FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Attention mechanisms to improve forecasting performance

António Gonçalo Silva Pinto da Cunha

Mestrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Prof. Doutor José Nuno Moura Marques Fidalgo

October 13, 2023

Resumo

No paradigma atual de regulação e operação dos sistemas de energia existe uma grande dependência de estimativas de diversos tipos de variáveis, tais como o preço da eletricidade - que é altamente volátil -, a produção de fontes de energia renováveis, o consumo e muitas outras. Especificamente, a previsão de carga é abrangentemente utilizada pelas empresas fornecedoras de energia para ajudar a prever a quantidade de energia necessária para fornecer num determinado período. Esta previsão requer a identificação dos fatores que possam potencialmente influenciar as grandezas a prever. Frequentemente, este impacto é dependente do contexto, como acontece com a dependência do consumo face à temperatura. A temperatura tem um impacto não-linear no consumo, porque, por exemplo, depende da época do ano. No verão, quando a temperatura aumenta, o consumo aumenta, já no inverno, quando a temperatura diminui, o consumo também aumenta.

Assim, o objetivo principal desta dissertação é a implementação de uma metodologia data-driven (baseada nos dados) que permita caraterizar, de forma totalmente automática e sem discricionariedade nem apriorismos, a dependência do consumo face à temperatura. O sistema a implementar deve, não apenas permitir interpretar a relação entre consumo e temperatura, como também melhorar a qualidade da previsão. A principal técnica que será utilizada nesta implementação são os mecanismos de atenção quando aplicados a redes neuronais artificiais, e um dos objectivos mais importantes deste trabalho é compreender o efeito desta técnica nos resultados globais das previsões.

Para isso, primeiramente, foi feita uma revisão do estado da arte relativamente aos mecanismos de atenção, à previsão de séries temporais e interpretabilidade dos modelos. A partir daí, considerando as suas vantagens e desvantagens, foi selecionada uma arquitetura que possibilitou o cálculo da atenção global de todas as variáveis e também o cálculo da matriz de atenção, o que permitiu encontrar os períodos em que a atenção é maior e, dessa forma, inferir o efeito da temperatura na carga.

Os resultados deste trabalho conduziram a várias conclusões interessantes relativamente ao desempenho dos modelos baseados na atenção e, mais importante ainda, relativamente à influência da temperatura nos padrões de consumo no caso italiano, de onde foram retirados os dados. Verificou-se que, com a inclusão da atenção, o desempenho dos modelos melhora, em geral, e, além disso, no que respeita ao impacto da temperatura no consumo, verificou-se que a atenção é maior em períodos de maior calor e, especialmente, quando estes apresentam grandes variações, algo corroborado pela inclusão de duas novas variáveis no modelo relativas a estas condições. Era também expetável um aumento da atenção em períodos de temperaturas muito baixas. Ou seja, pode-se concluir que a atenção não foi capaz de detetar este caso, o que pode indicar uma limitação. No entanto, isto pode ser justificado pelo facto de o sistema de aquecimento em Itália depender fortemente do gás e, por isso, não ter o mesmo impacto que os sistemas de ar condicionado durante os períodos de calor. Portanto, foi possível inferir que o consumo é mais sensível a esses períodos, precisamente devido a essa elevada utilização de aparelhos de ar condicionado.

Abstract

In the current paradigm of energy systems regulation and operation there is a very high dependency on estimations of several types of variables, such as the price of electricity - which is highly volatile -, the production of renewable energy sources, the demand and many others. Specifically, load forecasting is widely used by energy-providing companies to help predict the demand of power required to supply in a certain period. This forecasting requires the identification of the main factors that may influence the variables that are needed to predict. More often than not, this influence is dependent on the context, as it is the case with the variation of the load consumption in relation to the temperature. Temperature has a non-linear effect regarding the consumption, because, for example, it depends on the time of year. In the summer when the temperatures rise, the demand also rises, however, in the winter, when the temperatures decrease, the demand also rises.

So, the main purpose of this thesis is to implement a data-driven methodology that allows the characterization, in a fully automatic fashion, of the dependency of the load consumption regarding the temperature. This implementation should, not only make possible the interpretation of the relation between consumption and temperature, but also improve the accuracy of the prediction. The main technique that will be used in this implementation is attention mechanisms when applied to artificial neural networks, and of the most important goals of this work is to understand the effect of this technique on the overall results of the predictions.

For that, first of all, a revision was made on the state of the art regarding attention mechanisms, time series forecasting and model interpretability. From that, considering its advantages and drawbacks, an architecture was selected that made possible the calculation of the global attention of all features and also the calculation of the attention matrix, which made it possible to find the periods in which the attention is higher and, that way, inferring the effect of temperature on the load.

The results of this work have led to several interesting conclusions regarding the performance of attention-based models and, more importantly, regarding the influence of temperature on consumption patterns in the Italian case, from which the data were drawn. In particular, it was found that with the inclusion of attention, the performance of the models generally improves and, in addition, regarding the impact of temperature on consumption, it was found that attention is higher in periods of greater heat and specially when these present great variations, something corroborated by the inclusion of two new variables in the model concerning these conditions. It was also expected an increase in attention in the initial part where there is a period of very low temperatures. Meaning that it be can concluded that attention was not able to detect this case, which may indicate a limitation. However, this can be justified by the fact that the heating system in Italy relies heavily on gas and, therefore, does not have the same impact as the air conditioning systems during periods of heat. It was therefore possible to infer that consumption is more sensitive to such periods, for example, precisely due to that high use of air-conditioners.

Agradecimentos

Agradeço ao meu orientador, Professor José Fidalgo, por todo o apoio e acompanhamento ao longo da elaboração desta tese, nomeadamente pelos constantes esclarecimentos que me foi prestando, a flexiblidade e disponibilidade que sempre demonstrou e, não menos importante, por toda a simpatia.

A todos os colegas que me acompanharam nesta jornada.

A todos os meus amigos, em especial aqueles que fiz ao longo deste percurso académico, e com os quais convivi diariamente nestes anos, pela amizade, companheirismo e fraternidade.

Aos meus parentes próximos pelo incentivo.

Finalmente, aos meus pais e irmãos pela paciência, suporte, consideração, carinho, e uma variedade infinita de gestos atenciosos, que sempre me deram, e que sem os quais garantidamente não teria alcançado o que alcancei.

António Gonçalo Silva Pinto da Cunha

"It is very difficult to predict
— especially the future."

Niels Bohr

Contents

List of Figures

List of Tables

Abbreviations

ARMA	AutoRegressive Moving Average
ARIMA	AutoRegressive Integrated Moving Average
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
ANN	Artificial Neural Network
FNN	Feedforward Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
AM	Attention Mechanism
MW	MegaWatt
MSE	Mean Squared Error
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error
MAPE	Mean Absolute Percentage Error

Chapter 1

Introduction

1.1 Motivation and goals

The influence of load forecasting in today's energy production and generation cannot be overstated. It is used to plan the reinforcement and expansion of the grid, to estimate financial profits, to better manage the demand, identifying the factors that have an effect on the consumption, to program the operation and the maintenance of the grid, to study the inclusion of distributed generation in the grid, and many others. With that being said, it is of a great concern the maximization of the quality of forecasts.

