysioNet dataset (SILVA *et al.*, 2012) was used for the task of mortality prediction and length of stay, and the MIMIC-III dataset for the tasks of mortality prediction with 20 categories of diagnoses for the next admission prediction. In both tests, a GRU-D recurrent neural network with only one layer was used, and, a softmax prediction layer was added to the recurrent network end for multi-label prediction. The predicted results of the diagnostic prediction task were compared with several GRU architectures and demonstrated to obtain better AUC results.

The method proposed in the work of Choi et al. (CHOI *et al.*, 2016b), although not focusing on the prediction of clinical trajectories, was an intersection between prediction and explainability since the experiments for model validation were done with future diagnoses prediction, the authors also proposed the model interpretation, since it works similarly with the Skip-gram model (MIKOLOV *et al.*, 2013). The proposed architecture, called Med2vec, aimed to learn representations related to clinical applications, *i.e.* diagnostics, procedures, prescriptions, demographic data, etc. To this end, the architecture was divided between codes and visits representation: first, a multi-hot encoded vector of codes was given as input to a MLP network with ReLU activation and its representation outputs were concatenated with demographic data and inserted again in an MLP network with ReLU activation and after that activated with a softmax layer. The model output was a dense matrix of visit representations and patients codes. Two datasets were used in experimentation: Children's Healthcare of Atlanta (CHOA) and CMS, and was performed the clinical trajectory prediction task by giving the previous visit tensor as input and the following visit tensor as the target. One of the performance metrics used by the authors was Recall@30, which obtained results superior to other similar methodologies such as Skip-gram and Glove (PENNINGTON; SOCHER; MANNING, 2014). The model interpretability was later explored and was discussed for the top 10 codes of each dimension within the embeddings dense tensor produced.

An interesting proposal was described in the paper of Choi et al. (CHOI *et al.*, 2016a), named Doctor AI. In this work, the authors described an architecture based on GRU-type recurrent neural networks aiming to predict subsequent admissions by modeling diagnostics and prescriptions in electronic medical records. The architecture consisted of multi-hot encoded vectors of diagnoses and procedures, passing through an embedding layer, then in a GRU-RNN layer, and lastly by a softmax layer. The authors presented results of four variations of architectures, namely: GRU-RNN of one and two layers with embedding weights and GRU-RNN of one and two layers with embedding weights trained by *Skip-gram*; the best-reported results were the two-layer model with Skip-gram. The dataset used for training was a private one provided by Sutter Health Palo Alto Medical Foundation. The authors also used the model trained for classification in the public MIMIC-III dataset through transfer learning. Finally,

the proposed method results were compared with logistic regression baselines, and MLPs and showed superior results for Recall@10, Recall@20, and Recall@30.

Another interesting methodology was the work of Rodrigues-Jr et al. (RODRIGUES-JR *et al.*, 2021) explored several designs of recurrent neural networks in the task of predicting clinical trajectories with the public dataset MIMIC-III and the private Instituto do Coração de São Paulo's dataset InCor. The authors explored the use of Feedforward networks, two variations of GRUs, two variations of LSTMs, the Jordan network, and also the architecture of Doctor AI described previously. At the end of the exploration, the architecture that achieved the best results for Recall@10, Recall@20, and Recall@30 was a single-layer bidirectional network composed of GRU or MinGRU units, followed by a densely connected layer with Leaky ReLU activation and a softmax activation layer. Sequences of both diagnostics and procedures codes, transformed into multi-hot encoding vectors, were used in both International Classification of Diseases, Ninth Revision (ICD-9) (ORGANIZATION, 1978) (more granular) and Clinical Classifications Software (CCS) (COST; PROJECT, 2015) (less granular) standards without going through an embedding layer, which according to the authors worsened the predictive results. Other authors concluded that the number of units in each recurring layer showed better results when it was closer to the size of the input, and by increasing the number of recurrent layers did not improve the prediction results, which according to the authors, was counterintuitive in comparison to other deep learning works results.

Lastly, the recent work of Florez et al. (FLOREZ *et al.*, 2021) proposed a transformer-based architecture named APEHR to predict next admission diagnoses over MIMIC-III and InCor datasets using both ICD-9 and CCS standards. The architecture consists of input of multi-hot encoding vectors representing diagnostics and procedures codes projected to a lower dimensional space through an embedding layer and then rearranged by a positional encoding layer, since transformers models does not use recurrence to process data. After the input embedding and arrangement, the decoder part of a transformer consisting of multi-head self-attention, addition & normalization, and densely connected layers outputs its context vectors to a densely connected layer and a softmax activation layer. In comparison with others state-of-the-art works, the authors reported competitive or better results for Recall@k ($k \in \{10, 20, 30\}$), Precision@n ($n \in \{1, 2, 3\}$) and AUC-ROC metrics.

3.2 Clinical phenotyping

The terms cohorts and phenotyping are sometimes used interchangeably in the literature (SCIENCES; INFORMATICS, 2020). However, these terms can be differentiated in the sense that while cohort means a group or subgroup of individuals belonging to a study, phenotyping can be understood as the observable state of an organism (HRIPCSAK; ALBERS, 2017). In this regard, we can consider that a clinical phenotyping task consists of identifying cohorts with the

determined desired phenotype (SHIVADE *et al.*, 2013). Another term that can be considered a synonym for clinical phenotyping is patient subtyping, which aims to identify groups of patients, that is, cohorts, with similar diagnostic progressions (BAYTAS *et al.*, 2017), commonly, with clustering techniques (ZINCHUK *et al.*, 2017). Here, both these terms are used interchangeably.

A methodology that does not consider the data in a longitudinal way, that is, ordered sequentially overtime, was the one described in the work of Vandromme et al. (VANDROMME *et al.*, 2019), in which the authors used a hierarchical clustering algorithm to perform clinical phenotyping in a cohort of patients with non-alcoholic fatty liver disease aiming for the identification of subtypes for this condition. Demographic characteristics, diagnoses, procedures, laboratory tests, and vital signs extracted from electronic medical records were selected. The dendrogram obtained was cut to obtain 5 clusters; the authors justified that this choice was made to balance the clusters' granularity and size. The best parameters were selected after several algorithm executions and a qualitative evaluation of the results. In the end, the methodology was validated against a labeled data set, and the clusters were compared with the classes of this new dataset. After ten runs of the clustering algorithm in the new set, the authors were able to verify that the results of the algorithm were similar to the labels of the new dataset.

Another non-longitudinal approach to clinical phenotyping discovery was performed in the work of Dai et al. (DAI *et al.*, 2017). First, the authors used natural language processing techniques in textual information of electronic medical records, aiming to transform them into a dictionary of code words in UMLS - Unified Medical Language System standard (LONG, 2005). Since the goal was to find topics in this dictionary, the authors chose the algorithm *Hierarchical Dirichlet process* (TEH *et al.*, 2006), which employs a Bayesian data grouping approach. According to the authors, the topics modeled by the algorithm captured the latent structure of the input data. After selecting only topics with at least 200 code words in the UMLS standard, the algorithm generated 27 topics with 9868 dimensions. Due to these topics' high dimensionality, the authors applied the t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm (MAATEN; HINTON, 2008) to reduce the dimensionality to two dimensions that could be observed. Lastly, after further investigation with the silhouette score technique, the K-means algorithm was used with K equals 26, empirically. The authors visually confirmed the methodology results of identifying the most relevant diagnoses in each cluster and the proximity between clusters, indicating relations according to medical literature.

In the work of Zhang et al. (ZHANG *et al.*, 2019), the authors conducted the subtyping of longitudinal data in a group of patients with Parkinson's disease, this study's goal was to characterize progressions groups of this disease and how their patients are characterized. For this purpose, the authors proposed an architecture consisting of extracting the temporal relationships through LSTMs, calculating the similarity between embeddings, and clustering the results of similarity to find similar trajectories. At first, demographic and clinical characteristics were extracted from patient records, these characteristics were concatenated to form dense vectors that

were used as input to a LSTM unit together with label representational vectors, which in this case were obtained in a previous study conducted by the authors and considered as the ground truth for disease subtyping. After training, the hidden states were used as an embedding representation of the original data; these embeddings were used to generate a matrix of similarities between each pair of patients with the Dynamic Time Warping algorithm (MÜLLER, 2007). Lastly, the similarities were reduced to two dimensions using the t-SNE algorithm and clustered with K-means with K equals 3. The authors justify their choice of K based on Hartigan's rule (HARTIGAN, 1975).

Another work in which patients were subtyped in longitudinal data was described in the paper of Baytas et al. (BAYTAS *et al.*, 2017). In this work, the authors proposed a new LSTM unit architecture: the Time-Aware LSTM, or T-LSTM. The particularity of this new unit was that it can model temporal irregularities in sequences in a more reliable way than the traditional LSTMs: before being combined with the long-term memory, a defined decay function weight is discounted from the short-term memory taking into account the number of days between the predictions. Then, the proposed T-LSTM was used in an auto-encoder architecture to phenotype patients' trajectories with Parkinson's disease. The representations generated by the auto-encoder were then clustered by K-means with K equal to 2, the authors justified choosing the value of K based on visual analysis. Lastly, the 2 clusters analysis was made by comparing the means of patients' characteristics in each cluster.

A similar proposal to the previous one was described in the work of Zhang et al. (ZHANG *et al.*, 2018): the phenotyping of patients with Parkinson's disease was also performed in the empirical values of 2 clusters and Alzheimer's disease in 3 clusters. They proposed a decay function with trainable weights as a modification in the operations of a GRU unit by multiplying the recurring weights in the update gate equation, the authors called this new unit as Time-Sensitive GRU or TS-GRU. Then, the phenotyping task was done in an auto-encoder architecture with a TS-GRU cell and the generated representations were clustered with the Weighted K-means algorithm. Unlike the previous work, in this one, the authors did not perform the clustering after the reduction of dimensionality. As a measure of comparison with previous studies, the p-value metric was used to verify the quality of the clustering results, according to the authors, the lower the p-values obtained for dataset characteristics, the better the clustering. The authors claimed that better results were obtained in comparison with previous state-of-the-art works.

Lastly, the methodology proposed in the paper of Wang et al. (WANG *et al.*, 2019) combined phenotyping in a cohort of chronic lymphoid leukemia with the predicted diagnoses interpretability through a medical services word cloud of each phenotype found. First, the authors proposed a model inspired by the Wide & Deep framework (CHENG *et al.*, 2016), in which the deep component of the model was replaced by a LSTM unit purposing sequential data modeling. The input characteristics given for the wide component were those related to demographic data and phenotype characteristics extracted through Non-negative matrix factorization - NMF

procedure (SRA; DHILLON, 2005), which is capable to cluster data. On the other hand, an embeddings tensor related to continuous diagnostics in the longitudinal electronic medical records was processed in the deep component and a global max-pooling operation was performed in the LSTM hidden states. Then, the concatenated outcomes of the wide and deep components were input to a three-layer neural network. Like previous works, one of main the difficulties was the definition of predicted phenotypes quantity, the authors reported that after several attempts, the quantity of three phenotypes was validated with the support of a specialist.

3.3 Clinical prediction explainability

In recent literature, the terms "explainability" (or "explicability") and "interpretability" are often used interchangeably when referring to *black box* models. Although the recent popularization of deep learning models have drawn attention to interpretability/explainability research, there are older methodologies in the literature, such as the work of Schmitz, Aldrich and Gouws (SCHMITZ; ALDRICH; GOUWS, 1999), which have proposed to explain models despite not using the terms "interpretability" or "explainability".

However, there are works that differentiate these terms and consider that while interpretability is a characteristic native of a model, that is, the predictions of a model are natively interpretable by analyzing their results, the explicability consists in the use of an interpretable model for explaining the predictions of an uninterpretable model, thus consisting of a *posthoc* explanation of an uninterpretable model (LIPTON, 2018; RUDIN, 2019).

In the medical field, it is mandatory that results from statistical analysis or machine learning algorithms to be interpretable or explainable to a specialist. Such a need became especially relevant recently due to the availability of electronic medical records and advances in machine learning (ZHANG *et al.*, 2018). Related works with the context of explainability of clinical predictions are described below.

In the work of Zhang et al. (ZHANG *et al.*, 2018), the authors proposed a prediction model and vector representation of patients' clinical trajectories. The methodology for trajectories' representation provides explicability to the predictions according to the authors. Their methodology evaluates the importance of features that contribute to the predicted diagnosis, identifying the patients' profiles. The proposed architecture was composed of a GRU recurrent bidirectional network and a hierarchical attention mechanism, which according to the authors provided superior predictions and contributed to the results' explicability. First, the medical records were given as input to a skip-gram model with an attentional convolutional layer to obtain a dense representation. These representations were then processed by a bidirectional GRU layer and then by a self-attention layer which outputted a context vector based on GRU's hidden states and the self-attention mechanism results. Lastly, the predictions were made after adding the obtained context vector with the patients' demographic representations vector. The dataset

used for tests was provided by the University of Virginia Health System, which contains 75 months of data about diagnoses, medications, and patient procedures.

A proposal that used the MIMIC-III dataset for diagnostic predictions and made possible the model interpretability was the one described in the work of Suresh et al. (SURESH *et al.*, 2017). The authors compared an architecture based on LSTMs with another based on convolutional neural networks (CNNs) to predict a set of 5 clinical interventions with their respective subcategories. Features were extracted manually from the dataset and grouped into numeric and static categories (such as demographic data), and narratives (such as medical notes). In the case of the narrative characteristics, the authors used the Latent Dirichlet allocation algorithm (BLEI; NG; JORDAN, 2003) to extract topics that became features. Both LSTMs and CNNs architectures presented results superior to the baseline defined by the authors, which was a logistic regression, taking into account the AUC evaluation metric. Lastly, the explainability in both models was based on features' occlusion techniques, in the LSTM scenario, each one of the features was replaced by an uniform distribution to verify which contributed more to the results based on problem labels.

On the other hand, the methodology described in the paper of Che et al. (CHE *et al.*, 2016) uses natively interpretable model imitating a deep learning model. The authors proposed two imitation approaches: in the first one, a deep learning model produced soft labels as outputs that were used as labels for an imitation model, while in the second one, the features learned by a deep model were used by an auxiliary classifier and then, the classifier predicted labels were considered as the target labels for an imitation model. The deep learning model used was GRU recurrent neural networks and MLPs, and the imitation model was the Gradient boosting trees algorithm (FRIEDMAN, 2001). The task, performed over the Pediatric ICU dataset (KHEMANI *et al.*, 2009) from Children's Hospital Los Angeles, was the prediction of mortality and free days without the use of ventilation, whose patients' features were static demographic data and daily monitoring measures. As result, the authors reported that the best imitation model was based on the first approach and the results of AUROC surpassed purely deep models. Lastly, the model interpretability was achieved by the visualization of features' importance and the decision rules of the ensemble's most important tree. Here, we pursue a similar rationale.

3.4 Considerations about related works with proposed work

Our work act as an intersection between the three presented groups of related works because our proposed methodology comprehends clinical trajectory predictions and predictions interpretability through patients phenotyping.

Accordingly, we compared related works with the methodology proposed in this qualification. In particular, we selected relevant characteristics of our proposed method and later

Table 1 – Comparison between related and proposed work.

	ICD Codes	CCS Codes	Longitudinal EHR	Explainability (Diagnostics)	Explainability (Demographics)	Future diagnosis prediction
Lipton et al. (LIPTON *et al.*, 2015)	✓	✗	✓	✓	✗	✓
Rajkomar et al. (RAJKOMAR *et al.*, 2018)	✓	✗	✓	✗	✗	✓
Care-LSTM (PHAM *et al.*, 2017)	✓	✗	✓	✓	✗	✓
Che et al. (CHE *et al.*, 2018)	✓	✗	✓	✗	✗	✗
Med2vec (CHOI *et al.*, 2016b)	✓	✗	✓	✓	✗	✓
Doctor AI (CHOI *et al.*, 2016a)	✓	✗	✓	✗	✗	✓
LIG-Doctor (RODRIGUES-JR *et al.*, 2021)	✓	✓	✓	✓	✗	✓
APEHR (FLOREZ *et al.*, 2021)	✓	✓	✓	✗	✗	✓
Vandromme et al. (VANDROMME *et al.*, 2019)	✓	✗	✓	✗	✓	✗
Dai et al. (DAI *et al.*, 2017)	✓	✗	✗	✓	✗	✗
Zhang et. al (ZHANG *et al.*, 2019)	✗	✗	✓	✓	✓	✗
T-LSTM (BAYTAS *et al.*, 2017)	✓	✗	✗	✓	✗	✗
TS-GRU (ZHANG *et al.*, 2018)	✗	✗	✓	✓	✗	✗
Wang et al. (WANG *et al.*, 2019)	✗	✗	✓	✓	✗	✓
Patient2vec (ZHANG *et al.*, 2018)	✓	✓	✓	✓	✗	✗
Suresh et al. (SURESH *et al.*, 2017)	✗	✗	✓	✓	✗	✗
Che et al. (CHE *et al.*, 2016)	✗	✗	✓	✓	✓	✗
Proposed methodology	✓	✓	✓	✓	✓	✓

attributed them to related works. They are:

- ICD codes: this characteristic refers to the model being able to make predictions using codes in the ICD-9 or ICD-10 standard;
- CCS codes: in the same sense as ICD codes, this characteristic refers to the model being able to make predictions with codes in the CCS standard;
- Longitudinal EHR: if the model makes predictions using longitudinal electronic medical records, that is, the patient's data have a progression over time;
- Interpretability (Diagnostics): this characteristic is related to the model's ability to provide the interpretability of the predicted diagnoses;
- Interpretability (Demographics): the models considered to have this characteristic are those that provided interpretability containing patients' demographic data;
- Future diagnosis prediction: if the model makes predictions of future diagnoses, such as patients' next admission.

CHAPTER

4

METHODOLOGY

In this chapter, we present our proposed framework describing materials, tasks to perform, proposed methods, and the methodology validation. Also, the framework code is public available at <https://github.com/grgau/msc-thesis>

4.1 Materials

Two datasets were used as materials, as described below:

- **MIMIC-III (Medical Information Mart for Intensive Care III)**: a publicly available dataset from MIT and Beth Israel Deaconess Medical Center researchers. This dataset consists of 58,976 patient admissions which were in Beth Israel Deaconess Medical Center's intensive care unit between 2001 and 2012. The data describe demographic features, vital signs, laboratory test results, diagnoses, procedures in the ICD-9 standard, medications, mortality, and caregivers' notes. The dataset makes it possible to carry out diverse tasks such as diagnosis prediction or mortality, and epidemiological studies.

  Some remarkable characteristics are: the patients' median age is 65.8 years, 55.9% of the patients are male, mortality of 11.5%, and ICU and hospital stay medians are 2.1 and 6.9 days, respectively. In addition, among patients over 16 years old, the three most common diagnoses are: "414.01 - Coronary atherosclerosis of native coronary artery" present in 7.1% of the admissions, "038.9 - Unspecified septicemia", and "410.71 - Subendocardial infarction, initial episode of care" present in 4.2% and 3.6% of admissions, respectively (JOHNSON *et al.*, 2016).

- **MIMIC-IV-ED (Medical Information Mart for Intensive Care IV - Emergency Department)**: also a public dataset, provided by Physionet in 2020. It covers medical admissions of the emergency department from the Beth Israel Deaconess Medical Center

between 2011 and 2019; it contains 448,972 admissions from a group of 216,877 patients; totaling 13,434 raw codes in both ICD-9 and ICD-10 standards.

The ICD-10 standard is a revision and extension of ICD-9; while the ICD-9 hierarchical structure is described as a 3 to 5 numeric code, ICD-10 describes conditions in the form of a 3-7 alphanumeric code, which extends the range of conditions and enriches their descriptions: for example, the total distinct codes in ICD-9 is 14,025 while in ICD-10, it is 69,823 (Centers for Disease Control and Prevention, 2015). Therefore, the diagnoses and procedures of the MIMIC-IV-ED dataset are more granular than the ones of MIMIC-III.

Datasets comparison

MIMIC-III is a smaller dataset in comparison with MIMIC-IV-ED, the patient's trajectories can have between 2 and 42 admissions, and the number of diagnosis codes varies between 1 and 39 per admission. The mean number of admissions per trajectory and the number of codes per admission are 11.89 and 16.89 respectively for this dataset. Meanwhile, for the larger MIMIC-IV-ED dataset, trajectories can have between 2 and 203 admissions and the number of diagnosis codes varies between 1 and 9 per admission; the mean number of admissions per trajectory and number of codes per admission are 38.74 and 4.2, respectively.

Despite the differences in size, the patients from MIMIC-III present shorter trajectories (*i.e.* fewer admissions), on average, in comparison with MIMIC-IV-ED; but their admissions present more diagnosis codes, on average. In terms of cardinality and granularity, that is, the quantity of admissions per patient and diagnoses per admission, MIMIC-IV-ED presents greater cardinality and greater granularity because it has longer trajectories and fewer diagnoses per admission than MIMIC-III.

4.2 Tasks description

4.2.1 Clinical trajectories prediction

As previously introduced in Section 3.1, clinical trajectories prediction (or longitudinal prediction) consists of predicting a patient's future diagnoses based on previous diagnoses. In this work, the trajectories prediction consists of predicting the diagnoses of a patient admission y_{t+1} considering his y_{t-r} previous admissions, for $(t+1) < r < 0$.

Formally, given a sequence of admissions $A = (a_0, a_1, a_2, ..., a_{m-1})$ with size $m \geq 2$, each patient admission is given as the pair $a_i = (t_i, Di)$, $a_i \in A$ where $i, 1 \leq i \leq (m-1)$ is the time order; $t_i, t_1 \leq t_i \leq t_{m-1}$ is the admission timestamp; $D_i = \{d_{i,0}, d_{i,1}, d_{i,2}, ..., d_{i,n-1}\}$, $0 \leq n \leq |\mathbb{D}|$ are the diagnoses; with $D \subset \mathbb{D}$, where $\mathbb{D}$ is the set of all possible predicted codes, usually in ICD-9 or ICD-10 standard. The probability $\mathscr{P}$ of predicting a set of diagnoses in admission y_{t+1}

is given by the following equation:

$$y_{t+1} = \{\ \mathscr{P}(D_{t+1,j} \mid a_{0:m-1})\ \},\ \ 0 \leq j \leq (|\mathbb{D}| - 1) \tag{4.1}$$

4.2.2 Clinical trajectories phenotyping and explicability

The previous subsection described the task of predicting the diagnosis for future admission y_{t+1}. After this task, the objective is the production of a matrix P of probabilities for all the patients. We can define the matrix $P \in \mathbb{R}^{p \times |\mathbb{D}|} = (y_{0,t+1},\ y_{1,t+1},\ y_{2,t+1},\ ...y_{p,t+1}), p \leq \mathbb{P}$, being $\mathbb{P}$ the set of all the patients.

Given matrix P with dimensions $p \times |\mathbb{D}|$ treated as features, it becomes possible to produce a clinical phenotyping by using clustering techniques (see Section 3.2. We compute a set of crisp clusters $C = \{(P_0, c_1), (P_1, c_1), (P_2, c_2), ..., (P_n, c_k)\}$, in which $0 \leq n \leq p$ and $1 \leq k \leq K$, K the maximum number of clusters, and c_k the cluster to which the patient's features P_n belongs. Thus, we can define a clustering function $\mathscr{C} : P \rightarrow C$ as:

$$\mathscr{C}(P) = \{\ (P, c) \mid c = Clust(P)\ \} \tag{4.2}$$

Where $c \in C$ and $Clust$ is a clustering algorithm to be defined.

Lastly, given a set of clusters C, the objective is to perform a classification task defined by a function $\mathscr{F} : C \rightarrow F$, given by:

$$\mathscr{F}(c) = \{\ (c, f) \mid f = Class(c)\ \} \tag{4.3}$$

Where $c \in C$ is an ordered pair of patients features and clusters treated as labels, $f \in F$ and $Class$ is an explainable classification algorithm to be defined, which are capable to produce interpretable results about the predictions using the provided patients features.

