2021

Master

Thesis on Neural Networks and Transfer-Learning for Blood Glucose Forecasting

POLYTECHNIC UNIVERSITY OF TURIN

POLYTEHCNINC UNIVERSITY OF TURIN

Master's Degree in Computer Science

Master's Degree Thesis

Thesis on Neural Networks and Transfer-Learning for Blood Glucose Forecasting

Supervisors

Prof. Edoardo Patti

Prof. Santa di Cataldo

Prof. Alessandro Aliberti

Candidate

Sebastián Gómez

March 2021

Acknowledgements

I wish to thank my supervisor, Alessandro Aliberti, for his guidance and support while developing this Thesis. Also, Ph.D. Edoardo Patti for giving me the opportunity of working in his research group, it has been a marvelous experience. Besides, I would like to thank the following companies for their assistance, COLFUTURO, without them this dream would have never been possible.

Also, I would like to express my gratitude to my parents, Gustavo, Luz, and Andres, my brother, I can not express how much happiness they bring into my life.

Dearly,

Juan Sebastian Gomez Vidal

Contents

List of Figures

List of Tables

Abstract

The purpose of this thesis is to shed some light on the effects of *Transfer Learning* on *Blood Glucose Forecasting* Neural Networks for diabetic individuals, with emphasis on people suffering from the same type of diabetes but constantly experimenting with a specific constant diabetic state, like long periods of hypoglycemia, and across different type of diabetes, such as gestational diabetes. Two Transfer-Learning methodologies are chosen, an *Inductive* and a *Domain Adaptation* approach, which are applied over three well-known SOTA for three different forecasting horizons of 30, 60, and 90 minutes. Results are evaluated in terms of statistical metrics and the Clark Error Grid Analysis.

Preliminary findings indicate that *Transfer Learning* works on both cases across all horizons plus the *Inductive Approach* was able to reach better performances compared to the ones offered by the *Domain Approach*.

Furthermore, experimental data tend to guide to the conclusion that complex architectures are well suited for 30 minutes and 60 minutes horizons while simpler ones are best at 90 minutes, and that the biggest impact of *Transfer Learning* occurs at larger time scenarios.

Chapter 1

Introduction

1.1 Overview

According to the World Health Organization (WHO), *diabetes* is defined as "a chronic, metabolic disease characterized by elevated levels of blood glucose, which leads over time to serious damage to the heart, blood vessels, eyes, kidneys, and nerves when it is left without control" [1].

This is because diabetic people suffers of either absence of insulin or resistance to process it. Insulin is a hormone produced by the pancreas and is in charged of control sugar levels in blood. Basically, people with a not normal functioning pancreas are diabetic since they cannot self-regulate their blood sugar level. Therefore they must recur into external strategies to sustain their glucose levels into acceptable values [2].

Diabetic lives across twos states, one where there is a lack of presence of insulin on blood, high level of glucose, known as hyperglycemia. In this state, a person experiments thirst, excessive sweating, excessive urination, headache, etc. When this condition is uncontrolled over a long period, glucose levels pile up in the body leading to an extreme case of Ketoacidosis where a person gets dizzy, losses consciousness, and then pass away.

To tackle glucose, people usually take artificial insulin, orally, or injected. An excess of insulin, low level of glucose, becomes into a second state known as hypoglycemia, common symptoms of hypoglycemia are paleness, weakness, dizziness. Extreme low glucose levels can lead to a diabetic coma and then death.

The dynamic of glucose is directly proportional to a person's diet/metabolism (genetics) and its input amount (quantity). Although other variables are important too when talking about glucose production such as exercise, stress, sleep, etc. [3]

There are three types of diabetes: type 1, type 2, and gestational. In type 1 diabetes, a body produces very little or no insulin and occurs most frequently in children and adolescents but can develop at any age. In type 2, the body does not make use of the insulin that produces and it accounts for almost 90% of all diabetes cases, finally, there is gestational diabetes, which consists of high blood glucose during pregnancy and is associated with complications to both mother and child [4].

Diabetes is not a minor illness when referring to the last statistics provided by the International Diabetes Federation during 2019, at that time were approximately 463 million adults (20 - 79 years) living with this disease, it is expected a rise to 700 million by 2045. [4] Moreover, just in the USA during 2018 was estimated an economic loss of 327 billion USD where 237 billion USD corresponded to direct medical costs and the remaining 90 billion USD to reduced productivity [5].

So studying and researching new strategies and techniques to help diabetic people control their illness is of vital importance but before delving into *Transfer learning and Neural Networks* is worth mentioning how, in real life, healthcare practitioners and diabetic people cope with Glucose issues, and it is nothing more than through Continuous Glucose Monitoring Systems or CGM's as indicated in Figure 1.1.

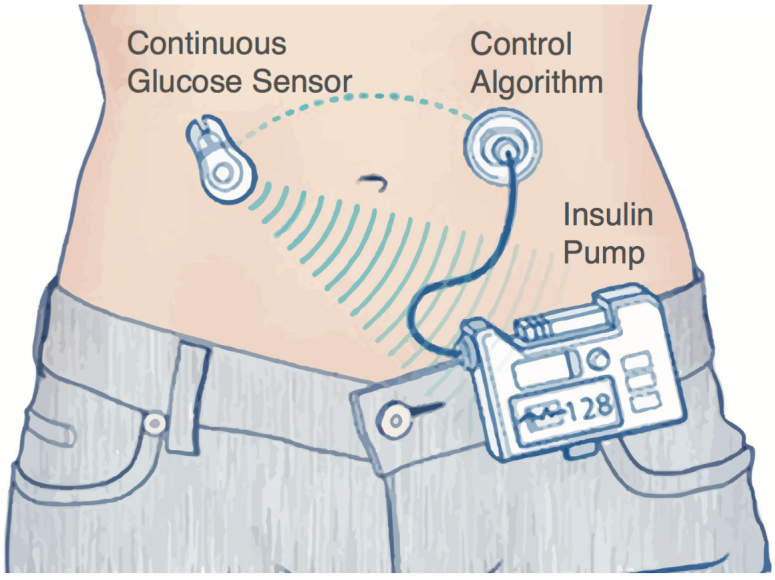


Figure 1.1: Diabetes Circuit

CGM's are the representation of closed-loops in Instrumented Systems, where there is a sensor measuring repeatedly a variable of interest, sugar levels in the blood, then transmitting this information into an embedded system, an electronic circuitry made of a battery, processors, and memory, which in turn runs a logic/program, commonly a Control Algorithm such as a Neural Network, to calculate current or future glucose values/trends, and producing results, decisions, or actions applied over some actuators like an insulin pump and/or a display.

Many investigations have been done in *Neural Networks* applied to diabetes and plenty of information on this topic is already available but little is know about *Transfer Learning*, one of the most promising techniques in the healthcare area, which is the focus of this thesis.

Usually, running studies and gathering information on the healthcare system is not a feasible task due to several complications such as patients are treated in multiple hospitals or clinics by different doctors, lacking availability of persons that meet certain selection criteria, privacy policies for treating individuals information, disposability of equipment and resources to gather information, follow-ups of people and measurements, so on and so forth.

Transfer-Learning helps in tackling the above-described pitfalls as its name implies by transferring a learned knowledge from one domain, let's say diabetes type I, to another different one like gestational diabetes, as to ease of breath, just as a human being extrapolates an ability to another field, for example, once it knows how to ride a bicycle, in no time will learn how to manage a motorbike. the advantages of transfer-Learning trebles because training time decrease, financial costs diminish and the volume of gathered information does not need to be high.

1.2 Related Work

As mentioned before, plenty of information has already been produced in the field of *Neural Networks* and Diabetes, researchers through extensive investigations, in past decades, have been able to categorize glucose predictors according to their underlying foundations, models explaining the pharmaco-dynamics of a human metabolism using complex mathematical formulas to correlate glucose kinetics with insulin are known as Physico-Chemical models, examples are the Bergman minimal model [6] or the Hovorka model [7], models based on Machine Learning, Artificial Intelligence and Time Series [8] [9] [10], are known as Data-Driven. Finally, blending the two previously described approaches gave birth to hybrid or compartmental models, which are predictors made up of cascading a physical model with a data-driven model [11] [12].

Results have proven that Data-Driven models surpass the other approaches in terms of accuracy, precision, implementation, generalization, and comprehension but they are demanding, both in terms of training data and computational resources.

Data-Driven models allow building predictors from a heterogeneous set of patients' information and recording devices, increasing the robustness of the model to unpredictable and unseen changes of the input signal [13]. Moreover, once a model is built, the device can be used on a new patient immediately, without re-training.

Fewer studies in literature explored the idea of creating a generalizable glucose level prediction model from a multi-patient training cohort. In [14] the authors propose an AR model with fixed coefficients (applying data filtering and Tikhonov regularisation [15]) and compare three different configurations: respectively i) models trained on each individual subject, ii) a model trained on different subjects using the same CGM's device, and iii) a model trained on different subjects using different devices. Their experimental results show comparable prediction errors for the three scenarios on a forecasting horizon of 30 min. Further developments of the same idea are presented by [16], this time using a model based on feed-forward ANN, and by [17], using a recurrent neural network(RNN). Nonetheless, the forecasting accuracy obtained by these works is still modest but acceptable despite the data used for the training is poor both in terms of number and type of patients, which intrinsically limits their generalization capability.

As opposed to *Neural Networks*, *Transfer Learning* is an uncharted area, few pieces of research have been created and few of them have focused on exploring its effects over a multi-patient predictor.

Stepping into *Transfer Learning*, one can define it as the process of first training a Neural-Network on an initial data-set to predict a domain task and then passing on the learned features (the network weights) to a similar secondary Neural Network to be trained on a target data-set to predict a different but related task. This concept has been thoroughly explored and applied mainly in Computer Vision, specifically to object classification, detection, segmentation, or even tracking, proving that Neural Networks' generalization capabilities can be improved.

For example, it has been shown how to positively use Transfer-Learning for brain tumor detection using models such as VGG-16, Inception, and Resnet-50 (Saxena et al., 2020 [18]). Also, utilizing pre-trained models like Resnet-50 along with Transfer-Learning has allowed achieving great results for pneumonia detection using x-rays images (Hossain et al., 2021 [19]).

Over and above this, some researchers have gone even deeper, into the transferability estimation for image classification problems, where they intend to develop a quantitative measure that ideally stresses how effective Transfer-Learning can be when moving from a source task to a target task *without training*. This would let knowing beforehand if Transfer-learning is applicable between two or more data-sets, detecting possible cases of positive or negative Transfer-Learning. That is, estimating some association degree among data-sets.

The most prolific outcomes had been developed by Tran, Nguyen, and Hassner [20], where they proposed their negative conditional entropy score, which relies on heavy statistical computations. Besides, on the same side Bao, Huang, Zheng [21] created another score metric called H score. Finally, The latest breakthrough on these types of measurements is the LEEP metric or Log Expected Empirical Prediction produced by Seeger, Hassner, and Nguyen [22] based on expected values and data-sets distributions.

Apart from Transfer-Learning applied to Computer Vision, little literature has been produced related to Time Series Classification (TSC) or even related to Glucose Prediction. Although, Neural Networks have gained some popularity for TSC tasks within the Time Series circle.

The most noticeable work for Transfer-Learning applied to Time Series is *Transfer-Learning for time series classification* done by Fawaz [23]. In this study, he explores the UCR data-set [24] which is the largest collection of information related to Time Series, containing around 85 different data collections, where he transformed state of the art networks used in Computer Vision to make them suitable for TSC, obtaining acceptable performances and confirming that Neural Networks are appropriate for time series.

When referring directly to glucose on blood, there are two main studies (Dubois et al., 2019 [25] and Bhimireddy et al., 2020 [26]). In Dubois's study, Transfer-Learning is applied from diabetes type-I patients to diabetes type-II patients, using shallow Neural Networks, mainly a simple Fully Convolutional Neural Networks comprised of a couple of layers connected to an output layer, whereas Bhimireddy focuses on a set of more complex architectures based on LSTM in a single patient approach, both studies used a single prediction horizon and the improvements achieved according to standard metrics were small.

1.3 Scope

The objective of this thesis is to explore and analyze the effects of *Transfer-Learning* over a well-known set of state of the art architectures for the glucose forecasting domain to establish a reference line, benchmark, that allow to measure and compare, in a fairly and equitable way, the real impact of Transfer-Learning on these architectures.

To accomplish this objective, the following premises are going to be taken:

1. Three prime architectures for glucose forecasting are going to be tested, the criteria for choosing these top-notch architectures are based on their performance achieved when comparing them against established metrics such as RMSE or MAPE.
   [Full explanation in chapter 4]

2. Three public available data-set will be used, one containing a large number of different glucose profiles (with hypoglycemia and hyperglycemia responses) related to people suffering from diabetes type-I across all ages and genders, a second data-set containing information for highly dominant hypoglycemia profiles on children and adolescents, and a third data-set holding governing hyperglycemia glucose profiles corresponding to women diagnosed with gestational diabetes.

3. Prediction horizons of 30 min, 60min, and 90 min are used to study responses and perform comparisons among architectures.

4. The amount of available data fetch for building predictors as well as doing Transfer-Learning, either for training or testing purposes, is going to be the same for all the three Neural Networks used in this study, regardless of the prediction horizon, locating all predictors and comparisons under equivalent conditions, drawing a line of reference to differentiate advantages and disadvantages among used models.
   [Full explanation in chapter 3]

5. Two *Transfer-Learning* techniques will be employed, one where predictors are fully re-train, known as an Inductive approach, and another one where training is made on a double stage, named Confusion approach.

The above premises would help to answer the next questions:

- Might transfer learning be successfully applied to data acquired with different equipment and conditions for patients with the same and different type of diabetes?
- Would model responses differ between the two Transfer Learning techniques?
- Which technique would reach better results in terms of performances and why?
- Will results have a logical consistency across prediction horizons?
- What is the best architecture inside the Inductive approach?
- How will performance change for the same architecture used in the Inductive and Confusion approach and how this compare to other responses?

Chapter 2

Data Analysis

2.1 Data-Sets

Three publicly available data-sets are used for the development of this thesis.The first two come from the JAEB CENTER FOR HEALTH RESEARCH [27], a nonprofit coordinating center for multi-center clinical trials and epidemiological research focusing on type 1 diabetes.

The first clinical trial contains data of 451 people with diabetes type 1 malady, with a variety of ages and therefore assorted with different glyacemic profiles, This data is gathered using continuous glucose monitoring systems (GCMS) of different commercially available brands (Abbott, Medtronic, and Dexcom) for a period of 6 Months, with a sampling rate of five minutes. This data-set is called "JAEB".

Gender	Age(years)	Patients(units)	Percentage(%)
male	8-14	72	15.96
	15-24	60	13.3
	>24	71	15.74
	subtotal	**203**	**45**
female	8-14	71	15.7
	15-24	83	18.4
	>24	94	20.86
	subtotal	**248**	**55**
	grandtotal	**451**	**100**

Table 2.1: JAEB Data-set

At first glance, one can say that there are a slightly higher number of women than men, 45 patients to be precise, the distribution of patients by sex and age are similar (between 15 - 20 % for each category) and it does not exist too many obese patients participating in the trial, just 16 (3.5 %), from which the majority are females whose age lies above 14 years old.

The second data-set is called "Tsalikian", it is a study developed on a cohort of 50 diabetic patients, with ages between 10 years old and 18 years old (youngsters), specialized in capture information about low blood sugar episodes at night after participants have exercised in the previous afternoon (Hypoglycemia profiles). The study collects information for each patient on 2 independent days (24 hours period), using as device a One-Touch Ultra GCM.

Gender	Age(years)	Patients(units)	Percentage(%)
	8-14	12	24
male	15-24	16	32
	subtotal	**28**	**56**
	8-14	7	14
female	15-24	15	30
	subtotal	**21**	**44**
	grandtotal	**50**	**100**

Table 2.2: TSALIKIAN Data-set

The above table depicts a higher number of men than those of women (28 versus 21) and that the population involved in the trial is frequently on the range from 15 to 24 years, around 62%, no obese people participated in this research.

Gender	Age(years)	Patients(units)	Percentage(%)
female	14-45	16	100

Table 2.3: AIDAS Data-set

Finally, the third data-set is a synthetic one produced with a computer-based software called AIDA [28], which is a freeware diabetes software simulator of blood glucose-insulin interaction, with the help of this software 16 different profiles for pregnant women with diabetes type I has been mimic according to different conditions like the number of carbohydrates eaten during a day, management of insulin (injections), patient's sensitivity to insulin, etc. The sampling rate is about every 10 minutes along one day (24 hours). All glucose generated profiles are highly hypoglycemics.

2.2 Data Wrangling

Before starting working directly with the previous data-sets, an initial inspection is done to check that all data-sets are structured similarly, which leads to look into features, which are :

1. PtID: Patient Identification Number, which is a positive integer, helps to conceal personal information about the participants who are simply associated with a number following data protection laws.

2. DeviceDtTm: Device Date Time, is a timestamp associated with every reading made by a device sensing glucose, its format corresponds to YY-MM-DD HH: mm: ss.

3. Glucose: Glucose readings are taken by a Continuous Glucose Monitoring Device, its unit is milligrams per deciliter of blood.

After verifying that all the information is structured in the same manner, a second inspection is done concerning the validity of the data. That is, a check for missing or null values, in case of detecting any abnormality, data is corrected following standard statistical procedures where no data pruning or rejection is allowed but instead replaced by mean values. Also, all features from all data-sets are expressed in the same primitive data types, leading to re-sampling Aidas data-set with a frequency of 5 minutes to guarantee data consistency among information.

Once all data is cleaned ans transformed, the remaining result is the JAEB data-set ends holding 772.061 records, the TSALIKIAN data-set 23976 records and AIDAS 4046 records, respectively.

Another important data factor to revise is the quality of glucose readings during the clinical trials, one way to empirically examine this factor is through samples continuity, in theory since the sampling rate is one sample every 5 min then on any test day a device should get a tally of 288 samples. Visually, this can be evinced by graphing information over the time axis, which is done for a random patient belonging to one of the above described data-sets.