In the case of this thesis, there are two main goals. The first one is to get a better understanding of the dependency of the power consumption regarding temperature. One of the main challenges, and characteristics, of this problem is that this relation is highly dependent on the circumstances. For example, the temperature has a non-linear effect on the power consumption, meaning, it depends on the time of the year: In the summer, when the temperature rises, the demand also rises, however, in the winter, when the temperature decreases, the demand is up again. Furthermore, it also depends on the cumulative effect, for instance, four days of intense heat and high temperatures in the summer causes an increase in the demand, which is not the same as having four days of intense heat spread across a month. It may also depend on the thermal inertia of the buildings. These are all possible deductions of the relationship between demand and temperature. The second goal is to improve the performance of the forecasts. To help achieve these two goals the plan is to implement a neural network-based model combined with an attention mechanism, which will require the study of specialized bibliography concerning the theory and implementation of said models and, if available, its characteristics when applied to field of load forecasting.

1.2 Dissertation structure

This document consists of five chapters.

The first chapter serves as an introduction, establishing the motivation and outlining the main goals to be achieved by the end of this work.

In the second chapter, titled "State of the Art", a brief overview of load forecasting characteristics and its various approaches is done. This is followed by an exploration of key theoretical concepts around the development and functioning of attention mechanisms. A more focused analysis is then conducted on specific models that are pertinent to time series forecasting and contain interpretability mechanisms.

Drawing on the investigation conducted in the second chapter, the third chapter explains the main model employed throughout the work, as well as the treatment of the data used.

In the fourth chapter, the primary results of this work are presented. These include comparisons of the main model's performance with other models using the same dataset; analysis of global attention importance of all features, extracted from the model, and comparison to other global feature importance methods; an in-depth analysis using the attention matrix, aiming to discern periods of heightened attention and infer the effect of temperature on the load.

Lastly, the fifth chapter concludes this work by drawing together the main findings and suggesting potential avenues for further analysis in future works.

Chapter 2

State of the art

This chapter presents a general description of the most common procedures, methods and techniques used in the scope of load forecasting. After that, the purpose is to dive deeper in how the attention mechanisms work and where they stand between all those previous techniques. Finally an investigation on interpretability methods in time series forecasting is done.

2.1 Load forecasting time horizon

One of the first things that should be taken into consideration when building models for these types of load forecasts is the time frame in which the predictions will be estimated. Usually, the time scale can be divided in: Very Short Term Forecasting, Short Term Forecasting, Medium Term Forecasting and Long Term Forecasting. Very short term typically means a span of a few minutes to a few hours. Short term can go from several minutes to some days. Medium term generally stands for forecasts that range from a few days to a few months. Finally, for a forecast that predicts results within a range of months, quarters, semesters, or even years, the time frame is said to be Long term [1]. In the case of this work, due to the nature of the meteorological predictions, which usually present bigger errors for larger forecasting horizons, the time scale used will be short term, otherwise, it's not possible to include the prediction of the temperatures in a consistent way.

2.2 Short-term forecasting techniques

Short term load forecasts can be modeled according to several kinds of techniques, such as: statistical methods, probabilistic methods, artificial intelligence-based methods, hybrid methods, and others [1]. In the following sections some of these methods will be described, particularly the artificial intelligence-based ones, since they are the main focus of this work.

2.3 Statistical approaches

Statistical methods utilize a mathematical combination between historical demand data and other variables, chronological ones and others regarding weather, to perform the forecasts. The predominant techniques within the statistical methods are ARMA (*AutoRegressive Moving Average*) and ARIMA (*AutoRegressive Integrated Moving Average*) [1].

ARMA is commonly used in the realm of time series analysis and, as the name implies, it combines the two basic models AR (*Auto-Regressive*), which depends on the past values of the series, and MA (*Moving Average*), which depends on past errors. This technique doesn't apply to all time series, it requires time series that are stationarized, by, for example, a process of differentiation. ARIMA models, on the other hand, already explicitly include this differentiation in its formulation [1].

2.4 Artificial Intelligence vs Machine Learning vs Deep Learning

Before proceeding to more advanced stages of this work, it is of great interest to, first of all, in a general way, introduce some of the fundamental notions in this field.

The term *Artificial Intelligence* (AI) was first introduced by Stanford Professor John McCarthy, who defined it as

> " the science and engineering of making intelligent machines." [2]

Although, initially, AI was created as way to solve complex, but straight-forward, mathematical problems, the biggest challenge is now the creation of tools to help solve tasks that are relatively easy for humans to do, but are hard to formulate, for example tasks that require intuition and are instinctively solved by us, such as the recognition of certain elements in a picture. These intelligent tools can be trained to perform a wide range of tasks, from recognizing patterns to making decisions, and even engaging in natural conversations with humans [3]. In terms of scientific research, these methods have been in applied in a number of fields, such as Statistics, Pattern Recognition, Signal and Image Processing, Computer-aided Medical Diagnosis, Machine Vision, Data Mining, etc [4].

It is known that a human being by gaining knowledge trough life is capable of making wiser and more rightful decisions. Therefore, the biggest challenge for AI is to give machines this ability to gain knowledge from something. So it emerged *Machine Learning* (ML), which is a subgroup of AI that enables computers, by using algorithms and statistical models, to improve their performance on a specific task trough experience, without explicit programming, which means that these kinds of algorithms can learn from data by extracting patterns and even make subjective decisions based on that data. This learning can be supervised and unsupervised. In supervised learning, a model is trained based on labels, eventually attributed by humans, and the goal is to label new unlabeled data based on the patterns of the input data. Consequently, in unsupervised learning all the data is unlabeled and the goal is for the machine to make its own predictions by discovering patterns by itself [4] [5].

Deep Learning (DL) is itself a subset of ML in which *Artificial Neural Networks* (ANN) are used to perform tasks by analyzing sets of data and finding complex patterns in them. It differs from ML in the sense that the feature extraction of DL is much more automated and so it removes even more of the human intervention, therefore making possible the use of even bigger sets of data. These tools are used in several different areas such as: Language processing, computer vision, speech recognition, and many others, influencing a large number of real-world situations by solving or facilitating the solution of complex problems [4] [1].

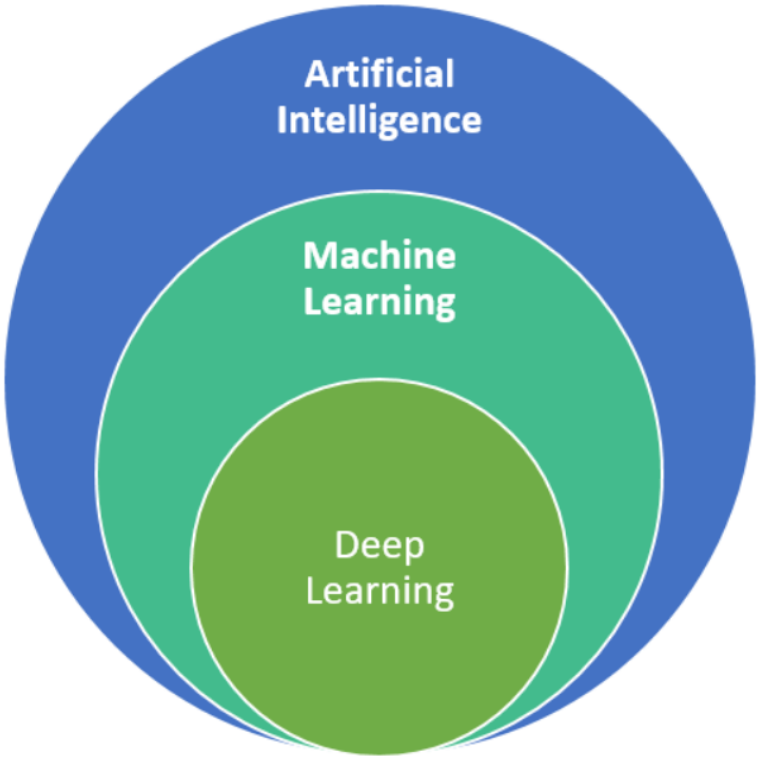

Figure 2.1: AI, ML and DL[6]