4.3 Input preprocessing and representation

Here we cover our preprocessing step; both our input preprocessing and representation were the same proposed and used in the works of Rodrigues-Jr et. al (RODRIGUES-JR *et al.*, 2021) and Florez et. al (FLOREZ *et al.*, 2021).

Firstly, we cleaned the patient's admissions data, converted the diagnosis codes into a simpler representation, and splited the data into training, validation, and testing sets. Next, we built multi-hot tensors representing the diagnoses of each admission for each patient.

In the cleansing, the admissions without diagnoses and the patients without admissions were discarded. Patients with a single admission were also discarded as it is not possible to use

them in the recurrent neural network learning process. After cleansing, we converted each ICD-9 or ICD-10 diagnosis into a sequential integer, or, otherwise, if the standard to be used was the CCS, firstly we mapped the ICD-9 codes into CCS codes.

Our model input was a tridimensional tensor of admissions $Y = (y_0, y_1, y_2, ..., y_{m-1})$, batches of patients $P = (p_0, p_1, p_2, ..., p_{p-1})$ and diagnoses $D_i = \{d_{i,0}, d_{i,1}, d_{i,2}, ..., d_{i,n-1}\}$. For each admission y_i, we defined a $|D|$-dimensional multi-hot vector, where D was the set of diagnosis codes, as defined by Equation (4.4):

$$x[i][h][j] = \begin{cases} 1, & \text{if } d_j \in x_{h,i} \\ 0, & \text{otherwise} \end{cases} \quad \text{for } 0 \leq i \leq (m-1), 0 \leq h \leq (p-1), 0 \leq j \leq (|D|-1) \tag{4.4}$$

In Equation (4.4), x is an input tensor in which the first dimension represents the admissions; the second represents the patients; and the third represents multi-hot vectors of diagnosis codes. We also used the length of the sequences as part of the input (*i.e.* the number of admissions per patient) as a method that prevents padding the sequences during the recurrent learning step. The binarization process of converting integer diagnosis codes to multi-hot vectors is illustrated in Figure 9.

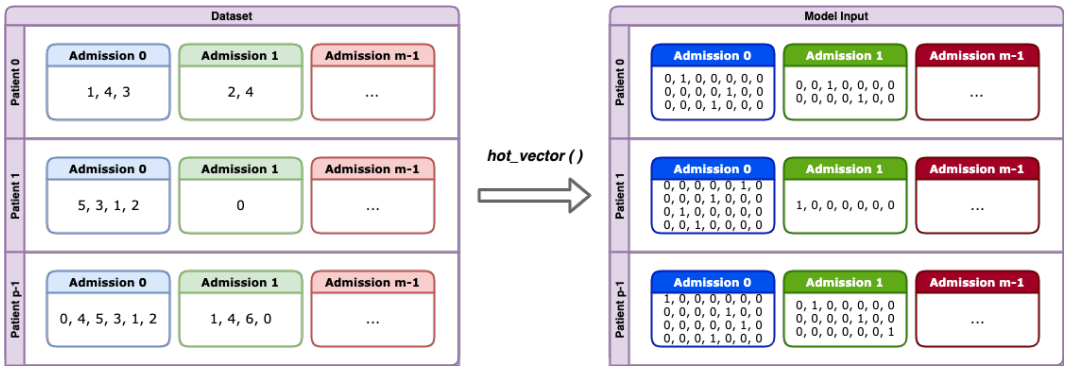

Figure 9 – Process of converting patient's data to our model input.

4.4 Proposed framework

Our proposed framework was composed of two stages, as illustrated in Figure 10, the framework performed a sequence of machine learning data processing tasks. The first stage performed clinical trajectories prediction through recurrent neural networks, and the second stage aimed at the explicability of the process through clustering and decision trees. The preprocessed data was inputed to an Encoder-Decoder recurrent neural network whose output were the predicted probabilities of each diagnostic code for each patient. In the second stage, these probabilities were clustered by a hierarchical clustering algorithm that produced clusters used as labels. Then, cohorts of patients were built by concatenating these predictions with the static

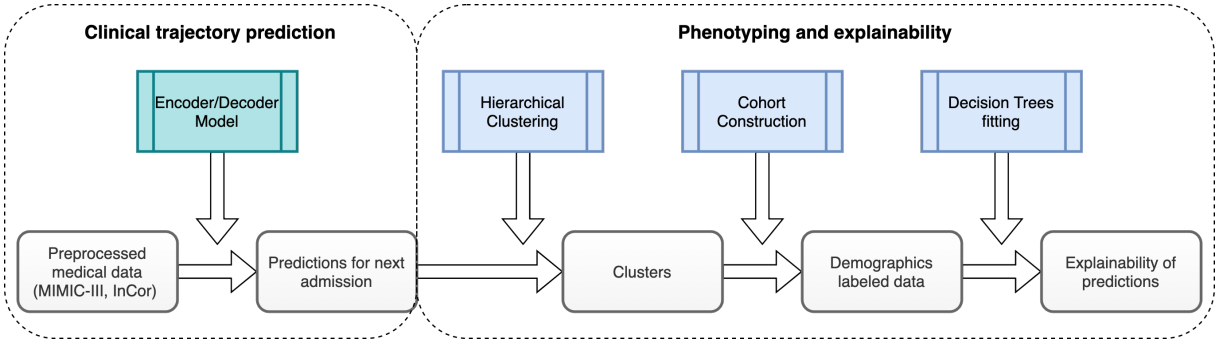

Figure 10 – Proposed framework.

demographic data of each patient. The last output of the explicability stage was the production of a decision tree with its respective visualization of decision rules.

4.4.1 Clinical trajectories prediction architecture: AttentionHCare

Clinical trajectories prediction corresponds to the first stage of the proposed framework. The challenge was the temporal recurrence of the input data, and, to learn from this aspect, we chose to use a recurrent neural network. More specifically, an Attentive Encoder-Decoder architecture composed of LSTM units. As far as we know, this architectural choice has not been explored yet for the longitudinal prediction of clinical trajectories.

This architecture, named by us *AttentionHCare*, also provided flexibility to the model, since it is a sequence-to-sequence model (or seq2seq), that is, given a sequence of inputs it is possible to predict an output sequence, which enables the prediction of diagnoses for $n > 1$ patients admissions. However, for comparative purposes with related works, we used $n = 1$ predictions for upcoming admissions and consider this improvement as future work.

AttentionHCare's architecture is presented in Figure 11. Its input corresponds to a multi-hot encoded tensor representing the presence or absence of a diagnosis in the current admission, according to a coding standard such as ICD-9 and for a batch of patients. The three-dimensional input tensors have dimensions (number of admissions $\times$ batch size $\times$ number of possible codes); they are passed through the Encoder, which is an n-layered LSTM recurrent neural network that outputs for each recurrent step, its hidden states together with a context vector that condenses the whole sequence (represented as Encoder States in the figure).

These states and outputs are input to a Monotonic Bahdanau Attention unit, whose purpose is to learn and capture the most relevant terms of the Enconder's states and outputs. Next, the Attention context vector is given as input to the Decoder layer with the ground truth tensor for decoding in a teacher-forcing manner. The Decoder structure is also composed of a n-layered LSTM network whose output is a tensor of hidden states that condenses the whole decoded sequence.

These states, which represent the long-term learning are given to a feed-forward layer

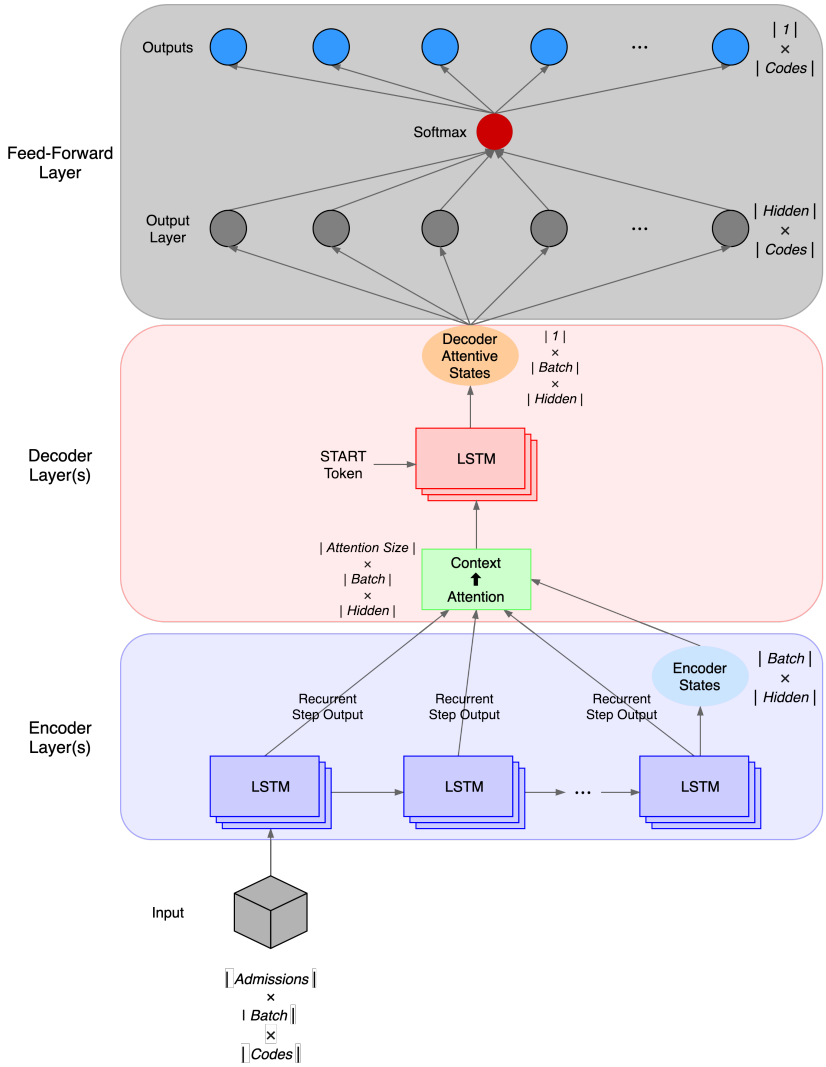

Figure 11 – *AttentionHCare* architecture.

that consists of a Leaky ReLU activated hidden layer and a Softmax activation for evaluation purposes. Equation (4.5) summarizes the model operations. For the input to the phenotyping architecture, the Softmax activation is removed aiming to produce the probabilities of each input code. In this sense, the last model output is a two-dimensional tensor of dimensions (batch size $\times$ number of codes in the adopted standard) describing the diagnostic probabilities of each code, for each patient.

$$\hat{y} = softmax(LReLU(w_{ff} * D_{h_{m-1}}(Att(E_{h_{m-1}}(x))) + b_{ff})) \tag{4.5}$$

Where w_{ff} and b_{ff} are the feed-forward weights and biases; Att is the Attention layer, whose energy function is given by Equation (2.7); and $D_{h_{m-1}}$ and $E_{h_{m-1}}$ are the hidden states of the LSTM networks given by Equations (2.1).

Our goal was to compute the following optimization of cross-entropy loss for the set of

model parameters θ_{EncDec}, multi-hot vector of target codes y, and vector of codes probabilities $\hat{y}$:

$$argmin_{\theta_{\text{EncDec}}}(Loss(y,\hat{y})) \tag{4.6a}$$

$$Loss(y,\hat{y}) = \sum_{j=0}^{|D|-1}(y_j log(\hat{y}_j) + (1-y_j) log(1-\hat{y}_j)) \tag{4.6b}$$

4.4.2 Phenotyping and explicability of clinical trajectories

This part of the framework, illustrated in Figure 12, corresponded to the second stage of the methodology. Here the purpose was to label trajectories predictions using a hierarchical clustering algorithm, to construct a cohort with demographic data, and to fit a decision tree to provide explainability of clinical predictions.

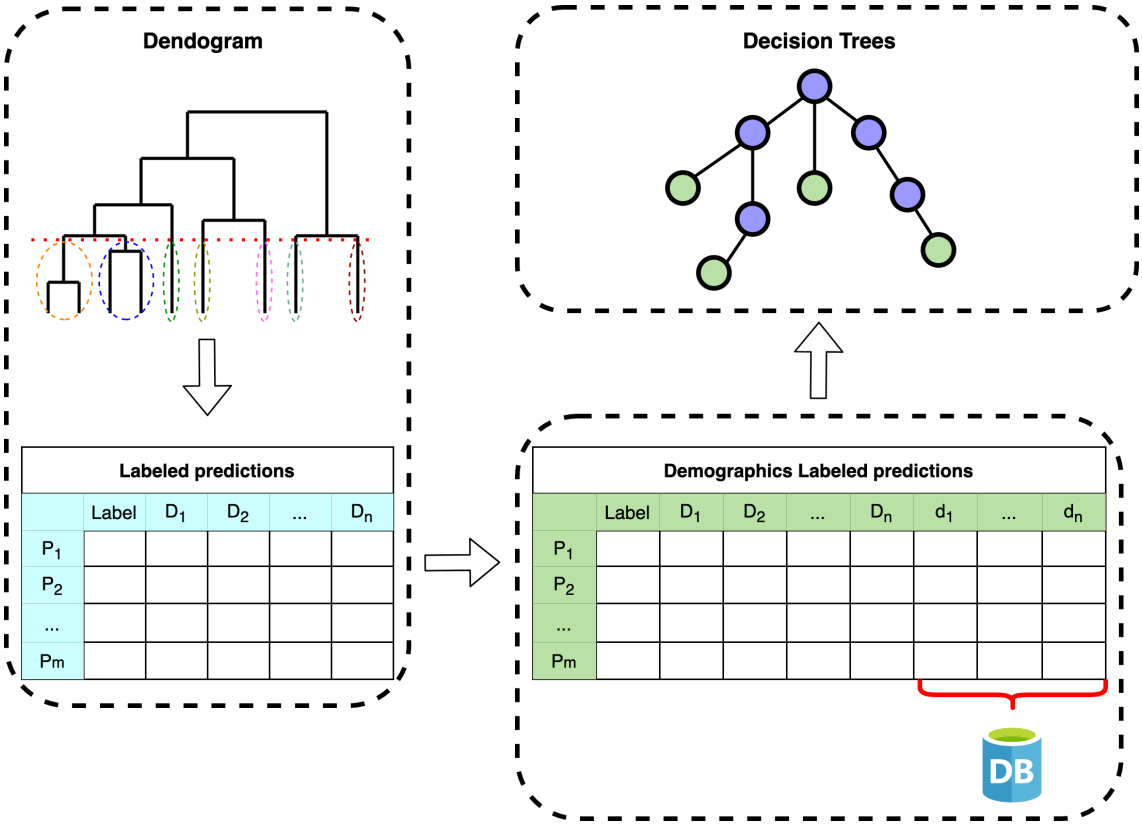

Figure 12 – Phenotyping and explicability architecture.

4.4.2.1 Hierarchical clustering

Given the predicted probabilities, the purpose of fitting a clustering algorithm was the production of similar data clusters as indicative of similar diagnosis groups, an example is: patients with a higher probability of heart diseases are expected to be closely related and

attributed to the same cluster, which becomes an indicator of patients with problems in the circulatory system.

When it comes to data clustering, a common decision is to use the K-means algorithm, and, despite methods that aim to help define the number of clusters, the clustering analysis is also based on the visualization of the obtained clusters. In this sense, we considered hierarchical clustering more suitable for analyzing the number of clusters, since it produces a dendrogram that can be visually inspected for the definition of parameters.

The clustering process is shown in Figure 13, with an example of a dendrogram produced after clustering the predictions. The cut was performed considering visual analysis, Silhouette score, and Davies-Bouldin index. With the cut, clusters numbered from 0 to $n-1$ were produced and, with the features vectors (*i.e.* the predictions) of each sample, these vectors were labeled with the cluster number to which it belongs.

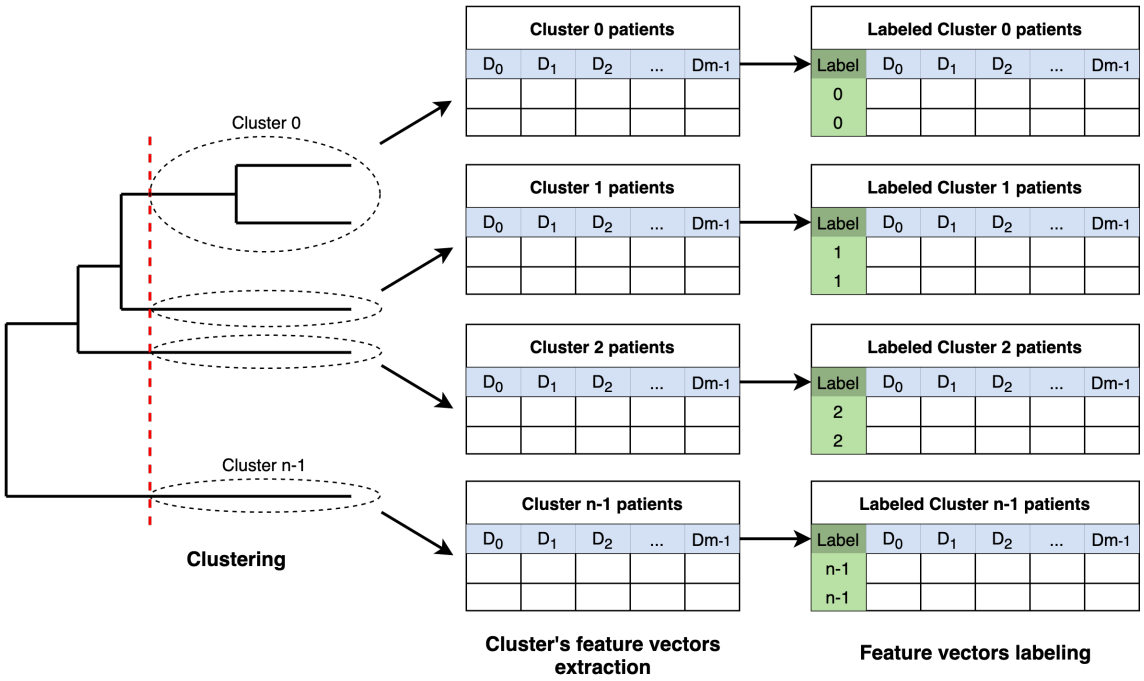

Figure 13 – Cluster number to label process.

4.4.2.2 Cohort representation

With labeled data, the cohort representation aimed to map the data to, or to be enriched with, demographic information relevant for patient metadata characterization. At the end of this processing sequence, our purpose was that these data contributed to the explainability of the decisions, *e.g.*, "patients with the probability of diagnosis x above y, age above k and medical plan type z were assigned to the disease group labeled as l".

We proposed two methods for cohort representation, as presented in Figures 14 and 15, respectively. Both methods are supposed to support the investigation of demographic features.

While the first one performed the concatenation with the probability matrix of the labeled diagnoses, the second one did the mapping between admissions and patients, ignoring the probabilities of diagnoses. In the case of MIMIC-III, the correct concatenation and mapping of demographic features were done using an unique identifier number for each patient.

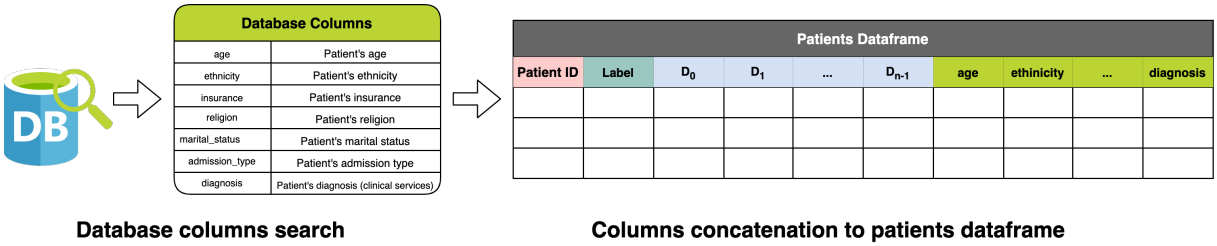

Figure 14 – Method 1 - Cohort representation (patients demographics concatenation).

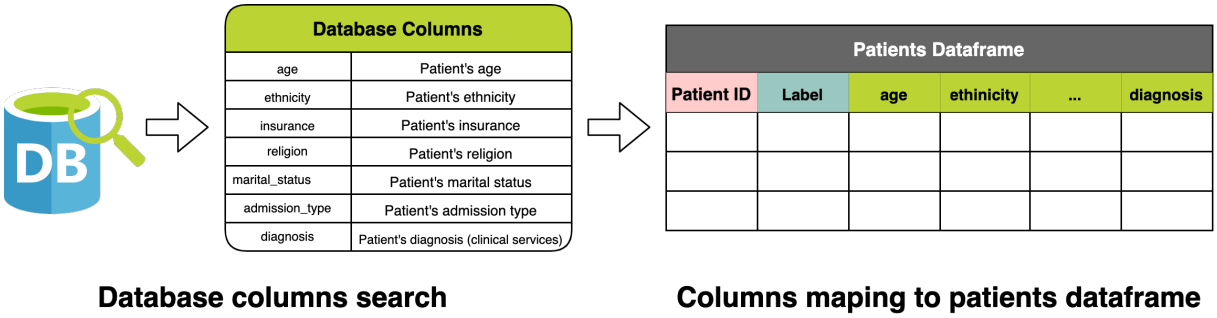

Figure 15 – Method 2 - Cohort representation (patients demographics mapping).

4.4.2.3 *Decision trees fitting*

The last step of our methodology was the fitting of a decision tree over the cohorts' representations. This decision tree aimed to provide explainability to the predictions made by the neural network model. After the tree fitting, our objective was to generate the visualization of the tree to identify the algorithmic decision rules and the importance of the features. With these rules, we expected to identify patients' habits, risk factors, and trends in the diagnoses through the traversal of the decision tree.

4.5 Validation

Each stage of the proposed methodology required its validation. For trajectories prediction, the validation occured with training and testing, followed by the computation of metrics Recall@k, with $k = \{10, 20, 30\}$, Precision@n, with $n = \{1, 2, 3\}$, AUC, and F1-score. With these metrics, we sought an overview of the prediction performance.

In the clustering step, the validation of the optimal number of clusters occured through the metrics of the Silhouette score and the Davies-Bouldin index, and through the projection

data in a two-dimensional space using the technique t-SNE visualization. The most appropriate linkage criteria for hierarchical clustering occured through dendrogram analysis.

The final results after the decision tree fitting were quantitatively validated using the metrics of accuracy, recall, precision, and F1-score; and qualitatively through the generated decision rules.

CHAPTER

5

RESULTS

This chapter reports the results obtained by each part of the proposed methodology over MIMIC-III and MIMIC-IV-ED datasets, as described in the previous Chapter 4. First, we report the results obtained by our model, then, we discuss the results that led us to our architectural decisions.

5.1 Setup

For the clinical trajectories prediction, we made the training and test split, respectively, with 90% and 10% of the patients, and we averaged the results of three model runs after randomization before each training and testing session. Also, we used early stopping with a tolerance of 10 consecutive validation cross-entropy iterations without improvements (loss reduction). The code was implemented in Python 3.7 over the framework Tensorflow 1.15.0. After training the model, we used the best model run to generate the predictions of 100% of the patients for the next framework parts.

For clustering, we used the Scipy 1.4.1 library methods “linkage” and “fcluster”, and the Scikit-learn 0.24.0 t-SNE for visualization. Finally, for fitting the decision trees, we used 10-fold cross-validation also obtained with the Scikit-learn 0.24.0 library, and the method graphviz for plotting. All the tests ran on a MacBook Pro Late 2013 with macOS Mojave 10.14.6, 8 GB of memory, and Intel Iris 1,536 MB graphics card.

5.2 Preprocessed input data

After the preprocessing step described in Section 4.3, our input data were transformed so that the diagnosis codes were arranged sequentially. Thus, for each dataset, we reached a number of distinct types of diagnoses, that is, labels to be predicted that were smaller than the raw number of diagnoses, as shown in Table 2, 272 codes for CCS, and 855 for ICD-9.