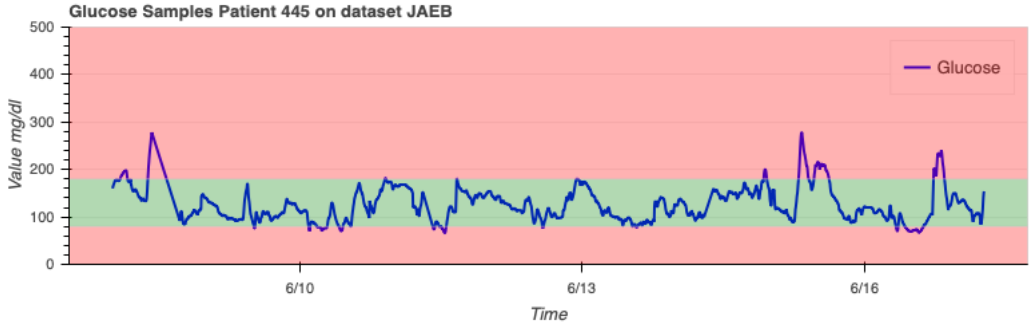


Figure 2.1: JAEB Data Continuity

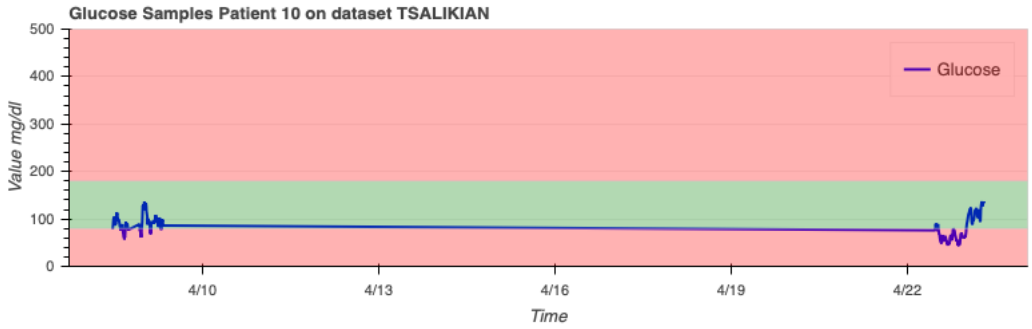


Figure 2.2: TSALIKIAN Data Continuity

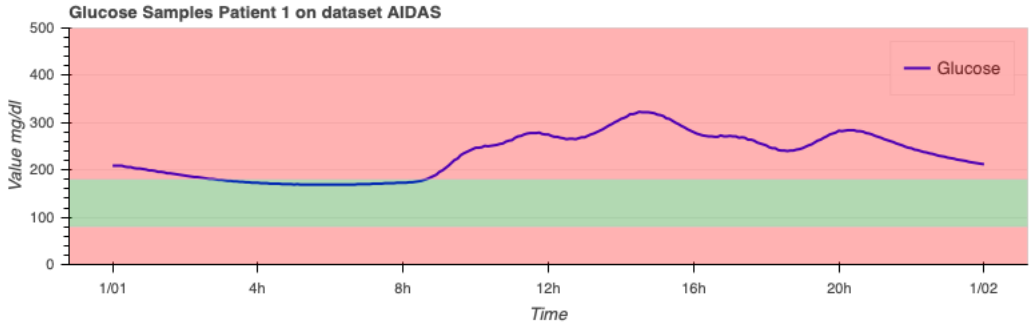


Figure 2.3: AIDAS Data Continuity

As shown in Figure 2.3, there is a beautifully smooth, straight line across the time axis, depicting a perfect continuous behavior for this random patient, nothing unusual since this data-set was synthetically generated. In contrast, Figures 2.1, 2.2 drew data with a non-continuous shape, this is inferred from the crooked and ragged lines, emphasizing a non-continuous nature on the samples taken for these random trial participants. This evidence is very important due to it will condition the way models ought to train and test in the future. A possible explanation for these anomalies can be participant's commitment or availability during the trial, failures on the monitoring devices related to calibration, power, memory, etc.

2.3 Data Statistics

To capture any relevant statistical information about glucose dynamics on diabetic patients, one ought to have in mind glucose variations are dependant on peoples' routines. Thus, it changes across the different parts of the day (morning, afternoon, and evening), something quite obvious taking into consideration, for example, the period when a person sleeps, here its body is not consuming large amounts of energy (does not need glucose, therefore, insulin) since it is static, on the other hand, during morning or afternoon times, the person is fully active and requiring a lot of energy to develop its day to day activities. This statement gets back up also with the carbohydrates consumption, at night generally is not high in contrast with day-light behaviors (not always the case).

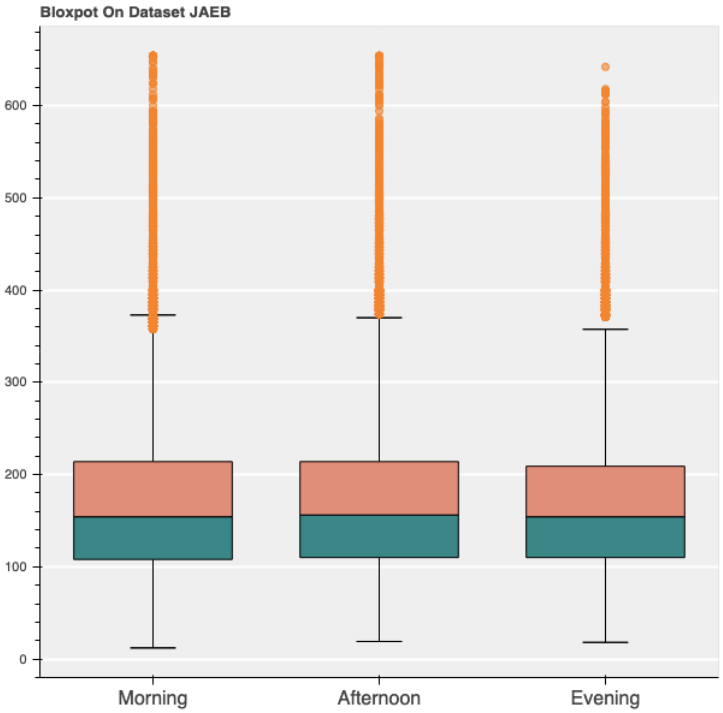


Figure 2.4: JAEB Glucose Box-plot

Metric	Value
count	772061
mean	167.32
std	76.92
max	654
75%	212
50%	154
25%	110
min	12

Table 2.4: JAEB Statistics

Figure 2.4 and Table 2.4 shows that in the JAEB data-set glucose profiles had a similar response during the three parts of the day (equal distribution of points), having average glucose mean around 167 mg/dl, which falls into a normal range between 150 mg/dl and 170 mg/dl. Also, dispersion is about 76 mg/dl (quite high). What's more, one can mention how close the 50 percent of readings are to the mean value, 154 mg/dl, thus patients belonging to this data-set had problems managing a stable glucose profile but it appears that they were able to stay on the safety zone. One interesting remark is how the fourth quartile accentuates the fact of predominant hyperglycaemic profiles presence over hypoglycaemic responses, only 25 percent of glucose values locate below 110 mg/dl and the minimum value corresponds to 12 mg/dl.

Another important remark is the presence of outliers in Figure 2.4, especially when you have a max value of 654 mg/dl, a thorough look could indicate that may be a subset of this study group were experiencing *"Hyperosmolar Hyperglycemic Nonketotic Syndrome"* which is a dangerous condition occurring when glucose goes over 500 mg/dl.

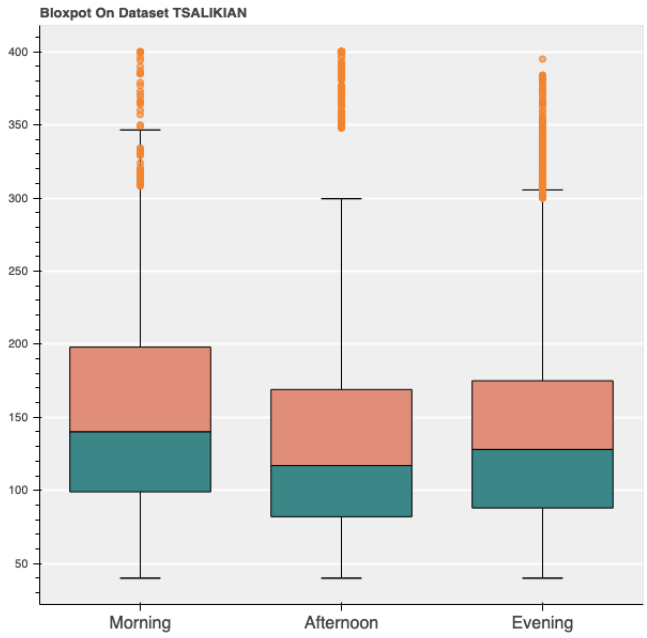


Figure 2.5: TSALIKIAN Box-plot

Metric	Value
count	23976
mean	142.85
std	69.24
max	400
75%	182
50%	129
25%	89
min	40

Table 2.5: TSALIKIAN Statistics

Passing to the TSALIKIAN data-set, which is mainly composed of children and adolescents, one can clearly distinguish a trend in which right after doing exercise (afternoon) glucose values remain on the safety zone, right after in the evening, these values rise a couple of mg/dl but in the morning reach its highest value, Figure 2.5. This could maybe lead to an empirical demonstration of how exercise can impact glucose responses, at least in children and adolescents. Nevertheless, other important points to highlight are the appearing no existence of *"Hyperosmolar Hyperglycemic Nonketotic Syndrome"*, the maximum value registered was 400 mg/dl. Besides, the minimum glucose value was 40 mg/dl, really close to the 58 mg/dl suggested by experts as the lowest value.

The mean average is about 142 mg/dl with a standard deviation of 69 mg/dl, relatively lower when compared to the statistics from the JAEB group.

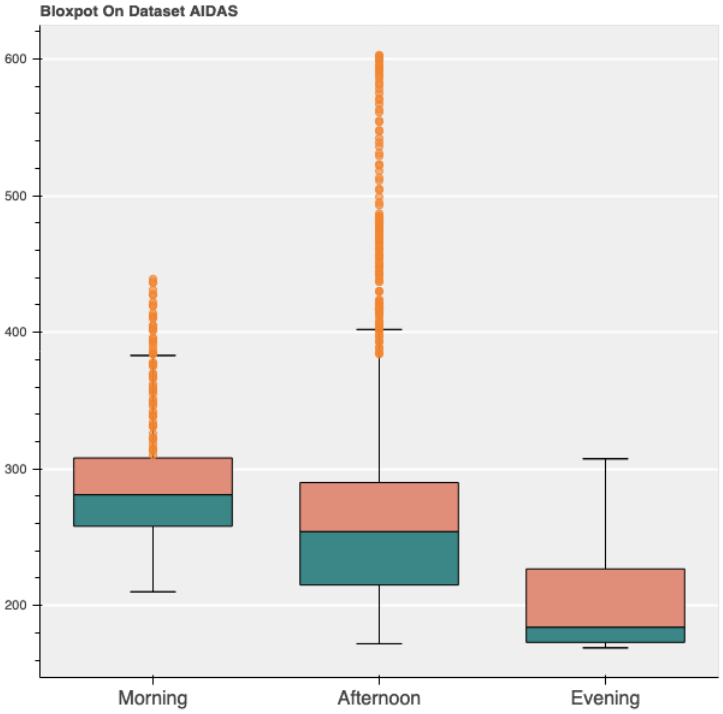


Figure 2.6: AIDAS Box-plot

Metric	Value
count	4046
mean	254.87
std	71.05
max	603
75%	288
50%	252
25%	196
min	169

Table 2.6: AIDAS Statistics

Figure 2.6 and Table 2.6 reveal interesting dynamics for pregnant women with diabetes type I. they point out a trend where glucose values decrement from morning to evening and increase from evening to morning. It appears these women never get to stay inside the glucose safe zone due to the majority of the data drops into the first and the second quartile, which is between 196 and 252 mg/dl glucose values, and the minimum value is 169 mg/dl. Although these profiles seem extreme, they are quite plausible since suffering from diabetes type I while bearing a child must not be an easy task to manage.

These DATA-SETS were carefully chosen to depict a close and far relationship among them, based on the basic statistical measurements prior explained has been probed this fact, where JAEB and TSALIKIAN are both similar and at the same time dissimilar to AIDAS, something very important for further analysis on transfer learning.

Even though the above statistical measurements threw some insight into the type of population involved in these studies and their behaviors, it is not enough for a complete understanding. To comprehend what real-life glycaemic profiles in diabetic individuals are like, one must first ask itself about how a normal glycaemic profile of a non-diabetic individual should be like, one can refer for example to some scientific studies such as Continuous Glucose Monitoring Profiles in Healthy Non-Diabetic Participants [29] to grasp some initial apprehension.

These studies had probed that in general non-diabetic people have to mean average blood glucose between 98 and 104 mg/dl with a coefficient of variance of 17 ± 3%, besides state that a no-diabetic individual spent around 96% of the time in a range between 70 to 140 mg/dl (what is called time in range), 2.1% of the time in ranges above 140 mg/dl (what is called time above range) and 1.8% of the time in ranges below 70 mg/dl (what is called time below range).

People suffering from any type of diabetes like type 1, type 2, gestational diabetes, etc, experiments variations in time of glucose, these are completely different from non-diabetic persons and even this type of fluctuating response varies from one diabetic person to another, principal because of two reasons, the first one relates to the inner physical workings associated to genetics such as metabolism, nervous systems, so on and so forth, and the second one related to external factors like food intakes - diet (carbo or fat-based), physical activity, undergoing medication and stress, as explained some lines above.

Taking into account these details, some important questions to ask are:

- How many people involved in these trials do get close or get away from the values of a normal person?
- How is the distribution of ranges among patients?
- How many people have similar glucose profiles ?

All these questions are important since intuitively would help us to evaluate the effectiveness of a transfer learning approach by classifying the type of people present in each data-set, in other words, will give us a guess on which type of people with diabetes can our prediction models be more precise and accurate, therefore narrowing the possibility of getting a negative transfer in favor of a positive transfer learning approach.

Although there isn't a consensus about to which degree should a diabetic might intend to behave as non-diabetic, many practitioners specialists agree that a diabetic ought to spent around 80% of its time in range, that is in values between 180 to 70 mg/dl, 3% of its time in values under 70 mg/dl or below range, and the remaining time on values above or above range. [30]

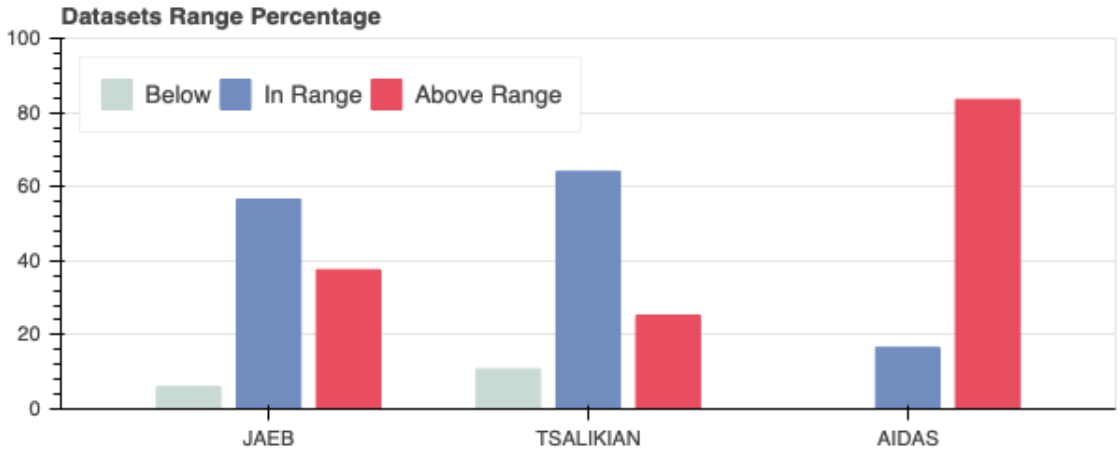


Figure 2.7: Glucose Time Range

Complementing the statistical information written before, one can say from Figure 2.7 that all participants' glucose profiles have a high degree of volatility or fluctuation, that is why the huge values on the standard deviation on each data-set, highlighting either type I diabetic is a disease hard to control and manage or that the control methods employed by these patients are not good enough or that patients were not so diligent taken care of its situation or might be a combination of all previous conditions.

In terms of numbers in JAEB data-set 17 out 451 patients had profiles similar to those of a non-diabetic, in the TSALIKIAN data-set only 2 out of 50 and from the AIDAS diabetic pregnant women, none.

This could imply that future prediction models could work relatively precise and accurate for people with non-stable glucose profiles (AIDAS) whereas would behave poorly for people with stable ones (TSALIKIAN).

Overall, the patients from all data sets spent less than 80 percent of their time in range in average, to be precised between 56 and 60 percent for the JAEB - TSALIKIAN pair and less than 16 percent for AIDAS. Moreover, it appears to be a tendency to spent around 36 and 28 percent of time above range in the cases of JAEB and TSALIKIAN while around 80 percent for AIDAS data-set. at last, TSALIKIAN and JAEB pass around 12 and 8 percent of their time below range, respectively. Pregnant women does not appear to suffer from hypoglycaemic periods.

2.4 Data-Sets Similarity

Comparing time-series is not an easy task. Traditional methods like Manhattan or L2-norm distances do not apply well to time-series since they are suitable across the y-axis (magnitude axis) but x-axis (time axis). Thus, Dynamic Time Warping is a good alternative due to it is a technique that allows a comparison of two time-series of different lengths and quantifies its similarity or dissimilarity through a distance calculation, solving the problem on the time axis [31]. Moreover, old-fashioned methods require that both time series have the same time length, another drawback that Dynamic Time Warping solves.

Dynamic Time Warping would correlate diabetic type I patients by associating an average blood glucose level with a corresponding DTW distance for each patient against patient 445 from JAEB. Both values will lead to the construction of a scatter graph.

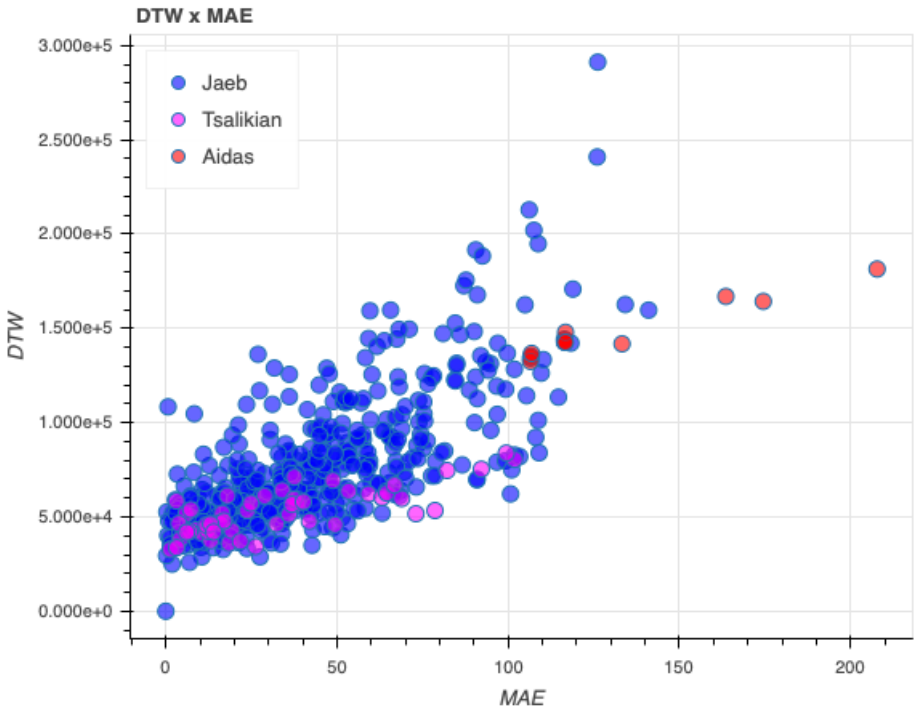


Figure 2.8: Dynamic Time Warping

As shown in Figure 2.8, JAEB and TSALIKIAN data overlap, which implicitly indicates that both data-sets are similar, in other words, contain blood glucose profiles akin. On the other hand, AIDAS' data-set has no similarity to TSALIKIAN but keeps some resemblance with JAEB, especially with some profiles that are on the outskirts of the JAEB' cluster

Chapter 3

Data Preprocessing

3.1 Data transformation

Deep Learning Neural Networks learn how to associate inputs to outputs from samples in a training data-set. Weights are initialized to small random numbers and updated via back-propagation through an optimization algorithm according to error estimation on the training data-set. When input and output data are unscaled and its magnitude is relatively high, computations to estimate an error and re-weight variables and biases with back-propagation get slower in time and less efficient in terms of prediction capability, guiding the Neural Network towards an unstable learning process or failure, e.g presence of exploding gradients, since networks with large weight values suffer from poor performance during learning and sensitivity to input values resulting in higher generalization error.