2.5 Artificial Neural Networks

ANN are defined as computational models that try to emulate an animal's central nervous system and therefore given the ability to "learn", in a broader sense. The building-block unit of these networks are nodes, also called neurons, that, by being linked to each other, are capable of, after processing the data locally, transmitting information across the neural network. This information is transmitted by the linking of these neurons, the synapses, characterized by connection weights, that represent the synaptic/association strength between neurons of the brain/system. By modifying these weights, with learning algorithms, networks acquire knowledge. In relation to the neurons, there are some important formulas, 2.1 and 2.2 that can represent it [7] [3]:

$$v_K = \sum_{j=0}^{m} \omega_{Kj}(X_j) \tag{2.1}$$

$$y_k = \Phi(v_K) \tag{2.2}$$

In which:

- ω_{Kj}, represents the weight and specifies the importance of the X_j signal in relation to the neuron K;
- v_K, represents the summation, by linear combination, of all input signals to determine the output of the neuron K;
- Φ, represents the activation function where v_K will be inserted into, resulting in the output, y_K.

There are a lot of different kinds of activation functions, and their importance should not be taken lightly, since they have a major influence in calculating the inputs of the neurons. It's important to note that, if the activation function is nonlinear the neural network will be able to modelize nonlinear systems. On the other hand, if the activation function is linear, the neural network will only be able to simulate linear systems. Here are two of the main activation functions [7] [3]:

$$Sigmoid: \quad (1+e^{-x})^{-1} \tag{2.3}$$

$$Softmax: \quad e^{x_i}(\sum_{j}^{n} e^{x_j})^{-1} \tag{2.4}$$

Although ANN proved to be very useful and enough for the majority of practical applications, as it is, its simpler version, it has some limitations. One example of those limitations is the fact that this kind of neural networks flow only one way, called *Feedforward Neural Networks* (FNN), (as seen in Figure 2.2) and therefore are not capable of retaining information from previous states which can be very pertinent to a big number of problems that require the processing of sequential data. Covering some of those needs the *Recurrent Neural Networks* (RNN) were created.

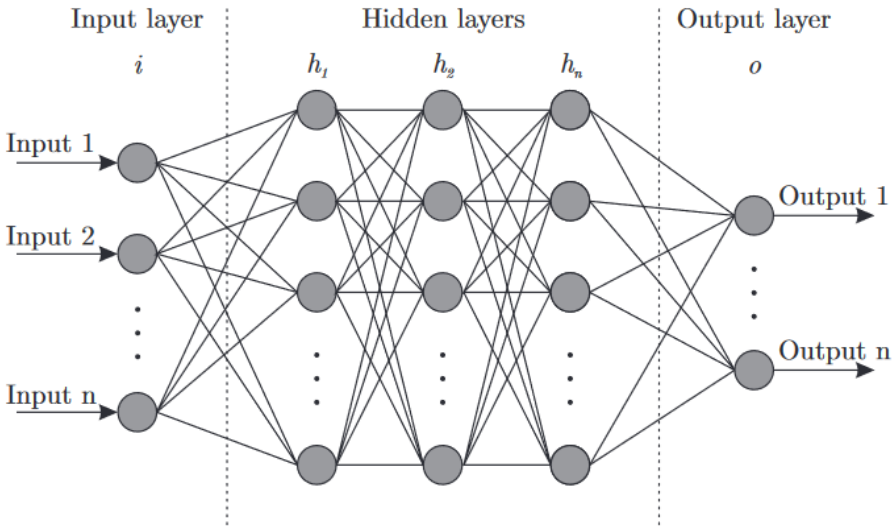

Figure 2.2: Example of FNN architecture [8]

2.6 Recurrent Neural Networks

According to [9], RNNs are the most popular deep learning method when it comes to short term load forecasting.

In RNNs hidden layers are created in order to feed its outputs to itself as seen in Figure 2.3, thus creating a cycle in which inputs of certain neurons are influenced by previous outputs of those same neurons.

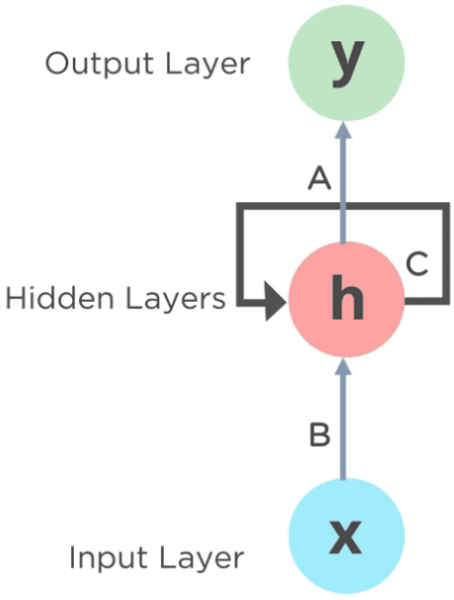

Figure 2.3: Example of simple RNN structure [10]

Analysing the simple representation of Figure 2.4 more carefully we can conclude that in each time step (represented by $(t-1)$, t and $(t+1)$) the result of the input neuron x is stored in a hidden layer , h, and then propagated to the outputs, o. The hidden layer can be expressed by:

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

In which a function f takes as arguments the input of the previous neuron stored in the hidden neuron h_{t-1} and the input of the current neuron x_t. U represents a weight matrix that does the parametrization of the input to hidden connections, W is a weight matrix that does the parametrization of the hidden to hidden connections and V is a weight matrix that does the parametrization of the hidden to output connections. L works as a loss function that is intended to be minimized in regards to the training targets y, this is accomplished by L computing 2.9 ($y^{'}$) and comparing it to y. With b and c as bias vectors, all these previous relations can be represented by [3]:

$$a_t = b + Wh_{t-1} + Ux_t \tag{2.6}$$
$$h_t = tanh(a_t) \tag{2.7}$$
$$o_t = c + Vh_t \tag{2.8}$$
$$y^{'} = softmax(o^t) \tag{2.9}$$

Although RNNs are very useful and accurate for situations where the information needed is located in time steps close to the current state, in practical terms, the further we go in a series the lower the effectiveness is. This happens because when computing the loss value necessary to obtain the gradients that will adjust the weights according to the change in the errors, since this operation happens according to the previous layer, small gradients tend to become even smaller in upcoming layers even if that information is of great relevance. Therefore, initial components tend to lose influence even though they might be important, culminating in short-term memory. For that reason, RNNs do not work as well in situations with large sets of data. This phenomenon is called *Vanishing gradient problem* and more can be read about it, specially regarding its theoretical aspect, in [11].