Table 2 – Preprocessed MIMIC-III and MIMIC-IV-ED comparison over the number of distinct codes, number of admissions per patient, and number of codes per admission.

	MIMIC-III	MIMIC-IV-ED
# of Distinct Codes	272 (CCS) 855 (ICD-9)	9,722
# of Admissions per Patient		
Min	2	2
Max	42	203
Mean	11.89	38.74
# of Codes per Admission		
Min	1	1
Max	39	9
Mean	16.39	4.2

5.3 Clinical trajectories prediction results

We compare our best prediction results over MIMIC-III and MIMIC-IV-ED datasets with the results presented by the related works; both the performance metrics and the preprocessing techniques were the same used in works LIG-Doctor (RODRIGUES-JR *et al.*, 2021) and APEHR (FLOREZ *et al.*, 2021). Table 3 shows these results for ICD-9 and CCS encodings of MIMIC-III, and ICD-9 with ICD-10 encodings of MIMIC-IV-ED.

Considering the MIMIC-III in both ICD-9 and CCS encodings, the best results of our proposed model (in bold) outperformed the baselines and related works. Concerning the MIMIC-IV-ED dataset, the proposed model also outperformed LSTM and GRU baseline models and some related works considering other large datasets indirectly comparable.

Our central assumption about the quality of these results is that they come from the use of the attention mechanism, and from the passage of long-term learning cell states, instead of hidden states, to the attention mechanism. The results obtained by these modifications were more evident in the MIMIC-IV-ED dataset, which has longer trajectories than the ones from MIMIC-III.

Understanding the model's learning behavior

Once trained, we used our model to predict diagnoses of a single patient in a real-time manner in order to better understand the model's recurrence learning. Given a three-dimensional input tensor $Z^{(a,1,d)}$, where a is the patient's number of admissions with d possible diagnosis codes, the next-admission (prediction) is an unidimensional tensor $A^{(n)}$, where n refers to the top n most probable diagnosis. Next, we compared its last admission with the model's predicted admission $t+1$.

Table 3 – Comparison between related works, baseline methods, and the proposed model *AttentionHCare*.

	Recall@10	Recall@20	Recall@30	Precision@1	Precision@2	Precision@3	AUC-ROC	F1-Score
Deepcare (PHAM *et al.*, 2017)								
Diabetes/Mental Datasets (1,369/1,318 codes)	-	-	-	0.66/0.52	0.59/0.46	0.53/0.40	-	-
Lipton et al. (LIPTON *et al.*, 2015)								
Children's Hospital LA Dataset (128 codes)	-	-	-	-	-	-	0.86/0.81	0.30/0.15
Rajkomar et al. (RAJKOMAR *et al.*, 2018)								
MIMIC-III ICD-9	-	-	-	-	-	-	-	0.40
UCSF/UCM (private - 14,025 codes)	-	-	-	-	-	-	0.90	-
DoctorAI (CHOI *et al.*, 2016a)								
MIMIC-III ICD-9 (767 codes)	-	-	0.64	-	-	-	-	-
MIMIC-III CCS (283 codes)	0.44	0.62	0.72	-	-	-	-	-
Sutter Health (private - 1,778 codes)	0.64	0.74	0.80	-	-	-	-	-
LIG-Doctor (RODRIGUES-JR *et al.*, 2021)								
MIMIC-III ICD-9 (855 codes)	0.48	0.64	0.72	0.78	0.74	0.70	0.95	0.48
MIMIC-III CCS (272 codes)	0.53	0.71	0.79	0.81	0.78	0.74	0.93	0.54
InCor (private - 3,133 codes)	0.47	0.56	0.61	0.21	0.17	0.14	0.94	0.23
APEHR (FLOREZ *et al.*, 2021)								
MIMIC-III ICD-9 (855 codes)	0.45	0.62	0.70	0.78	0.76	0.73	0.95	-
MIMIC-III CCS (272 codes)	0.53	0.70	0.79	0.81	0.78	0.76	0.94	-
InCor (private - 3,133 codes)	0.71	0.75	0.77	0.65	0.40	0.25	0.97	-
GRU								
MIMIC-III ICD-9 (855 codes)	0.47	0.62	0.71	0.77	0.73	0.70	0.95	0.52
MIMIC-III CCS (272 codes)	0.51	0.69	0.78	0.79	0.76	0.74	0.93	0.56
MIMIC-IV-ED (9,722 codes)	0.47	0.56	0.61	0.31	0.25	0.21	0.97	0.28
LSTM								
MIMIC-III ICD-9 (855 codes)	0.45	0.61	0.70	0.77	0.72	0.69	0.95	0.50
MIMIC-III CCS (272 codes)	0.50	0.68	0.77	0.80	0.77	0.74	0.93	0.55
MIMIC-IV-ED (9,722 codes)	0.48	0.56	0.61	0.31	0.25	0.21	0.93	0.28
AttentionHCare								
MIMIC-III ICD-9 (855 codes)	**0.49**	**0.65**	**0.74**	**0.80**	**0.77**	**0.74**	**0.96**	**0.55**
MIMIC-III CCS (272 codes)	**0.54**	**0.72**	**0.80**	**0.82**	**0.79**	**0.77**	**0.94**	**0.59**
MIMIC-IV-ED (9,722 codes)	**0.56**	**0.64**	**0.68**	**0.35**	**0.29**	**0.24**	**0.97**	**0.33**

We selected a patient having 9 admissions with a number of diagnoses varying between 9 and 20. We ran *AttentionHCare* trained with MIMIC-III standard ICD-9 and adjusted it to return the top-20 most probable diagnoses. We also computed the frequency according to which each code appeared in the past admissions – right-most column of Table 4. As seen in the table, for example, "Diabetes mellitus" has a 0.78 probability, indicating that it has been observed in 78% of the past admissions of our random patient. Still in the table, we present the ranked most-probable diagnoses as predicted by our model – second column "Probability ranking". By comparing columns "Probability ranking" and "Past admissions' frequency", it is possible to observe that the model was able to learn what were the most frequent codes (top-ranked by the model); the model also learned the codes that, while not being too frequent, were predicted due to the more complex learning implicit to the neural network as, for example, "Septicemia", not so frequent as "Complications..." and "Nephritis...", but ranked above by the model.

Furthermore, given the history of the patient's conditions (Past admissions' frequency), we could observe previous conditions related to kidney ("Hypertensive chronic kidney disease", "Chronic kidney disease", "Disorders resulting from impaired renal function" and "Nephritis and nephropathy not specified as acute as chronic"), and to diabetes ("Diabetes mellitus" and "Polyneuropathy in diabetes"). Given these conditions, the model was able to predict two new

Table 4 – Diagnoses predicted for a random patient. Predicted diagnoses in comparison with the frequency of diagnoses found in past admissions.

ICD-9	Probability ranking	Predicted diagnoses	Past admissions' frequency
403	1	Hypertensive chronic kidney disease	1.0
585	2	Chronic kidney disease	0.78
428	3	Heart failure	0.89
285	4	Other and unspecified anemias	0.89
250	5	Diabetes mellitus	0.78
276	6	Disorders of fluid electrolyte and acid-base balance	0.44
311	7	Depressive disorder, not elsewhere classified	0.44
038	8	Septicemia	0.33
995	9	Certain adverse effects not elsewhere classified	0.33
996	10	Complications peculiar to certain specified procedures	0.56
V12	11	Personal history of certain other diseases	0.22
583	12	Nephritis and nephropathy not specified as acute or chronic	0.44
V58	13	Encounter for other and unspecified procedures and aftercare	0.0
588	14	Disorders resulting from impaired renal function	0.11
357	15	Polyneuropathy in diabetes	0.56
E870	16	Accidental cut puncture perforation or hemorrhage during medical care	0.11
416	17	Chronic pulmonary heart disease	0.0
458	18	Hypotension	0.33
286	19	Coagulation defects	0.0
041	20	Bacterial infection in conditions classified elsewhere and of unspecified site	0.11

diagnoses related to these past conditions: "Disorders resulting from impaired renal function" and "Coagulation defects" that, despite being uncommon, appeared as future conditions.

In Figure 16, we present the intersection of codes between the last admission (left) and the predicted admission (right). The last admission is composed of 14 diagnoses, mostly related to kidney diseases and infections. From these 14 diagnoses, our model predicted 7 of them to appear in a future admission. The model also included related diagnoses "038 - Septicemia", and "041 - Bacterial infection in conditions classified elsewhere and of unspecified site", which may indicate a worsening of the patient's condition; "996 - Complications peculiar to certain specified procedures", and "V58 - Encounter for other and unspecified procedures and aftercare" related to last admission's diagnosis "E87 - Accidental cut puncture perforation or hemorrhage during medical care"; and other kidney issues: "583 - Nephritis and nephropathy not specified as acute or chronic", and "588 - Disorders resulting from impaired renal function".

Discussion

Foremost, *AttentionHCare* surpassed both baseline models, such as GRUs and LSTMs networks, and related works, such as DoctorAI, LIG-Doctor, and APEHR. Our comparison was based on the metrics of Recall@10, Recall@20, Recall@30, Precision@1, Precision@2, Precision@3, AUC-ROC, and F1-Score; and we reproduced the related works experiments when ever possible [1]. For this reason, the works Deepcare of Pham et al. (PHAM *et al.*, 2017), Lipton et al. (LIPTON *et al.*, 2015), Rajkomar et al. (RAJKOMAR *et al.*, 2018) and DoctorAI of Choi et al. (CHOI *et al.*, 2016a) were less straightforwardly comparable to our model than the works LIG-Doctor of Rodrigues-Jr et al. (RODRIGUES-JR *et al.*, 2021), APEHR of Florez

[1] When not possible we compared to the authors' published results

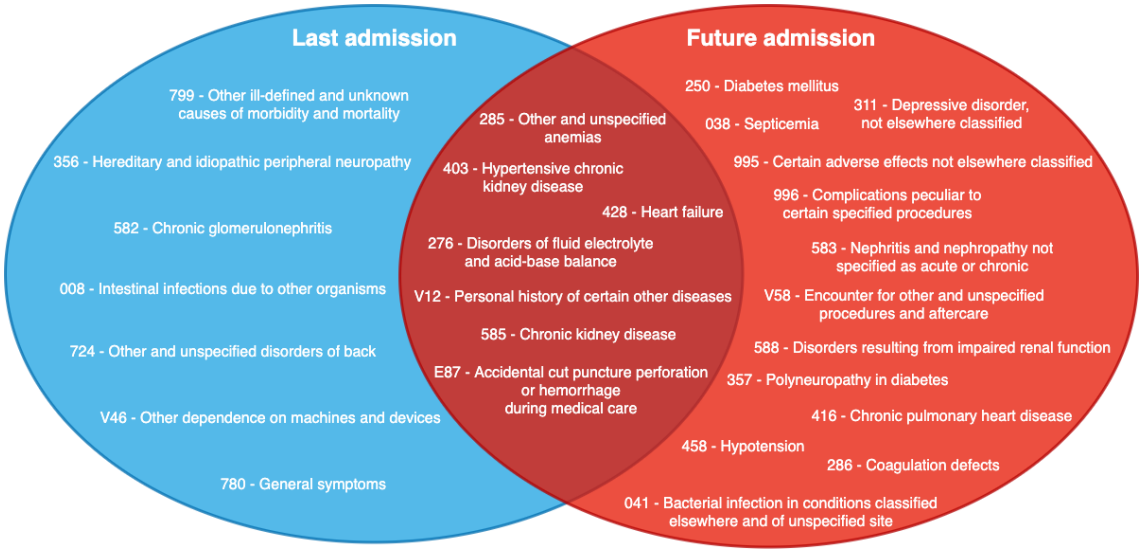

Figure 16 – Venn diagram of last admission and future admission diagnoses of a randomly selected patient.

et al. (FLOREZ *et al.*, 2021) and the baseline models of GRU and LSTM which used the same preprocessing steps and number of predicted codes.

In comparison with the baseline models, the proposed model obtained better results for both datasets considering all the metrics: the improvement of the proposed model over MIMIC-III using ICD-9 and CCS encodings was about 2%to 3% for Recall and Precision metrics and about 3% to 5% for F1-Score – the lower differences were in comparison with the GRU network. In the comparison over dataset MIMIC-IV-ED, the prediction improvement was pronouncedly upper: about 8% for Recall metrics, 4% for Precision metrics, and 5% for the F1-Score metric. We consider that, since MIMIC-IV-ED is more extensive and has more diagnosis codes, and longer trajectories than MIMIC-III, the results over this dataset provide significant indications regarding the improvements obtained with the Encoder-Decoder's attention mechanism.

When compared with related works, the proposed model also obtained better results: Pham et al. (DeepCare) report that their best Precision@1, @2, and @3 were 0.66, 0.59, and 0.53, respectively, while our best results were 0.80, 0.78, and 0.75. Compared with dataset MIMIC-IV-ED, we obtained better results for metric AUC-ROC; meanwhile, MIMIC-IV-ED is more challenging, as it has more diagnosis codes than the Diabetes/Mental datasets employed by Pham et al.; it has 9,722 diagnoses while the Diabetes/Mental datasets have 243 and 247 diagnoses respectively. Compared with Rajkomar et al. and Choi et al. (DoctorAI), our model obtained better results considering all the metrics for MIMIC-III ICD-9 and CCS encodings, Rajkomar et al. report that their best results for AUC-ROC and F1-Score were 0.90 over the UCSF/UCM dataset and 0.40 over the MIMIC-III ICD-9 dataset; in comparison, our best results were 0.96 and 0.52 for the same datasets and metrics. In comparison to Choi et al. (DoctorAI), our results considering metrics Recall@10, @20, and @30 were superior to DoctorAI over datasets MIMIC-III CCS and ICD-9, but inferior in comparison to dataset Sutter Health, which is private,

however, this dataset contains 14 million admissions and 1,183 codes, while the MIMIC-III ICD-9 has 58,976 admissions and 855 diagnoses; that is, the results are not comparable. Compared with Florez et al. and Rodrigues-Jr et al. (LIG-Doctor), which are straightly comparable to our work, our model obtained superior results for all the metrics considering datasets MIMIC-III ICD-9 and CCS.

Further discussions about this model and results are presented on our published paper *AttentionHCare: Advances on computer-aided medical prognosis using attention-based neural networks* (BARROS; RODRIGUES, 2022).

5.4 Phenotyping results

Given the prediction results, that is, the diagnosis probabilities of each patient in their next admission, we describe the results of clustering and t-SNE projecting these predictions to obtain medical phenotypes. To obtain these phenotypes, illustrated in Figures 17 and 18, and shown in Tables 7 and 5, we first calculated the patients' accumulated probabilities of diagnoses for each cluster in comparison to the general probabilities of the patients' diagnoses ρ_j^c as described in Equation 5.1; where n is the number of patients, $|\mathbb{D}|$ is the number of distinct predicted diagnoses, p_j is the predicted probability of diagnosis j for patient i and $p_{i,j}^c$ is the predicted probability of diagnosis j for patient i from cluster c, which is a value between 0 and k (the maximum cluster number). By dividing the sum of probabilities of grouped clusters by the overall sum of probabilities, we avoid common diagnoses to all the clusters in the most probable diagnosis per cluster.

$$\rho_j^c = \frac{\sum_{i=1}^{n} p_{i,j}^c}{\sum_{i=1}^{n} p_{i,j}}, \quad 0 \leq c \leq k \text{ and } 0 \leq j \leq (|\mathbb{D}| - 1) \tag{5.1}$$

Lastly, for each cluster c of ρ_j^c, $0 \leq c \leq k$, we selected the top 3 most significant values of ρ_j, which represent, for each cluster, the top 3 most probable diagnoses penalizing the ones that are common for all the clusters.

Table 5 – Top 3 diagnoses for each MIMIC-III ICD-9 phenotyping.

Cluster (Phenotype)	Top-1 Diagnosis	Top-2 Diagnosis	Top-3 Diagnosis
0	Hypertensive chronic kidney disease	Chronic kidney disease (ckd)	Disorders resulting from impaired renal function
1	Angina pectoris	Old myocardial infarction	Other forms of chronic ischemic heart disease
2	Diffuse diseases of connective tissue	Other diseases of respiratory system	Chronic bronchitis
3	Inflammatory and toxic neuropathy	Diabetes mellitus	Arthropathy associated with other disorders classified elsewhere
4	Cardiomyopathy	Diseases of other endocardial structures	Fitting and adjustment of other device
5	Suicide and self-inflicted poisoning by solid or liquid substances	Multiple sclerosis	Late effects of injuries to the nervous system
6	Malignant neoplasm of brain	Hemiplegia and hemiparesis	Epilepsy and recurrent seizures
7	Other perinatal jaundice	Endocrine and metabolic disturbances specific to the fetus and newborn	Observation and evaluation of newborns for suspected conditions not found
8	Secondary malignant neoplasm of other specified sites	Secondary malignant neoplasm of respiratory and digestive systems	Malignant neoplasm of female breast
9	Observation and evaluation of newborns for suspected conditions not found	Need for other prophylactic vaccination and inoculation against single diseases	Conditions involving the integument and temperature regulation of fetus and newborn
10	Epilepsy and recurrent seizures	Unspecified intellectual disabilities	Infantile cerebral palsy
11	Aortic aneurysm and dissection	Other aneurysm	Arterial embolism and thrombosis
12	Other rheumatic heart disease	Diseases of mitral and aortic valves	Organ or tissue replaced by other means
13	Acute myocardial infarction	Atherosclerosis	Diabetes mellitus
14	Diseases of esophagus	Disorders of lipoid metabolism	Occlusion and stenosis of precerebral arteries
15	Other and ill-defined cerebrovascular disease	Subarachnoid hemorrhage	Migraine
16	Asthma	Chronic sinusitis	Other diseases of upper respiratory tract
17	Varicose veins of other sites	Liver abscess and sequelae of chronic liver disease	Chronic liver disease and cirrhosis
18	Malignant neoplasm of trachea bronchus and lung	Secondary and unspecified malignant neoplasm of lymph nodes	Secondary malignant neoplasm of other specified sites
19	Abscess of anal and rectal regions	Other appendicitis	Other arthropod-borne diseases
20	Acquired hypothyroidism	Lymphoid leukemia	Disorders of the pituitary gland and its hypothalamic control
21	Other and ill-defined cerebrovascular disease	Subarachnoid hemorrhage	Strabismus and other disorders of binocular eye movements
22	Hematological disorders of newborn	Other respiratory conditions of fetus and newborn	Other and ill-defined conditions originating in the perinatal period
23	Asymptomatic human immunodeficiency virus [HIV] infection status	Viral hepatitis	Drug dependence
24	Alcohol-induced mental disorders	Housing household and economic circumstances	Alcohol dependence syndrome
25	Personal history of malignant neoplasm	Other specified personal exposures and history presenting hazards to health	Secondary and unspecified malignant neoplasm of lymph nodes

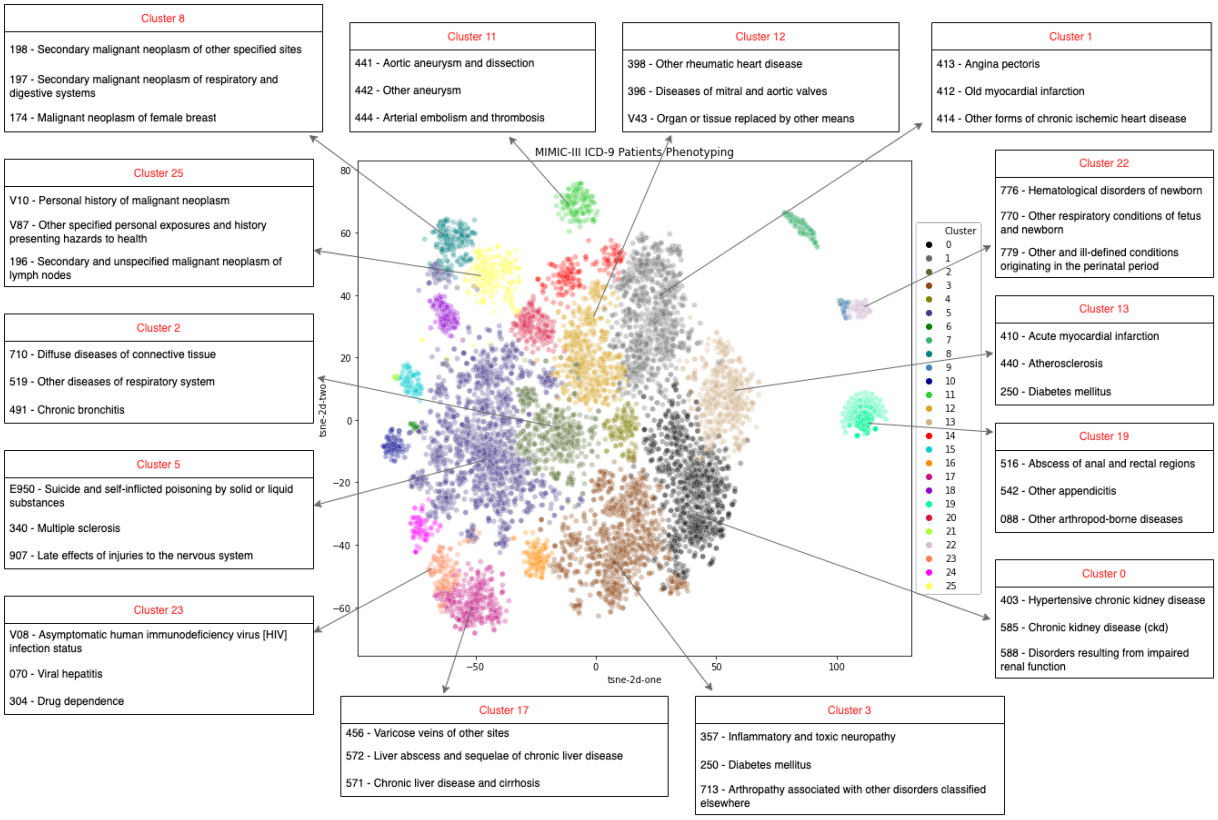

Figure 17 – MIMIC-III ICD-9 patients phenotyping.

We also calculated the Kullback–Leibler divergence (CSISZAR, 1975) after 250 and 5000 iterations of t-SNE algorithm in order to measure the projection results, the Kullback–Leibler divergence is a type of statistical distance commonly used to measure the difference between two probability distributions. The results are shown in Table 6.

Table 6 – Comparison of Kullback–Leibler divergence after 250 and 5000 iterations of t-SNE algorithm.

Iterations	250	5000
MIMIC-III ICD-9	82.903908	1.816837
MIMIC-III CCS	81.494095	1.875908

Observing the Figures and Tables, we can see that each cluster (*i.e.* phenotype) represents a set of more or less related conditions. Diagnoses of phenotypes of ICD-9 encoding, presented in Figure 17, for example, are more related to each other than the ones presented in Figure 18 of CCS encoding.

Also, some phenotypes of related conditions are spatially close, that is the case of clusters 1 and 12 in Figure 17, which describe heart-related conditions, and clusters 1 and 2 of Figure 18, which are related to alcohol mental disorders and disorders of the biliary tract.