One option to overcome this difficulty is scaling down data, in the world of Machine Learning there is a variety of methods to achieve this but there are two favorite methods: Normalization and Standardization.

In Normalization, data is scaled according to the maximum and minimum values present on the data-set, meanwhile on Standardization, data is origin-centered using the mean value and the standard deviation of the samples. A visual demonstration is plotted next for the JAEB data-set.

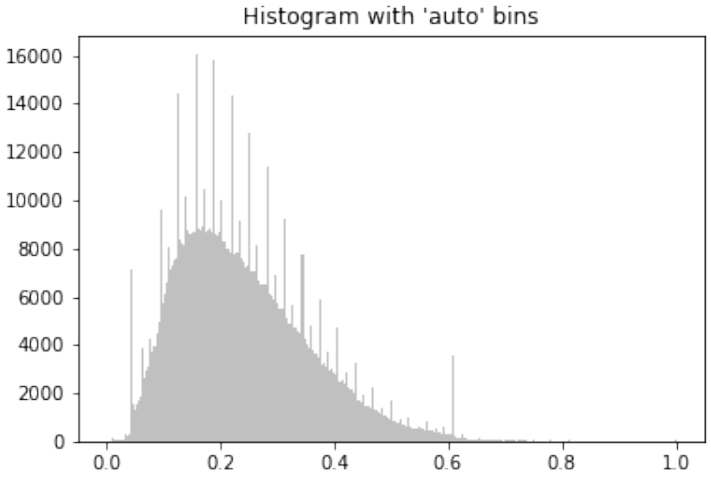


Figure 3.1: JAEB Normalized

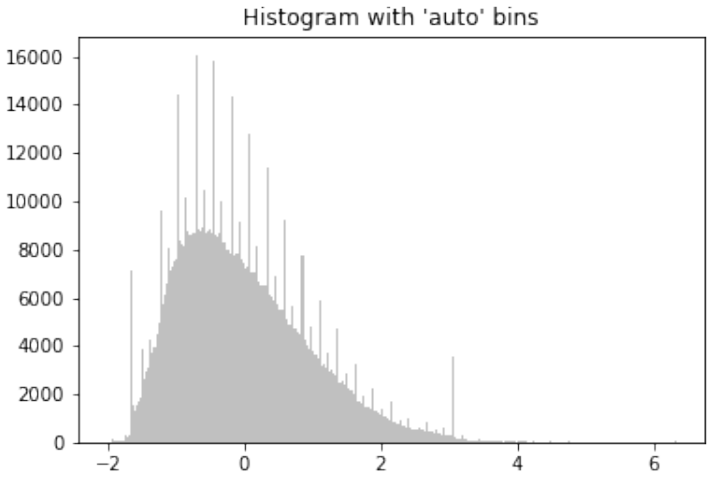


Figure 3.2: JAEB Standardized

As seen in Figures 3.1 and 3.2 , normalized data values span from 0 to 1 whereas standardized data goes from -2 to 4. In order to pick a method, a little experiment was put in place through a simple ANN using a part of the JAEB data-set and measuring RMSE and R2. Results probe that standardization works a little better than normalization judging by metrics, RMSE was down almost 1.5 units and R2 gained 0.8 percent when compared, respectively.

This can be explained by a simple fact, as mention earlier, diabetic people could experience "Hyperosmolar Hyperglycemic Nonketotic Syndrome" or what is the same, have glucose readings above 500 mg/dl. The maximum reading on JAEB is about 654 mg/dl, in TSALIKIAN is around 400 mg/dl, and in AIDAS somewhere near 600 mg/dl. Taking as a reference point the previous Figure 3.1, it would be the same for the other data-sets, one can see that the vast majority of points concentrate far from the maximum value, so when normalization is used, the gruesome part of the points would be very small when compared to the maximum value since they are divided by a huge number, this situation affects the Network because weights would fluctuate between large and small weight values when training, therefore the Neural Network performance inevitably should worsen.

As a repercussion of the above revelation in this Thesis is going to be used the Standardization method.

3.2 Selection of Training and Testing Samples

Frequently in clinical trials data, the gathered information is split by patient, that is, each participant's data is divided into training, validation, and testing when working with Machine Learning models. A different edge is selected in this study. Data for training and testing is gonna be picked on a patient's basis rather than dividing each patient's information. This approach is convenient from the point of view of data management and necessary due to the non-continuous nature of the data itself. In other words, a fraction of patients are selected for training only and the remaining part for testing only, increasing the chances of avoiding overfitting or underfitting situations. the selection ratio for splitting data-sets is 70% for training and 30% for testing. Selection is done randomly, over each band of ages within the given data-sets.

JAEB training set:

[2, 3, 4, 6, 8, 9, 11, 12, 16, 17, 19, 20, 22, 24, 26, 28, 31, 32, 33, 37, 39, 42, 43, 46, 47, 49, 50, 53, 54, 56, 57, 59, 60, 61, 62, 63, 64, 66, 69, 72, 73, 76, 77, 78, 81, 85, 87, 88, 90, 93, 94, 95, 96, 97, 100, 101, 104, 105, 107, 109, 112, 113, 114, 116, 117, 118, 121, 122, 123, 127, 128, 131, 132, 133, 135, 136, 137, 138, 139, 140, 142, 143, 144, 145, 146, 147, 148, 149, 151, 152, 154, 155, 156, 157, 158, 161, 164, 165, 167, 168, 169, 172, 173, 175, 176, 177, 178, 180, 181, 183, 186, 187, 189, 190, 191, 194, 195, 196, 197, 198, 199, 200, 202, 203, 204, 205, 207, 209, 210, 211, 212, 213, 214, 216, 218, 219, 220, 221, 223, 225, 226, 227, 229, 231, 232, 235, 236, 237, 239, 240, 241, 243, 244, 245, 246, 247, 248, 250, 251, 253, 254, 255, 256, 259, 260, 261, 263, 264, 265, 266, 267, 268, 269, 270, 273, 279, 280, 283, 286, 287, 290, 291, 293, 294, 296, 300, 301, 304, 305, 306, 307, 308, 311, 312, 313, 314, 315, 317, 318, 319, 320, 322, 323, 324, 325, 327, 331, 333, 335, 339, 340, 341, 342, 343, 344, 346, 347, 351, 353, 354, 355, 357, 359, 360, 362, 363, 364, 368, 369, 370, 373, 374, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 388, 390, 394, 395, 396, 397, 398, 399,

400, 401, 402, 404, 405, 407, 409, 410, 413, 416, 418, 421, 422, 423, 425, 428, 429, 430, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 445, 447, 449, 450, 451, 454, 459, 460, 461, 462, 463, 464, 466, 468, 469, 470, 472, 473, 474, 475, 476, 477, 479, 484, 485, 486, 488, 490, 491, 495, 497, 498, 501, 502]

JAEB testing set:

[1, 5, 7, 10, 13, 14, 15, 18, 23, 25, 27, 29, 34, 35, 38, 40, 41, 44, 45, 51, 52, 55, 58, 65, 67, 70, 71, 74, 75, 79, 80, 82, 83, 84, 86, 89, 91, 92, 98, 99, 102, 103, 108, 111, 115, 119, 120, 130, 134, 150, 153, 160, 162, 163, 170, 174, 185, 188, 192, 193, 206, 208, 215, 217, 222, 230, 233, 238, 242, 252, 257, 258, 262, 271, 275, 276, 277, 281, 282, 284, 288, 289, 295, 298, 302, 303, 309, 310, 321, 326, 336, 337, 338, 350, 352, 356, 358, 361, 367, 371, 372, 386, 387, 389, 391, 393, 403, 406, 411, 417, 419, 420, 424, 426, 427, 431, 444, 446, 448, 455, 456, 457, 458, 465, 467, 471, 478, 481, 482, 483, 487, 489, 494, 496, 499, 500, 503]

TSALIKIAN training set:

[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 23, 24, 25, 26, 27, 28, 32, 33, 35, 36, 38, 43, 44, 48, 50, 52, 53, 54]

TSALIKIAN testing set:

[1, 13, 19, 20, 21, 22, 29, 30, 31, 37, 39, 40, 41, 42, 45, 46]

AIDAS training set:

[1, 2, 3, 8, 10, 12, 13, 14, 15, 16]

AIDAS testing set:

[4, 5, 6, 7, 9, 11]

3.3 Segmentation

As delineated several times before, data present on the data-sets are not continuous, but what does it mean is not continuous? simply means that on any random patient, time samples collected are discontinuous, that is, it might have few readings in the morning, then other more on the evening, or even some readings in one day, none on the following one, and so forth. This makes data splitting a cumbersome task since data is rather heterogeneous because the amount of time samples varies not only within any specific day but between consecutive days, hence no uniform form can be implemented to train and test Neural Networks.

A proposed solution in preceding studies over Time Series Classification like [32] [33] suggests the creation of segments holding a minimum amount of sequential samples. This number is got by considering the minimum number of consecutive time steps that a Neural Network needs to be fed plus the adjacent values that will serve as predictions.

Fellow researchers have experimented with several values for time steps input, obtaining valuable results demonstrating the trade-off between the input size and the accuracy of predictions. Perez-Gandia et al., (2010) [16] chose 20 input steps, Daniels et al., (2019) [33] picked 24 and Pupillo et al., (2019) [32] selected 30.

In this research, the approach taken by Pupillo et al., (2019) [32] is going to be followed for consecutive input step selection due to their models' outstanding responses when compared to other works. Now, given that different prediction horizons are intended for evaluation, then at least 18 extra samples are needed as prediction, which corresponds to the largest horizon of 90 min, for a total of 48-time samples per segment.

Considering the inner working of a Neural Network now is a whole complete story, the best way Neural Networks learn patterns is by sliding a time window over a segment one step at a time, with 48 samples this is not possible, the stride equals zero, thus an increase in the number of samples is required. Thinking on the quality of the segments, heuristically an appropriate number criteria for segment generation is about 70 continuous adjacent samples which are equivalent to have six continuous hours of data. This number would guarantee that at least a stride equals to 22, enhancing its pattern retention capability.

JAEB	**Training**	**Testing**
Segments	913	338

Table 3.1: JAEB segments

TSALIKIAN	**Training**	**Testing**
Segments	68	32

Table 3.2: TSALIKIAN segments

AIDAS	**Training**	**Testing**
Segments	7	5

Table 3.3: AIDAS segments

3.4 Data Windows

Once the respective segments are computed for each of the given data-sets, it is time to produce slices of time windows. Having in mind that one of the most important goals is to train and test every Neural Network with the same amount of data, every segment is sliced taking into account a feed input equal to 30-time steps plus an offset of 18 sequential time-steps. To put in practice the premise that all Networks are going to be fed with the same amount of data when evaluating for different prediction horizons, each of these produced slices will be sub-sliced again with an offset matching the proper prediction horizon understudy, this would place all predictors under equivalent conditions for future comparison.

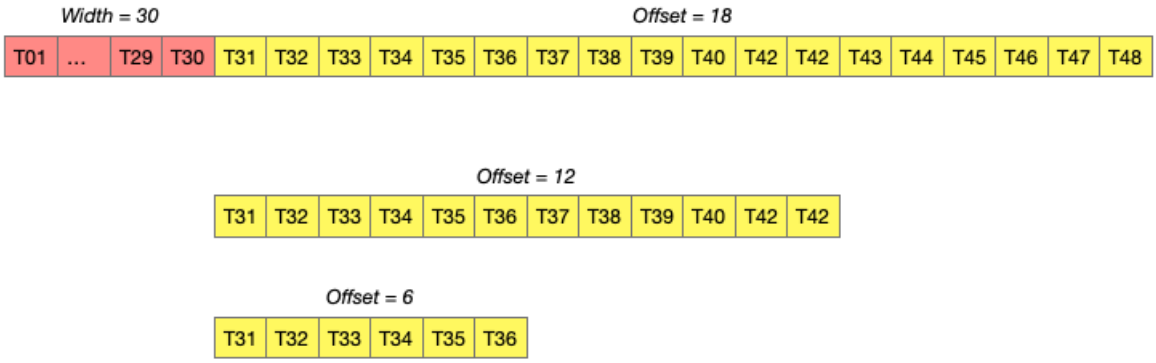


Figure 3.3: Data window slice

Special consideration needs to be made when generating data-window slices, the case when several slices have the same time-step values as input but different offset values, this is dangerous because a Neural Network would get confused estimating weights for different predictions. To avoid this complication, slices with similar input values but different offset values are purged, leaving only one slice per multiple repetitions.

Table 3.4: Data Window slices

JAEB	Slices
train	268502
test	112050

TSALIKIAN	Slices
train	12219
test	5786

AIDAS	Slices
train	1692
test	1210

Chapter 4

Machine Learning

This work aims to set a fair base for comparison among three top-notch Deep Learning Architectures applied to Blood Glucose Predictions when looking at their *Transfer Learning Capabilities* under different horizons (30min - 60min and 90min).

Initially, a specific review of model selections and architectures is going to be done. After that, the models will be trained and tested over a huge data-set, the JAEB, which contains glucose profiles belonging to all kinds of persons suffering from diabetes type I, an analytical and clinical assessment is going to be conducted for each horizon, the goal is to build models as accurate as possible within acceptable metrics and their known values.

Once models are ready for performing glucose predictions, two Transfer Learning Approaches are going to carry out. An Inductive approach, where the same models will be re-trained over the remaining data-sets and a Confusion approach, based on the work of Ganin et al., (2016) [34], where an unchanged CRNN architecture is going to be trained in two steps, first the whole network is trained with samples from the JAEB data-set and right after only the convolutional part (feature extractor) is trained to employ domain confusion or the Gradient Reversal method, the idea is to confuse the convolutional network to be able to recognize not only samples from JAEB but the other two data-sets, respectively.

Results will be evaluated with absolute and relative error metrics such as MAE, RMSE, R2, MAPE, MD, and FIT.

4.1 Models

4.1.1 Selection

As debated in the Related Work section of the Introduction chapter, it was found that data driven models are better than others, as was openly argued and proven. After carefully reviewing scientific papers on Data-Driven models applied to Glucose Forecasting, below is a table displaying the most relevant information linked to the state of art Data-Driven architectures:

Research	Method	PH (min)	RMSE (mg/dl)
Pappada et al. [8]		75	43.9
Pérez-Gandía et al. [16]	FFNN	15 30 45	9.7 17.5 27.1
Zecchin et al. [35]		30	14
Albertetti et al. [36]		30 60	17.45 33.67
Mougiakakou et al. [37]	RNN	5	13.65
Robertson et al. [38]		15	10.09
Bhimireddy et al. [26]		30	20.6
Pupillo et al. [32]	LSTM	30 60 90	19.47 32.38 41.54

Table 4.1: Comparison among architectures

This information is corroborated in Benaly et al., (2018) [39], table [4] and Pupillo et al. (2019) [32], table [3].

Criteria for choosing architecture amidst methods described in Table 4.1 are complexity, RMSE values, and accessible information for reproducing responses. Consequently, the architectures picked are Perez-Gandia [16], Albertetti [36] and Bhimireddy [26].

4.1.2 Architectures

Artificial neural networks are a form of machine-learning algorithm with a structure roughly based on that of the human brain. Like other kinds of machine-learning algorithms, they can solve problems through trial and error without being explicitly programmed with rules to follow.

Neural networks were first developed in the 1950s to test theories about the way that interconnected neurons in the human brain store information and react to input data. As in the brain, the output of an artificial neural network depends on the strength of the connections between its virtual neurons – except in this case, the "neurons" are not actual cells, but connected modules of a computer program. When the virtual neurons are connected in several layers, this is known as Deep learning nothing else that "stacking neural networks".

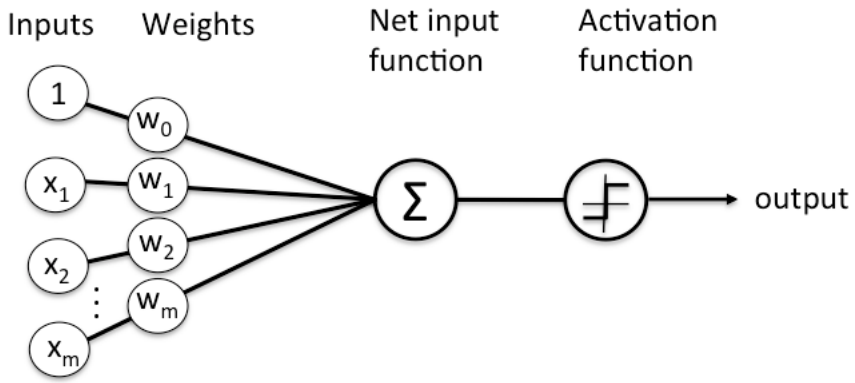


Figure 4.1: Artificial Neuron

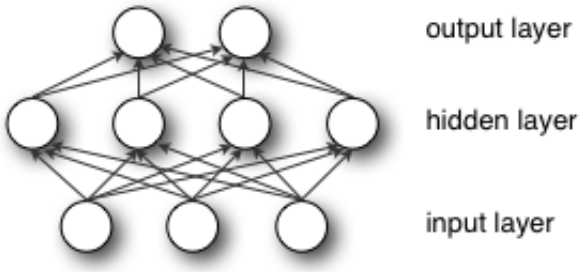


Figure 4.2: Feed Forward Network

The layers are made of nodes. A node is just a place where computation happens, loosely patterned on a neuron in the human brain, which fires when it encounters sufficient stimuli. A node combines input from the data with a set of coefficients, or weights, that either amplify or dampen that input, thereby assigning significance to inputs with regard to the task the algorithm is trying to learn; e.g. which input is most helpful is classifying data without error? These input-weight products are summed and then the sum is passed through a node's so-called activation function, to determine whether and to what extent that signal should progress further through the network to affect the ultimate outcome, say, an act of classification. If the signals passes through, the neuron has been "activated."

Deep-learning networks are distinguished from the more commonplace single-hidden-layer neural networks by their depth; that is, the number of node layers through which data must pass in a multi-step process of pattern recognition [40].

A more sophisticated variation of a Neural Network can be the Convolutional Neural Network, or CNN for short, is a specialized type of neural network model designed for working with two-dimensional image data, although they can be used with one-dimensional and three-dimensional data.

Central to the convolutional neural network is the convolutional layer that gives the network its name. This layer performs an operation called a “convolution“.

In the context of a convolutional neural network, a convolution is a linear operation that involves the multiplication of a set of weights with the input, much like a traditional neural network. Given that the technique was designed for two-dimensional input, the multiplication is performed between an array of input data and a two-dimensional array of weights, called a filter or a kernel.