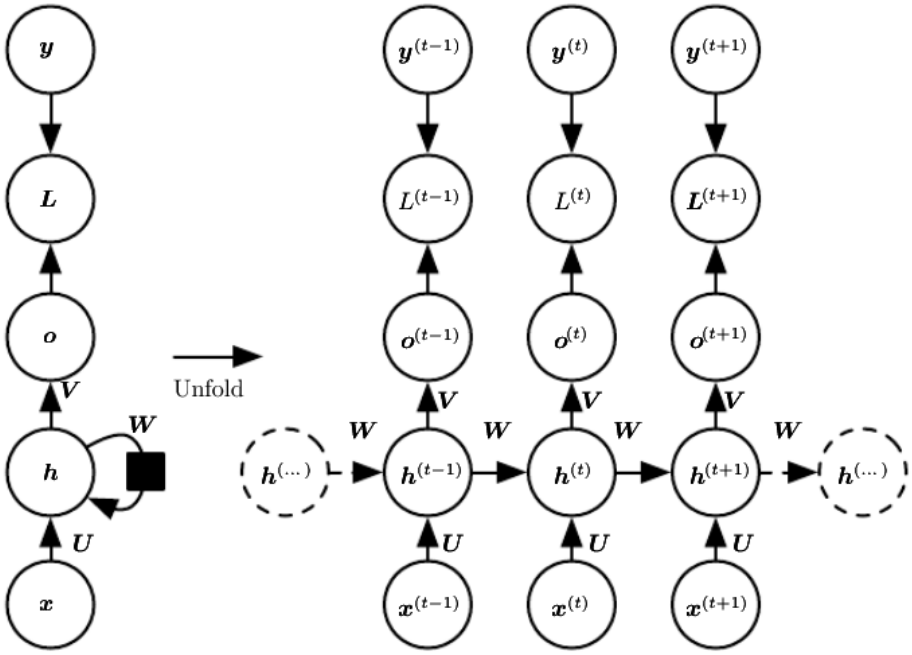

Figure 2.4: RNN training representation [3]

To cover the previously stated problem several solutions were proposed. One of those solutions was a variant of the RNN, known as *Long Short Term Memory* (LSTM) networks.

2.7 LSTM

According to [1], LSTM-RNN models present some of the lowest errors of all load forecasting techniques currently being used.

LSTMs make long term memory more reliable, with the introduction of gates. These gates dictate the passing of information, by controlling self loops and its weights in which the gradient flows, therefore making possible the dynamic change of the time scale of integration [3]. A representation of an isolated cell of this type can be seen in Figure 2.5.

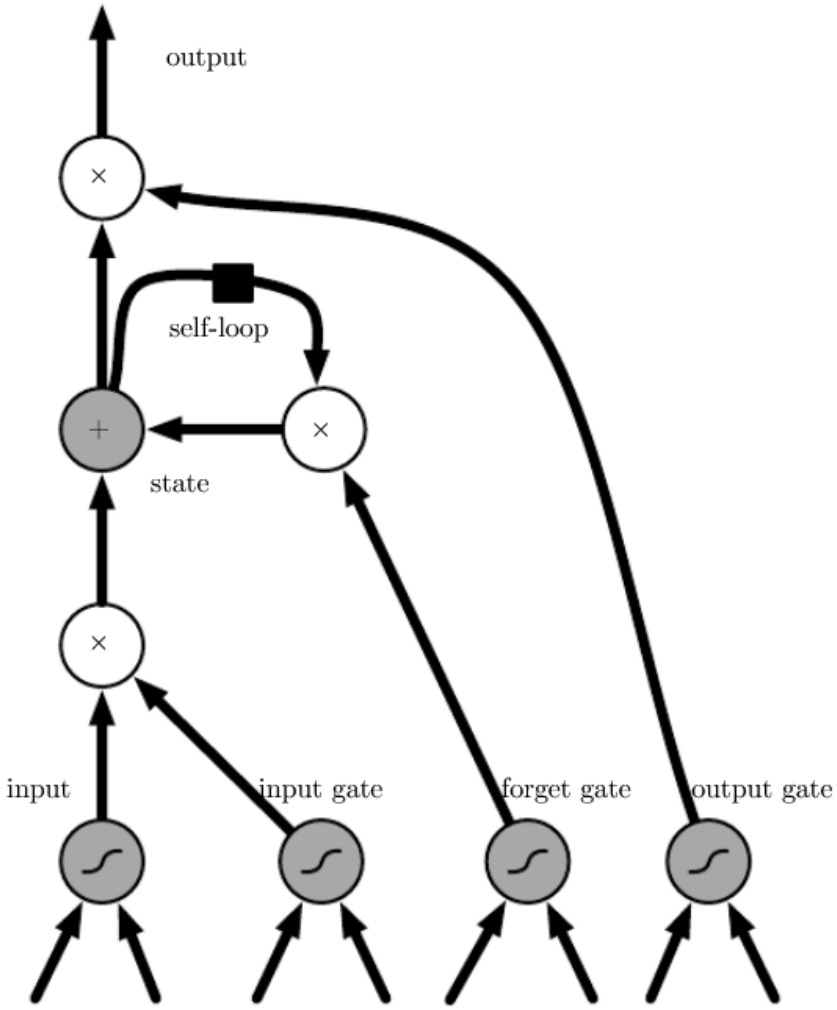

Figure 2.5: Isolated cell representation [3]

In Figure 2.5 it can also be observed the so-called gates [3]:

- *Forget Gate*, as the name hints, is responsible for deciding what parts of information should be "forgotten". It accomplishes that by using a sigmoid (σ) which normalizes weights of the self-loops to values between 0 and 1, meaning 0 the deletion and 1 the keeping of the information as it stands. This proceeding can be expressed as stated in 2.10, where b is bias, U is input weights, x_t is the current input vector, W is recurrent weights and h_t is the current hidden layer vector.

$$f_t = \sigma(b_f + U_f x_t + W_f h_{t-1}) \tag{2.10}$$

- *Input Gate* is responsible for the decision of adding new data back into the cell, therefore,

like the Forget Gate, it also uses a sigmoid:

$$g_t = \sigma(b_g + U_g x_t + W_g h_{t-1}) \tag{2.11}$$

- *Output Gate* is responsible for deciding what information is important and to present it as an output:

$$q_t = \sigma(b_o + U_o x_t + W_o h_{t-1}) \tag{2.12}$$

Finally, it is possible to represent the output of one cell, h_t, which is the result of the computing of the cell state (2.13) in an activation function (in this case, the hyperbolic tangent) multiplied by the output gate function (2.12):

$$s_t = f_t s_{t-1} + g_t \sigma(b_s + U_s x_t + W_s h_{t-1}) \tag{2.13}$$

$$h_t = tanh(s_t) q_t \tag{2.14}$$

For some types of models, mainly sequential ones, it is common to use an *encoder-decoder* architecture (see chapter 2.8) in which inputs are passed to an RNN, functioning as the encoder, and then a context vector, c, is created from the last hidden state of the encoder which in turn is passed as an input to the decoder (Figure 2.6). In cases, though, where the input sets are considerably large, some problems arise. Since the encoder has to compress all the required information into just a vector some information may be lost. Additionally, the decoder needs to "decode" all the information received, which is a complex function. With this necessity, *Attention Mechanisms* were created [3].