As seen in the reported results, our clustering method produced consistent phenotypes, especially in the ICD-9 standard data. These phenotypes captured patients with related conditions,

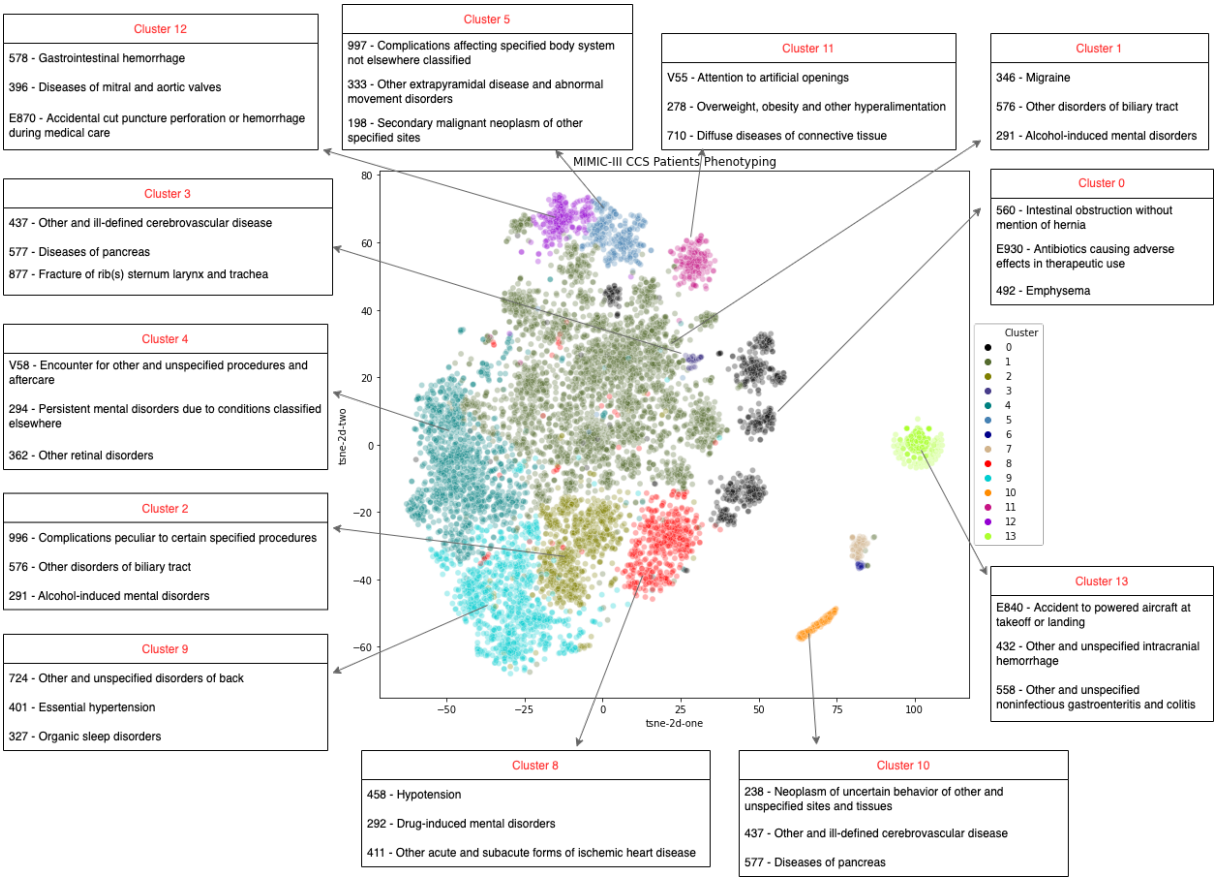

Figure 18 – MIMIC-III CCS patients phenotyping.

Table 7 – Top 3 diagnoses for each MIMIC-III CCS phenotyping.

Cluster (Phenotype)	Top-1 Diagnosis	Top-2 Diagnosis	Top-3 Diagnosis
0	Intestinal obstruction without mention of hernia	Antibiotics causing adverse effects in therapeutic use	Emphysema
1	Migraine	Other disorders of biliary tract	Alcohol-induced mental disorders
2	Complications peculiar to certain specified procedures	Chronic pulmonary heart disease	Viral hepatitis
3	Occlusion and stenosis of precerebral arteries	Other hernia of abdominal cavity without mention of obstruction or gangrene	Other symptoms involving abdomen and pelvis
4	Encounter for other and unspecified procedures and aftercare	Persistent mental disorders due to conditions classified elsewhere	Other retinal disorders
5	Complications affecting specified body system not elsewhere classified	Other extrapyramidal disease and abnormal movement disorders	Secondary malignant neoplasm of other specified sites
6	Other and ill-defined cerebrovascular disease	Diseases of pancreas	Fracture of rib(s) sternum larynx and trachea
7	Nephritis and nephropathy not specified as acute or chronic	Diseases of pancreas	Disorders of the autonomic nervous system
8	Hypotension	Drug-induced mental disorders	Other acute and subacute forms of ischemic heart disease
9	Other and unspecified disorders of back	Essential hypertension	Organic sleep disorders
10	Neoplasm of uncertain behavior of other and unspecified sites and tissues	Other and ill-defined cerebrovascular disease	Diseases of pancreas
11	Attention to artificial openings	Overweight, obesity and other hyperalimentation	Diffuse diseases of connective tissue
12	Gastrointestinal hemorrhage	Diseases of mitral and aortic valves	Accidental cut puncture perforation or hemorrhage during medical care
13	Accident to powered aircraft at takeoff or landing	Other and unspecified intracranial hemorrhage	Other and unspecified noninfectious gastroenteritis and colitis

placing them in the same phenotype; also, the clusters were closer to other phenotypes that represented other similar conditions.

Discussion

The phenotyping experiments were performed with our model predictions over the MIMIC-III dataset in both CCS and ICD-9 encodings. The importance of these results was to group and label patients with similar clinical trajectories to better understand which were the most relevant diagnoses for each group.

The main challenge was to choose the best clustering method and the best number of clusters for each one of the encodings. After exploring some methods and the number of clusters

we employed an agglomerative hierarchical clustering algorithm using the ward criteria and projected the clustering results in two dimensions with the t-SNE algorithm.

In comparison with the MIMIC-III CCS phenotyping experiments, reported in Figure 18 and Table 7, the MIMIC-III ICD-9 phenotyping results, reported in Figure 17 and Table 5, presented more consistent phenotypes, that is, phenotypes whose top-3 most relevant diagnoses were more related with each other from a medical point of view; for example, the phenotypes 0, 8 and 24 are related to renal issues, neoplasms and alcohol abuse issues, respectively. Also, we could see that similar, or related, patients from phenotypes were projected closely in Figure 17; for example, the phenotypes 17, 23, and 24 corresponded to liver diseases caused by alcohol abuse, sexually transmitted diseases, and alcohol abuse only, respectively; and the phenotypes 9 and 22 that represented newborn related issues.

On the other hand, the diagnoses from phenotypes obtained over the MIMIC-III CCS encoding were less related to each other; here we argue that our model predictions over this less granular type of encoding contributed to producing less separable data points on clusters, what is visible when we compare Figures 34 and 35.

However, in Figure 18, we could see relationships between some phenotypes; for example, both phenotypes 1 and 3 presented pancreas-related issues (576 - Other disorders of billiary tract in Cluster 1 and 577 - Diseases of pancreas in Cluster 3), and these phenotypes were visually close. Also, two of the top 3 diagnoses from phenotype 12 presented a relationship with each other about hemorrhage issues (578 - Gastrointestinal hemorrhage and E870 - Accidental cut puncture perforation of hemorrhage during medical care).

5.5 Cohort representation and decision trees' fitting

Next, we proceed with the two methods of cohort representation and fitting decision trees to obtain interpretable decision rules about the diagnoses categorized by cohorts: either by diagnoses enriched with demographic features or by demographic features only. As described in Section 2.3, fitting a decision tree on clustering results by treating each cluster as a label is a technique to obtain interpretable decision rules based on the clustering algorithm groupings.

These demographic data include the type of admission, location of admission, discharge location, insurance, spoken language, religion, marital status, ethnicity, gender, date of birth, date of death (if exists), date of death as recorded in the hospital (if exists), and date of death from social security (if exists).

First of all, we evaluated the metrics of accuracy, weighted recall, weighted precision, and weighted F1 score. Since this stage lacks related works for comparison, we considered a dummy classifier that predicted the most frequent class and a Naive Bayes classifier, to evaluate our decision tree results. Tables 8 and 9 present results related to the fitting of a decision tree

with balanced class weight, minimum samples split, and minimum samples leaf set to 50, to methods of demographic mapping and demographic concatenation, respectively. After some exploring, we chose these hyperparameters because they were a middle ground between the evaluation metrics performance and the tree interpretability: trees with better evaluation metrics were larger and less interpretable, and vice versa.

Table 8 – Comparison of Decision Trees evaluation, Dummy Classifier (most frequent) and and Naive Bayes over a constructed cohort of demographic features only.

Model	Accuracy	Recall (weighted)	Precision (weighted)	F1-Score (weighted)
MIMIC-III CCS				
Decision Tree	0.15	0.15	0.28	0.13
Dummy classifier (Most Frequent)	0.34	0.34	0.11	0.17
Naive Bayes	0.30	0.30	0.22	0.21
MIMIC-III ICD-9				
Decision Tree	0.09	0.09	0.17	0.09
Dummy classifier (Most Frequent)	0.17	0.17	0.03	0.05
Naive Bayes	0.16	0.16	0.11	0.11

Table 9 – Comparison of Decision Trees evaluation, Dummy Classifier (most frequent) and Naive Bayes over a constructed cohort of concatenating diagnoses and demographic features.

Model	Accuracy	Recall (weighted)	Precision (weighted)	F1-Score (weighted)
MIMIC-III CCS				
Decision Tree	0.81	0.81	0.84	0.82
Dummy classifier (Most Frequent)	0.33	0.33	0.11	0.17
Naive Bayes	0.30	0.30	0.27	0.25
MIMIC-III ICD-9				
Decision Tree	0.74	0.74	0.78	0.74
Dummy classifier (Most Frequent)	0.17	0.17	0.03	0.05
Naive Bayes	0.11	0.11	0.16	0.09

Since the method of demographic mapping did not performed well, and obtained worse results than both the baseline methods of Dummy and Naive Bayes classifiers, we chose the method of concatenating demographic features to obtain decision rules such as the ones presented in Figures 19, 20, 21 and Figures 22, 23 and 24 respectively. Here we highlight that the "class" attribute illustrated in these Figures corresponds to the phenotypes from the previous clustering results. Notice that the colors attributed for each tree node do not match the ones used in Figures 18 and 17, this is because these colors are automatically selected by the Scikit-learn graphviz library.

In Figure 19, we can see the main diagnosis that differentiated phenotypes 2 and 10 is the high probability (over 0.392) of "Neoplasm of uncertain behavior of other and unspecified sites and tissues", which is the top-1 diagnosis for this phenotype according to Table 7.

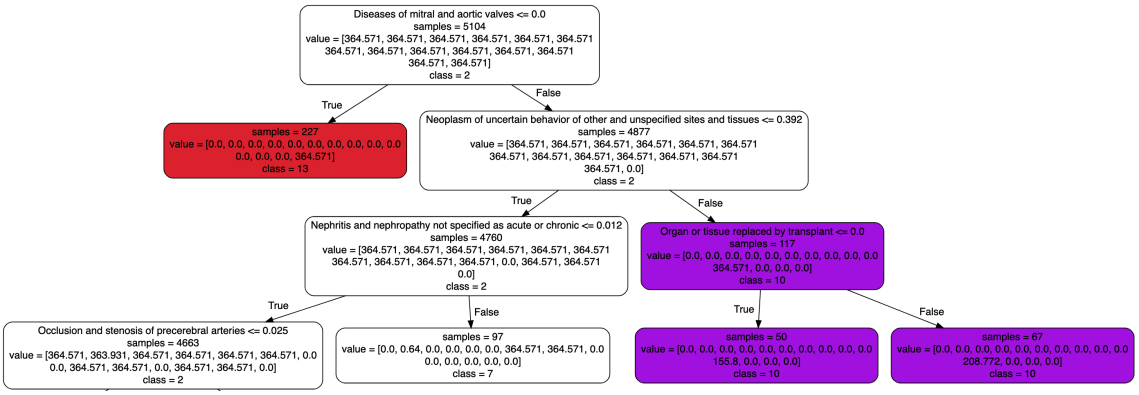

Figure 19 – Sample of decision tree fitted over MIMIC-III CCS showing decision rules for classes 2, 7, 10, and 13.

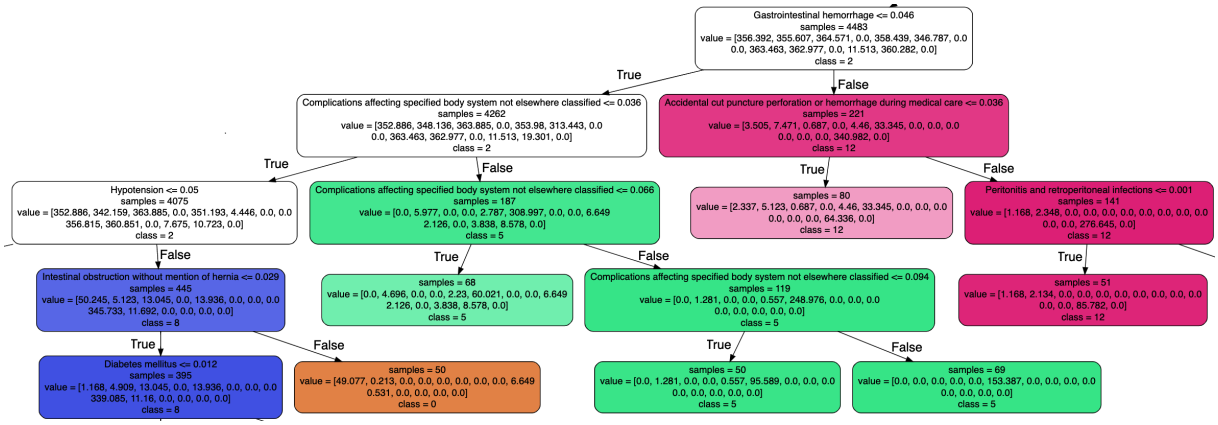

Figure 20 – Sample of decision tree fitted over MIMIC-III CCS showing decision rules for classes 0, 2, 5, 8, and 12.

Figure 20 shows, for the same phenotype 2, which were the determinant diagnoses for classification between phenotypes 12, 5, and 8. For phenotype 12, it was a probability over 0.046 of "Gastrointestinal hemorrhage"; for phenotype 5, it was a probability over 0.036 of "Complications affecting specified body system not elsewhere classified"; and for phenotypes 8, it was a probability over 0.05 of "Hypotension". Furthermore, one can see the relationships between diagnoses of the same class; for example, all the diagnoses from class 12 are hemorrhage related, as well as the top diagnoses of phenotype 12.

Figure 21 shows a chain of decision rules that distinguished diagnoses between phenotypes 1, 4, 0 and 2. The main point of this Figure is that the chain of diagnoses with low probabilities starting with "Encounter for other and unspecified procedures and aftercare" and ending with "Diffuse diseases of connective tissue" assign patients to phenotype number 1. This way, if patients have already shown probabilities below 0.049 for the first decision rule "Encounter for other and unspecified procedures and aftercare", these decision rules showed that there is already enough information to assign them to phenotype number 1, and the other rules only served as a confirmation.

Figure 22, shows decision rules for the classification of phenotypes 4, 7, 14 and 16 on

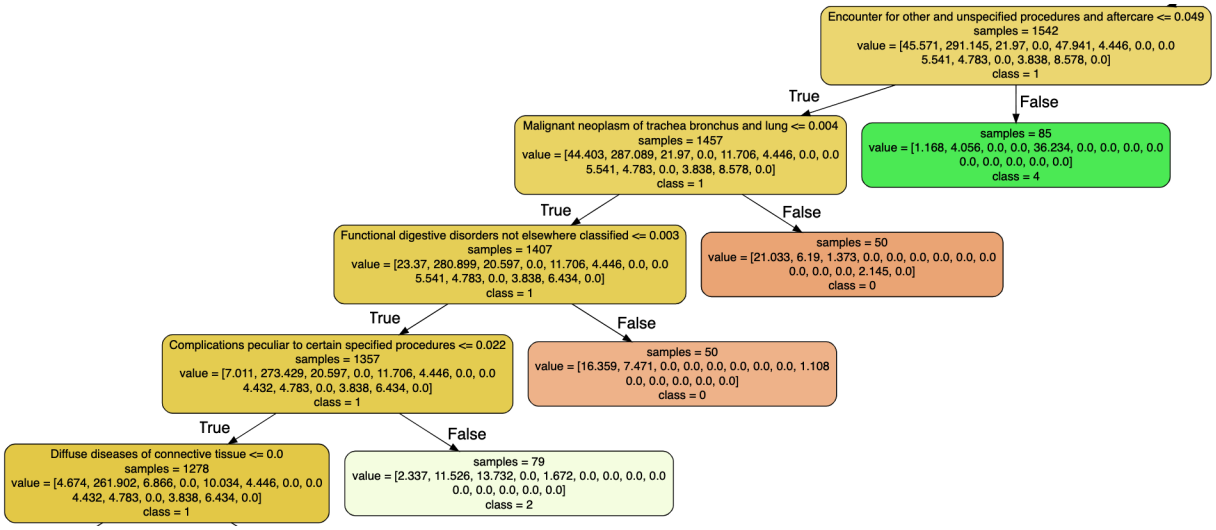

Figure 21 – Sample of decision tree fitted over MIMIC-III CCS showing decision rules for classes 0, 1, 2, and 4.

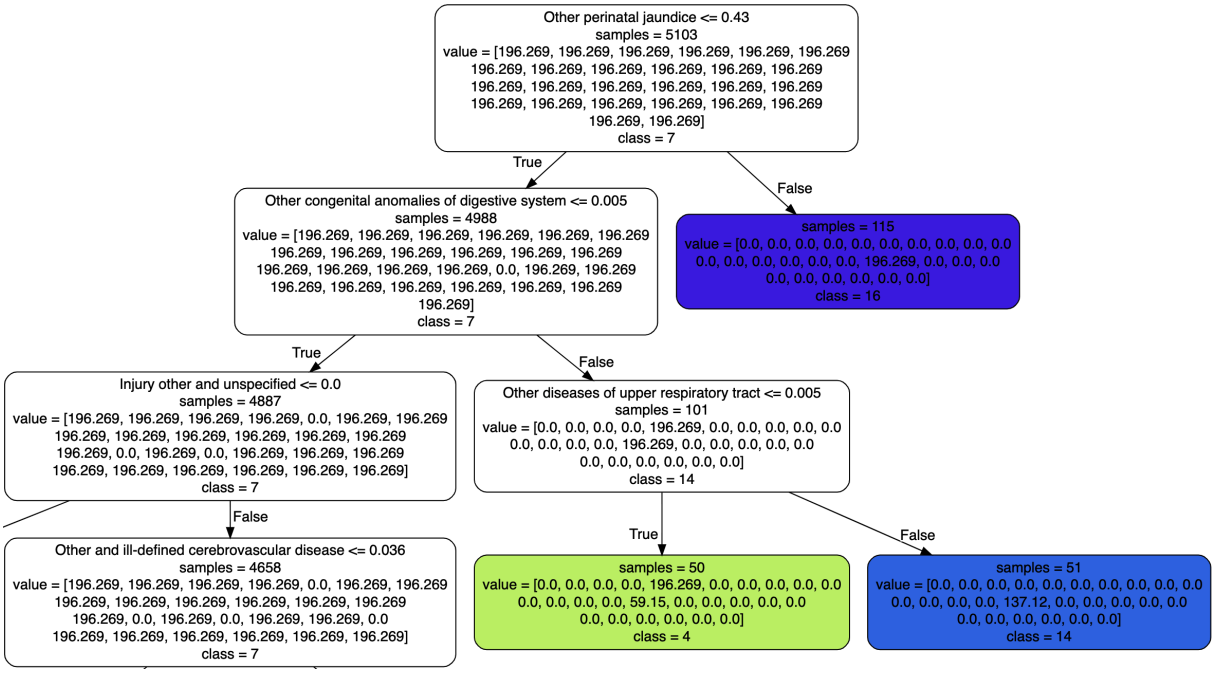

Figure 22 – Sample of decision tree fitted over MIMIC-III ICD-9 showing decision rules for classes 4, 7, 14, and 16.

MIMIC-III ICD-9. Firstly, a probability below 0.43 distinguished between phenotypes 7 and 16. However, for patients initially assigned as belonging to phenotype 7, if they also presented a high (over 0.005) probability for “Other congenital anomalies of digestive system” they were more specifically assigned to phenotype 14. The same occurs for the patients assigned to phenotype 14 if they also presented a low probability (below or equal to 0.005) of “Other diseases of upper respiratory tract”, and they were more specifically assigned to phenotype 4. This example shows one more time how a combination of nested diagnoses was able to change a previously assigned phenotype.

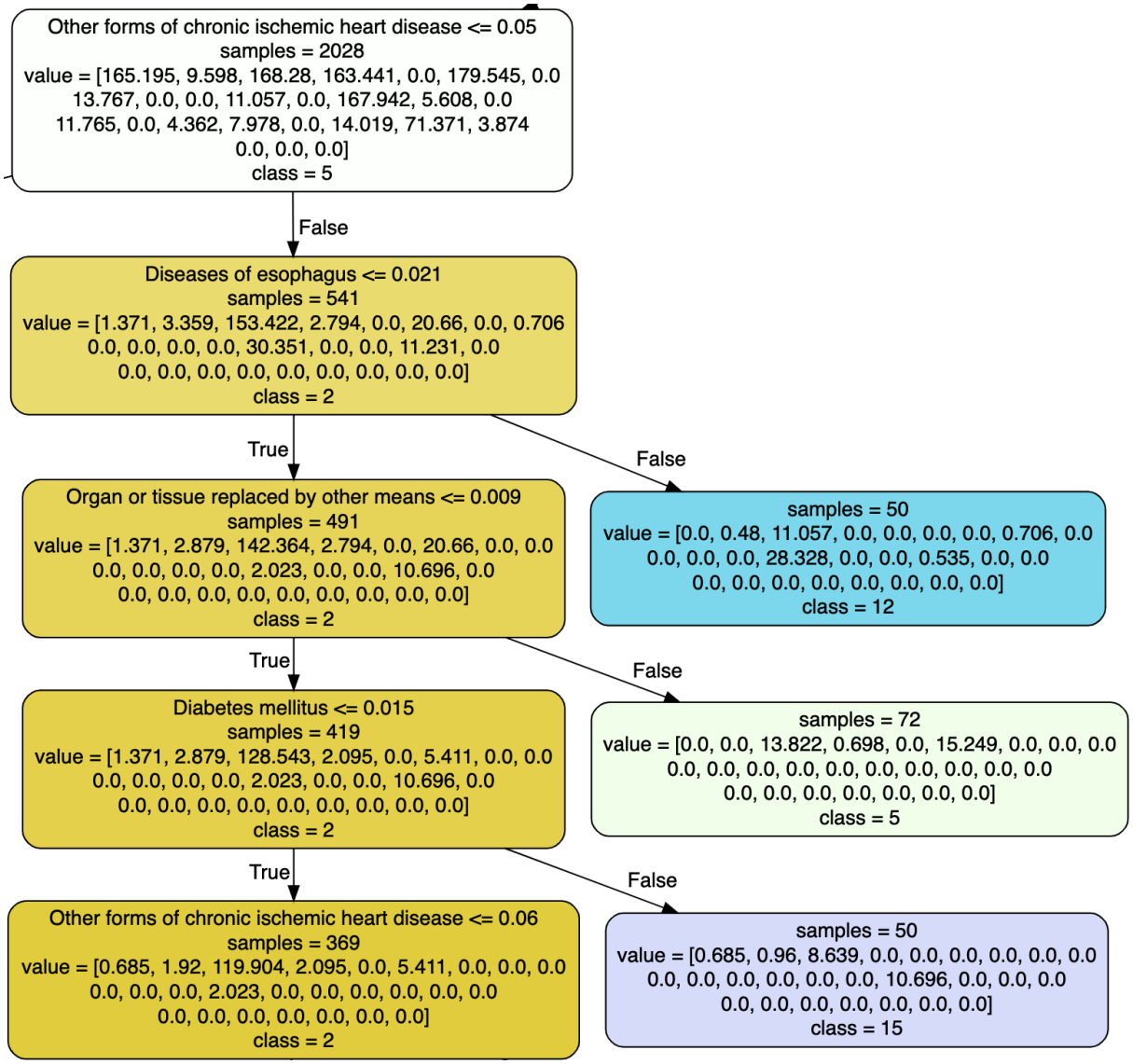

Figure 23 – Sample of decision tree fitted over MIMIC-III ICD-9 showing decision rules for classes 2, 5, 12, and 15.