The filter is smaller than the input data and the type of multiplication applied between a filter-sized patch of the input and the filter is a dot product. A dot product is the element-wise multiplication between the filter-sized patch of the input and filter, which is then summed, always resulting in a single value. Because it results in a single value, the operation is often referred to as the “scalar product“.

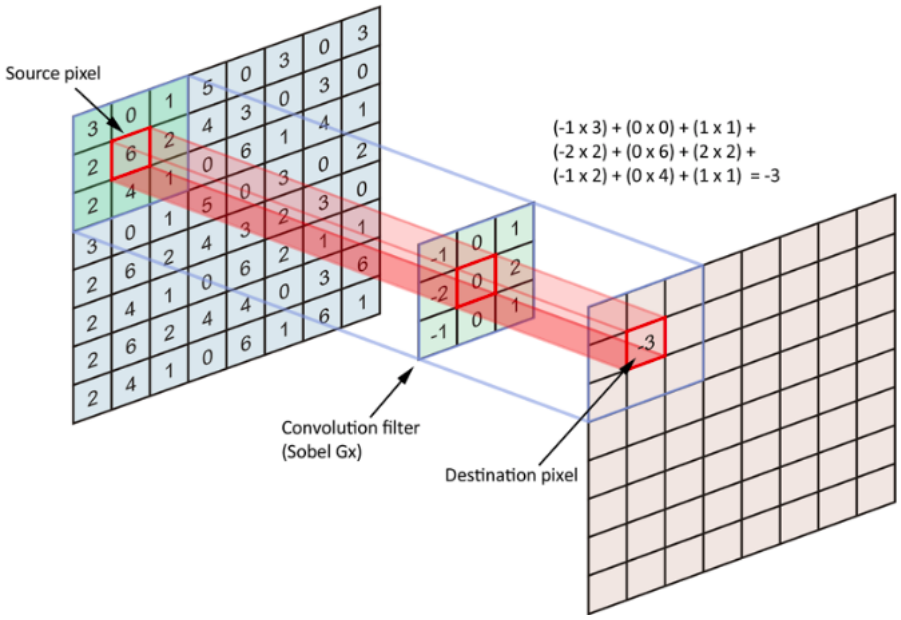


Figure 4.3: Convolutional Neural Network

Using a filter smaller than the input is intentional as it allows the same filter (set of weights) to be multiplied by the input array multiple times at different points on the input. Specifically, the filter is applied systematically to each overlapping part or filter-sized patch of the input data, left to right, top to bottom.

This systematic application of the same filter across an input is a powerful idea. If the filter is designed to detect a specific type of feature in the input, then the application of that filter systematically across the entire input allows the filter an opportunity to discover that feature anywhere. This capability is commonly referred to as translation invariance [41].

Another improvement to simple Neural Networks are Long Short Term Memory or LSTM, these are a type of recurrent neural network capable of learning order dependence in sequence prediction problems.The main important thing is that they have internal mechanisms called gates that can regulate the flow of information.

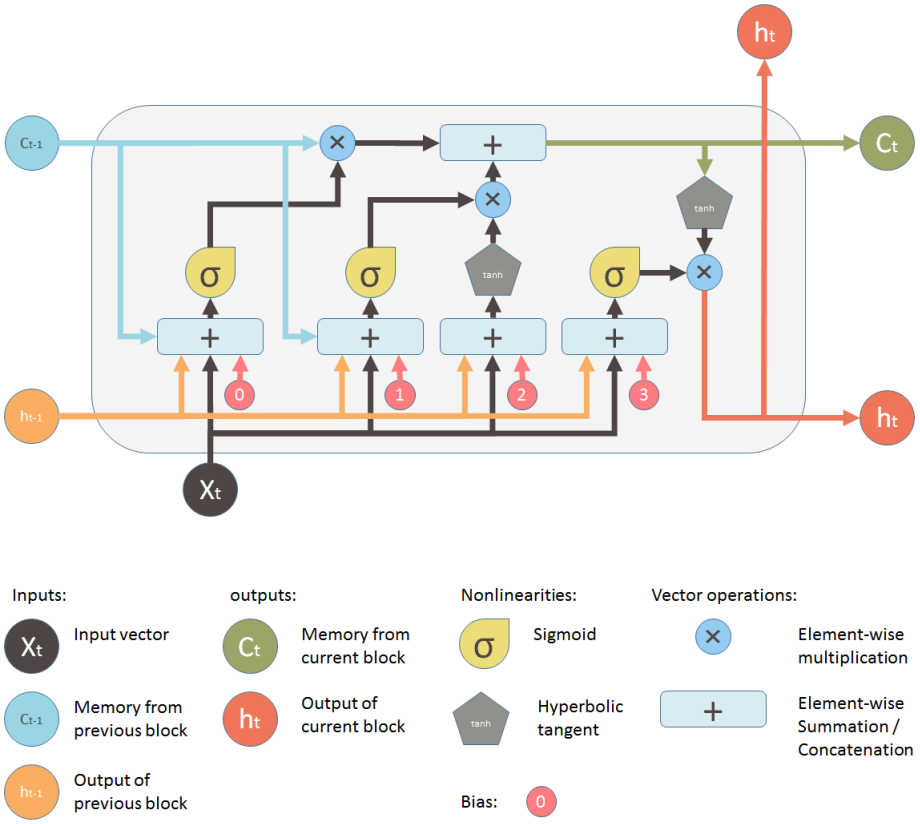


Figure 4.4: LSTM Network

These gates can learn which data in a sequence is valuable to keep or throw away. by doing that, it can pass relevant information down the long chain of sequences to make predictions [42].

In the case of this thesis, the specific models' architectures used are precisely denoted next:

- **Feed Forward Neural Network**

  Comprised of one input layer plus 2 hidden layers of 10 and 5 Neurons respectively, and a multi-step output layer. This model was introduced by Perez-Gandia [16].

Layer description	Output dimension
Input	(None, 30)
Dense	(None, 10)
Dense	(None, 5)
Dense	(None, output)

Figure 4.5: FFNN architecture

- **CRNN**

  Composed of a Convolutional Feature Extractor coupled with a Recurrent Feed-Forward Neural Network. The convolutional part is made up of 3 Convolutional 1D layers, each one followed by a 1D Max-Pooling Layer then coupling is done through an LSTM layer of 64 cells followed by 2 Dense layers of 256, 32 neurons connected to a multi-step output. (Albertetti et al., 2020 [36])

Layer description	Output dimension
Convolution 1D	(Batch size, 28, 8)
Max pooling 1D	(Batch size, 14, 8)
Convolution 1D	(Batch size, 12, 6)
Max pooling 1D	(Batch size, 6, 16)
Convolution 1D	(Batch size, 4, 32)
Max pooling 1D	(Batch size, 2, 32)
LSTM	(Batch size, 64)
Dense	(Batch size, 256)
Dense	(Batch size, 32)
Dense	(Batch size, output)

Figure 4.6: CRNN architecture

- **LSTM**

  Consist of an encoder coupled with a decoder, each one of 200 cells, followed by 2 Dense Layers containing 150 neurons and a multi-step output, presented by Bhimireddy [26].

Layer description	Output dimension
Input	(Batch size, 30, 1)
LSTM	(Batch size, 200)
Repeat Vector	(Batch size, output, 200)
LSTM	(Batch size, 200)
Time Distributed Dense	(Batch size, output,150)
Time Distributed Dense	(Batch size, output, 1)

Figure 4.7: LSTM architecture

4.1.3 Parameters

- **Optimization Algorithm**

  Adaptative Moment Estimation (Adam) is employed as an optimization algorithm, known properties of Adam are low memory requirements, invariant to diagonal rescale of gradients, appropriate for either non-stationary objectives, just to cite a few advantages [43].

  In addition, empirical results demonstrate that Adam compares favorably to other stochastic optimization methods [44], due to combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems. Also, it is easy to configure where the default configuration parameters perform acceptably on most problems.

- **Loss function**

  As seen in chapter 3 [section 3.1], all data-sets have a skewed Gaussian shape distribution with an ample range of values between minimum and maximum glucose points. This could indicate a possible existence of outliers. Now, as reviewed before, diabetic individuals are prone to live events where glucose levels ramp up over 500 mg/dl values, a temporal state called *"Hyperosmolar Hyperglycemic Nonketotic Syndrome"*, which could resemble outliers Figure 3.2.

  For regression estimation, the customary loss functions are Mean Squared Logarithmic Error, use when data is not scaled, Mean Squared Error, used with scaled data but not good for dealing with outliers due to the squaring factor, and the Mean Absolute Error Loss function, apt for managing large or small values far from the mean value, giving some robustness, and therefore aligned with the scope of this study [45].

- **Epochs and Batches**

  An early stop epoch policy has been adopted for detecting either global or local minima points which translates into optimum models, the exact criterion used for validation-based early stopping was selected following Prechelt et al., (2002) [46] who recommends choosing a value around 4 percent of the total number of epochs, also early-stopping is selected as hyper-parameter tuning technique. Complementing this policy is the model checkpoint and the adoption of the mini-batch paradigm.

  Since there is not a golden rule for epoch nor batches selection, a value of 2000 epochs is chosen for the the feed-forward and convolutional network and 80 epochs for the LSTM with the induction technique while 1000 epochs for the confusion one, this goes in consonance with what stated researchers have done in each architecture. Also 4096 batches are used for training, since this an intermediate number that ensures stability when calculating average values for losses and metrics.

- **Metrics**

  To validate the prediction performance of these architectures, both for *Glucose Forecasting* and *Transfer Learning*, the most common statistical metrics used and defined in literature [47] [48] are employed in this study.

  On the side of the absolute errors we can count on the Mean Absolute Error (MAE), the Root Mean Square Error (RMSE) and the Sum of Square of Glucose Prediction Errors (SSGPE).

  The Mean Absolute Error is defined as the average value of the sum of the absolute difference between pairs of real and predicted values, since the absolute operation is used, the direction of the difference is neglected, therefore, it is appropriate to use it when there is a known presence of outliers. It is expressed as:

$$MAE = (\frac{1}{n}) \sum_{i=1}^{n} (|G_i - \hat{G}_i|) \tag{4.1}$$

  The Root Mean Square Error is defined as the root square of the average of the sum of square differences between pairs of real and predicted values. It is good because it depends on the variance of a frequency distribution error, it is written as:

$$RMSE = \sqrt{(\frac{1}{n}) \sum_{i=1}^{n} (G_i - \hat{G}_i)^2} \tag{4.2}$$

  SSGPE is a special statistical measurement for quantifying the discrepancy between pairs of real and predicted points, it is often applied as an optimal criterion for model selection. Mathematically is defined as:

$$SSGPE = \sqrt{\frac{\sum_{i=1}^{n} (G_i - \hat{G}_i)^2}{\sum_{i=1}^{n} (G_i)^2}} \tag{4.3}$$

  On the other side, for relative errors, we can rely on the Squared Correlation (R2), the Mean Absolute Percentage Error (MAPE), the Mean Absolute Difference (MD) and the Fitness Error (FIT).

R^2 coefficient, also known as the coefficient of correlation, is a percentage indicator of how close predicted values are to real values, being R^2 = 100 % the goal, meaning that both real and predicted values superimpose or are equal.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(G_i - \hat{G}_i)^2}{\sum_{i=1}^{n}(G_i - \bar{G})^2} \tag{4.4}$$

The Mean Absolute Percentage Error is the percentage variation of the sum of absolute values between pairs of real and predicted figures against its real part, because it accumulates the percentage variations, it is a good measure to obtain a sight on a variational estimate. It is defined as:

$$MAPE = (\frac{100}{n})\sum_{i=1}^{n}|\frac{G_i - \hat{G}_i}{G_i}| \tag{4.5}$$

The Mean Absolute Difference is described as a measure of statistical dispersion equal to the average absolute difference between real and forecasted values. Mathematically is written as:

$$MD = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|G_i - \hat{G}_j|}{n(n-1)} \tag{4.6}$$

The Fitness Percentage Error is the division of the root square error between the square difference of pairs of observed and predicted values with the root square of the square difference between the observed values and its mean value.

$$FIT = 1 - \frac{\sqrt{\sum_{i=1}^{n}(G_i - \hat{G}_i)^2}}{\sqrt{\sum_{i=1}^{n}(G_i - \bar{G})^2}} \tag{4.7}$$

However, Analytical metrics do not suffice at demonstrating the quality of models because they are only good from a theoretical point of view. To explore models' precision in a real-life application there is a method called *Clarke Error Grid Analysis* (CEGA) [49], This method is a graphical tool for assessing models' glucose forecasts from a clinical perspective, here a *Cartesian Plane* is divided into five zones as depicted in the following image.

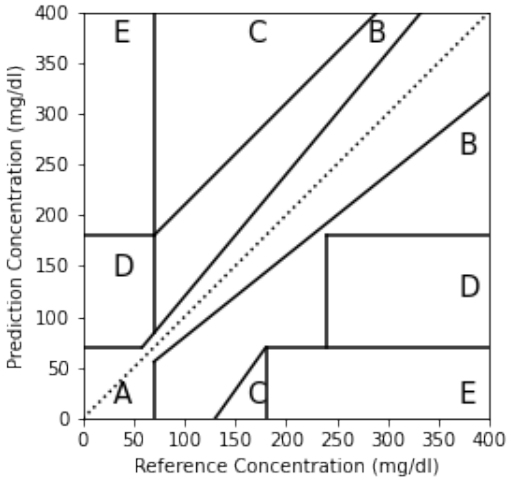


Figure 4.8: Cega Zones

✓Zone A: It characterizes the predicted blood glucose levels that are deviated from the actual blood glucose levels by no more than 20% of the reference sensor.

✓Zone B: It characterizes the predicted blood glucose levels that are outside of 20% of the reference sensor but would not lead to inappropriate treatment.

✓Zone C: It characterizes a good medication adjustment of blood glucose levels (or unnecessary treatment because these levels are in the range [70 mg/dl, 180 mg/dl]).

✓Zone D: It characterizes dangerous cases to identify and to assess significant clinical mistakes and errors.

✓Zone E: It characterizes the false treatment zone (wrong medication adjustment).

Predicted points are plotted over the graph, then it is tally the number of points that hit each zone. Model's with a high percentage of points laying over zone A and zone B are said to be accurate otherwise models are considered as inaccurate.

4.1.4 Blood Glucose Predictors Results

Following the guidelines in Perez-Gandia [16], Albertetti [36], and Bhimireddy [26], the three models are reproduced according to the reported information using the JAEB data-set, with variations in some specific cases to adapt them to the current analysis. At first glance, all models' metrics deteriorate with the increase of the prediction horizon but keep a high degree of similarity across forecasting horizons. Besides when comparing the reported RMSE metrics with the RMSE metrics obtained after training and testing, for each horizon, is clear how the RMSE improved, for example at 30 minutes Perez-Gandia et al. reported an RMSE value of 17.5 mg/dl and the trained models' RMSE range from 11.8 to 12.15 mg/dl, a diminution of 5 mg/dl, at 60 minutes Albertteti et al. published an RMSE value of 33.67 mg/dl while models obtained values between 19.89 and 20.75 mg/dl, a reduction of 13 mg/dl, finally at 90 minutes Pupillo et al. stated an RMSE of 41.54 mg/dl whereas SOTA got figures around 28.44 mg/dl, a contraction of 14 mg/dl. These results are remarkable because they check that all models are fine-tuned with regard to the conditions previously established by these researchers.

PH	Metrics	FFNN	CRNN	LSTM
30 min				
	MAE	7.64459	7.78691	7.38907
	RMSE	12.15289	12.10014	11.80638
	SSGPE	0.00581	0.00573	0.00551
	R2	96.46	96.51	96.64
	MAPE	5.21	5.42	4.98
	MD	8.23	8.08	8.51
	FIT	81.18	81.31	81.67
60 min				
	MAE	13.67811	13.83973	13.32239
	RMSE	20.26366	20.75422	19.89391
	SSGPE	0.01650	0.01731	0.01598
	R2	89.96	89.47	90.27
	MAPE	9.35	9.30	9.06
	MD	4.60	4.54	4.72
	FIT	68.30	67.53	68.80
90 min				
	MAE	18.68273	19.43397	18.64639
	RMSE	27.12637	28.44312	27.12733
	SSGPE	0.02934	0.03222	0.02953
	R2	82.16	80.41	82.05
	MAPE	12.69	13.09	12.74
	MD	3.36	3.23	3.37
	FIT	57.75	55.72	57.59

Table 4.2: Glucose Predictors Results

Going deeper into other statistical measurements from table 4.2, it can be seen how the Squared Correlation (R2) computation gives a high resemblance between real and predicted data, starting at 96% at 30 minutes, passing to 90% at 60 minutes, and ending at 82% for 90 minutes. This goes in line with the augmentation of the MAPE from 5.42% to 9.35%, indicating that the spread variation raises over prediction trends and the growth of MAE from 7.78 mg/dl to 19.43% showing a climb on the bias, something logical since the more points are needed to be predicted the more error is expected to be embedded into predictions.

Now when comparing models among themselves, at the short and medium horizon LSTM outperforms the others but in the long run MLP does better. The above statement can be supported in detail observing the loss and loss validation behavior under training for each model through each prediction horizon in Figure 4.9.