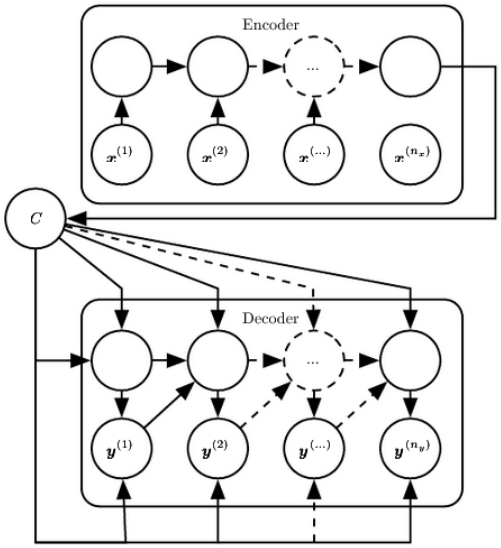

Figure 2.6: Encoder-decoder representation [3]

According to [9], LSTM-RNN based-models outperform the majority of other load forecasting tools, such as ARIMA, support vector regression (SVR) and conventional ANN or feed-forward neural networks (FNN).

2.8 Attention Mechanisms

To put it simply, attention mechanisms (AM) are techniques utilized within the scope of Deep Learning, that look to imitate human attention. These mechanisms take a set of inputs and determine which set of those inputs should be given more importance, reducing the less important ones, purposefully concentrating on smaller but more relevant terms, giving them more "attention", hence the term. The difference between the encoder-decoder architecture explained beforehand and this new architecture with the implementation of an attention mechanisms is that without the AM the decoder has to make predictions based only on the compressed information on vector c, however, with "attention" the mechanisms look to all information inside the hidden states of the encoder, at each time step, and then attributes values according to the importance of each one, and only after that the decoder processes that information (Figure 2.7) [7] [12].

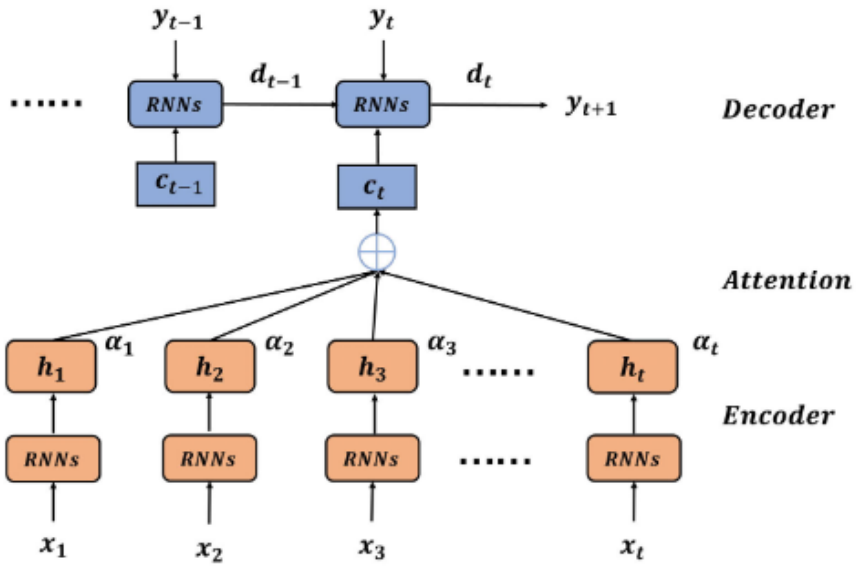

Figure 2.7: Encoder-decoder with Attention Mechanism [9]

The application of attention mechanisms can be implemented according to the next formulas [7] [9]:

$$Att_t = \sum_{j=1}^{T} \alpha_{tj} h_j \tag{2.15}$$

$$\alpha_{tj} = \frac{exp(e_{tj})}{\sum_{j'=1}^{T} exp(e_{tj'})} \tag{2.16}$$

$$e_{tj} = softmax(s_t, h_j) \tag{2.17}$$

Generally, attention mechanisms can be formulated as the taking of a vector of T arguments, and the hidden state of the decoder, s, and then the returning of a vector output, the attention. This attention is defined as a weighted summation of the T arguments, where weights are selected for each individual element of the vector T, h_i (hidden encoder unit), according to an alignment function, or compatibility function, in this case $softmax$, which indicates the importance of each element h_i taking into consideration s [7] [3]. With that being said, this representation can be seen in picture form in Figure 2.8.

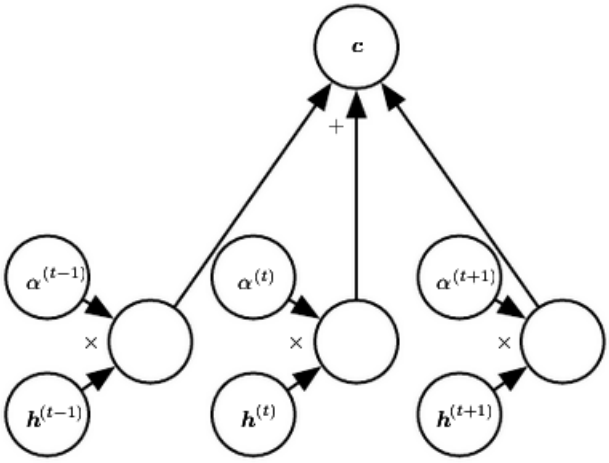

Figure 2.8: Attention Mechanism in detail [3]

As concluded by [9], short term load forecasting problem, the inclusion of attention mechanisms in RNNs proves to enhance greatly the performance compared to "state-of-the-art baselines in both accuracy and efficiency", being instrumental in the accomplishment of said performances.

In [13], which was one of the first works "to show the power of neural network attention mechanisms in the domain of time series forecasts", the purpose was to predict the demand of certain commodities over several stores in the United States of America. With the usage of attention mechanisms it was possible the automatic incorporation of data from several external sources that could be potentially important, like weather, holidays, and many kinds of events. In this case, a baseline forecast was done and it was adjusted based on observations made related to the aforementioned external data, having an interpretative additive consequence on the baseline, achieved with the attention mechanisms. It was concluded that the utilized soft attention mechanism that learns attention weights based on a classifier on top of the hidden representations achieved better results than just the attention weights being based only on the input representation. This model averaged almost 24% better performance with the incorporation of this external data.

In [14], an attention mechanism employed into the encoder-decoder architecture of neural networks was used to model the time series forecast of the behaviour of temperature of an electric arc furnace side-wall. With the historical behaviour of 49 variables the model was able to determine which of the variables were important and proved to be a good tool to apply in any multivariable problem to predict the behaviour of a given variable.

2.8.1 Multi-Head Attention

In 2017 [15] introduced a new architecture for translation tasks, called $Transformer$, which solely works based on attention mechanisms, specifically a new concept named *Multi-head attention* which, essentially, rather than having just one layer attention function running, has various layers of attention running at the same time. The purpose of this is to have each "head" focusing on information from various representational spaces at different locations, enabling the model to do it simultaneously.

$$MultiHead(Att_t) = Concat(head_1, ..., headk)W^O \tag{2.18}$$

Where W^O projects the concatenation of the k heads to the output space [16].