Figure 23 shows a more straightforward chain of decision rules than the ones from the previous Figure. These decision rules assigned patients between phenotypes 2, 5, 12, and 15: After patients presented a high probability (over 0.05) of "Other forms of chronic ischemic heart disease" and were assigned to phenotype 2, a nested chain of high probabilities (over 0.021, 0.009 and 0.015, respectively) for diagnoses "Diseases of esophagus", "Organ or tissue replaced by other means" and "Diabetes mellitus" distinguished assignments of patients to phenotypes 12, 5 and 15, respectively.

Another example of a straightforward chain of decision rules is illustrated in Figure 24, where these nested rules distinguished assignments of patients to phenotypes 9, 24, 6, and 17 if these patients presented high probabilities (over 0.04, 0.015, 0.052 and 0.044) for diagnoses "Malignant neoplasm of trachea bronchus and lung", "Alcohol-induced mental disorders", "Secondary malignant neoplasm of other specified sites" and "Cardiomyopathy", respectively.

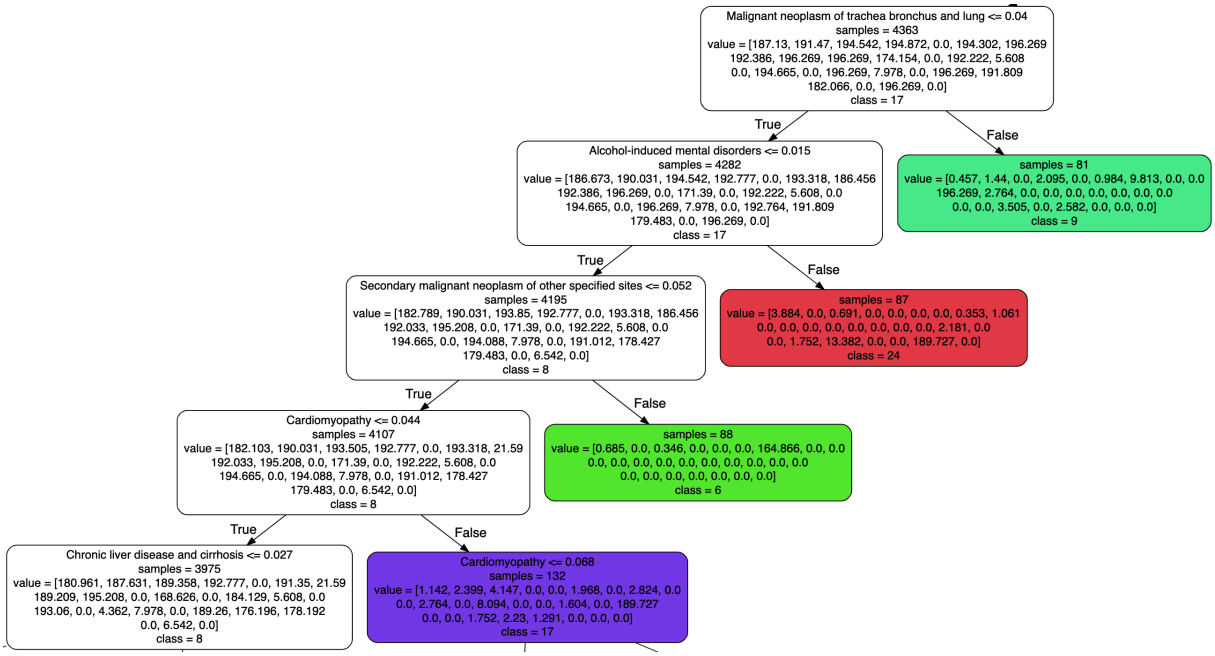

Figure 24 – Sample of decision tree fitted over MIMIC-III ICD-9 showing decision rules for classes 6, 8, 9, 17, and 24.

However, in this chain of rules, following the path of low probabilities (the True values presented in Figure) changed the patients' phenotype from 17 to 8 after the diagnosis of "Alcohol-induced mental disorders". This particular case shows how a more granular set of predicted diagnoses, like the ones from MIMIC-III ICD-9 in comparison with MIMIC-III CCS, is capable to specify more phenotypes since these types of chain of rules are rarer for MIMIC-III CCS dataset.

Discussion

Our final results were the ones obtained by the combination of the cohort representation and the decision trees fitting. These results were presented both quantitative in Table 9 and qualitative in Figures 19 to 24. Since there is a lack of related works to quantitatively compare our performance metric results with, we proposed a dummy and a Naive Bayes classifier to draw a baseline for the decision trees metric results; the pruned decision trees outperformed both the classifiers results by 0.48, 0.48, 0.73, and 0.65 on metrics accuracy, weighted recall, weighted precision, and weighted F1-Score, respectively, over the MIMIC-III CCS encoding; and by 0.57, 0.57, 0.75 and 0.69 on metrics accuracy, weighted recall, weighted precision, and weighted F1 score, respectively, over the MIMIC-III ICD-9 encoding. This superiority of the decision tree results indicated the effectiveness of employing the method of cohort construction of demographic features with diagnoses and decision trees fitting.

However, the method of cohort construction of only demographic features performed worse than both the dummy and Naive Bayes classifiers for both MIMIC-III CCS and MIMIC-III ICD-9 encodings. This particular result demonstrated the incapacity of using only demographic features to explain diagnoses and to attribute patients to phenotypes.

Regarding the qualitative results presented in Figures 19 to 24, our first remark was that, in comparison with the diagnoses, the demographic features were not relevant to the decision trees construction, the evidence to that conclusion was that these features were not used by any tree's decision rule. This particular result has indicated that, at least for the demographic data of MIMIC-III dataset, using only these features is not enough to explain why patients were assigned to certain phenotypes.

In general, we focused on tuning the decision trees hyperparameters to balance the tree interpretability (*i.e.* the number of decision rules) with the performance metrics presented early in Table 9. By observing samples of these MIMIC-III CCS and MIMIC-III ICD-9 decision trees we could draw insights into the relations between diagnoses of one same phenotype and into the most relevant diagnoses to discern between patients' phenotypes assignments.

CHAPTER

6

CONCLUSIONS

This study focused in three distinct areas: (i) prediction of clinical trajectories, (ii) prediction-based phenotyping, and (iii) explanation of phenotypes by decision trees. It also made possible to produce results with the combination of the three areas as to explaining clinical trajectories predictions. The longitudinal prediction results obtained by the Attentive Encoder-Decoder model demonstrated state-of-the-art results, surpassing related works. Also, our method demonstrated promising results when trained with the recently published MIMIC-IV-ED dataset.

Regarding the phenotyping and explainability results, we did an extensive experimentation with clustering methods and linkage criteria in order to find the most suitable clustering for the datasets and problem setup. We found that the hierarchical agglomerative clustering with ward criterion demonstrated the best performance after comparing different linkage criteria and clustering methods by (i) Silhouette, (ii) DB index score, and (iii) t-SNE visualizations. We found 14 phenotypes for MIMIC-III dataset on CCS encoding, and 26 for ICD-9 encoding; we also found the top-3 most relevant diagnoses of each one of these phenotypes. Finally, we concluded that the diagnoses of ICD-9 phenotypes were more consistent than the ones of CCS, *i.e.*, the diagnoses of one same phenotype seemed to be more related to each other and neighbor phenotypes also demonstrated to have more similar diagnoses.

Our study provided explainability about the diagnosis roles through cohort representation and by the fitting of decision trees. The cohort representation enriched the patients' diagnosis data with demographic features, which we found to be irrelevant to decision rules. Our decision trees performance metrics surpassed the ones of both dummy and Naive Bayes classifiers used for comparison, and we presented samples of the decision rules for MIMIC-III dataset on CCS and ICD-9 encodings that presented relationships between the phenotypes' top-3 most relevant diagnoses. These findings have the potential of revealing to physicians and healthcare professionals an outlook for the next admission of a patient: the most probable diagnosis, which was the phenotype it most likely belongs to, and what were the decision rules to attribute that patient to its specific phenotype.

Finally, we believe our approach will support other researchers in the fields of clinical trajectories prediction, phenotyping, and explainability of medical predictions, besides physicians and healthcare professionals.

BIBLIOGRAPHY

AGGARWAL, C. C.; HINNEBURG, A.; KEIM, D. A. On the surprising behavior of distance metrics in high dimensional space. In: BUSSCHE, J. Van den; VIANU, V. (Ed.). **Database Theory — ICDT 2001**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. p. 420–434. ISBN 978-3-540-44503-6. Citation on page 33.

AGRAWAL, R.; GEHRKE, J.; GUNOPULOS, D.; RAGHAVAN, P. Automatic subspace clustering of high dimensional data for data mining applications. In: **Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data**. New York, NY, USA: Association for Computing Machinery, 1998. (SIGMOD '98), p. 94–105. ISBN 0897919955. Available: <https://doi.org/10.1145/276304.276314>. Citation on page 31.

ANAVI, Y.; KOGAN, I.; GELBART, E.; GEVA, O.; GREENSPAN, H. Visualizing and enhancing a deep learning framework using patients age and gender for chest x-ray image retrieval. In: TOURASSI, G. D.; III, S. G. A. (Ed.). **Medical Imaging 2016: Computer-Aided Diagnosis**. SPIE, 2016. v. 9785, p. 978510. Available: <https://doi.org/10.1117/12.2217587>. Citation on page 19.

APILETTI, D.; BARALIS, E.; CERQUITELLI, T.; GARZA, P.; VENTURINI, L. Safe-nec: A scalable and flexible system for network data characterization. In: **NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium**. [S.l.: s.n.], 2016. p. 812–816. Citation on page 41.

ARAUJO, R. A semi-supervised approach for kernel-based temporal clustering. University of Waterloo, 2015. Citation on page 31.

ARDILA, D.; KIRALY, A. P.; BHARADWAJ, S.; CHOI, B.; REICHER, J. J.; PENG, L.; TSE, D.; ETEMADI, M.; YE, W.; CORRADO, G. *et al.* End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. **Nature Medicine**, Nature Publishing Group, v. 25, n. 6, p. 954–961, 2019. Citation on page 19.

BAEHRENS, D.; SCHROETER, T.; HARMELING, S.; KAWANABE, M.; HANSEN, K.; MÜLLER, K.-R. How to explain individual classification decisions. **Journal of Machine Learning Research**, JMLR.org, v. 11, p. 1803–1831, 2010. Citation on page 20.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. arXiv, 2014. Available: <https://arxiv.org/abs/1409.0473>. Citations on pages 28 and 29.

BARROS, P. H. F. D.; RODRIGUES, J. F. Attentionhcare: Advances on computer-aided medical prognosis using attention-based neural networks. In: **2022 International Joint Conference on Neural Networks (IJCNN)**. [S.l.: s.n.], 2022. p. 1–8. Citations on pages 7, 9, and 70.

BAYTAS, I. M.; XIAO, C.; ZHANG, X.; WANG, F.; JAIN, A. K.; ZHOU, J. Patient subtyping via time-aware lstm networks. In: **Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: Association

for Computing Machinery, 2017. (KDD '17), p. 65–74. ISBN 9781450348874. Available: <https://doi.org/10.1145/3097983.3097997>. Citations on pages 49, 50, and 53.

BENGIO, Y.; SIMARD, P.; FRASCONI, P. Learning long-term dependencies with gradient descent is difficult. **IEEE Transactions on Neural Networks**, v. 5, n. 2, p. 157–166, 1994. Citation on page 24.

BERKHIN, P. A survey of clustering data mining techniques. In: KOGAN, J.; NICHOLAS, C.; TEBOULLE, M. (Ed.). **Grouping Multidimensional Data: Recent Advances in Clustering**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. p. 25–71. ISBN 978-3-540-28349-2. Available: <https://doi.org/10.1007/3-540-28349-8_2>. Citation on page 32.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. **J. Mach. Learn. Res.**, JMLR.org, v. 3, n. null, p. 993–1022, mar 2003. ISSN 1532-4435. Citation on page 52.

BOLLMANN, M.; SøGAARD, A. Improving historical spelling normalization with bi-directional lstms and multi-task learning. arXiv, 2016. Available: <https://arxiv.org/abs/1610.07844>. Citation on page 25.

BOLSHAKOVA, N.; AZUAJE, F. Cluster validation techniques for genome expression data. **Signal Processing**, v. 83, n. 4, p. 825–833, 2003. ISSN 0165-1684. Genomic Signal Processing. Available: <https://www.sciencedirect.com/science/article/pii/S0165168402004759>. Citation on page 39.

BREIMAN, L.; FRIEDMAN, J.; STONE, C. J.; OLSHEN, R. A. **Classification and Regression Trees**. [S.l.]: Routledge, 1984. Citations on pages 21 and 40.

CARUANA, R.; LOU, Y.; GEHRKE, J.; KOCH, P.; STURM, M.; ELHADAD, N. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In: **Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: Association for Computing Machinery, 2015. (KDD '15), p. 1721–1730. ISBN 9781450336642. Available: <https://doi.org/10.1145/2783258.2788613>. Citations on pages 20 and 21.

Centers for Disease Control and Prevention. **International Classification of Diseases, (ICD-10-CM/PCS) Transition - Background**. 2015. <https://www.cdc.gov/nchs/icd/icd10cm_pcs_background.htm>. Acessed: 2021-06-13. Citation on page 56.

CHAUDHRY, B.; WANG, J.; WU, S.; MAGLIONE, M.; MOJICA, W.; ROTH, E.; MORTON, S. C.; SHEKELLE, P. G. Systematic review: Impact of health information technology on quality, efficiency, and costs of medical care. **Annals of Internal Medicine**, American College of Physicians, v. 144, n. 10, p. 742–752, 2006. Citation on page 19.

CHE, Z.; PURUSHOTHAM, S.; CHO, K.; SONTAG, D.; LIU, Y. Recurrent neural networks for multivariate time series with missing values. **Scientific Reports**, Nature Publishing Group, v. 8, n. 1, p. 1–12, 2018. Citations on pages 19, 46, and 53.

CHE, Z.; PURUSHOTHAM, S.; KHEMANI, R.; LIU, Y. Interpretable deep models for icu outcome prediction. In: AMERICAN MEDICAL INFORMATICS ASSOCIATION. **AMIA Annual Symposium Proceedings**. [S.l.], 2016. v. 2016, p. 371. Citations on pages 52 and 53.

CHENG, H.-T.; KOC, L.; HARMSEN, J.; SHAKED, T.; CHANDRA, T.; ARADHYE, H.; ANDERSON, G.; CORRADO, G.; CHAI, W.; ISPIR, M.; ANIL, R.; HAQUE, Z.; HONG, L.; JAIN, V.; LIU, X.; SHAH, H. Wide & deep learning for recommender systems. In: **Proceedings of the 1st Workshop on Deep Learning for Recommender Systems**. New York, NY, USA: Association for Computing Machinery, 2016. (DLRS 2016), p. 7–10. ISBN 9781450347952. Available: <https://doi.org/10.1145/2988450.2988454>. Citation on page 50.

CHIU, C.-C.; SAINATH, T. N.; WU, Y.; PRABHAVALKAR, R.; NGUYEN, P.; CHEN, Z.; KANNAN, A.; WEISS, R. J.; RAO, K.; GONINA, E.; JAITLY, N.; LI, B.; CHOROWSKI, J.; BACCHIANI, M. State-of-the-art speech recognition with sequence-to-sequence models. In: **2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.: s.n.], 2018. p. 4774–4778. Citation on page 28.

CHO, K.; MERRIENBOER, B. van; GULCEHRE, C.; BAHDANAU, D.; BOUGARES, F.; SCHWENK, H.; BENGIO, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv, 2014. Available: <https://arxiv.org/abs/1406.1078>. Citations on pages 23, 27, and 28.

CHOI, E.; BAHADORI, M. T.; SCHUETZ, A.; STEWART, W. F.; SUN, J. Doctor ai: Predicting clinical events via recurrent neural networks. In: DOSHI-VELEZ, F.; FACKLER, J.; KALE, D.; WALLACE, B.; WIENS, J. (Ed.). **Proceedings of the 1st Machine Learning for Healthcare Conference**. Northeastern University, Boston, MA, USA: PMLR, 2016. (Proceedings of Machine Learning Research, v. 56), p. 301–318. Available: <https://proceedings.mlr.press/v56/Choi16.html>. Citations on pages 19, 21, 47, 53, 67, and 68.

CHOI, E.; BAHADORI, M. T.; SEARLES, E.; COFFEY, C.; THOMPSON, M.; BOST, J.; TEJEDOR-SOJO, J.; SUN, J. Multi-layer representation learning for medical concepts. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 1495–1504. ISBN 9781450342322. Available: <https://doi.org/10.1145/2939672.2939823>. Citations on pages 47 and 53.

CHOI, E.; BAHADORI, M. T.; SUN, J.; KULAS, J.; SCHUETZ, A.; STEWART, W. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. Curran Associates, Inc., v. 29, 2016. Available: <https://proceedings.neurips.cc/paper/2016/file/231141b34c82aa95e48810a9d1b33a79-Paper.pdf>. Citation on page 20.

COST, H.; PROJECT, U. **Clinical Classifications Software**. [S.l.], 2015. Citation on page 48.

CSISZAR, I. *i*-divergence geometry of probability distributions and minimization problems. **The Annals of Probability**, Institute of Mathematical Statistics, v. 3, n. 1, p. 146–158, 1975. ISSN 00911798. Available: <http://www.jstor.org/stable/2959270>. Citation on page 71.

DAI, Y.; LOKHANDWALA, S.; LONG, W.; MARK, R.; LEHMAN, L.-w. H. Phenotyping hypotensive patients in critical care using hospital discharge summaries. In: **2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)**. [S.l.: s.n.], 2017. p. 401–404. Citations on pages 49 and 53.

DAVIES, D. L.; BOULDIN, D. W. A cluster separation measure. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, PAMI-1, n. 2, p. 224–227, 1979. Citation on page 39.

ELLSON, J.; GANSNER, E.; KOUTSOFIOS, L.; NORTH, S. C.; WOODHULL, G. Graphviz—open source graph drawing tools. In: MUTZEL, P.; JÜNGER, M.; LEIPERT, S. (Ed.). **Graph Drawing**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002. p. 483–484. ISBN 978-3-540-45848-7. Citations on pages 11 and 40.

ESTER, M.; KRIEGEL, H.-P.; SANDER, J.; XU, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: **Proceedings of the Second International Conference on Knowledge Discovery and Data Mining**. [S.l.]: AAAI Press, 1996. (KDD'96), p. 226–231. Citation on page 31.

EuroRec. 2009. Available: <https://www.eurorec.org/eurorec-seal/>. Accessed: 31/01/2020. Citation on page 19.

FISHER, R. A. The use of multiple measurements in taxonomic problems. **Annals of Eugenics**, v. 7, n. 2, p. 179–188, 1936. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-1809.1936.tb02137.x>. Citations on pages 11 and 40.

FLOREZ, A. Y.; SCABORA, L.; ELER, D. M.; RODRIGUES, J. F. Apehr: Automated prognosis in electronic health records using multi-head self-attention. In: **2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)**. [S.l.: s.n.], 2021. p. 277–282. Citations on pages 48, 53, 57, 66, 67, 69, and 97.

FRIEDMAN, J. H. Greedy function approximation: A gradient boosting machine. **The Annals of Statistics**, Institute of Mathematical Statistics, v. 29, n. 5, p. 1189–1232, 2001. ISSN 00905364. Available: <http://www.jstor.org/stable/2699986>. Citation on page 52.

GERS, F.; SCHMIDHUBER, J. Recurrent nets that time and count. In: **Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium**. [S.l.: s.n.], 2000. v. 3, p. 189–194 vol.3. Citation on page 25.

GERS, F. A.; SCHMIDHUBER, J.; CUMMINS, F. Learning to forget: Continual prediction with lstm. **Neural Computation**, v. 12, n. 10, p. 2451–2471, 2000. Citation on page 24.

GERS, F. A.; SCHRAUDOLPH, N. N.; SCHMIDHUBER, J. Learning precise timing with lstm recurrent networks. **Journal of Machine Learning Research**, JMLR.org, v. 3, n. Aug, p. 115–143, 2002. Citation on page 26.

GHOSH, P.; NEUFELD, A.; SAHOO, J. K. Forecasting directional movements of stock prices for intraday trading using lstm and random forests. **Finance Research Letters**, v. 46, p. 102280, 2022. ISSN 1544-6123. Available: <https://www.sciencedirect.com/science/article/pii/S1544612321003202>. Citation on page 24.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: TEH, Y. W.; TITTERINGTON, M. (Ed.). **Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics**. Chia Laguna Resort, Sardinia, Italy: PMLR, 2010. (Proceedings of Machine Learning Research, v. 9), p. 249–256. Available: <https://proceedings.mlr.press/v9/glorot10a.html>. Citation on page 25.

GRAVES, A.; WAYNE, G.; DANIHELKA, I. Neural turing machines. arXiv, 2014. Available: <https://arxiv.org/abs/1410.5401>. Citation on page 28.

GRAY, S.; RADFORD, A.; KINGMA, D. P. Gpu kernels for block-sparse weights. arXiv, v. 3, 2017. Citation on page 24.

GREFF, K.; SRIVASTAVA, R. K.; KOUTNíK, J.; STEUNEBRINK, B. R.; SCHMIDHUBER, J. Lstm: A search space odyssey. **IEEE Transactions on Neural Networks and Learning Systems**, v. 28, n. 10, p. 2222–2232, 2017. Citation on page 25.

GRIRA, N.; CRUCIANU, M.; BOUJEMAA, N. Unsupervised and semi-supervised clustering: a brief survey. **A Review of Machine Learning Techniques for Processing Multimedia Content**, 09 2005. Citation on page 31.

GUNTER, T. D.; TERRY, N. P. The emergence of national electronic health record architectures in the united states and australia: Models, costs, and questions. **Journal of Medical Internet Research**, JMIR Publications Inc., Toronto, Canada, v. 7, n. 1, p. e3, 2005. Citation on page 19.

HANCOCK, T.; COOMANS, D.; EVERINGHAM, Y. Supervised hierarchical clustering using cart t. In: **MODSIM 2003. International Congress on Modelling and Simulation**. [S.l.: s.n.], 2003. p. 1880–1885. Citation on page 41.

HARTIGAN, J. A. **Clustering algorithms**. [S.l.]: John Wiley & Sons, Inc., 1975. Citation on page 50.

HOCHREITER, S. Untersuchungen zu dynamischen neuronalen netzen. **Diploma, Technische Universität München**, v. 91, n. 1, 1991. Citation on page 24.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. **Neural computation**, MIT Press, v. 9, n. 8, p. 1735–1780, 1997. Citation on page 23.

HORN, M.; MOOR, M.; BOCK, C.; RIECK, B.; BORGWARDT, K. Set functions for time series. arXiv, 2019. Available: <https://arxiv.org/abs/1909.12064>. Citation on page 24.

HRIPCSAK, G.; ALBERS, D. J. High-fidelity phenotyping: richness and freedom from bias. **Journal of the American Medical Informatics Association**, v. 25, n. 3, p. 289–294, 10 2017. ISSN 1527-974X. Available: <https://doi.org/10.1093/jamia/ocx110>. Citation on page 48.

HU, L.; LI, J.; NIE, L.; LI, X.-L.; SHAO, C. What happens next? future subevent prediction using contextual hierarchical lstm. In: **Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence**. [S.l.]: AAAI Press, 2017. (AAAI'17), p. 3450–3456. Citation on page 25.

HUANG, Z.; XU, W.; YU, K. Bidirectional lstm-crf models for sequence tagging. arXiv, 2015. Available: <https://arxiv.org/abs/1508.01991>. Citation on page 25.

JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: A review. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 31, n. 3, p. 264–323, sep 1999. ISSN 0360-0300. Available: <https://doi.org/10.1145/331499.331504>. Citation on page 31.

JOHNSON, A. E.; POLLARD, T. J.; SHEN, L.; LI-WEI, H. L.; FENG, M.; GHASSEMI, M.; MOODY, B.; SZOLOVITS, P.; CELI, L. A.; MARK, R. G. Mimic-iii, a freely accessible critical care database. **Scientific Data**, Nature Publishing Group, v. 3, n. 1, p. 1–9, 2016. Citations on pages 46 and 55.

KARIM, F.; MAJUMDAR, S.; DARABI, H.; HARFORD, S. Multivariate lstm-fcns for time series classification. **Neural Networks**, v. 116, p. 237–245, 2019. ISSN 0893-6080. Available: <https://www.sciencedirect.com/science/article/pii/S0893608019301200>. Citation on page 24.

KHEMANI, R. G.; CONTI, D.; ALONZO, T. A.; BART, R. D.; NEWTH, C. J. Effect of tidal volume in children with acute hypoxemic respiratory failure. **Intensive Care Medicine**, Springer, v. 35, n. 8, p. 1428–1437, 2009. Citation on page 52.

KIM, J.; EL-KHAMY, M.; LEE, J. Residual lstm: Design of a deep recurrent architecture for distant speech recognition. arXiv, 2017. Available: <https://arxiv.org/abs/1701.03360>. Citation on page 25.

KIMBLE, C. Electronic health records: Cure-all or chronic condition? **Global Business and Organizational Excellence**, Wiley Online Library, v. 33, n. 4, p. 63–74, 2014. Citation on page 19.

KRATZERT, F.; HERRNEGGER, M.; KLOTZ, D.; HOCHREITER, S.; KLAMBAUER, G. Neuralhydrology – interpreting lstms in hydrology. In: SAMEK, W.; MONTAVON, G.; VEDALDI, A.; HANSEN, L. K.; MÜLLER, K.-R. (Ed.). **Explainable AI: Interpreting, Explaining and Visualizing Deep Learning**. Cham: Springer International Publishing, 2019. p. 347–362. ISBN 978-3-030-28954-6. Available: <https://doi.org/10.1007/978-3-030-28954-6_19>. Citation on page 25.

KRISHNAN, R.; SIVAKUMAR, G.; BHATTACHARYA, P. Extracting decision trees from trained neural networks. **Pattern Recognition**, v. 32, n. 12, p. 1999–2009, 1999. ISSN 0031-3203. Available: <https://www.sciencedirect.com/science/article/pii/S0031320398001812>. Citation on page 41.

LAKKARAJU, H.; BACH, S. H.; LESKOVEC, J. Interpretable decision sets: A joint framework for description and prediction. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 1675–1684. ISBN 9781450342322. Available: <https://doi.org/10.1145/2939672.2939874>. Citation on page 20.

LAKKARAJU, H.; KAMAR, E.; CARUANA, R.; LESKOVEC, J. Interpretable & explorable approximations of black box models. arXiv, 2017. Available: <https://arxiv.org/abs/1707.01154>. Citation on page 41.

LETHAM, B.; RUDIN, C.; MCCORMICK, T. H.; MADIGAN, D. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. **The Annals of Applied Statistics**, Institute of Mathematical Statistics, v. 9, n. 3, p. 1350 – 1371, 2015. Available: <https://doi.org/10.1214/15-AOAS848>. Citation on page 20.

LI, S.; LI, W.; COOK, C.; ZHU, C.; GAO, Y. Independently recurrent neural network (indrnn): Building a longer and deeper rnn. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2018. Citation on page 98.

LIPTON, Z. C. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. **Queue**, Association for Computing Machinery, New York, NY, USA, v. 16, n. 3, p. 31–57, jun 2018. ISSN 1542-7730. Available: <https://doi.org/10.1145/3236386.3241340>. Citations on pages 20 and 51.

LIPTON, Z. C.; KALE, D. C.; ELKAN, C.; WETZEL, R. Learning to diagnose with lstm recurrent neural networks. arXiv, 2015. Available: <https://arxiv.org/abs/1511.03677>. Citations on pages 21, 45, 53, 67, and 68.

LIU, X.; TIZHOOSH, H.; KOFMAN, J. Generating binary tags for fast medical image retrieval based on convolutional nets and radon transform. In: **2016 International Joint Conference on Neural Networks (IJCNN)**. [S.l.: s.n.], 2016. p. 2872–2878. Citation on page 19.

LONG, W. Extracting diagnoses from discharge summaries. In: AMERICAN MEDICAL INFORMATICS ASSOCIATION. **AMIA Annual Symposium proceedings**. [S.l.], 2005. v. 2005, p. 470. Citation on page 49.

LUONG, M.-T.; PHAM, H.; MANNING, C. D. Effective approaches to attention-based neural machine translation. arXiv, 2015. Available: <https://arxiv.org/abs/1508.04025>. Citation on page 28.

MAATEN, L. van der; HINTON, G. Visualizing data using t-sne. **Journal of Machine Learning Research**, v. 9, n. 86, p. 2579–2605, 2008. Available: <http://jmlr.org/papers/v9/vandermaaten08a.html>. Citation on page 49.

MACQUEEN, J. Some methods for classification and analysis of multivariate observations. In: OAKLAND, CA, USA. **Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability**. [S.l.], 1967. v. 1, n. 14, p. 281–297. Citation on page 31.

MANDEL, J. C.; KREDA, D. A.; MANDL, K. D.; KOHANE, I. S.; RAMONI, R. B. SMART on FHIR: a standards-based, interoperable apps platform for electronic health records. **Journal of the American Medical Informatics Association**, v. 23, n. 5, p. 899–908, 02 2016. ISSN 1067-5027. Available: <https://doi.org/10.1093/jamia/ocv189>. Citation on page 46.

MCQUITTY, L. L. Elementary linkage analysis for isolating orthogonal and oblique types and typal relevancies. **Educational and Psychological Measurement**, v. 17, n. 2, p. 207–229, 1957. Available: <https://doi.org/10.1177/001316445701700204>. Citation on page 34.

METZ, J. **Interpretação de clusters gerados por algoritmos de clustering hierárquico**. Master's Thesis (Master's Thesis) — Universidade de São Paulo, 2006. Citation on page 32.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed representations of words and phrases and their compositionality. Curran Associates, Inc., v. 26, 2013. Available: <https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf>. Citation on page 47.

MILLER, D. D.; BROWN, E. W. Artificial intelligence in medical practice: The question to the answer? **The American Journal of Medicine**, Elsevier, v. 131, n. 2, p. 129–133, 2018. Citation on page 20.

MORICHETTA, A.; CASAS, P.; MELLIA, M. Explain-it: Towards explainable ai for unsupervised network traffic analysis. In: **Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks**. New York, NY, USA: Association for Computing Machinery, 2019. (Big-DAMA '19), p. 22–28. ISBN 9781450369992. Available: <https://doi.org/10.1145/3359992.3366639>. Citation on page 41.

MÜLLER, M. Dynamic time warping. Springer Berlin Heidelberg, Berlin, Heidelberg, p. 69–84, 2007. Available: <https://doi.org/10.1007/978-3-540-74048-3_4>. Citation on page 50.

MURTAGH, F.; CONTRERAS, P. Algorithms for hierarchical clustering: an overview. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, Wiley Online Library, v. 2, n. 1, p. 86–97, 2012. Citation on page 32.

ORGANIZATION, W. H. **International classification of diseases : [9th] ninth revision, basic tabulation list with alphabetic index**. [S.l.]: World Health Organization, 1978. 331 p. p. Citation on page 48.

PASCANU, R.; GULCEHRE, C.; CHO, K.; BENGIO, Y. How to construct deep recurrent neural networks. arXiv, 2013. Available: <https://arxiv.org/abs/1312.6026>. Citation on page 25.

PASCANU, R.; MIKOLOV, T.; BENGIO, Y. On the difficulty of training recurrent neural networks. In: DASGUPTA, S.; MCALLESTER, D. (Ed.). **Proceedings of the 30th International Conference on Machine Learning**. Atlanta, Georgia, USA: PMLR, 2013. (Proceedings of Machine Learning Research, 3), p. 1310–1318. Available: <https://proceedings.mlr.press/v28/pascanu13.html>. Citation on page 24.

PENNINGTON, J.; SOCHER, R.; MANNING, C. D. Glove: Global vectors for word representation. In: **Empirical Methods in Natural Language Processing (EMNLP)**. [s.n.], 2014. p. 1532–1543. Available: <http://www.aclweb.org/anthology/D14-1162>. Citation on page 47.

PHAM, T.; TRAN, T.; PHUNG, D.; VENKATESH, S. Predicting healthcare trajectories from medical records: A deep learning approach. **Journal of Biomedical Informatics**, v. 69, p. 218–229, 2017. ISSN 1532-0464. Available: <https://www.sciencedirect.com/science/article/pii/S1532046417300710>. Citations on pages 19, 21, 46, 53, 67, and 68.

PLS 474. 2008. Available: <https://www25.senado.leg.br/web/atividade/materias/-/materia/88695>. Citation on page 19.

QUIROS, J. V. **Information-theoretic anomaly detection and authorship attribution in literature**. Master's Thesis (Master's Thesis) — Utrecht University, 2017. Citation on page 41.

RAFFEL, C.; LUONG, M.-T.; LIU, P. J.; WEISS, R. J.; ECK, D. Online and linear-time attention by enforcing monotonic alignments. In: PRECUP, D.; TEH, Y. W. (Ed.). **Proceedings of the 34th International Conference on Machine Learning**. PMLR, 2017. (Proceedings of Machine Learning Research, v. 70), p. 2837–2846. Available: <https://proceedings.mlr.press/v70/raffel17a.html>. Citation on page 30.

RAI, A. Explainable ai: from black box to glass box. **Journal of the Academy of Marketing Science**, v. 48, 12 2019. Citation on page 41.

RAJKOMAR, A.; OREN, E.; CHEN, K.; DAI, A. M.; HAJAJ, N.; HARDT, M.; LIU, P. J.; LIU, X.; MARCUS, J.; SUN, M. *et al.* Scalable and accurate deep learning with electronic health records. **NPJ Digital Medicine**, Nature Publishing Group, v. 1, n. 1, p. 1–10, 2018. Citations on pages 46, 53, 67, and 68.

RANZATO, M.; CHOPRA, S.; AULI, M.; ZAREMBA, W. Sequence level training with recurrent neural networks. arXiv, 2015. Available: <https://arxiv.org/abs/1511.06732>. Citation on page 27.

REDDY, C. K.; VINZAMURI, B. A survey of partitional and hierarchical clustering algorithms. **Data clustering: Algorithms and applications**, v. 87, 2013. Citations on pages 13, 33, 34, and 35.

RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. "why should i trust you?": Explaining the predictions of any classifier. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 1135–1144. ISBN 9781450342322. Available: <https://doi.org/10.1145/2939672.2939778>. Citation on page 20.

____. Anchors: High-precision model-agnostic explanations. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2018. v. 32, n. 1. Citation on page 20.

RICHETTE, P.; CLERSON, P.; PÉRISSIN, L.; FLIPO, R.-M.; BARDIN, T. Revisiting comorbidities in gout: a cluster analysis. **Annals of the Rheumatic Diseases**, BMJ Publishing Group Ltd, v. 74, n. 1, p. 142–147, 2015. Citation on page 32.

RODRIGUES-JR, J. F.; GUTIERREZ, M. A.; SPADON, G.; BRANDOLI, B.; AMER-YAHIA, S. Lig-doctor: Efficient patient trajectory prediction using bidirectional minimal gated-recurrent networks. **Information Sciences**, v. 545, p. 813–827, 2021. ISSN 0020-0255. Available: <https://www.sciencedirect.com/science/article/pii/S002002552030935X>. Citations on pages 21, 48, 53, 57, 66, 67, 68, and 97.

ROKACH, L. A survey of clustering algorithms. In: MAIMON, O.; ROKACH, L. (Ed.). **Data Mining and Knowledge Discovery Handbook**. Boston, MA: Springer US, 2010. p. 269–298. ISBN 978-0-387-09823-4. Available: <https://doi.org/10.1007/978-0-387-09823-4_14>. Citations on pages 31, 32, and 35.

ROKACH, L.; MAIMON, O. Top-down induction of decision trees classifiers - a survey. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, v. 35, n. 4, p. 476–487, 2005. Citation on page 41.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. **Psychological Review**, American Psychological Association, v. 65, n. 6, p. 386, 1958. Citation on page 23.

ROUSSEEUW, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. **Journal of Computational and Applied Mathematics**, v. 20, p. 53–65, 1987. ISSN 0377-0427. Available: <https://www.sciencedirect.com/science/article/pii/0377042787901257>. Citation on page 38.

RUDIN, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. **Nature Machine Intelligence**, Nature Publishing Group, v. 1, n. 5, p. 206–215, 2019. Citations on pages 20, 21, and 51.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning Internal Representations by Error Propagation**. [S.l.], 1985. Citation on page 23.

____. Learning representations by back-propagating errors. **Nature**, Nature Publishing Group, v. 323, n. 6088, p. 533–536, 1986. Citation on page 23.

SAK, H.; SENIOR, A. W.; BEAUFAYS, F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. p. 338–342, 2014. Citation on page 26.

SCHIMIDINGER, N. 2020. Available: <https://www.niklasschmidinger.com/posts/2020-09-09-lstm-tricks>. Accessed: 20/10/2020. Citation on page 26.

SCHMIDHUBER, J.; HUBER, R. **Learning to generate focus trajectories for attentive vision**. [S.l.]: Institut für Informatik, 1990. Citation on page 28.

SCHMITZ, G.; ALDRICH, C.; GOUWS, F. Ann-dt: an algorithm for extraction of decision trees from artificial neural networks. **IEEE Transactions on Neural Networks**, v. 10, n. 6, p. 1392–1401, 1999. Citation on page 51.

SCIENCES, O. H. D.; INFORMATICS. 2020. Available: <https://ohdsi.github.io/TheBookOfOhdsi/Cohorts.html#what-is-a-cohort>. Accessed: 08/12/2020. Citation on page 48.

Scikit-learn Examples. 2020. Available: <https://scikit-learn.org/stable/auto_examples/cluster/plot_linkage_comparison.html>. Accessed: 26/11/2020. Citation on page 34.

____. 2021. Available: <https://scikit-learn.org/stable/auto_examples/cluster/plot_agglomerative_clustering.html#sphx-glr-auto-examples-cluster-plot-agglomerative-clustering-py>. Accessed: 29/01/2022. Citation on page 37.

SENNHAUSER, L.; BERWICK, R. C. Evaluating the ability of lstms to learn context-free grammars. arXiv, 2018. Available: <https://arxiv.org/abs/1811.02611>. Citation on page 25.

SHALEV-SHWARTZ, S.; BEN-DAVID, S. **Understanding Machine Learning: From Theory to Algorithms**. [S.l.]: Cambridge University Press, 2014. 212–218 p. Citation on page 41.

SHI, X.; CHEN, Z.; WANG, H.; YEUNG, D.-Y.; WONG, W.-k.; WOO, W.-c. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: CORTES, C.; LAWRENCE, N.; LEE, D.; SUGIYAMA, M.; GARNETT, R. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2015. v. 28. Available: <https://proceedings.neurips.cc/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf>. Citation on page 25.

SHIVADE, C.; RAGHAVAN, P.; FOSLER-LUSSIER, E.; EMBI, P. J.; ELHADAD, N.; JOHNSON, S. B.; LAI, A. M. A review of approaches to identifying patient phenotype cohorts using electronic health records. **Journal of the American Medical Informatics Association**, v. 21, n. 2, p. 221–230, 11 2013. ISSN 1067-5027. Available: <https://doi.org/10.1136/amiajnl-2013-001935>. Citation on page 49.

SILVA, I.; MOODY, G.; SCOTT, D. J.; CELI, L. A.; MARK, R. G. Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012. In: **2012 Computing in Cardiology**. [S.l.: s.n.], 2012. p. 245–248. Citation on page 47.

SOKAL, R. R. A statistical method for evaluating systematic relationships. **University of Kansas science bulletin.**, v. 38, p. 1409–1438, 1958. Citation on page 35.

____. Numerical taxonomy. **Scientific American**, JSTOR, v. 215, n. 6, p. 106–117, 1966. Citations on pages 34 and 35.

SPANHOL, F. A.; OLIVEIRA, L. S.; PETITJEAN, C.; HEUTTE, L. Breast cancer histopathological image classification using convolutional neural networks. In: **2016 International Joint Conference on Neural Networks (IJCNN)**. [S.l.: s.n.], 2016. p. 2560–2567. Citation on page 19.

SRA, S.; DHILLON, I. Generalized nonnegative matrix approximations with bregman divergences. In: WEISS, Y.; SCHöLKOPF, B.; PLATT, J. (Ed.). **Advances in Neural Information Processing Systems**. MIT Press, 2005. v. 18. Available: <https://proceedings.neurips.cc/paper/2005/file/d58e2f077670f4de9cd7963c857f2534-Paper.pdf>. Citation on page 51.

STEINBACH, M.; ERTÖZ, L.; KUMAR, V. The challenges of clustering high dimensional data. In: WILLE, L. T. (Ed.). **New Directions in Statistical Physics: Econophysics, Bioinformatics, and Pattern Recognition**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. p. 273–309. ISBN 978-3-662-08968-2. Available: <https://doi.org/10.1007/978-3-662-08968-2_16>. Citation on page 31.

STOLLENGA, M. F.; BYEON, W.; LIWICKI, M.; SCHMIDHUBER, J. Parallel multi-dimensional lstm, with application to fast biomedical volumetric image segmentation. arXiv, 2015. Citation on page 19.

SURESH, H.; HUNT, N.; JOHNSON, A.; CELI, L. A.; SZOLOVITS, P.; GHASSEMI, M. Clinical intervention prediction and understanding using deep networks. arXiv, 2017. Available: <https://arxiv.org/abs/1705.08498>. Citations on pages 52 and 53.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to sequence learning with neural networks. Curran Associates, Inc., v. 27, 2014. Available: <https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf>. Citation on page 27.

SZMRECSANYI, B. **Grammatical Variation in British English Dialects: A Study in Corpus-Based Dialectometry**. [S.l.]: Cambridge University Press, 2012. (Studies in English Language). Citation on page 36.

TAN, P.-N.; STEINBACH, M.; KARPATNE, A.; KUMAR, V. **Introduction to Data Mining (2nd Edition)**. 2nd. ed. [S.l.]: Pearson, 2018. ISBN 0133128903. Citation on page 38.

TEH, Y. W.; JORDAN, M. I.; BEAL, M. J.; BLEI, D. M. Hierarchical dirichlet processes. **Journal of the american statistical association**, Taylor & Francis, v. 101, n. 476, p. 1566–1581, 2006. Citation on page 49.

The Python Graph Gallery. 2017. Available: <https://python-graph-gallery.com/dendrogram/#prettyPhoto>. Accessed: 24/11/2020. Citation on page 33.

TinaGongting. 2019. Available: <https://medium.com/@penggongting/implementing-decision-tree-from-scratch-in-python-c732e7c69aea>. Accessed: 25/01/2021. Citation on page 40.

Trisha Torrey. **ICD 10 Codes and How to Look Them Up**. 2022. <https://www.verywellhealth.com/finding-icd-codes-2615311>. Acessed: 2022-12-04. Citation on page 43.

U.S.HHS. 2009. Available: <https://www.commonwealthfund.org/publications/newsletter-article/federal-government-has-put-billions-promoting-electronic-health>. Citation on page 19.

VANDROMME, M.; JUN, T.; PERUMALSWAMI, P.; DUDLEY, J. T.; BRANCH, A.; LI, L. Automated phenotyping of patients with non-alcoholic fatty liver disease reveals clinically relevant disease subtypes. In: **Biocomputing 2020**. [s.n.], 2019. p. 91–102. Available: <https://www.worldscientific.com/doi/abs/10.1142/9789811215636_0009>. Citations on pages 49 and 53.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L. u.; POLOSUKHIN, I. Attention is all you need. Curran Associates, Inc., v. 30, 2017. Available: <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>. Citation on page 27.

WANG, Y.; HUANG, M.; ZHU, X.; ZHAO, L. Attention-based LSTM for aspect-level sentiment classification. In: **Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing**. Austin, Texas: Association for Computational Linguistics, 2016. p. 606–615. Available: <https://aclanthology.org/D16-1058>. Citation on page 25.

WANG, Y.; WU, T.; WANG, Y.; WANG, G. Enhancing model interpretability and accuracy for disease progression prediction via phenotype-based patient similarity learning. In: **Biocomputing 2020**. [s.n.], 2019. p. 511–522. Available: <https://www.worldscientific.com/doi/abs/10.1142/9789811215636_0045>. Citations on pages 50 and 53.

WARD, J. H. Hierarchical grouping to optimize an objective function. **Journal of the American Statistical Association**, [American Statistical Association, Taylor & Francis, Ltd.], v. 58, n. 301, p. 236–244, 1963. ISSN 01621459. Available: <http://www.jstor.org/stable/2282967>. Citations on pages 32 and 36.

WERBOS, P. J. Generalization of backpropagation with application to a recurrent gas market model. **Neural Networks**, v. 1, n. 4, p. 339–356, 1988. ISSN 0893-6080. Available: <https://www.sciencedirect.com/science/article/pii/089360808890007X>. Citation on page 23.

WILLIAMS, R. J.; ZIPSER, D. Gradient-based learning algorithms for recurrent networks and their computational complexity. L. Erlbaum Associates Inc., USA, p. 433–486, 1995. Citation on page 23.

World Health Organization. **International Statistical Classification of Diseases and Related Health Problems (ICD)**. 2022. <https://www.who.int/standards/classifications/classification-of-diseases>. Acessed: 2022-12-04. Citation on page 43.

WU, L.; TAN, X.; HE, D.; TIAN, F.; QIN, T.; LAI, J.; LIU, T.-Y. Beyond error propagation in neural machine translation: Characteristics of language also matter. arXiv, 2018. Available: <https://arxiv.org/abs/1809.00120>. Citation on page 28.

XU, J.; XIANG, L.; HANG, R.; WU, J. Stacked sparse autoencoder (ssae) based framework for nuclei patch classification on breast cancer histopathology. In: **2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)**. [S.l.: s.n.], 2014. p. 999–1002. Citation on page 19.

XU, Y.; HOSNY, A.; ZELEZNIK, R.; PARMAR, C.; COROLLER, T.; FRANCO, I.; MAK, R. H.; AERTS, H. J. Deep learning predicts lung cancer treatment response from serial medical imaging. **Clinical Cancer Research**, American Association for Cancer Research, v. 25, n. 11, p. 3266–3275, 2019. Citation on page 19.

ZHANG, J.; KOWSARI, K.; HARRISON, J. H.; LOBO, J. M.; BARNES, L. E. Patient2vec: A personalized interpretable deep representation of the longitudinal electronic health record. **IEEE Access**, v. 6, p. 65333–65346, 2018. Citations on pages 51 and 53.

ZHANG, X.; CHOU, J.; LIANG, J.; XIAO, C.; ZHAO, Y.; SARVA, H.; HENCHCLIFFE, C.; WANG, F. Data-driven subtyping of parkinson's disease using longitudinal clinical records: A

cohort study. **Scientific Reports**, Nature Publishing Group, v. 9, n. 1, p. 1–12, 2019. Citations on pages 49 and 53.

ZHANG, Y.; CHEN, G.; YU, D.; YAO, K.; KHUDANPUR, S.; GLASS, J. Highway long short-term memory rnns for distant speech recognition. In: **2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.: s.n.], 2016. p. 5755–5759. Citation on page 98.

ZHANG, Y.; ZHOU, H.; LI, J.; SUN, W.; CHEN, Y. A time-sensitive hybrid learning model for patient subgrouping. In: **2018 International Joint Conference on Neural Networks (IJCNN)**. [S.l.: s.n.], 2018. p. 1–8. Citations on pages 50 and 53.