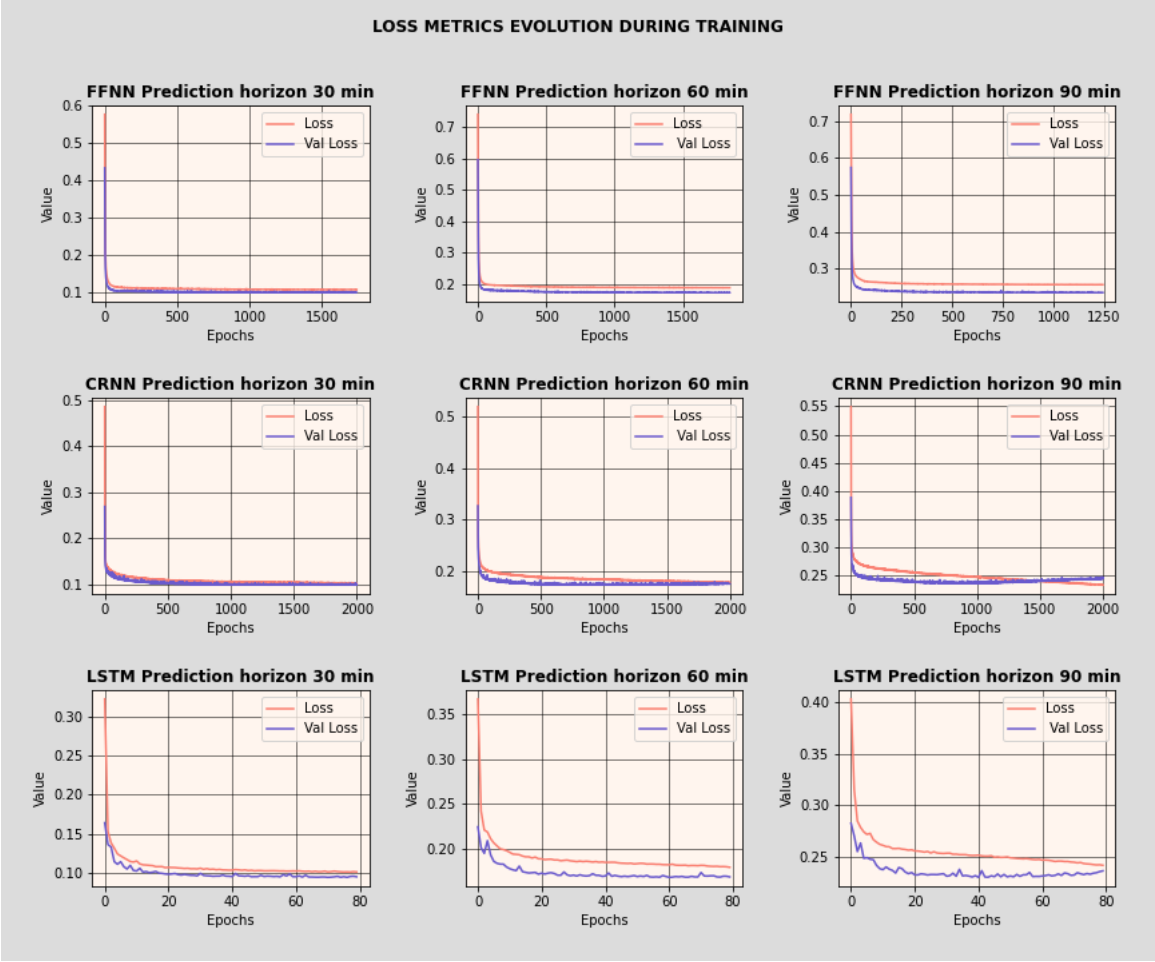


Figure 4.9: Loss Training Comparison

One important clarification in Figure 4.9 is despite of models look like they are overfitting in some instances, they are not [50]. The reasons for obtaining validation losses greater than or equal to regular losses are:

1. Regularizations are applied during training but not under validation/test.
2. Training loss is measured during each epoch while validation loss is measured after each epoch.
3. Even though it appears there is a data leakage from the training set to the validation set, there is none, as proven in the first chapters of this thesis, the training data-set holds a lot of glucose profiles that have a big degree of resemblance among them.

The most relevant issues are how in the CRNN architecture loss validation line overlaps to some degree over the loss line during the whole iteration at 30 minutes, then both lines tend to overlap again from 1500 epochs onward for the 60 minutes case, and finally, at 90 minutes both lines cross over each other around 1500 epochs.

In contrast, the LSTM architecture had problems reaching convergence quicker as the prediction horizon increases, albeit the gap between the loss and loss validation is visually clear, a point to highlight is how the fluctuation on the lines seem to be knit and stable due to architecture capacity of calculating weights more smoothly when looking at the other models.

The behavior of the FFNN architecture is similar to the CRNN with the difference that loss validation never overlaps training loss, and it reaches convergence fast.

To understand if these models are useful in real-life situations, the Clark Error Grid Analysis is run.

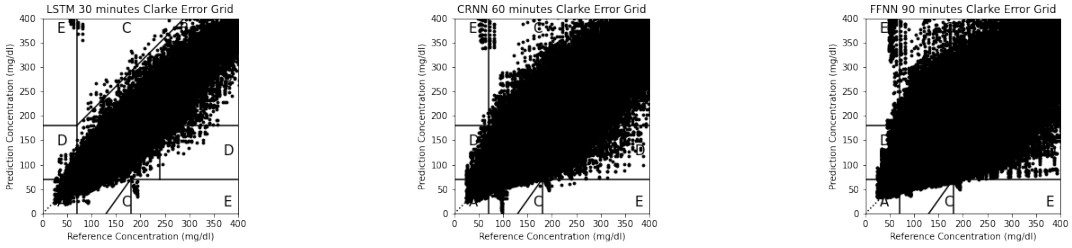


Figure 4.10: Clark Error Grid for Glucose Predictors

The computations per zone are displayed down below,

PH	Metrics	FFNN	CRNN	LSTM
30 min				
	A	96.11617	95.95285	96.39432
	B	3.58069	3.56374	3.36100
	C	0.01130	0.01175	0.00997
	D	0.28618	0.46735	0.22996
	E	0.00565	0.00431	0.00476
60 min				
	A	87.44422	87.31452	87.87342
	B	11.14666	11.26975	10.81043
	C	0.07497	0.09899	0.08516
	D	1.32389	1.30381	1.22304
	E	0.01026	0.01294	0.00796
90 min				
	A	79.87535	79.00000	79.69166
	B	17.62323	18.33195	17.62115
	C	0.23809	0.34325	0.29416
	D	2.23913	2.29050	2.36864
	E	0.02420	0.03431	0.02439

Table 4.3: Predictors' Clark Error Grid

Values in Table 4.3 keep ratifying what has already been saying before, models' precision tends to decrease with the prediction horizon, yet the Clark Error Grid Analysis validates models' acceptance as they scored above 97% of precision when adding Zone A and Zone B numbers across horizons, that is, all models are able to predict values below of 20% of deviation from real values, and when models go far this cusp, does not affect treatment and zone D is below 2%, meaning that from all readings around 2% are mispredictions.

4.2 Transfer Learning Results

Transfer learning (TL) is a research problem in machine learning (ML) that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem.

The definition of transfer learning is given in terms of domains and tasks. A domain $\mathcal{D}$ consists of: a feature space $\mathcal{X}$ and a marginal probability distribution P(X), where $X = \{x_1, ..., x_n\} \in \mathcal{X}$. Given a specific domain, $\mathcal{D} = \{\mathcal{X}, P(X)\}$, a task consists of two components: a label space $\mathcal{Y}$ and an objective predictive function $f : \mathcal{X} \rightarrow \mathcal{Y}$. The function f is used to predict the corresponding label f(x) of a new instance x. This task, denoted by $\mathcal{T} = \{\mathcal{Y}, f(x)\}$, is learned from the training data consisting of pairs $\{x_i, y_i\}$, where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$.

Given a source domain $\mathcal{D}_\int$ and a learning task $\mathcal{T}_\int$, a target domain $\mathcal{D}_\sqcup$ and a learning task $\mathcal{T}_\sqcup$, where $\mathcal{D}_\int \neq \mathcal{D}_\sqcup$ and $\mathcal{T}_\int \neq \mathcal{T}_\sqcup$,transfer learning aims to help improve the learning of the target predictive function $f_t(.)$ in $\mathcal{T}_\sqcup$ using knowledge in $\mathcal{D}_\int$ and $\mathcal{T}_\int$ [51].

There are many types of *Transfer-Learning*, in this thesis two methodologies are followed, one related to *Induction Transfer Learning*, that reuses parts of a previously trained model on a new network tasked for a different but similar problem by retraining the whole model with new unseen and different data.

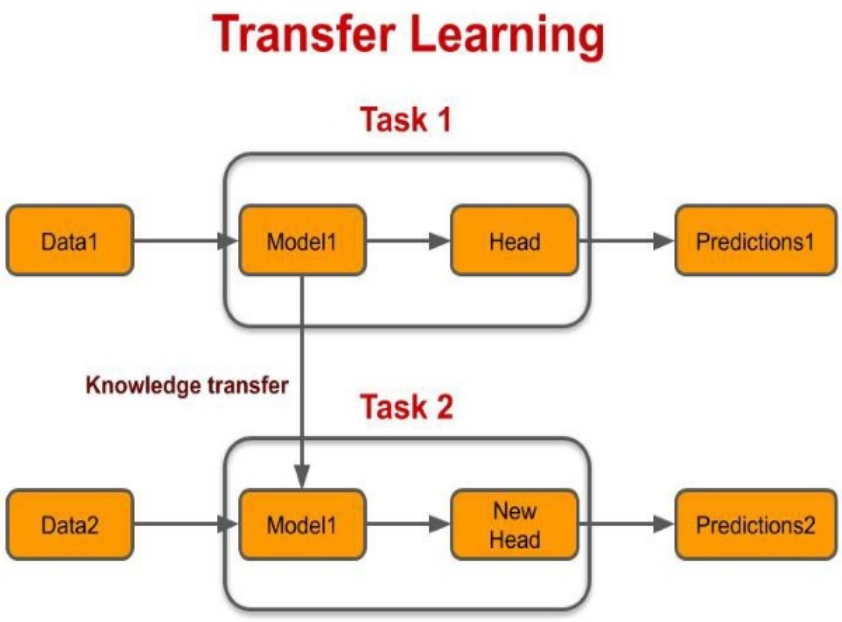


Figure 4.11: Inductive Transfer Learning

And *Domain Adaptation Transfer Learning*, which is usually applied over a CNN network focus on retraining only the extractor segment utilizing domain confusion [34]. The idea is to initially train the whole network, figure 4.12, in two stages. In stage one, the feature extractor and the regressor segments are trained on the source domain data, green and blue blocks on figure 4.12. In stage two, data is mixed between a source and a target domain then this mingled data is fed through the network, including a classifier, this classifier is paired with a mathematical trick called gradient reverse, that is, as long as the classifier is learning to distinguish between source and target domains, errors are calculated and then with back-propagation weights are updated. The particularity of this is that the estimation of gradients weight is multiplied by a negative factor Lambda, the purpose here is when the classifier sees data from the source domain, the whole multiplied gradients shall be small since the extractor already posses information from training in stage one, therefore, the weighing update is negligible but when it sees data from the target domain, the multiplied gradients shall be big to update effectively the feature extractor weights, giving it the power to recognize or extract patterns from this new target domain.

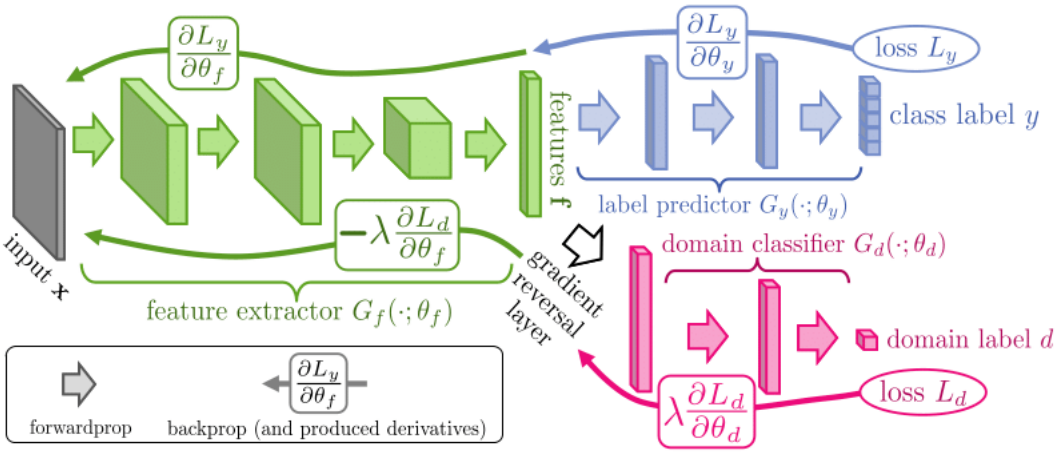


Figure 4.12: Domain Adaptation Transfer Learning

4.2.1 Inductive Transfer Learning Results

Starting from the pre-trained glucose predictors, new full training is ran but now taking as input feed the new target domain on which there is an interest to get new predictions. Since the amount of available data in the target domain is smaller compared to the source domain, then there is uncertainty about how to fine-tune the hyper-parameters of each architecture, the number of epochs is taken as a reference point through an early stop policy. The main idea is to compare the prediction capacity of each model *Transfer Learning* against the prediction capacity of the new model after the induction method is applied to.

Beginning with the Tsalikian data-set, concerning to statistical measures, the most important marks are described in the next table:

		FFNN		CNN		LSTM	
PH	**Metrics**	**Standard**	**Inductive**	**Standard**	**Inductive**	**Standard**	**Inductive**
30 min							
	MAE	12.91102	11.95373	12.79891	12.02580	12.31130	11.74999
	RMSE	18.36937	17.06684	17.97491	17.25181	17.69827	16.93861
	SSGPE	0.01530	0.01312	0.01441	0.01327	0.01409	0.01289
	R2	91.40	92.63	91.90	92.54	92.08	92.75
	MAPE	10.16	9.34	10.35	9.30	9.54	9.10
	MD	5.35	5.78	5.40	5.74	5.61	5.88
	FIT	70.74	72.91	71.63	72.76	71.93	73.15
60 min							
	MAE	19.00912	17.90358	19.10767	18.17952	18.55055	17.80845
	RMSE	26.51143	25.13413	26.76900	25.64014	25.99941	24.71903
	SSGPE	0.03166	0.02842	0.03239	0.02946	0.03040	0.02741
	R2	82.09	83.92	81.67	83.33	82.80	84.49
	MAPE	15.46	14.07	15.32	14.13	15.11	14.25
	MD	3.64	3.87	3.62	3.81	3.73	3.89
	FIT	57.86	60.06	57.35	59.33	58.68	60.77
90 min							
	MAE	23.16893	22.34438	25.00482	23.42065	23.07927	22.74294
	RMSE	31.86989	30.73599	34.79888	32.55209	31.75722	31.62373
	SSGPE	0.04534	0.04214	0.05447	0.04732	0.04518	0.04509
	R2	74.30	76.11	69.11	73.17	74.39	74.43
	MAPE	19.13	17.85	20.60	18.47	19.24	18.59
	MD	2.99	3.10	2.77	2.96	3.01	3.05
	FIT	49.50	51.32	44.64	48.40	49.58	49.62

Table 4.4: Inductive Transfer Learning over Tsalikian

Metrics on table 4.4 proved two important findings, the first one that usually is taking for granted is *Transfer-Learning* is highly dependant on the tuning condition of the base model, here across the different forecasting scenarios can be seen how this dependency hold still, the latter finding is that *Transfer-Learning* seems to be more effective as the horizons grow.

At 30 minutes, When looking at each model individually, statistics show that the model that has the biggest improvement is the FFNN, this can be confirmed by looking at the change of the R2 correlation metric, gaining a 0.8 % marginally or a reduction of 1 mg/dl in MAE and RMSE, this implies that the new model was tuned with a lower bias and variance than the reference model. Following the same procedure, LSTM has a change of the R2 correlation metric 0.63 %, MAE drops 0.5 mg/dl or MAPE drops 0.44 %, CRNN has R2 variation similar to the LSTM but the MAE falls 0.77 mg/dl or MD slides 0.34 %. Visual confirmation of this information is in Figure 4.14, the best plot belongs to FFNN, followed by CRNN and LSTM.

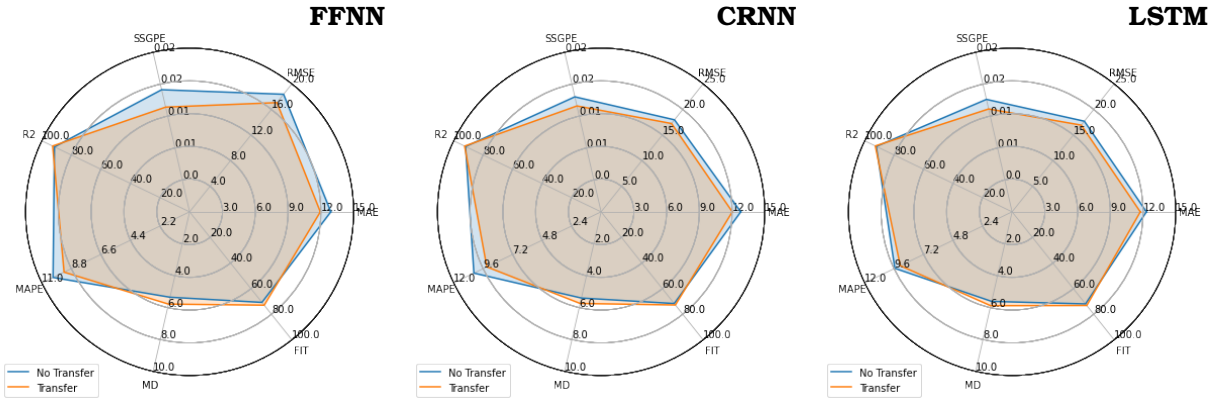


Figure 4.13: Tsalikian Radar Charts at 30 min

When looking at all models together, in terms of quality the LSTM model succeeded in obtaining the cleanest numbers when considering all architectures with the lowest MAE and RMSE with values of 11.74 and 16.93 mg/dl or highest R2 and FIT with values of 92.75 % and 73.15%, as depicted in Figure 4.14.

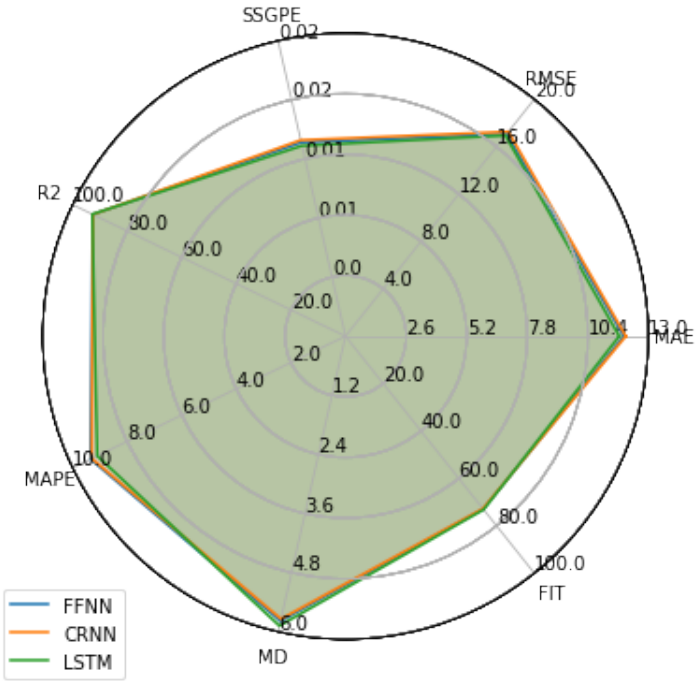


Figure 4.14: Inductive Radar Chart over Tsalikian at 30 min

At 60 minutes, the same tendency that appeared at 30 min keeps appearing, the simpler architecture keeps improving marginally on variance and bias with a jump of below 2 mg/dl on MAE or 1.2 mg/dl on RMSE, whilst complex architectures get a more neutral impact, for example, CRNN gets a reduction of 1 mg/dl in MAE or RMSE, or LSTM can lessen 0.7 mg/dl its MAE and RMSE respectively.