2.9 Interpretability

The primary challenge that prevails in most contemporary architectures is their 'black-box' nature. The complexity of these models, due to nonlinear interactions between numerous parameters, makes it challenging to understand how these models appraise their predictions. This inherent opaqueness limits the trust users can place in these models' outputs and impedes effective debugging. Moreover, as the influence of the inputs on the output is not clear, the user cannot be sure if the model would completely diverge for some specific inputs combination.

A promising direction is the adoption of inherently interpretable modeling approaches that build feature selection directly into the architecture. However, since those kinds of architectures are normally used in sequence-based problems, like translation or speech, different types of input features are not considered in those problems, like they are in time series forecasting ones. That means that the attention on those architectures is implemented sequence-wise, giving information about the attention given to a certain timestep of a sequence, in which every sample/timestep will have an attention value to all the other values of the sequence (including itself, hence the name $self-attention$), instead of feature-wise, which means that it cannot distinguish the importance given to different features at each timestep [17].

Attention mechanisms, while extensively used in language translation and other applications like image classification and tabular learning, have been adapted recently for time series forecasting. Attention-based methods have been leveraged to enhance the selection of relevant timesteps

from historical data. Research such [18] has employed direct methods based on sequence-to-sequence models, like LSTM encoders, to summarize past inputs. The authors introduced a multi-modal attention mechanism with LSTM encoders to construct context vectors for a bi-directional LSTM decoder, outperforming traditional LSTM-based iterative methods. However, the same challenge of interpretability remains unresolved in such direct methods. Moreover, models such as RETAIN [19] and "Attend and Diagnose" [20] have introduced attention-based mechanisms to offer instance-specific interpretations. They offer the advantage of identifying salient portions of input for each instance using the magnitude of attention weights. However, these approaches still fall short of adequately considering static covariates due to blending variables at each input, and they also fail to provide insights into global temporal dynamics. Another example is the Interpretable Multi-Variable LSTM, which partitions the hidden state such that each variable contributes uniquely to its own memory segment and weights these segments to determine variable contributions. This approach represents a significant step towards enhancing model transparency in time series forecasting, although much progress is still required [17].

In [17] the Temporal Fusion Transformer is introduced, which aims to provide better performance and interpretability for multi-horizon time series forecasting by, for example, establishing specific encoders for the problem of the static covariates, has a variable selection network to discard irrelevant inputs and also temporal attention layers for long-term (temporal) relationships, and many more particular nuances. Despite being so sophisticated, it may be overly complex for certain tasks that do not require multivariate and multi-horizon forecasting.

For the specific case of this work, it might be preferable not to use such intricate, and experimental, architectures and maybe stick to a simpler model that could facilitate the execution of the model and thereby make possible a wider range of analysis specific to the temperature and consumption. For example, by not using the sequence-based models and treating the problem as a tabular learning one, it would be way more straightforward the extraction of the attention scores. One example of this kind of model is [21] in which global and local, or instance-based, attention scores are obtained. The global attention can determine the global importance/significance of all features, while the local attention (calculated from the attention weight matrix) makes possible the identification of specific conditions in the data in which the inputs acquire greater relevance.

Post-hoc explanation methods like SHAP (SHapley Additive exPlanations) are also heavily used for determining feature importance in several machine learning and deep learning regression and classification tasks [22]. SHAP makes use of shapley values([23]) and game theory to calculate the contribution of each feature to the final predicted value. The shapley values assist on how to fairly do that distribution. Specifically for deep learning models, SHAP takes advantage of the DeepLIFT method [24] and the shapley values by taking the discrepancy between the predicted value and the average prediction and backpropagating this value across the layers of the neural network [25] [26] [27].

Chapter 3

Methodology

In the forthcoming chapter, an in-depth exploration of the foundation of this research unfolds, detailing the procedures, techniques, and models utilized throughout the study. The focus primarily lies on the chosen data and the developed models. Insight is provided into the steps taken to process and prepare the data, revealing the precautions undertaken to ensure its validity and relevance. Furthermore, the analytical models used in the study are mentioned, especially describing the model from which the attention scores were taken, including the rationale behind that choice. The underlying principle of the model, its implementation, and its contribution to the study's results and findings are also clarified. Ultimately, this chapter serves as a comprehensive guide to the scientific methods and strategies that form the core of this research, emphasizing its robustness and validity.

3.1 Utilized Models

This prediction involves the use of several features for the purpose of demand forecasting, such as chronological and meteorological variables, for example. Based on the analysis of the previous chapter, it is mentioned that most of the existing state-of-the-art technologies, like transformers, utilize attention for sequence-based problems, which could be the case here. However, this type of attention only allows for an inspection of the attention weights across the time steps of said sequences and therefore are not completely suitable for the analysis to be carried out, which is the study of the global impact of the different features in the prediction and, in particular, the effect of temperature on the consumption. Other models, like the Temporal Fusion Transformer, use other types of more intricate technologies relating to variable selection and temporal attention but are extremely complex, more geared towards multi-horizon forecasting, and do not allow more direct, easier, and repeatable experiments on the attention instances. For that reason, another approach was needed and another type of attention had to be implemented, fundamentally just one simple mechanism applied across the features sample-by-sample (instead of one applied across a sequence of time steps). It is worth mentioning that the use of the previously mentioned sequence-based models would most likely improve the quality of the predictions, however, a trade-off has

to be done in order to get a simpler model that is able to provide a more direct look into the actual feature importance and that further allows to make a more in-depth analysis for this case study, the impact of temperature. The goal is to extract from the model an importance score for each feature. Generally, these scores range from 0, which means no importance at all, to 1, which means heavily important, and the sum of all scores has to be 1. After that, it is viable to delete or change variables that are deemed unimportant and keep the significant ones unchanged. By doing this, the results of the predictions should improve, the model should work in a more efficient way, with less computational complexity, and the interpretability should be easier to do. With that being said, the main model utilized in this work for the acquisition of attention scores is described below.

Firstly, The inputs x are fed into the Multi-head Attention part of the network. For each attention head, called k, the inputs will be passed through a fully connected layer with input and output sample sizes equal to the input size (that is the number of features, with the purpose of trying to keep relations between features) suffering a linear transformation, therefore each of those heads will have its set of weights and biases. The resulting matrices of the previous operation will go through a $softmax$ function and the results will be multiplied, using the $Hadamard$ product, $\otimes$, by the input, x. After that, all the matrices, representing each head, will be added together, using the $Hadamard$ summation, $\oplus$, and then the cumulative sum is divided by the number of heads, getting an average of the values of all heads. The idea behind this is that each head has its own weights and biases, which are learned independently, and this results in a form of ensemble learning where each head should learn a different feature representation, and all are aggregated to give a final output. This means that each head can learn to pay attention to different parts of the input, giving the model the ability to focus on multiple aspects of the data at the same time and also learn and combine different types of relationships. This attention part can be represented by:

$$Att_L(X) = \frac{1}{k}\oplus_k [X \otimes softmax(W^k_{att_h} X + b^k_{att_h})] \tag{3.1}$$

In which:

- X, represents the input space
- $W^k_{att_h}$, represents the weight matrix of the attention layer for attention head k
- $b^k_{att_h}$, represents the bias vector of the attention layer for attention head k

After the attention component, the model is then passed through a fully connected layer, with a particular hidden size, then through a dropout layer, for regularization, then through an activation function, and finally by another dense layer which will output the final vector containing the predictions. The following diagram, Figure 3.1, aims to represent the general structure of the model.