ZHOU, Z.; SIDDIQUEE, M. M. R.; TAJBAKHSH, N.; LIANG, J. Unet++: A nested u-net architecture for medical image segmentation. In: **Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support**. [S.l.]: Springer, 2018. p. 3–11. Citation on page 19.

ZIMMERMANN, H.-G.; TIETZ, C.; GROTHMANN, R. Forecasting with recurrent neural networks: 12 tricks. In: MONTAVON, G.; ORR, G. B.; MÜLLER, K.-R. (Ed.). **Neural Networks: Tricks of the Trade: Second Edition**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 687–707. ISBN 978-3-642-35289-8. Available: <https://doi.org/10.1007/978-3-642-35289-8_37>. Citation on page 25.

ZINCHUK, A. V.; GENTRY, M. J.; CONCATO, J.; YAGGI, H. K. Phenotypes in obstructive sleep apnea: a definition, examples and evolution of approaches. **Sleep Medicine Reviews**, Elsevier, v. 35, p. 113–123, 2017. Citation on page 49.

APPENDIX

A

SUPPORTING EXPERIMENTS

We conducted supporting experiments to set the best number of layers, neurons, and other hyperparameters for the neural model; and the best clustering method, number of clusters, and hyperparameters for the phenotyping architecture composed by clustering and decision trees.

In general, we used the MIMIC-III dataset to conduct these experiments due to the computational cost and time to replicate the same experiments over the larger MIMIC-IV-ED dataset. The only exception is the experiment of attention's alignments due to the objective of analyzing long trajectories, which are not frequent in the MIMIC-III dataset.

A.0.1 Recurrent network deepness

To define the most suitable number of neurons for each model architecture with a single layer, we explored three different setups for the number of neurons over dataset MIMIC-III with ICD-9 and CCS encodings; the results are presented in Tables 10 and 11.

To confirm or contradict previous literature results about the model width, we choose the same number of neurons used (271, 542, 1084). The results presented in Tables 10 and 11 provided us an indication of the optimal number of neurons for the proposed and baseline models. In general, the number of neurons did not significantly affected the prediction results after a given number of neurons, which our exploratory results showed was 271. These results confirm the ones reported by Rodrigues-Jr et al. (RODRIGUES-JR *et al.*, 2021): although the model width did not significantly influenced the prediction results in general for the MIMIC-III dataset, the number of neurons equal to 271 was the most ideal.

Regarding the model depth, we obtained conclusions similar to the ones reported by Rodrigues-Jr et al. (RODRIGUES-JR *et al.*, 2021) and Florez et al. (FLOREZ *et al.*, 2021): increasing the number of layers not only presented no improvements but significantly worsened the results for all sizes of the model tested. At a first glance, this particular result seems counter-intuitive in the context of deep learning models, however, stacked recurrent networks' spatial

Table 10 – Comparison of Recall@k and Precision@n between the number of neurons per layer for AttentionHCare over dataset MIMIC-III with ICD-9 and CCS encodings.

	Recall@10	Recall@20	Recall@30	Precision@1	Precision@2	Precision@3
1 Layer						
271	0.49/0.54	0.65/0.72	0.73/0.80	0.80/0.81	0.77/0.79	0.74/0.77
542	0.48/0.53	0.64/0.72	0.73/0.80	0.80/0.82	0.76/0.79	0.74/0.77
1084	0.48/0.53	0.64/0.71	0.72/0.80	0.79/0.81	0.76/0.79	0.73/0.76
2 Layers						
271	0.47/0.52	0.63/0.70	0.71/0.78	0.79/0.81	0.76/0.79	0.73/0.76
542	0.47/0.52	0.63/0.70	0.71/0.78	0.79/0.81	0.76/0.78	0.72/0.76
1084	0.47/0.52	0.63/0.70	0.71/0.78	0.80/0.81	0.76/0.79	0.73/0.76
3 Layers						
271	0.43/0.49	0.58/0.66	0.67/0.76	0.77/0.81	0.73/0.78	0.69/0.74
542	0.43/0.49	0.58/0.67	0.67/0.76	0.78/0.81	0.73/0.78	0.69/0.74
1084	0.44/0.49	0.58/0.66	0.67/0.76	0.78/0.81	0.73/0.78	0.70/0.74

Table 11 – Comparison of Recall@k and Precision@n between the number of neurons per layer for LSTM and GRU over dataset MIMIC-III with ICD-9 encoding only.

	Recall@10	Recall@20	Recall@30	Precision@1	Precision@2	Precision@3
2 Layers						
271	0.42/0.44	0.57/0.59	0.66/0.68	0.75/0.75	0.71/0.71	0.67/0.68
542	0.43/0.44	0.58/0.60	0.66/0.68	0.75/0.74	0.71/0.71	0.67/0.68
1084	0.43/0.43	0.58/0.58	0.67/0.67	0.79/0.75	0.74/0.70	0.69/0.67
3 Layers						
271	0.40/0.40	0.55/0.56	0.64/0.65	0.73/0.73	0.67/0.68	0.63/0.64
542	0.40/0.40	0.55/0.55	0.64/0.64	0.74/0.71	0.68/0.68	0.63/0.63
1084	0.40/0.38	0.55/0.53	0.64/0.62	0.71/0.69	0.67/0.64	0.63/0.60

deepness is still an open question in deep learning literature (LI *et al.*, 2018), (ZHANG *et al.*, 2016).

A.0.2 Attention's alignments scores for trajectories

At each decoding step, the attention mechanism produces an alignment score by considering previous decoding steps, encoder outputs of each time step (in our case each time step is a patient admission), and the last state of the encoder. This score is a representation of the amount of attention given to each sequence part regarding previous decoding steps; it is used as a form to interpret the model's decisions about relations between sequences in temporal data. In our case, for example, by analyzing alignment scores for trajectories with different lengths, we can

determine which lengths were more or less relevant to the prediction results.

To produce Figure 25, we trained our model over the MIMIC-IV-ED dataset, extracted the scores produced for the test split, and grouped into three: trajectories with up to 10 admissions (orange line in the figure), trajectories with up to 20 admissions (purple line in the figure), and trajectories with more than 20 admissions (green line in the figure). Our goal was to visualize, for each group of trajectories, what was the number of admissions where the attention scores were more relevant to the predicted next admission. We opted for the MIMIC-IV-ED dataset for this analysis instead of MIMIC-III because the former presents longer trajectories.

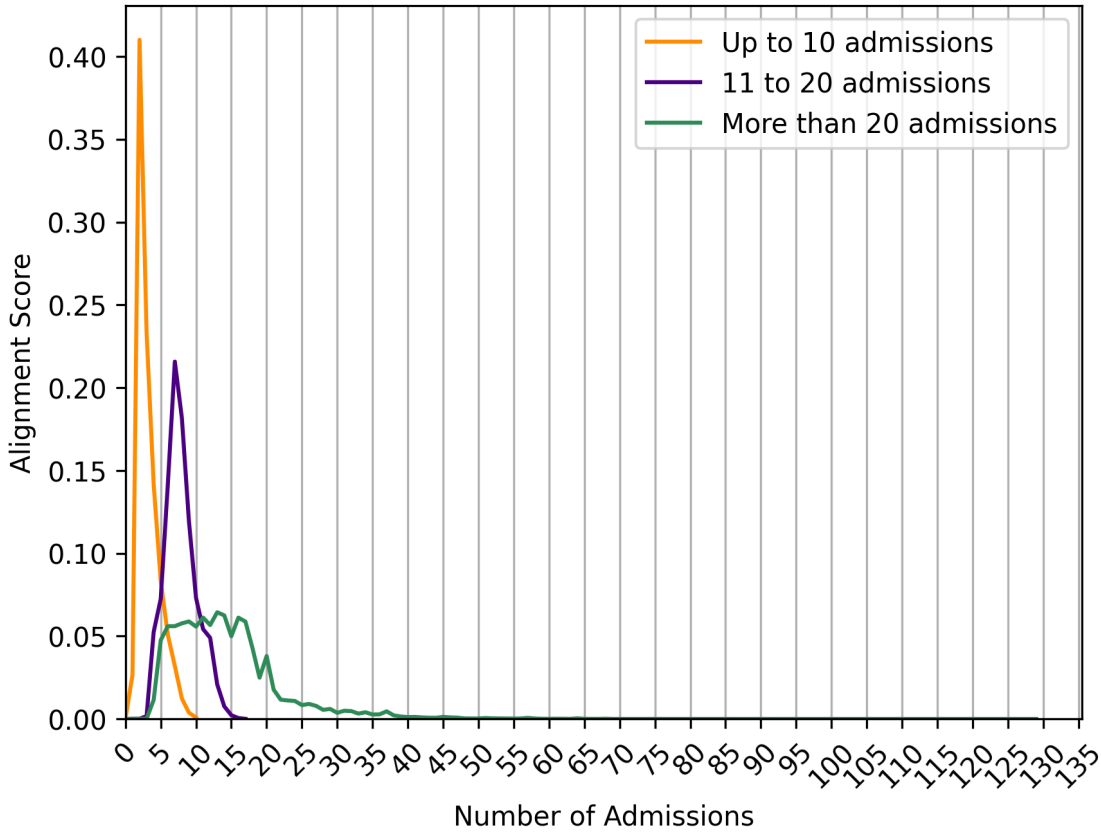

Figure 25 – Attention's alignments scores for trajectories with different numbers of admissions over MIMIC-IV-ED dataset.

Given Figure 9, we hypothesize that these results are derived from the number of relevant admissions for each one of the three groups of trajectories. For trajectories with up to 10 admissions, the first 5 admissions were the most relevant ones for the prediction task; for trajectories with up to 20 admissions, the attention focused on the first 5 to 10 admissions; lastly, for the longer admissions, with up to 20 admissions, the most relevant admissions were evenly distributed between the first 5 to 20 admissions. We consider that this evidence is an indication that long trajectories are independent of each other and could be split.

A.0.3 Hierarchical clustering linkages

We evaluated the dendrograms produced by each linkage criteria described previously in Chapter 2.2.1 to the best one. Figures 26 and 27 show the dendrograms obtained by each linkage criteria tested over the model predictions of MIMIC-III ICD-9, which were in Figure 26, from row 1 through row 4: Single, Complete, Average and Weighted linkages respectively, and from columns 1 through 3, distances Euclidean, City Block, and Minkowski, respectively; and in Figure 27, from row 1 through row 3: Centroid, Median and Ward linkages respectively, and from columns 1 through 3, distances Euclidean, City Block, and Minkowski, also respectively. Similarly, Figures 28 and 29 shows the same set of dendrograms over the model predictions of MIMIC-III CCS.

One point to notice is that some linkage criteria does not support other distance metrics besides Euclidean distance, e.g. Median Linkage with City Block distance, in this scenario we filled these plots with the text "Linkage criteria does not support this distance" in the figures.

Our goal with these figures was to select the best pair of linkage criteria with calculated distance. Our criterion was that the larger the distance between both sibling clusters and parent-to-child clusters the better, as described in Chapter 2.2.1. In this sense, we saw that some linkage criteria shown to be not suitable for our input data, *e.g.* Weighted linkage with City Block distance which presented a short distance between parent-to-child clusters, while other linkages as the Ward one with the Euclidean distance shown to be better suitable because presented both large parent-to-child clusters distance and large distance between sibling clusters.

Accordingly, we selected Ward linkage with Euclidean distance as the chosen linkage criteria for our hierarchical clustering method. This was because we visually verified that this pair distance-criterion was the one among all the other pairs distance-criterion with larger parent-to-child clusters distance and larger distance between sibling clusters simultaneously.

A.0.4 Optimal number of clusters

After choosing ward linkage, we ran an exploratory test to consider the optimal number of clusters for dendrogram cut of standard hierarchical clustering, hierarchical clustering with connection constraints, and other clustering methods such as K-means and Spectral clustering. Results are shown in Figures 30 and 31 for MIMIC-III ICD-9 predictions, while 32 and 33 shows the experiments for MIMIC-III CCS predictions. We ran our experiments in a range from 10 to 40 clusters because more or fewer clusters would not be suitable according to our domain understanding of the data.

Finally, we chose hierarchical clustering with connection constraints of 20-nearest neighbors and K-means clustering for qualitative analysis with t-SNE visualization (Figures 34 and 35 for MIMIC-III ICD-9 and MIMIC-III CCS predictions, respectively).

Our decision about these two clustering methods, and so the number of clusters chosen,

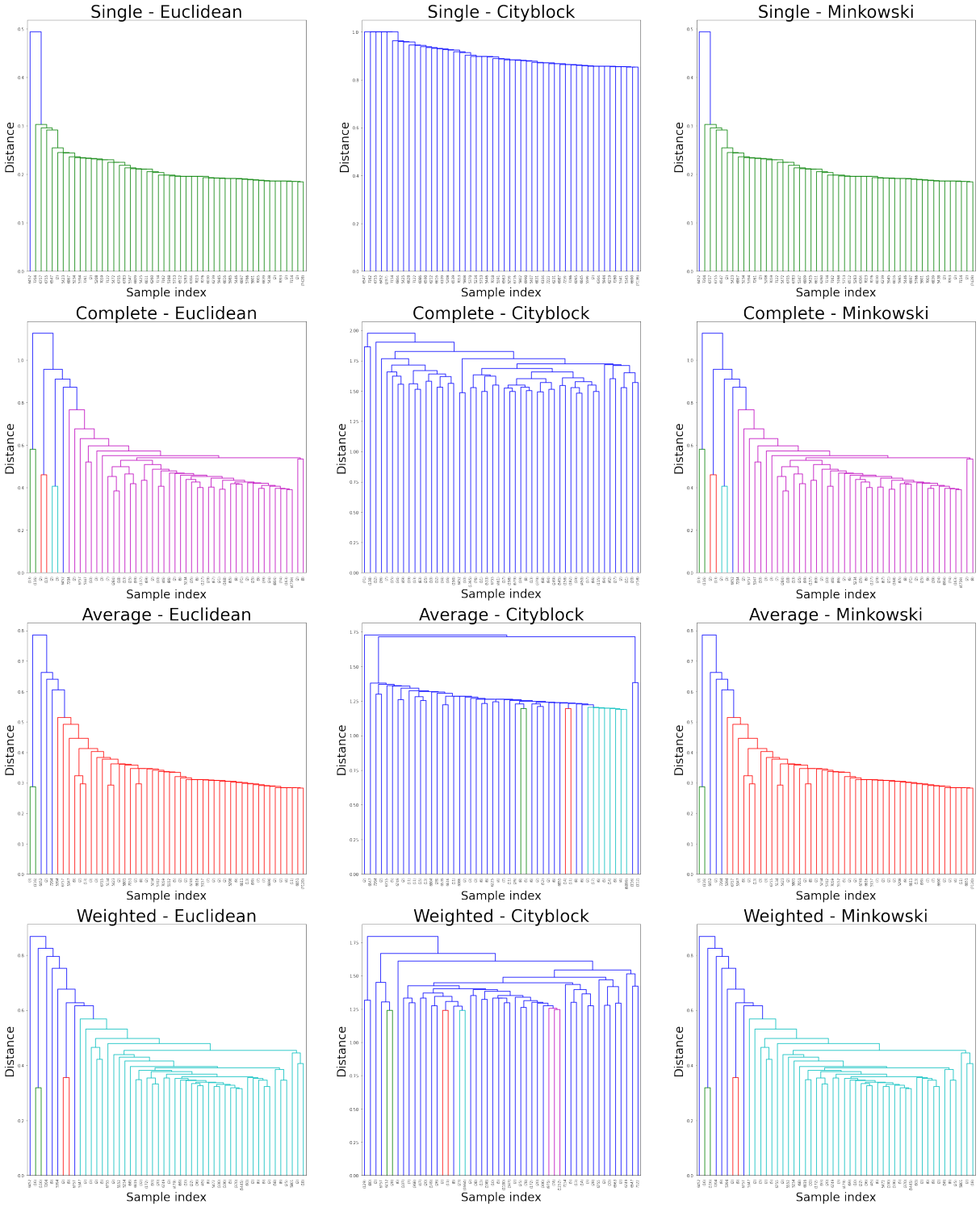

Figure 26 – Clustering linkage criteria over MIMIC-III ICD-9 comparison Part. 1.

were based on which method presented a consensus about the Silhouette score and DB index metrics for each number of clusters *i.e.* for which number of clusters the clustering method presented the maximum, or close to maximum Silhouette score and the minimum, or close to minimum DB index. Given these criteria, the number of 26 clusters for hierarchical clustering with connection constraints of 20-nearest neighbors; and 24 clusters for K-means over MIMIC-

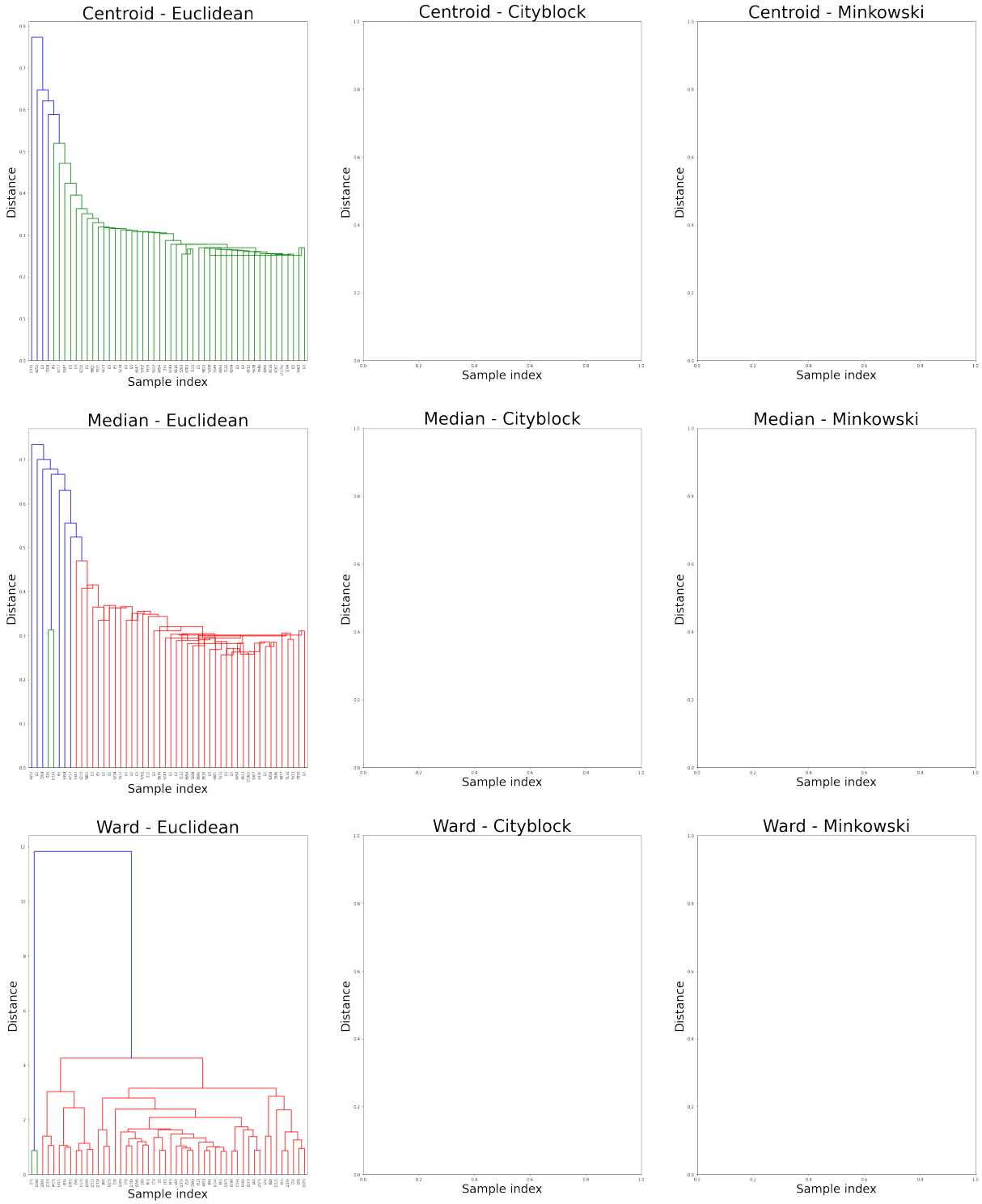

Figure 27 – Clustering linkage criteria over MIMIC-III ICD-9 comparison Part. 2.

III ICD-9 were considered the best results. Similarly, the number of 14 clusters for hierarchical clustering with connection constraints of 20-nearest neighbors; and 20 clusters for K-means were considered the best results over MIMIC-III CCS.

Furthermore, choosing fewer numbers of clusters for MIMIC-III CCS than for MIMIC-III ICD-9 is in accordance with the purpose of the CCS encoding standard of describing diagnoses

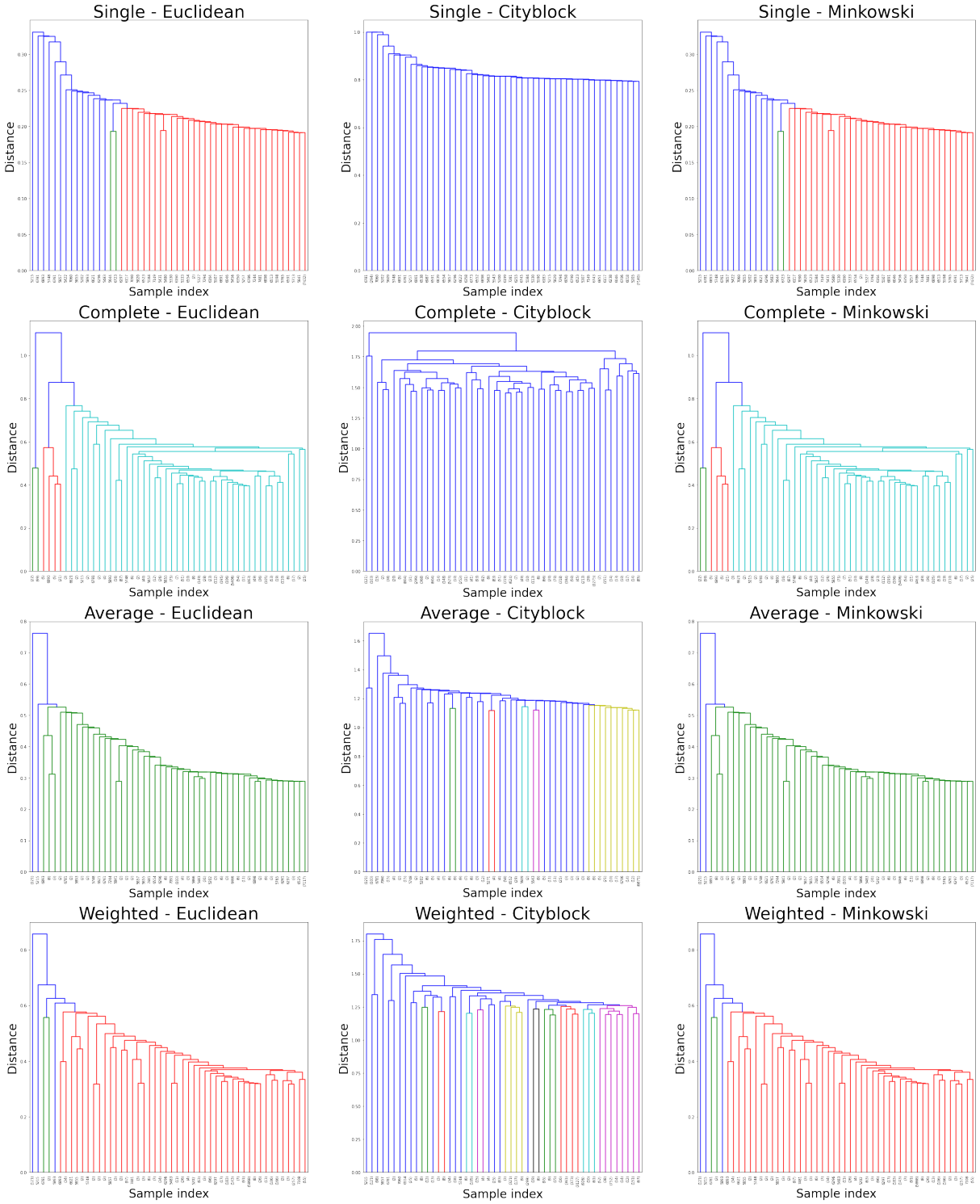

Figure 28 – Clustering linkage criteria over MIMIC-III CCS comparison Part. 1.

in a less granular way than the ICD-9.