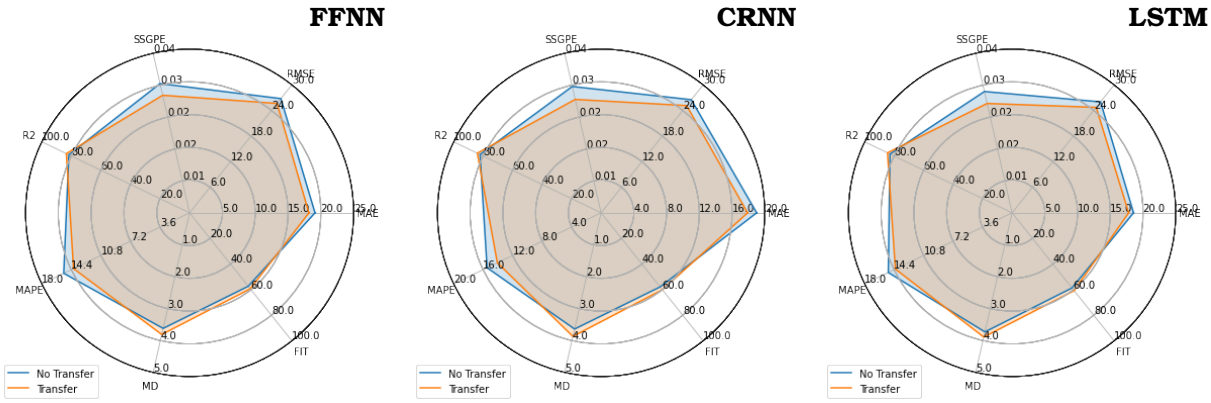


Figure 4.15: Tsalikian Radar Charts at 60 min

Comparing all models, LSTM still holds the best statistical marks but this time is followed by FFNN and CRNN, correspondingly, as Figure 4.16 draws.

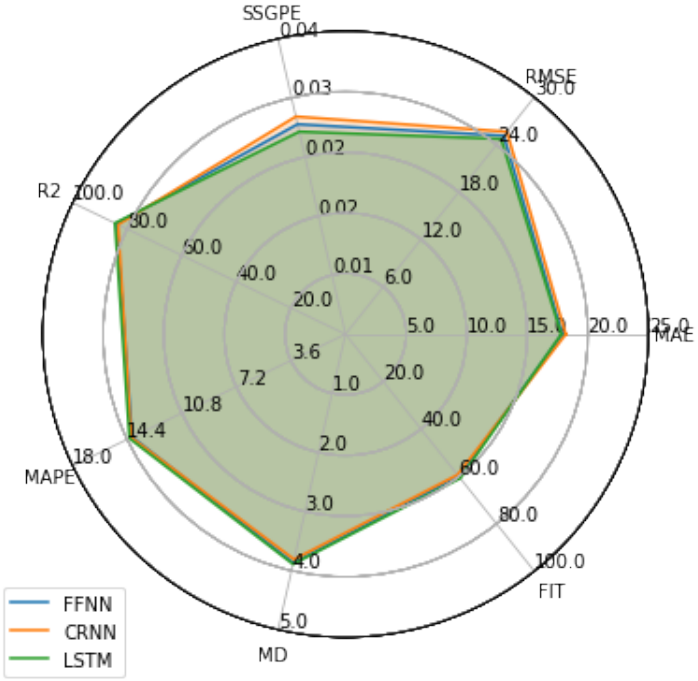


Figure 4.16: Inductive Radar Chart over Tsalikian at 60 min

At 90 minutes on a one by one analysis, statistics show that the model that has the biggest improvement is the CRNN, this can be confirmed by looking at the change of the R2 correlation metric, gaining a 4 % marginally or a reduction around of 2.22 mg/dl in MAE and RMSE, this implies that the new model was tuned with a lower bias and variance than the reference model. Following the same procedure, FFNN has a change of the R2 correlation metric of 2 %, MAE drops 1 mg/dl or MAPE decreases 1.1 %, LSTM has an R2 variation of 0.1 %, MAE falls 0.3 mg/dl or MD slides 0.04 %, as shown on Figure 4.18, the best plot belongs to CRNN, followed by FFNN and LSTM.

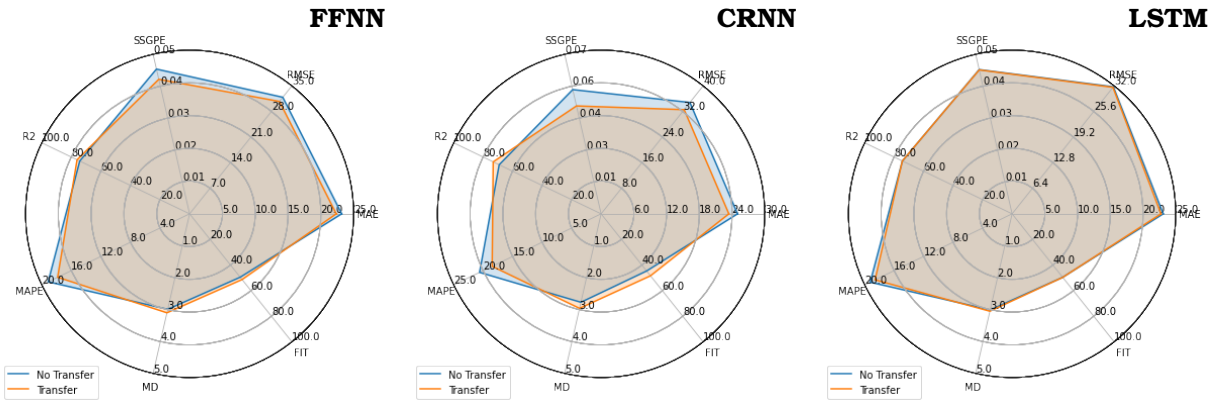


Figure 4.17: Tsalikian Radar Charts at 90 min

On the global perspective the model with the best performance is FFNN, followed by CRNN and LSTM.

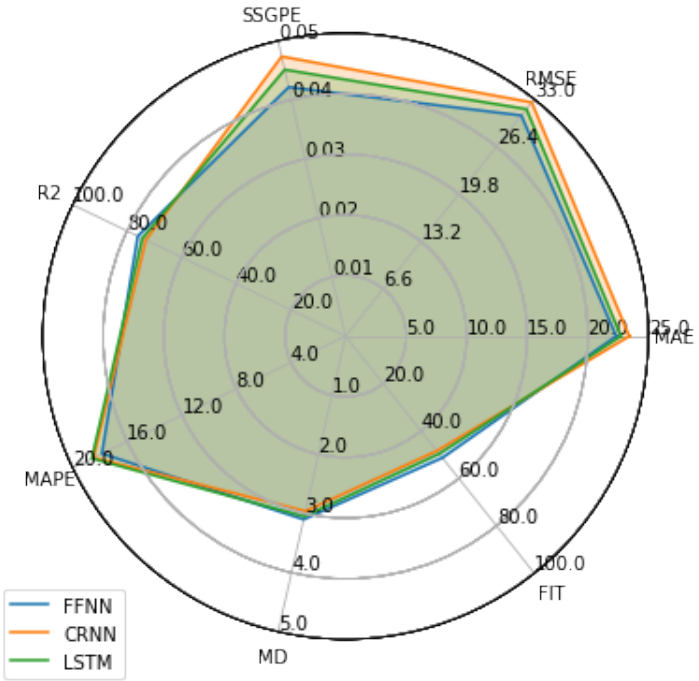


Figure 4.18: Inductive Radar Chart over Tsalikian at 60 min

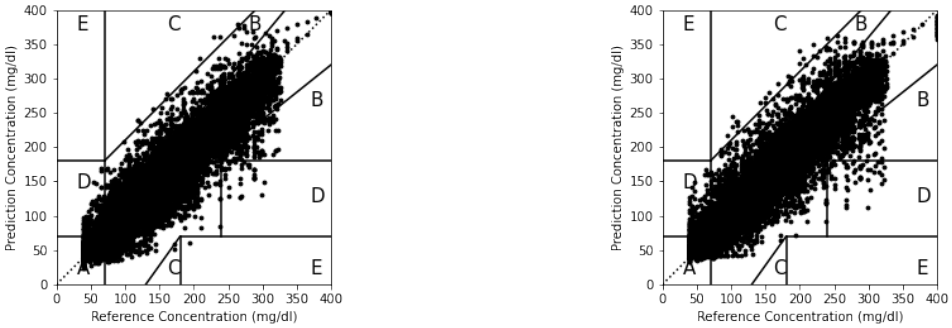
Pointing to the clinical assessment, after running the Clark Error Grid Analysis, the foremost marks found are listed beneath:

		FFNN		CNN		LSTM	
PH	**Zones**	**Standard**	**Inductive**	**Standard**	**Inductive**	**Standard**	**Inductive**
30 min							
	A	87.57633	88.72854	87.02327	89.07420	88.16108	89.31617
	B	10.76161	9.60940	10.67807	9.49418	10.39002	9.30695
	C	0.04609	0.08930	0.00864	0.00576	0.03457	0.00864
	D	1.61309	1.56988	2.29001	1.42297	1.41145	1.36536
	E	0.00288	0.00288	0.00000	0.00288	0.00288	0.00288
60 min							
	A	75.46808	77.93957	76.04563	78.28235	76.97603	78.18009
	B	20.23995	18.44106	19.38011	18.18182	18.85874	17.68493
	C	0.23908	0.21460	0.30966	0.23476	0.31110	0.13250
	D	4.02552	3.39613	4.21996	3.28811	3.82389	3.99672
	E	0.02736	0.00864	0.04465	0.01296	0.03025	0.00576
90 min							
	A	69.60287	70.05511	69.74498	69.45981	69.73346	70.42862
	B	24.36053	24.41141	23.77866	24.66682	24.25587	23.51461
	C	0.48201	0.57994	0.41864	0.69036	0.40999	0.58282
	D	5.48642	4.87479	5.97515	5.11676	5.47490	5.37120
	E	0.06817	0.07873	0.08257	0.06625	0.12578	0.10274

Table 4.5: Inductive Clark-Error Grid over Tsalikian

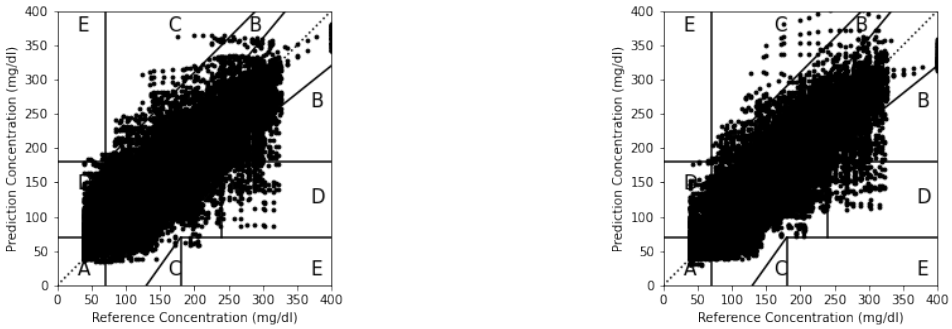
From table 4.5 can be inferred that once *Transfer Learning* is applied, the quality of the predicted values diminishes with the forecasting horizon, although, in general, models can still predict correct values and values with a deviation below of 20 %, Zone A and Zone B, above 93%, the number of values of Zone A decreases and values in Zone B increases even when comparing to Glucose Predictors without *Transfer Learning* and values of Zone D, predictions that can lead to wrong treatments, rise as well, both for the case of cross horizons and Glucose Predictors without *Transfer Learning*.

Some CEGA graphs are plotted for informative purposes.



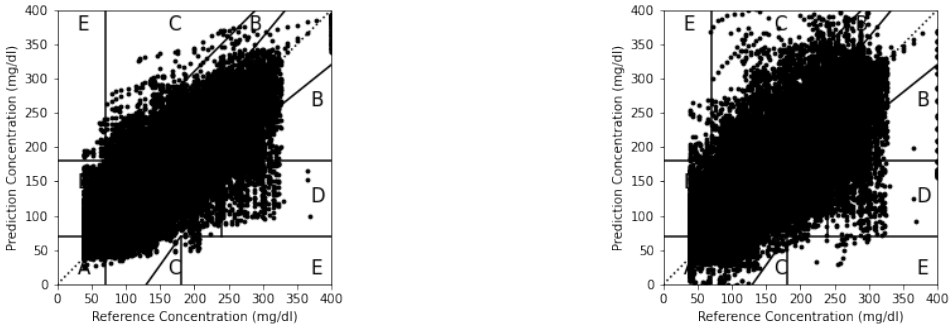
(a) Standard **(b)** Inductive

Figure 4.19: LSTM Clark Error Grid over Tsalikian at 30 min



(a) Standard **(b)** Inductive

Figure 4.20: FFNN Clark Error Grid over Tsalikian at 60 min



(a) Standard **(b)** Inductive

Figure 4.21: CRNN Clark Error Grid over Tsalikian at 90 min

Moving to the AIDAS' data-set, the main statistical computations collected after running several experiments are outlined in the latter table:

		FFNN		CNN		LSTM	
PH	**Metrics**	**Standard**	**Inductive**	**Standard**	**Inductive**	**Standard**	**Inductive**
30 min							
	MAE	3.37031	2.48604	3.11608	2.44913	3.16370	1.86235
	RMSE	4.56293	3.42671	4.23732	3.36478	4.18502	2.55397
	SSGPE	0.00045	0.00025	0.00039	0.00023	0.00038	0.00014
	R2	98.76	99.31	98.94	99.37	98.97	99.62
	MAPE	1.40	1.03	1.28	1.01	1.30	0.78
	MD	12.51	16.96	13.53	17.21	13.32	22.63
	FIT	88.90	91.69	89.72	92.05	89.86	93.84
60 min							
	MAE	7.39861	5.47172	6.96720	4.67358	7.10958	4.56554
	RMSE	10.22373	7.61852	9.56594	6.49287	9.67521	6.29632
	SSGPE	0.00244	0.00147	0.00218	0.00102	0.00219	0.00098
	R2	93.26	95.95	93.98	97.18	93.95	97.29
	MAPE	3.00	2.22	2.80	1.95	2.88	1.88
	MD	5.69	7.69	6.04	9.00	5.92	9.22
	FIT	74.30	79.93	75.62	83.26	75.56	83.60
90 min							
	MAE	11.45222	8.71806	11.94808	8.71806	10.62830	7.57168
	RMSE	15.42458	11.59308	15.90960	11.99489	14.29082	10.97773
	SSGPE	0.00567	0.00327	0.00601	0.00341	0.00482	0.00301
	R2	84.19	90.88	83.26	90.50	86.56	91.61
	MAPE	4.55	3.41	4.81	3.57	4.20	3.04
	MD	3.66	4.97	3.51	4.81	3.95	5.54
	FIT	60.93	69.91	59.89	69.31	63.61	71.20

Table 4.6: Inductive Transfer Learning over AIDAS

As listed in Table 4.6, all metrics values are lower than those reported for the base reference glucose predictors in all horizons, also the predictive power is higher comparing to the data captured with the TSALIKIAN data-set across different time scenarios.

At 30 minutes, LSTM individually and globally overworks the other two models, something particular is how all architectures have an optimum response without *Transfer Learning* and are still able to gain performance after it, bias and variance almost caress perfect punctuation. For example, LSTM has a marginal earning over R2 statistic of 0.62 %, MAE diminishes 1.3 mg/dl, or RMSE shrinkages 1.7 mg/dl. Following LSTM's performance are FFNN and then CRNN.

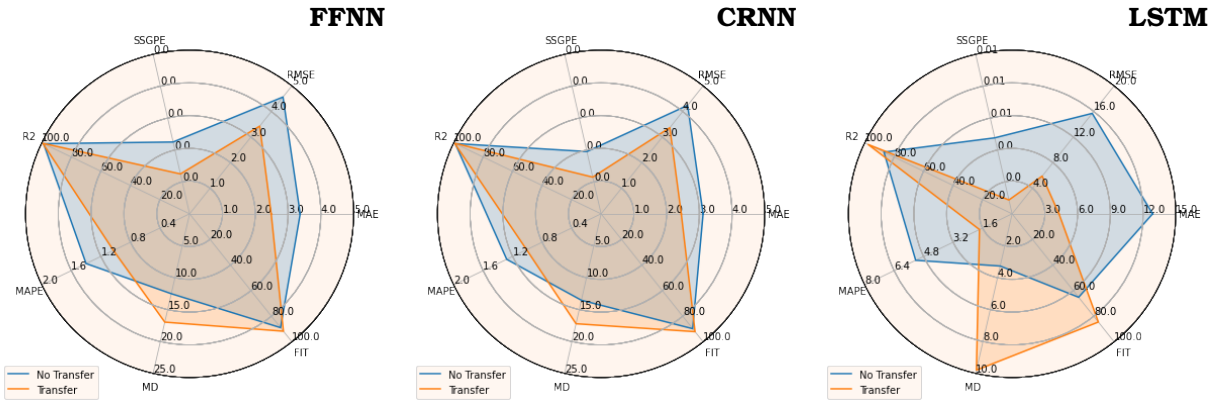


Figure 4.22: Metric Radar Chart 30 min

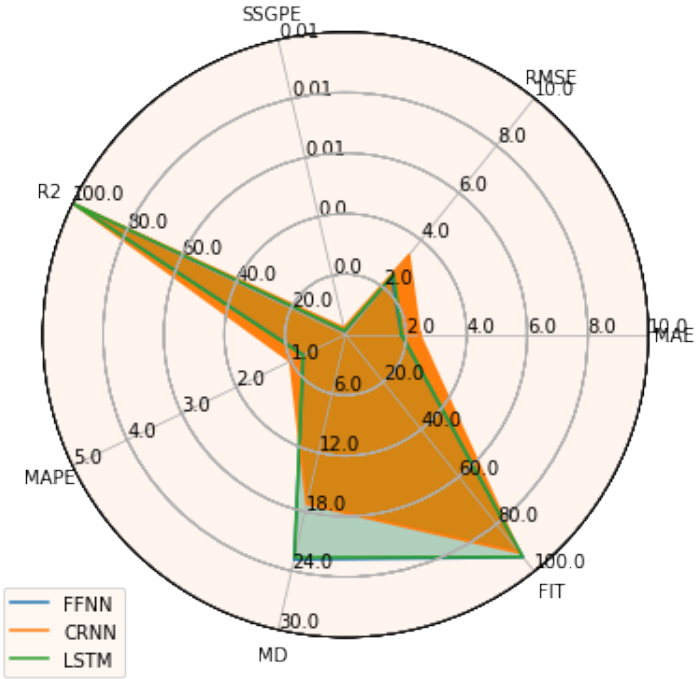


Figure 4.23: Overall Radar Comparison 30 min

At 60 minutes, the correction on the bias and variance keeps happening, the greater mark shifts are still attained by the LSTM network where its MAE decrements in 3.5 mg/dl, or its RMSE falls 3.3 mg/dl. CRNN and FFNN are able to correct its variance but not their bias as well as LSTM. e.g. RMSE goes down 3 mg/dl and 2.6 mg/dl correspondingly whilst their MAE just 2 mg/dl.