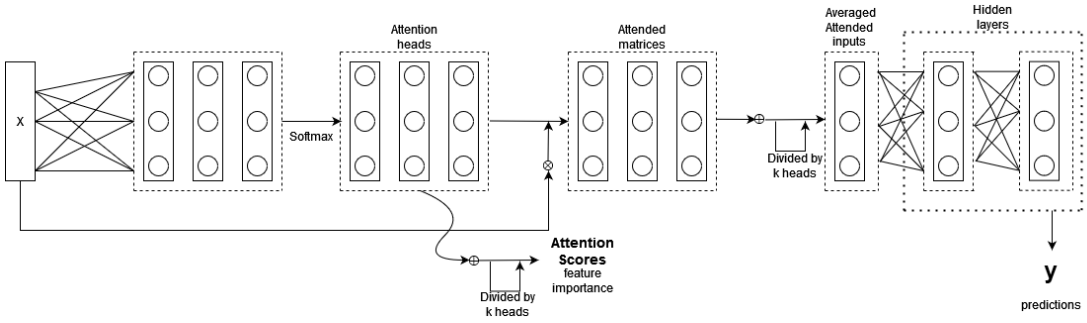

Figure 3.1: Representation of the structure of the model

With that said, the output of the model is calculated by:

$$Fc_2 = W_{Fc_2} \cdot (a(W_{Fc_1} \cdot Att_L(X) + b_{Fc_1})) + b_{Fc_2} \tag{3.2}$$

Where:

- Att_L, represents the attention layer
- W_{Fc_2}, represents the weight matrix of the last hidden layer
- W_{Fc_1}, represents the weight matrix of the second to last fully connected layer
- b_{Fc_2}, represents the bias vector of the last hidden layer
- b_{Fc_1}, represents the bias vector of the second to last fully connected layer
- a, represents the activation function

After considering several functions for the activation function, such as *Sigmoid*, *ReLU*, *SELU*, and *swish*, the choice eventually fell on the *Sigmoid* function, because it seemed slightly better in terms of the convergence of the models for this specific case [28].

In regard to the actual calculation of the feature importance, through the attention scores, two methods were utilized:

The first, and most obvious one, is simply to, by looking at the mathematical expression 3.1, factor out the X, which can be done since all the mathematical operations implemented are linear transformations, and take out the remaining component, the attention matrix. This is represented in 3.3. This method will return the attention matrix regarding each input for all training examples, and the analysis of specific training cases (e.g. cases where the temperature is very high) allows the highlighting of the attention of certain conditions.

$$Att = \frac{1}{k} \oplus_k [softmax(W^k_{att_h} X + b^k_{att_h})] \tag{3.3}$$

The general implementation of this model was loosely inspired on [21], however, the following method of calculating global attention was actually taken in its entirety from the previously cited paper. This method takes, for each head, the diagonal of the respective weight matrix, applies a $softmax$ function to it, which turns it into an activated diagonal, stacks all the activated diagonals into a matrix, and then simply calculates the (global) average along the columns, and, by doing so, returns the global mean attention scores of each feature. This can be represented by:

$$M_{GA} = \frac{1}{k} \oplus_k [softmax(diag(W^k_{att_h}))] \tag{3.4}$$

In order to assess if the actual results of the predictions of the previous model are acceptable, other different models were tested on the same data and comparisons will be made, in the next chapter, between those models. This is particularly interesting to understand how the model explained in this section actually ranks among other models, specially sequence-based ones. In order to make it easier to refer to the different models they have been labeled as follows:

- **Model 1**, the model explained in this section
- **Model 2**, similar to Model 1, but without the attention mechanism
- **Model 3**, a vanilla LSTM model with just two layers
- **Model 4**, a single layer LSTM layer followed by a traditional multi-head attention layer (applied to the sequences)
- **Model 5**, model utilizing XGBoost (eXtreme Gradient Boosting) Reggressor, which is based on gradient boosted trees algorithms [29] [30]

3.1.1 Hyperparameter tuning

For the purpose of keeping this work focused and concise, this subsection will only explain the chosen hyperparameters of Model 1, and also because this is the main model utilized in the analysis of the results. For the other models, the procedure is very similar.

Firstly, since this work was all done on a personal laptop, and not with the external help of more powerful hardware, the hyperparameters were chosen on a trade-off relationship between performance and computational cost (and therefore execution time).

In terms of hyperparameters directly related to the model itself, for the number of hidden neurons, it was observed that, as expected, the bigger the number of neurons, the higher the execution time of a single epoch. The improvement in actual prediction results was not substantial in order to require a high number of hidden neurons, therefore, the chosen number was the standard amount of 32. For the number of attention heads, on the other hand, a small increase would make a huge difference in execution time and the prediction results would improve quite a bit. With that being said, after several experiments, the chosen number of attention heads was 8.

Regarding the hyperparameters related to the training, the obvious choice for the error criterion of the loss function is the *MSE*, which measures the mean squared error, for each input, regarding the target. Those will be the values later verified in the plots of the training and validation errors. *Adam* was the chosen optimization algorithm for gradient descent in training, since it is widely used in the context of regression problems. Due to the large size of the utilized dataset, there was a need to divide the dataset into batches, in order to get a lower execution time of the model. The size of the batch is quite important in the convergence of the model, decreasing or increasing this value will impact the execution time of the training and validation processes. For this model, it was used a batch size of 64, a standard size for deep learning models. Something that also has quite a big impact on the convergence of the models is the learning rate, which is usually adjusted conjointly with the batch size. In order to obtain a first estimate of this value, a function of $Pytorch - Lightning$ was used [31]. The estimate returned by this algorithm was around 0.01, as seen in Figure 3.2.

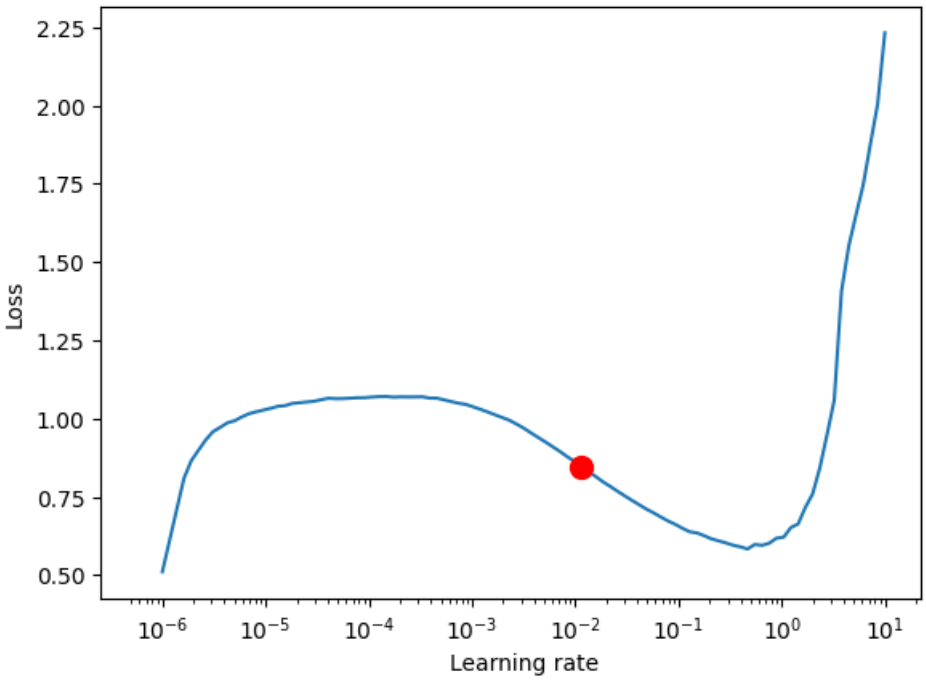

Figure 3.2: Representation of the results of the algorithm that finds good initial estimations for the learning rate