Lastly, Figures 34 and 35 make it possible to visually analyze the better clustering formation, in these figures we saw that the two-dimensional data representation was condensed in a larger group of points, which was divided into less separable clusters, and in smaller better separable clusters.

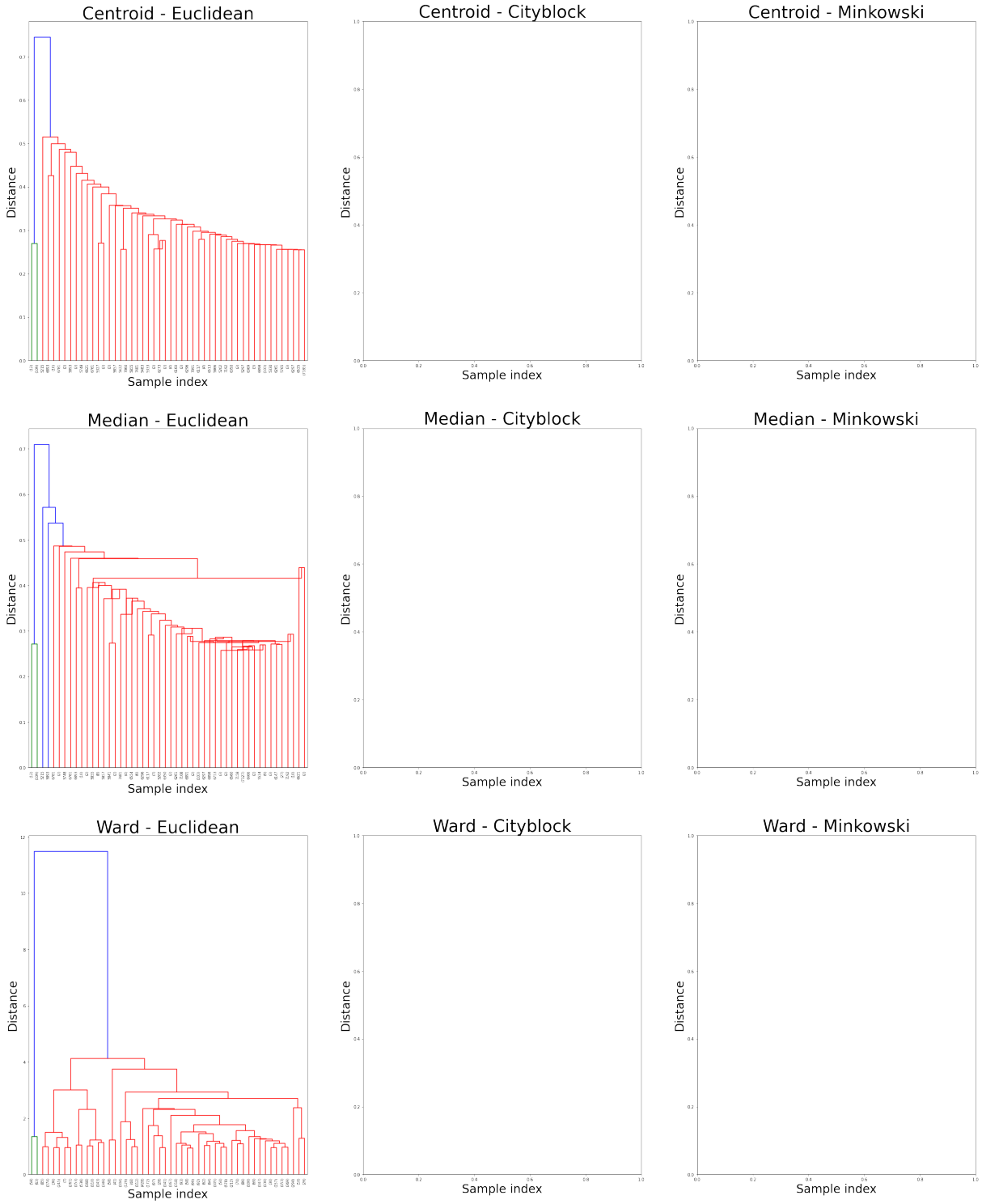

Figure 29 – Clustering linkage criteria over MIMIC-III CCS comparison Part. 2.

The main characteristic we noticed in both figures was that the Ward Agglomerative Clusterings visually presented more well-defined clusterings, and fewer outlier points, *i.e.* points which were attributed to a certain cluster but are visually distant to the other points from the same cluster than the K-Means Clusterings. These visual representations of the agglomerative method indicate to us a better clustering result in comparison with K-means since, despite the

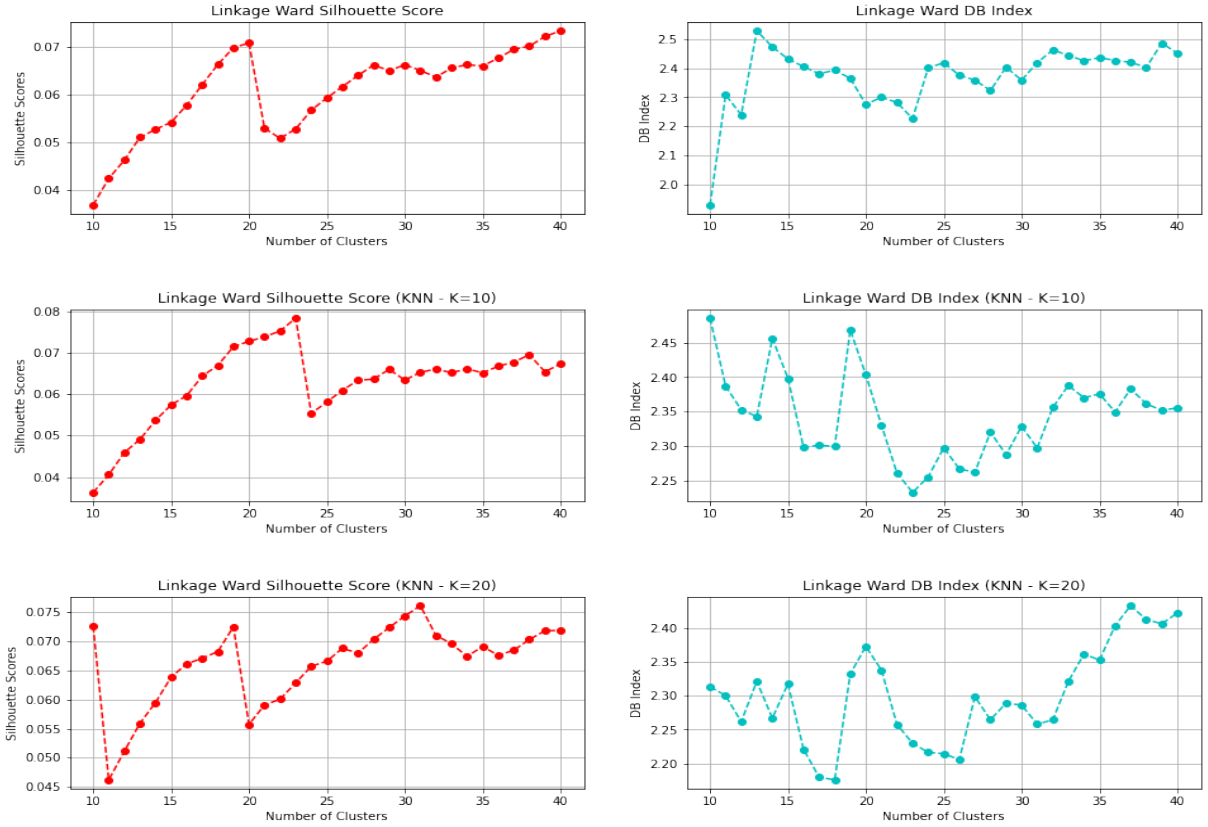

Figure 30 – Evaluation of the optimal number of clusters by cluster method over MIMIC-III ICD-9 Part. 1.

proximity between different clusters, the edge between these clusters is better defined.

As a result of the clustering method and the number of clusters we opted to use the less granular numbers of 26 clusters and 14 clusters for MIMIC-III ICD-9 and MIMIC-III CCS respectively, both of then obtained by the Ward Agglomerative Clustering. Moreover, we considered that these more well-defined clusters and the smaller number of clusters, which produced fewer labels, were a better choice considering the next step of our methodology.

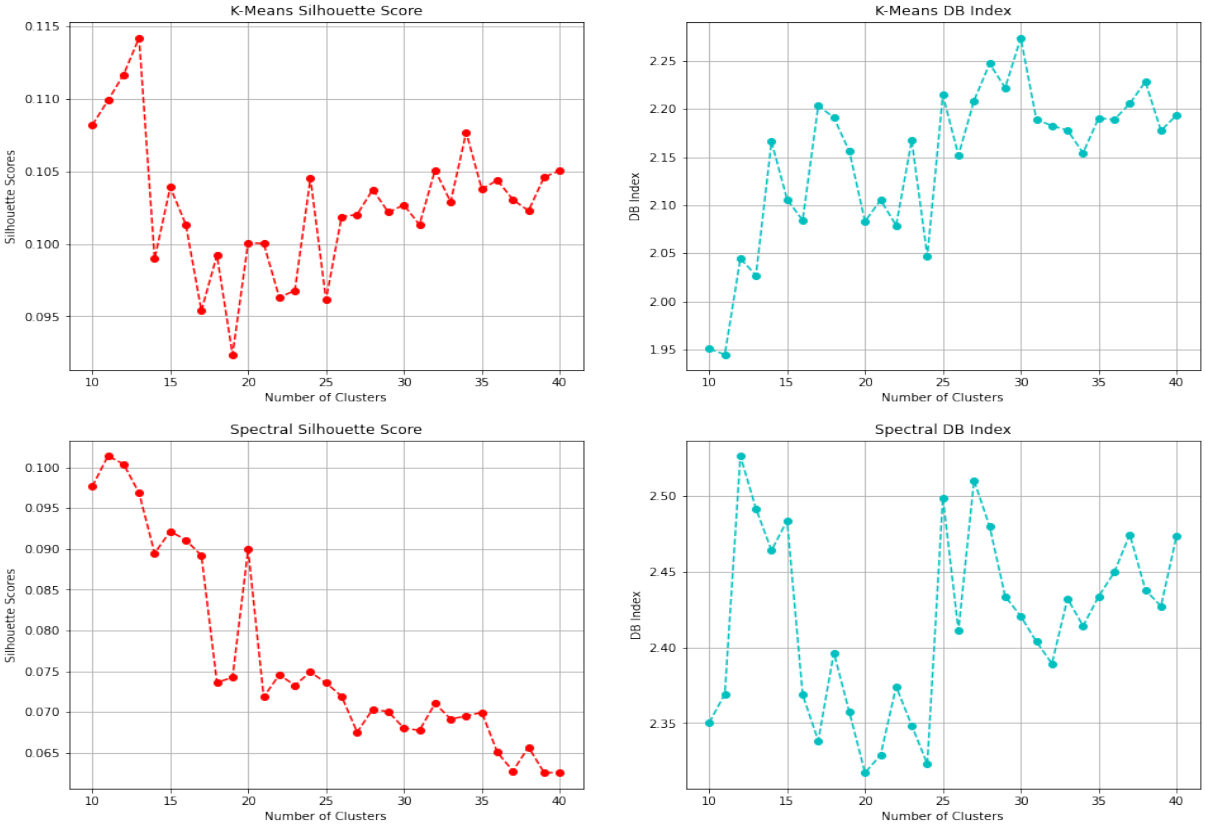

Figure 31 – Evaluation of the optimal number of clusters by cluster method over MIMIC-III ICD-9 Part. 2.

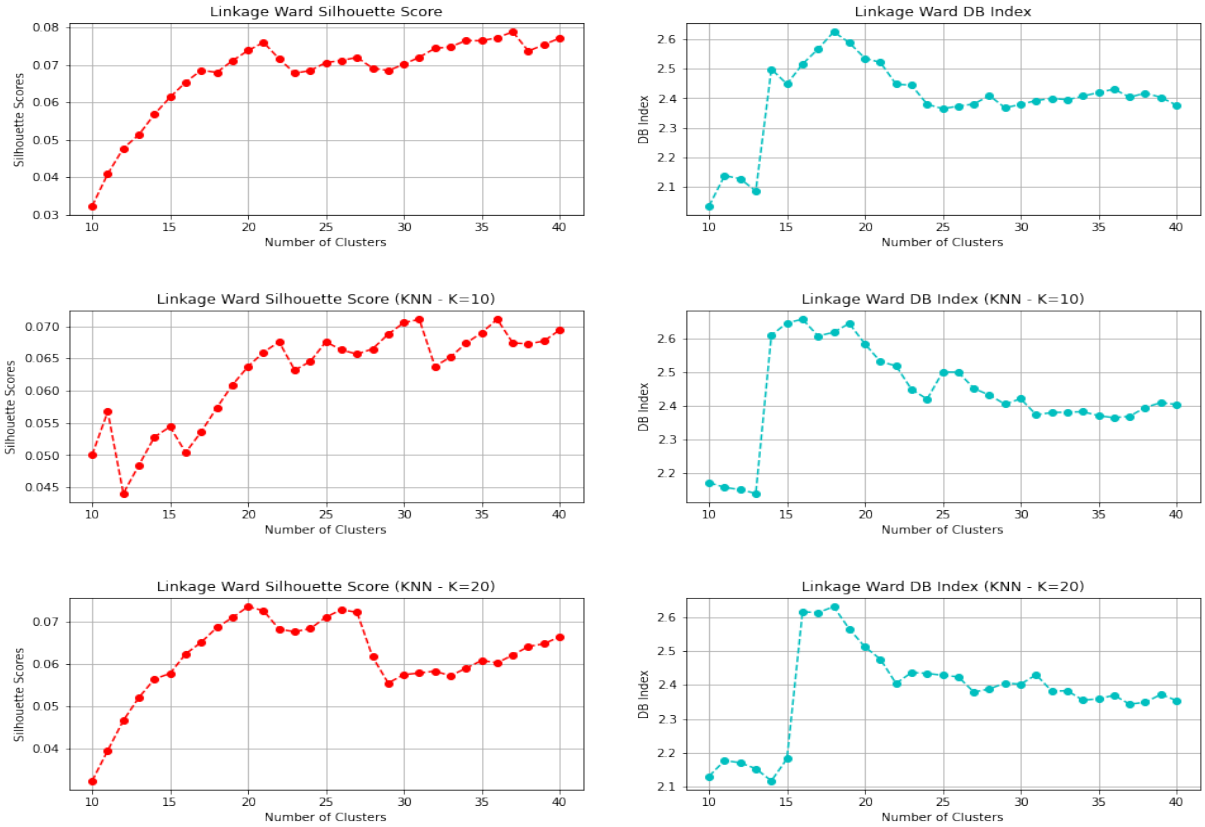

Figure 32 – Evaluation of the optimal number of clusters by cluster method over MIMIC-III CCS Part. 1.

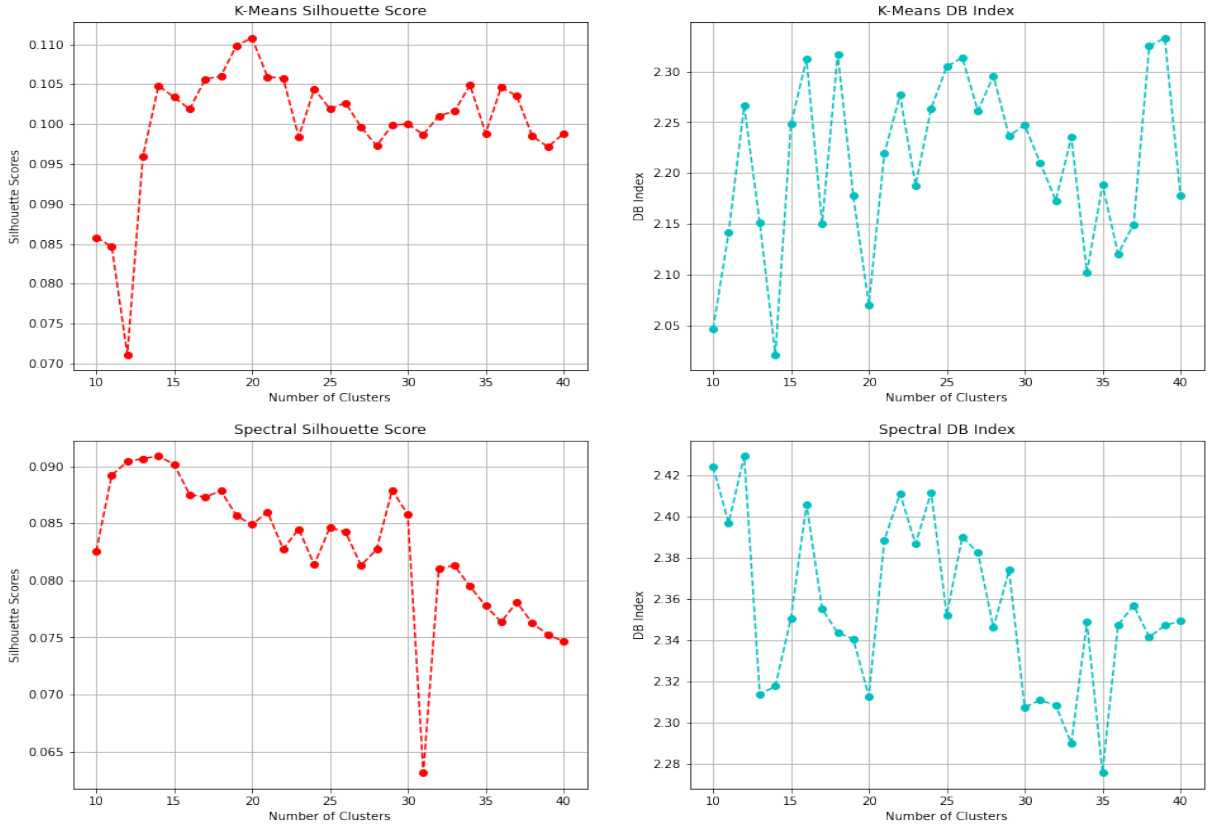

Figure 33 – Evaluation of the optimal number of clusters by cluster method over MIMIC-III CCS Part. 2.

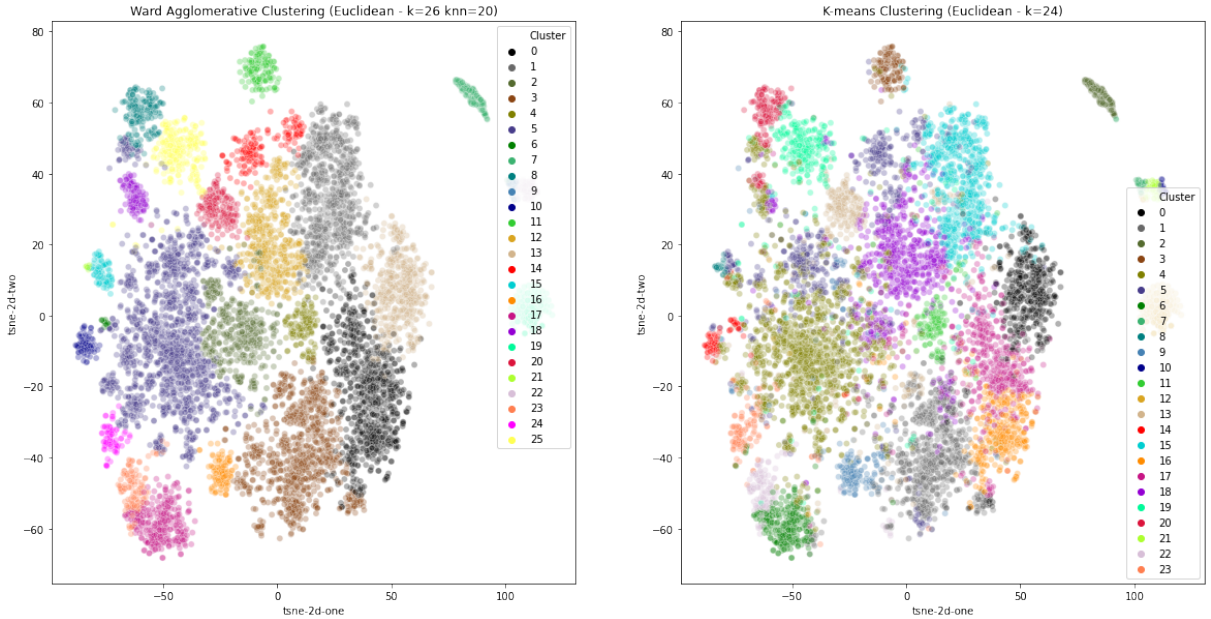

Figure 34 – t-SNE visualizations for methods hierarchical clustering with connection constraints and K-means over MIMIC-III ICD-9.

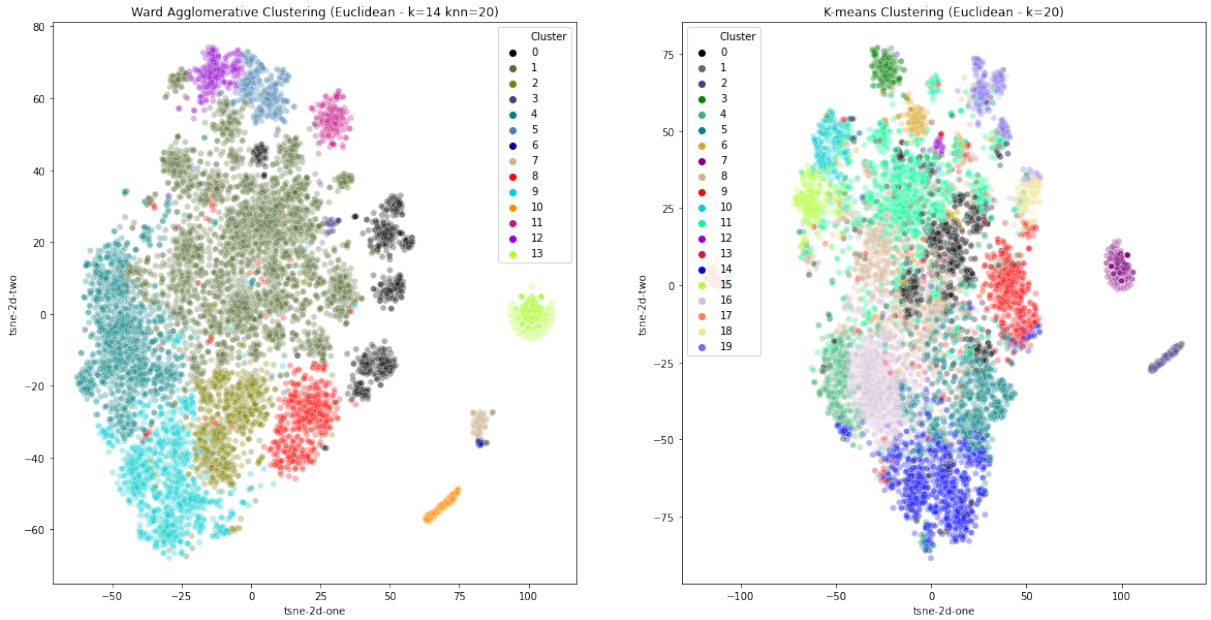

Figure 35 – t-SNE visualizations for methods hierarchical clustering with connection constraints and K-means over MIMIC-III CCS.


ICMC
USP
SÃO CARLOS