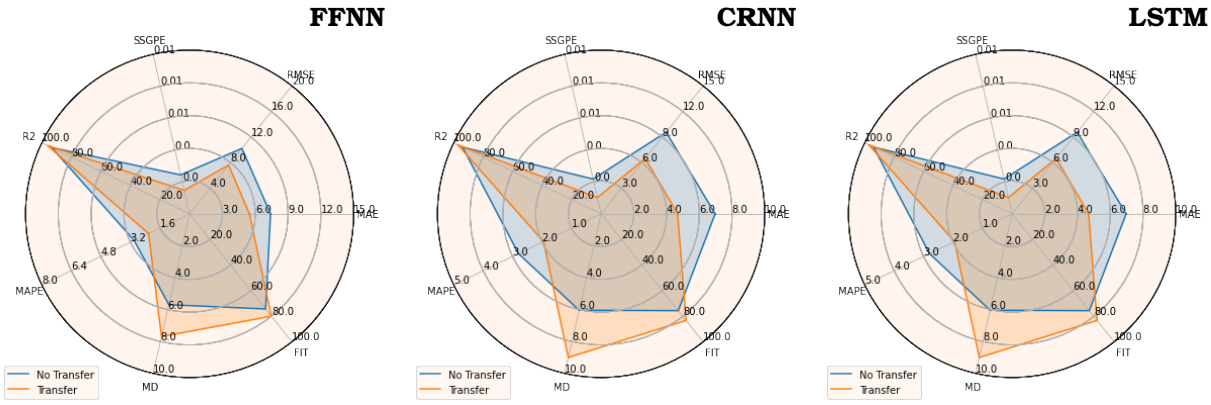


Figure 4.24: Metric Radar Chart 60 min

Nevertheless, in Figure 4.25, CRNN's polygon try to superimpose over LSTM's polygon, meaning both responses are alike but FFNN gets in last place.

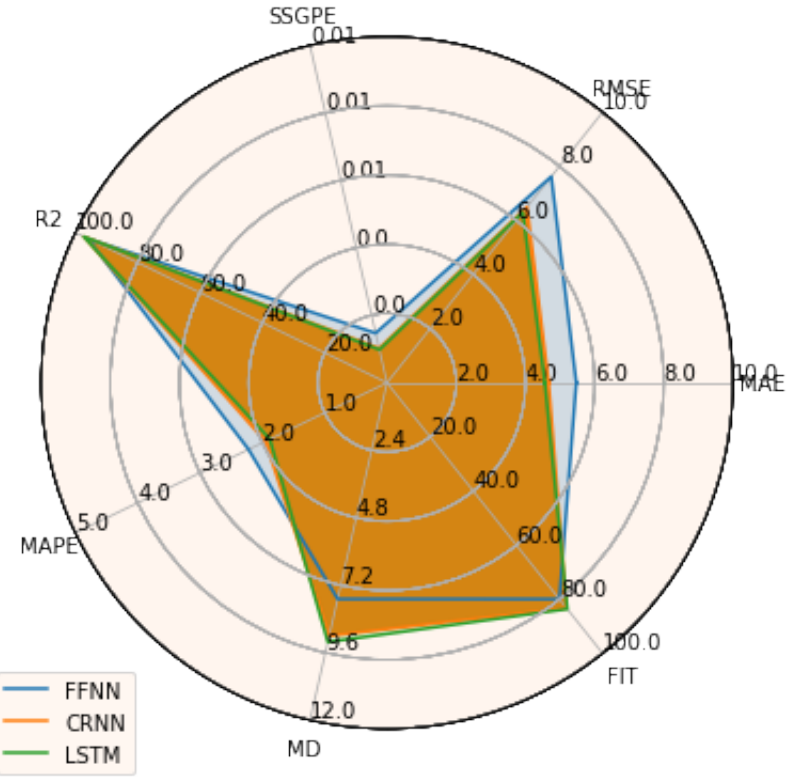


Figure 4.25: Overall Radar Comparison 60 min

At 90 minutes all models are able to sharply cut on errors in the same proportion, both MAE and RMSE reduces their value on 3 mg/dl but when analyzing on a one on one basis, CRNN is the model that accomplish the biggest marginal change on R2, about 6.5 % gain.

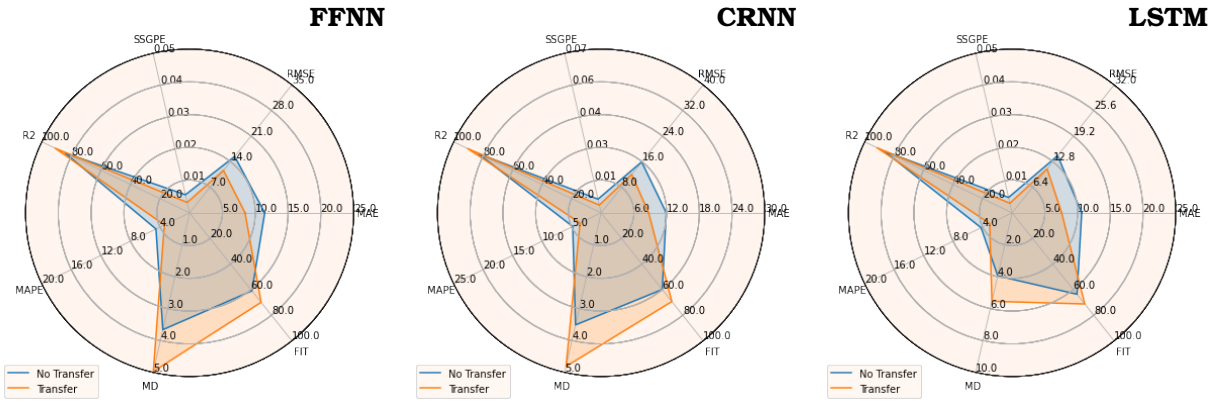


Figure 4.26: Metric Radar Chart 90 min

On the global panorama, LSTM is still the best network and FFNN with CRNN produces similar responses, both polygons barely superimposed over each and other.

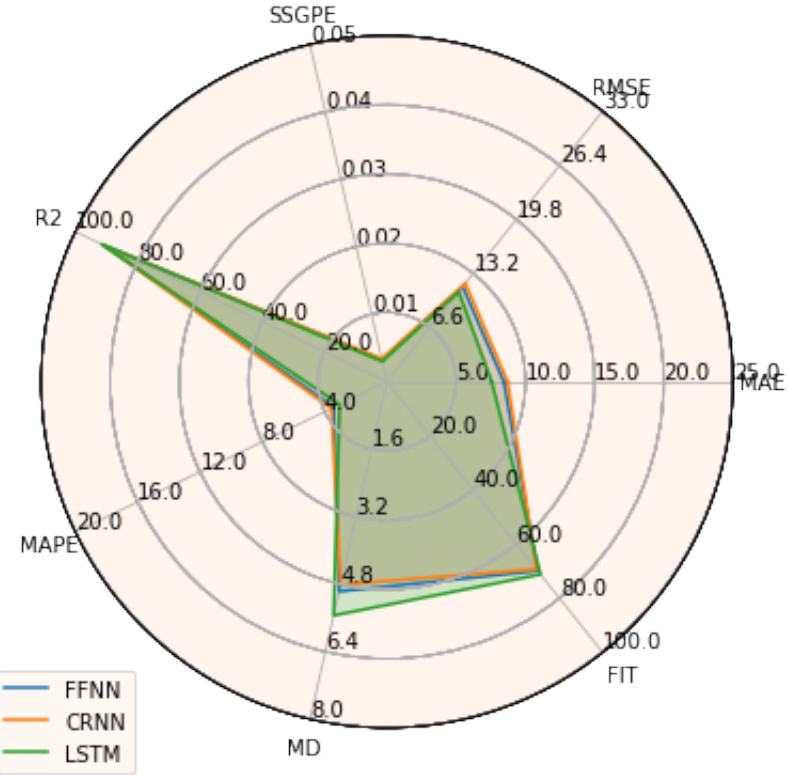


Figure 4.27: Overall Radar Comparison 90 min

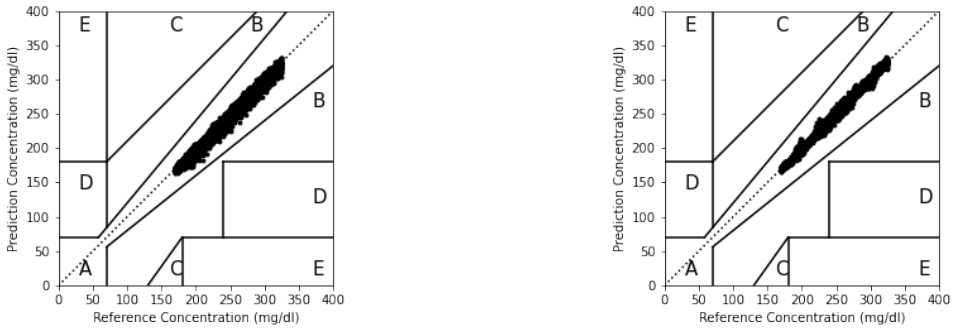
All the preceding statements get backed up with the data thrown by the Clark Error Grid Analysis.

		FFNN		CNN		LSTM	
PH	**Zone**	**Standard**	**Inductive**	**Standard**	**Inductive**	**Standard**	**Inductive**
30 min							
	A	100.00000	100.00000	100.00000	100.00000	100.00000	100.00000
	B	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	C	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	D	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	E	0.00000	0.00000	0.00000	0.00000	0.00288	0.00000
60 min							
	A	99.83471	99.95868	99.85537	99.91047	99.93113	99.96556
	B	0.16529	0.04132	0.1446	0.08953	0.06887	0.03444
	C	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	D	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	E	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
90 min							
	A	97.51607	99.00826	97.11203	98.79247	98.36547	98.90266
	B	2.40129	0.99174	2.82828	1.20753	1.59780	1.09734
	C	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	D	0.08264	0.00000	0.05969	0.00000	0.03673	0.00000
	E	0.00000	0.00000	0.00000	0.00000	0.00288	0.00000

Table 4.7: Inductive Transfer Learning over AIDAS

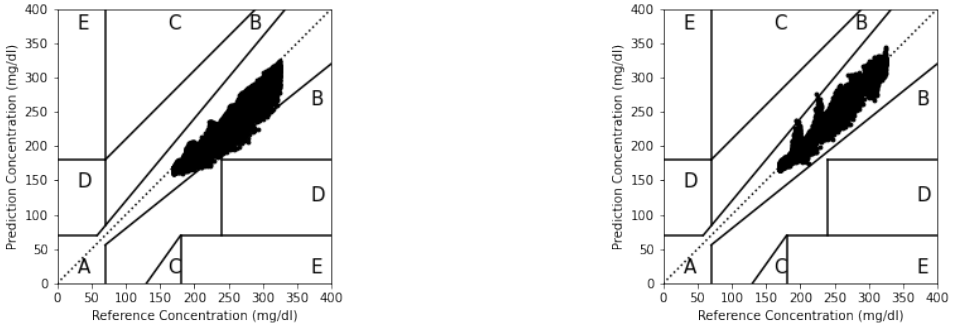
As show in Table 4.7, *Transfer Learning* helps models to improve their accuracy, without *Transfer Learning* models reported values on Zone C, D and E but once this technique is applied, these Zones' values get to zero. Another important remark is how despite statistical metrics classify LSTM as the best model, this medical analysis tells that at 90 minutes the FFNN is able to predict more accurate values since its value of Zone A is the highest among all and its value of Zone B is the lowest one.

Some CEGA graphs are plotted for informative purposes.



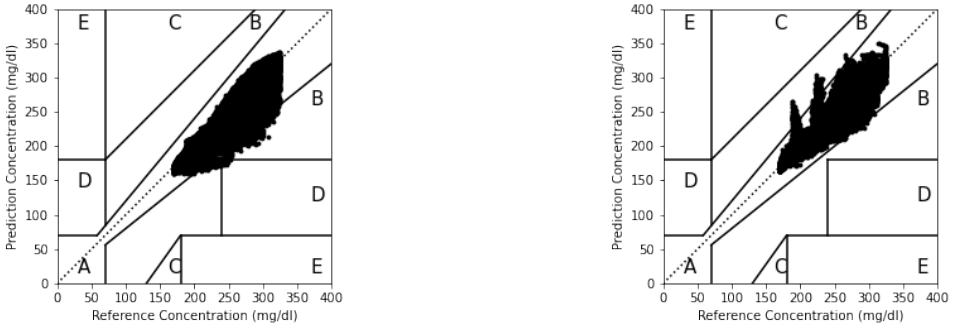
(a) Standard **(b)** Inductive

Figure 4.28: FFNN Clark Error Grid over Aidas at 30 min



(a) Standard **(b)** Inductive

Figure 4.29: FFNN Clark Error Grid over Aidas at 60 min



(a) Standard **(b)** Inductive

Figure 4.30: CRNN Clark Error Grid over Aidas at 90 min

After analyzing model architectures from the performance view through statistical and clinical measurements, now it is time to take a different analysis angle, training time. Although models' training time had not been measure directly, considering training epochs as a substitute metric seems like an acceptable approach. The following Figure would display some information relates to this matter with respect the TSALIKIAN data-set.

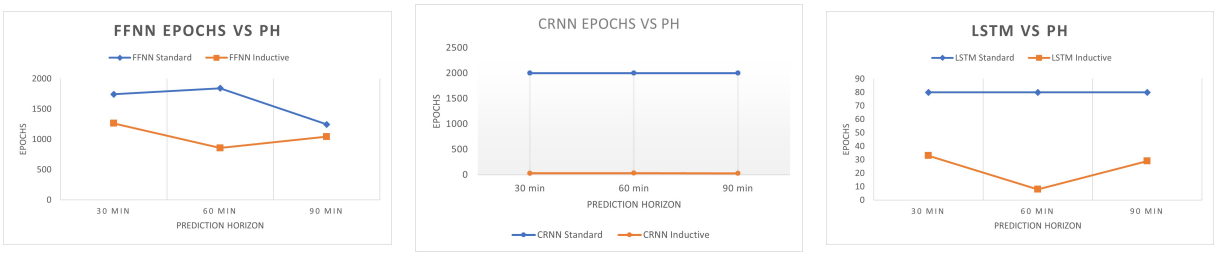


Figure 4.31: Epochs vs PH

As shown in Figure 4.31, There is a reduction in training epochs on both Standard and Inductive Techniques as the prediction horizon grows. In the FFNN case, it starts with 1742 and ends with 1264 epochs, a marginal gain of 30 % at 30 minutes, then at 90 minutes, begins with 1245 and finishes with 1043, a reduction of 16 %.

On the other hand, for the CRNN and LSTM case, the variation between standard and inductive training epochs is more stable, they start with 2000 and 80 epochs, respectively, and end with 30 epochs on average, tallying a decrease of 85 % and 50 % correspondingly.

4.2.2 Domain Adaptation Transfer Learning Results

The CRNN proposed by Albertetti et al. is used as a feature extractor and regressor predictor. This architecture is paired with a simple one-layer Neural Network Classifier which in its input layer receives the output of the feature extractor then this information is passed to a hidden layer of 100 Neurons connected to an output of 1 neuron, this simple structure would work as a binary classifier.

The analysis shall start with the TSALIKIAN data-set for which the main collected statistics are described in the following table.

PH	Metrics	Standard	Inductive	Domain
30 min				
	MAE	12.79891	12.02580	12.44822
	RMSE	17.97491	17.25181	17.66656
	SSGPE	0.01441	0.01327	0.01383
	R2	91.90	92.54	92.23
	MAPE	10.35	9.30	9.79
	MD	5.40	5.74	5.55
	FIT	71.63	72.76	72.19
60 min				
	MAE	19.10767	18.17952	18.92672
	RMSE	26.76900	25.64014	26.44825
	SSGPE	0.03239	0.02946	0.03084
	R2	81.67	83.33	82.56
	MAPE	15.32	14.13	15.14
	MD	3.62	3.81	3.66
	FIT	57.35	59.33	58.38
90 min				
	MAE	25.00482	23.42065	22.96895
	RMSE	34.79888	32.55209	31.56944
	SSGPE	0.05447	0.04732	0.04399
	R2	69.11	73.17	75.06
	MAPE	20.60	18.47	18.93
	MD	2.77	2.96	3.02
	FIT	44.64	48.40	50.24

Table 4.8: Domain Transfer Learning over TSALIKIAN

Marks in table 4.8 shows how Domain Adaption for the short and medium horizons has results sitting between those achieved by the Albertetti's structure in *Inductive Transfer Learning* and no *Transfer Learning* but in the long term performs better, however, this result do not match the best ones obtained in the *Inductive Transfer Learning* counterpart.

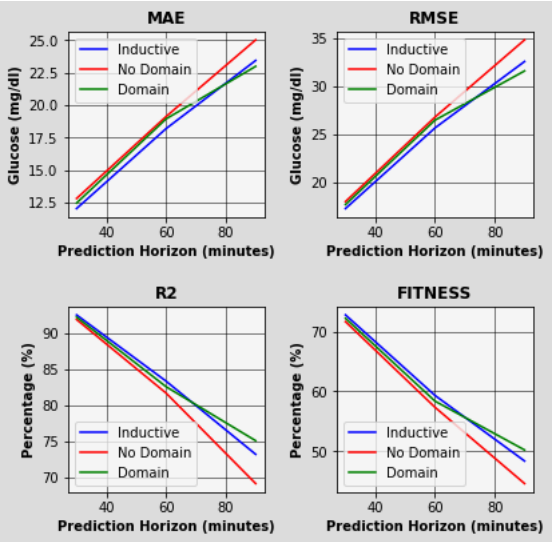


Figure 4.32: Metrics for Domain over TSALIKIAN

In the 30 and 60 minutes horizon, *Inductive Transfer Learning* outperforms *Domain Adaptation Transfer Learning*, both approaches tend to lower the bias and variance according to the reported metrics, for example, MAE slips at 30 min to 12 mg/dl in the inductive case and 12.44 mg/dl in the domain case, also RMSE decreases to 17.25 mg/dl and 17.66 mg/dl, respectively.

At 90 minutes, *Domain Adaptation Transfer Learning* can surpass in performance the *Inductive Transfer Learning* in a manner, for example, R2 metric jump to 75.06 % while the inductive only reaches 73.17 %. Even though this result is outstanding when comparing at the best response under the *Inductive Transfer Learning*, that is accomplished by the FFNN, is not a match, since this architecture achieves an R2 of 76.11 %, the same analogy goes to the other metrics.

The above statements has been proven with the respective Clark Error Grid Analysis listed down.

PH	Zone	Standard	Inductive	Domain
30 min				
	A	87.02327	89.07420	88.12363
	B	10.67807	9.49418	9.85425
	C	0.00864	0.00576	0.02016
	D	2.29001	1.42297	1.99908
	E	0.00000	0.00288	0.00288
60 min				
	A	76.04563	78.28235	76.71103
	B	19.38011	18.18182	19.26057
	C	0.30966	0.23476	0.27221
	D	4.21996	3.28811	3.72883
	E	0.04465	0.01296	0.02736
90 min				
	A	67.20820	69.45981	69.55390
	B	25.64812	24.66682	24.14257
	C	0.94481	0.69036	0.51273
	D	6.00972	5.11676	5.71686
	E	0.18915	0.06625	0.07393

Table 4.9: Clark Error Grid Analysis over TSALIKIAN

When comparing values at 90 minutes between FFNN from *Inductive* and *Domain Adaptation*, it is obvious how FFNN has higher Zone A and B predictions than the CRNN, 69 % to FNN and 67 % for CRNN when looking at Zone A.

Running the same procedure with AIDAS data-set, main gathered statistics are:

PH	Metrics	Standard	Inductive	Domain
30 min				
	MAE	3.11608	2.44913	3.06094
	RMSE	4.23732	3.36478	4.01335
	SSGPE	0.00039	0.00023	0.00033
	R2	98.94	99.37	99.09
	MAPE	1.28	1.01	1.26
	MD	13.53	17.21	13.77
	FIT	89.72	92.05	90.49
60 min				
	MAE	6.96720	4.67358	6.12929
	RMSE	9.56594	6.49287	8.50551
	SSGPE	0.00218	0.00102	0.00163
	R2	93.98	97.18	95.51
	MAPE	2.80	1.95	2.47
	MD	6.04	9.00	6.87
	FIT	75.62	83.26	78.89
90 min				
	MAE	11.94808	8.71806	11.10953
	RMSE	15.90960	11.99489	15.02677
	SSGPE	0.00601	0.00341	0.00495
	R2	83.26	90.50	86.20
	MAPE	4.81	3.57	4.45
	MD	3.51	4.81	3.78
	FIT	59.89	69.31	63.31

Table 4.10: Domain Transfer Learning over AIDAS

Table 4.10 highlights the same pattern occurring over TSALIKIAN data-set, with the difference that know is more evident that across all horizons, metrics belonging to *Inductive Transfer* are better than those of *Domain Transfer*.

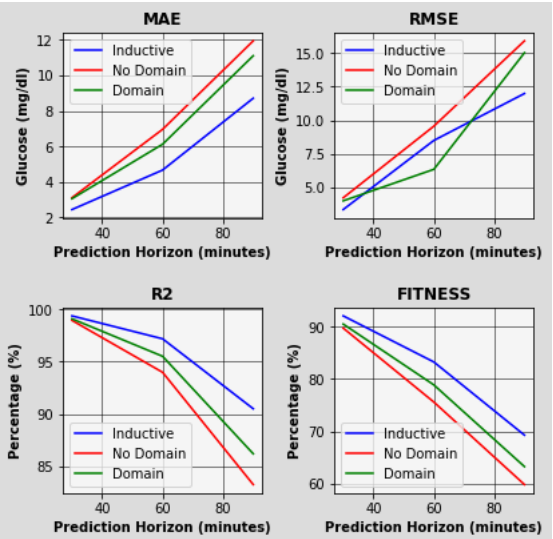


Figure 4.33: Metrics for Domain over AIDAS

Which gets confirmed after collecting marks from driving the Clark Error Grid Analysis over AIDAS,

PH	Zone	Standard	Inductive	Domain
30 min				
	A	100.00000	100.00000	100.00000
	B	0.00000	0.00000	0.00000
	C	0.00000	0.00000	0.00000
	D	0.00000	0.00000	0.00000
	E	0.00000	0.00000	0.00000
60 min				
	A	99.95868	99.91047	99.99311
	B	0.04132	0.08953	0.00689
	C	0.00000	0.00000	0.00000
	D	0.00000	0.00000	0.00000
	E	0.00000	0.00000	0.00000
90 min				
	A	97.11203	98.79247	98.22314
	B	2.82828	1.20753	1.73095
	C	0.00000	0.00000	0.00000
	D	0.00000	0.00000	0.04591
	E	0.00000	0.00000	0.00000

Table 4.11: Clark Error Grid Analysis over AIDAS

Chapter 5

Conclusions

The goal of this research was to study the impact of *Transfer-Learning* on *Blood Glucose Forecasting* across people suffering from different types of diabetes. Thus, there were proposed the analysis of two methodologies, inductive and domain adaptation, to see how lack of clinical data, both in terms of quality and quantity, could be addressed, as explained at the end of [Section 1.1].

As seen in the results section, it was confirmed that *Transfer-Learning* is highly dependant on the fine-tuning of the glucose reference models, poor models would transfer poor knowledge and vice-versa, and that predictions worsen, from the statistical and clinical view, with the growth of the prediction horizon.

Also, empirical data links the effectiveness of *Transfer-Learning*, for both approaches, to the length of the prediction horizon, the larger the more effective *Transfer-Learning* gets over the SOTA selected on this thesis.

On the other side, although *Transfer-Learning* allowed to fine-tune SOTA glucose models over a small amount of data in a positive manner, one peculiarity is that from the clinical perspective, predictions lose grip, regardless of the architecture used, since values on positive Zones different from A and B get bigger, when looking at glucose base reference models, which implies that despite models statistical metrics get better after applying *Transfer-Learning*, there is space for mispredictions that could lead to a bad diagnosed or treatment.

Furthermore, it was demonstrated that despite both *Transfer-Learning* work, the *Inductive Transfer-Learning* provides better performances, on specific cases, than those reported for the *Domain Adaptation Transfer-Learning*, plus, experimental data tends to show that complex architectures such as LSTM functions better for the short and medium scenarios whereas simpler ones such as FFNN or CRNN works better at large ones. One simple explanation is that at large horizons the number of parameters from simpler architectures are smaller when compared to complex ones, so the number of computations decreases, thus the precision rise.

From an analytical point of view is hard to determine which type of SOTA is better than others due to the marginal difference among measurements, all were responsive to the *Transfer-Learning* approaches proposed.

Finally, *Transfer-Learning* permitted to diminishes training time over small data-sets, as shown with the variation of epochs on the *Induction Transfer Learning* section.

Thus, the objectives of this thesis were fulfilled, since it was proved that SOTA can be used for Transferring Knowledge on the Glucose prediction field with excellent performances over small data sets with an important reduction of training times.

Chapter 6

Future Work

Further studies shall be made on *Incremental Learning* applied to *Blood Glucose Forecasting*, this is important because there is an implicit assumption inside the *Machine Learning* community that always exist a large data-set on the domain source to train an algorithm but in real-life applications, like *Glucose Prediction*, this is not always the case.

Incremental Learning is a machine learning paradigm that allows an algorithm to adjust its learning process as soon as new data emerges, so counting with big data sets is not a requirement anymore. Also, it helps to overcome the problem of forgetting previously learned knowledge, as it happens in *Classical Transfer Learning*, enabling the possibility of acquiring a large complex set of patterns while retaining the corresponding data.

This would be an extraordinary trait for a Deep Learning Algorithm applied to Glucose Prediction due to its flexibility to cover a whole ample range of diverse profiles.

Bibliography

[1] WHO. *Diabetes*. URL: https://www.who.int/health-topics/diabetes#tab=tab_3. (accessed: 12.12.2020) (cit. on p. 1).

[2] The Pancreas Center Columbia Surgery. *The pancreas and its functions*. URL: https://columbiasurgery.org/pancreas/pancreas-and-its-functions. (accessed: 12.12.2020) (cit. on p. 1).

[3] Marius Eriksen. *Model Predictive Control for Closed-Loop Insulin Delivery*. URL: https://nbviewer.jupyter.org/gist/mariusae/18a62db9cc32d09dc691fd4f78dcdbfa. (accessed: 12.12.2020) (cit. on p. 1).

[4] International Diabetes Federation. *What is diabetes*. URL: https://www.idf.org/aboutdiabetes/what-is-diabetes/facts-figures.html. (accessed: 12.12.2020) (cit. on p. 2).

[5] Anthony J Webb and Brent D Cameron. «Noninvasive in vivo Glucose Sensing using an iris based technique». In: *Optical Diagnostics and Sensing XI: Toward Point-of-Care Diagnostics; and Design and Performance Validation of Phantoms Used in Conjunction with Optical Measurement of Tissue III*. Vol. 7906. International Society for Optics and Photonics. 2011, p. 79060C (cit. on p. 2).

[6] S Welch, SSP Gebhart, RN Bergman, and LS Phillips. «Minimal model analysis of intravenous glucose tolerance test-derived insulin sensitivity in diabetic subjects». In: *The Journal of Clinical Endocrinology & Metabolism* 71.6 (1990), pp. 1508–1518 (cit. on p. 4).

[7] Steen Andreassen, Roman Hovorka, Jonathan Benn, Kristian G Olesen, and Ewart R Carson. «A model-based approach to insulin adjustment». In: *AIME 91*. Springer, 1991, pp. 239–248 (cit. on p. 4).

[8] Scott M Pappada, Brent D Cameron, Paul M Rosman, Raymond E Bourey, Thomas J Papadimos, William Olorunto, and Marilyn J Borst. «Neural network-based real-time prediction of glucose in patients with insulin-dependent diabetes». In: *Diabetes technology & therapeutics* 13.2 (2011), pp. 135–141 (cit. on pp. 4, 27).

[9] Taiyu Zhu, Xi Yao, Kezhi Li, Pau Herrero, and Pantelis Georgiou. «Blood glucose prediction for type 1 diabetes using generative adversarial networks». In: *CEUR Workshop Proceedings*. Vol. 2675. 2020, pp. 90–94 (cit. on p. 4).

[10] Jun Yang, Lei Li, Yimeng Shi, and Xiaolei Xie. «An ARIMA model with adaptive orders for predicting blood glucose concentrations and hypoglycemia». In: *IEEE journal of biomedical and health informatics* 23.3 (2018), pp. 1251–1260 (cit. on p. 4).

[11] Ivan Contreras, Silvia Oviedo, Martina Vettoretti, Roberto Visentin, and Josep Vehi. «Personalized blood glucose prediction: A hybrid approach using grammatical evolution and physiological models». In: *PloS one* 12.11 (2017), e0187754 (cit. on p. 4).

[12] Arthur Bertachi, Lyvia Biagi, Ivan Contreras, Ningsu Luo, and Josep Vehi. «Prediction of Blood Glucose Levels And Nocturnal Hypoglycemia Using Physiological Models and Artificial Neural Networks.» In: *KHD@ IJCAI*. 2018, pp. 85–90 (cit. on p. 4).

[13] Douglas M Hawkins. «The problem of overfitting». In: *Journal of chemical information and computer sciences* 44.1 (2004), pp. 1–12 (cit. on p. 4).

[14] Adiwinata Gani, Andrei V Gribok, Srinivasan Rajaraman, W Kenneth Ward, and Jaques Reifman. «Predicting subcutaneous glucose concentration in humans: data-driven glucose modeling». In: *IEEE Transactions on Biomedical Engineering* 56.2 (2008), pp. 246–254 (cit. on p. 4).

[15] Chris M Bishop. «Training with noise is equivalent to Tikhonov regularization». In: *Neural computation* 7.1 (1995), pp. 108–116 (cit. on p. 4).

[16] Carmen Perez-Gandia, A Facchinetti, G Sparacino, C Cobelli, EJ Gomez, M Rigla, Alberto de Leiva, and ME Hernando. «Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring». In: *Diabetes technology & therapeutics* 12.1 (2010), pp. 81–88 (cit. on pp. 4, 24, 27, 31, 36).

[17] Fayrouz Allam, Zaki Nossai, Hesham Gomma, Ibrahim Ibrahim, and Mona Abdelsalam. «A recurrent neural network approach for predicting glucose concentration in type-1 diabetic patients». In: *Engineering Applications of Neural Networks*. Springer, 2011, pp. 254–259 (cit. on p. 4).

[18] Priyansh Saxena, Akshat Maheshwari, and Saumil Maheshwari. «Predictive modeling of brain tumor: A Deep learning approach». In: *Innovations in Computational Intelligence and Computer Vision*. Springer, 2021, pp. 275–285 (cit. on p. 5).

[19] Sheikh Md Hanif Hossain, SM Raju, and Amelia Ritahani Ismail. «Predicting Pneumonia and Region Detection from X-Ray Images using Deep Neural Network». In: *arXiv preprint arXiv:2101.07717* (2021) (cit. on p. 5).

[20] Anh T Tran, Cuong V Nguyen, and Tal Hassner. «Transferability and Hardness of Supervised Classification Tasks—Supplemental material—». In: () (cit. on p. 5).

[21] Yajie Bao, Yang Li, Shao-Lun Huang, Lin Zhang, Lizhong Zheng, Amir Zamir, and Leonidas Guibas. «An information-theoretic approach to transferability in task transfer learning». In: *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2019, pp. 2309–2313 (cit. on p. 5).

[22] Cuong Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. «Leep: A new measure to evaluate transferability of learned representations». In: *International Conference on Machine Learning*. PMLR. 2020, pp. 7294–7305 (cit. on p. 5).

[23] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. «Transfer learning for time series classification». In: *2018 IEEE international conference on big data (Big Data)*. IEEE. 2018, pp. 1367–1376 (cit. on p. 5).

[24] Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. «The UCR time series archive». In: *IEEE/CAA Journal of Automatica Sinica* 6.6 (2019), pp. 1293–1305 (cit. on p. 5).

[25] Maxime De Bois, Mouni m A El Yacoubi, and Mehdi Ammi. «Adversarial multi-source transfer learning in healthcare: Application to glucose prediction for diabetic people». In: *Computer Methods and Programs in Biomedicine* 199 (2021), p. 105874 (cit. on p. 6).

[26] Ananth Bhimireddy, Priyanshu Sinha, Bolu Oluwalade, Judy Wawira Gichoya, and Saptarshi Purkayastha. «Blood Glucose Level Prediction as Time-Series Modeling using Sequence-to-Sequence Neural Networks». In: *CEUR Workshop Proceedings*. 2020 (cit. on pp. 6, 27, 31, 36).

[27] JAEB CENTER FOR HEALTH RESEARCH. *JDRF Continuous Glucose Monitoring (JDRF CGM RCT)*. URL: https://www.jaeb.org/projects/. (accessed: 13.12.2020) (cit. on p. 9).

[28] ED Lehmann, T Deutsch, ER Carson, and PH Sonksen. «AIDA: an interactive diabetes advisor». In: *Computer methods and programs in biomedicine* 41.3-4 (1994), pp. 183–203 (cit. on p. 10).

[29] R Borg, JC Kuenen, B Carstensen, H Zheng, DM Nathan, RJ Heine, J Nerup, K Borch-Johnsen, and DR Witte. «Real-life glycaemic profiles in non-diabetic individuals with low fasting glucose and normal HbA 1c: the A1C-Derived Average Glucose (ADAG) study». In: *Diabetologia* 53.8 (2010), pp. 1608–1611 (cit. on p. 16).

[30] diaTribelearn. *CGM and Time-in-Range: What Do Diabetes Experts Think About Goals?* URL: https://diatribe.org/cgm-and-time-range-what-do-diabetes-experts-think-about-goals. (accessed: 16.12.2020) (cit. on p. 17).

[31] Thales Sehn Körting. *How DTW (Dynamic Time Warping) algorithm works*. URL: https://www.youtube.com/watch?v=_K1OsqCicBY. (accessed: 16.12.2020) (cit. on p. 18).

[32] Alessandro Aliberti, Irene Pupillo, Stefano Terna, Enrico Macii, Santa Di Cataldo, Edoardo Patti, and Andrea Acquaviva. «A multi-patient data-driven approach to blood glucose prediction». In: *IEEE Access* 7 (2019), pp. 69311–69325 (cit. on pp. 24, 27).

[33] Kezhi Li, John Daniels, Chengyuan Liu, Pau Herrero, and Pantelis Georgiou. «Convolutional recurrent neural networks for glucose prediction». In: *IEEE journal of biomedical and health informatics* 24.2 (2019), pp. 603–613 (cit. on p. 24).

[34] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario Marchand, and Victor Lempitsky. «Domain-adversarial training of neural networks». In: *The journal of machine learning research* 17.1 (2016), pp. 2096–2030 (cit. on pp. 26, 41).

[35] Chiara Zecchin, Andrea Facchinetti, Giovanni Sparacino, and Claudio Cobelli. «How much is short-term glucose prediction in type 1 diabetes improved by adding insulin delivery and meal content information to CGM data? A proof-of-concept study». In: *Journal of diabetes science and technology* 10.5 (2016), pp. 1149–1160 (cit. on p. 27).

[36] Jonas Freiburghaus, Aicha Rizzotti-Kaddouri, and Fabrizio Albertetti. «A Deep Learning Approach for Blood Glucose Prediction of Type 1 Diabetes». In: () (cit. on pp. 27, 31, 36).

[37] Stavroula G Mougiakakou, Aikaterini Prountzou, Dimitra Iliopoulou, Konstantina S Nikita, Andriani Vazeou, and Christos S Bartsocas. «Neural network based glucose-insulin metabolism models for children with type 1 diabetes». In: *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. 2006, pp. 3545–3548 (cit. on p. 27).

[38] Gavin Robertson, Eldon D Lehmann, William Sandham, and David Hamilton. «Blood glucose prediction using artificial neural networks trained with the AIDA diabetes simulator: a proof-of-concept pilot study». In: *Journal of Electrical and Computer Engineering* 2011 (2011) (cit. on p. 27).

[39] Jaouher Ben Ali, Takoua Hamdi, Nader Fnaiech, Veronique Di Costanzo, Farhat Fnaiech, and Jean-Marc Ginoux. «Continuous blood glucose level prediction of Type 1 Diabetes based on Artificial Neural Network». In: *Biocybernetics and Biomedical Engineering* 38.4 (2018), pp. 828–840 (cit. on p. 27).

[40] A.I. Wiki. *A Beginner's Guide to Neural Networks and Deep Learning*. URL: https://wiki.pathmind.com/neural-network. (accessed: 23.04.2021) (cit. on p. 29).

[41] Machine Learning Mastery. *How Do Convolutional Layers Work in Deep Learning Neural Networks?* URL: https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/. (accessed: 23.04.2021) (cit. on p. 30).

[42] Towards DataScience. *Illustrated Guide to LSTM's and GRU's: A step by step explanation*. URL: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21. (accessed: 23.04.2021) (cit. on p. 30).

[43] Sebastian Ruder. «An overview of gradient descent optimization algorithms». In: *arXiv preprint arXiv:1609.04747* (2016) (cit. on p. 32).

[44] Diederik P Kingma and Jimmy Ba. «Adam: A method for stochastic optimization». In: *arXiv preprint arXiv:1412.6980* (2014) (cit. on p. 32).

[45] Machine Learning Mastery. *How to Choose Loss Functions When Training Deep Learning Neural Networks*. URL: https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learning-neural-networks/. (accessed: 23.04.2021) (cit. on p. 32).

[46] Lutz Prechelt. «Early stopping-but when?» In: *Neural Networks: Tricks of the trade*. Springer, 1998, pp. 55–69 (cit. on p. 32).

[47] Kamuran Turksoy, Elif S Bayrak, Lauretta Quinn, Elizabeth Littlejohn, Derrick Rollins, and Ali Cinar. «Hypoglycemia early alarm systems based on multivariable models». In: *Industrial & engineering chemistry research* 52.35 (2013), pp. 12329–12336 (cit. on p. 33).

[48] Yoichi Hayashi and Rudy Setiono. «Combining neural network predictions for medical diagnosis». In: *Computers in biology and medicine* 32.4 (2002), pp. 237–246 (cit. on p. 33).

[49] Boris P Kovatchev, Linda A Gonder-Frederick, Daniel J Cox, and William L Clarke. «Evaluating the accuracy of continuous glucose-monitoring sensors: continuous glucose–error grid analysis illustrated by TheraSense Freestyle Navigator data». In: *Diabetes Care* 27.8 (2004), pp. 1922–1928 (cit. on p. 34).

[50] Aurelien Geron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019 (cit. on p. 38).

[51] Sinno Jialin Pan and Qiang Yang. «A survey on transfer learning». In: *IEEE Transactions on knowledge and data engineering* 22.10 (2009), pp. 1345–1359 (cit. on p. 40).