It's worth noting that during the model training process, an early stopping checkpoint was introduced to prevent overfitting and excessive training time. This checkpoint continually saves the best model - defined as the one with the lowest validation error - throughout the training. Training stops if, after a designated number of epochs, no improved model is found. This strategy assumes that if, for instance, no superior model emerges after 50 epochs, it's likely that none will be found in further training. This can be seen in Figure 4.1.

3.2 Evaluation metrics

To assess the performance of each model in the test dataset three types of errors were considered, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) [1]. Each of those errors is calculated by:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Actual_i - Predicted_i| \tag{3.5}$$

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{(Predicted_i - Actual_i)^2}{n}} \tag{3.6}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}|\frac{Actual_i - Predicted_i}{Actual_i}| \tag{3.7}$$

3.3 Brief overview of the data

The data used in this work relate to energy consumption in Italy, obtained over a period of about 5 years. These data are organized on an hourly basis, meaning each consumption value corresponds to one hour. It is worth mentioning that the data contains very few occasional gaps, which should not pose a problem at all due to the large size of the dataset, and it was made sure that all days had full 24 hours, so the missing data are just a small amount of days across the set. Figure 3.3 presents the evolution of said consumption throughout the whole dataset. Although not represented in the plots, all the consumption values throughout this work are in MW (Megawatt).

A decision was made to perform the forecasts 48 hours into the future, therefore for the following two days, which lands on the "short-term forecasting" time frame described in chapter 2.

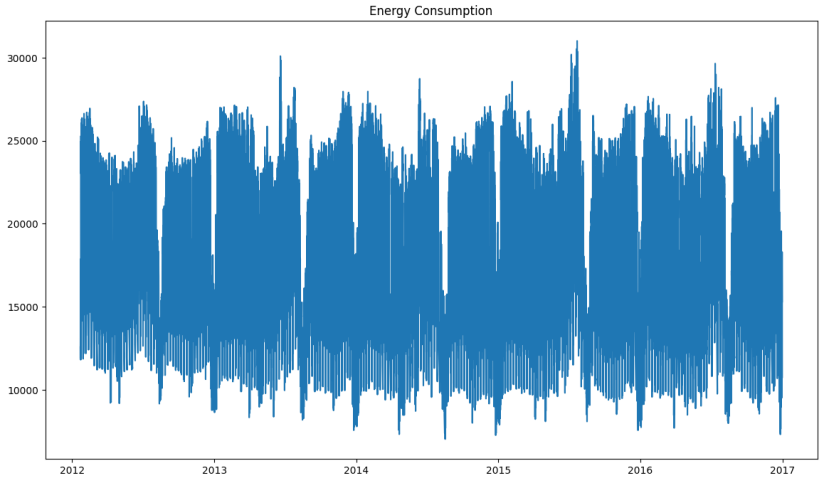

Figure 3.3: Energy consumption across the entire dataset

Due to the nature of this type of load forecasting, and due to the characteristics of model 1 described in chapter 3.1, several kinds of variables are needed. First, chronological variables, such as hour, day of the week, and month. The idea behind the use of these types of variables is to try to capture the different patterns, or seasonality, of the demand over time, for example, the fact that on certain days of the week, such as weekends, the consumption is usually lower, or that at certain hours of the day, like dawn there is little consumption, as opposed to other sections of the day. Chronological values can help identify that. Moreover, to try to capture odd periods of the year, for example, national holidays, which can introduce irregularities in the consumption patterns (if a holiday occurs on a weekday, it may disturb the results of the predictions, since on normal weekdays the demand is usually different than on holidays), binary variables were introduced, named from A to E, representing national holidays and other types of unusual days, with the intention of trying to help the model identify those cases. As previously mentioned, one of the main objectives of this work is to study the influence of temperature on consumption so, of course, temperature was introduced as a variable, making it the only meteorological variable used. Apart from those, as was mentioned in section 3.1, the traits of model 1 do not allow the use of sequences, like in the traditional models used for natural language processing, for example. Because of this, it is of great importance the use of lagged inputs, which are essentially features with built-in delay. Since this is a load forecasting problem, several lagged features of consumption were added, such as consumption from the previous four days, the previous week, the previous two weeks, and the previous three weeks.

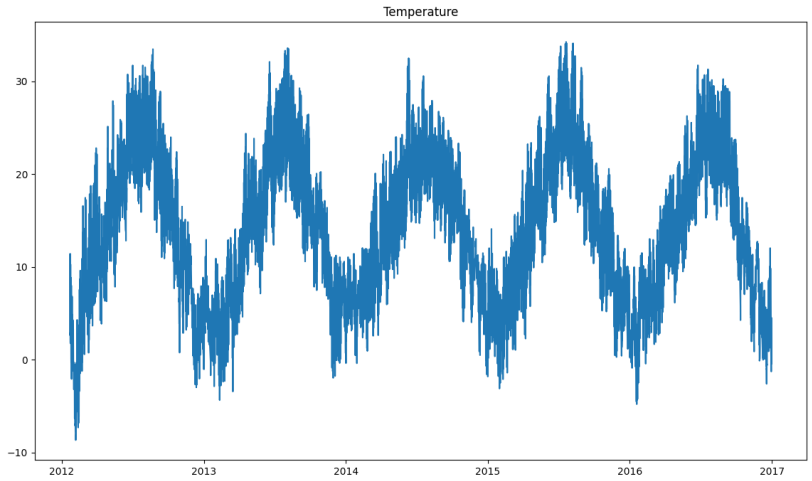

Figure 3.4: Evolution of the temperature across the entire dataset

Figure 3.4 shows the evolution of the temperature along the dataset, starting from 2012 up until 2017 as seen in the x-axis, with a clear definition of different periods of heat, representing the different seasons. Although not represented in the plots, all the temperature values throughout this work are in degrees Celsius.

As mentioned, the temperature will be especially relevant in this work, so it is of great interest to assure that its values follow a normal progression, as is the case.

3.4 Feature creation

With the notions explained in the previous section in mind, the input features of the model were established. Starting off with a particular case, the chronological variables do need special concern when considered in the model. These variables are only sequences of numbers repeated several times across the dataset, for instance, the hour of the day is the numbers 0 through 23 replicated multiple times along the data, the same for the days of the week with numbers 1 through 7 (where 1 represents Sunday and 7 represents Saturday) and the same for the month of the year with numbers 1 through 12 (where 1 represents January and 12 represents December). However, if those variables were fed into the model as they are, some considerable oversights would be overlooked. For example, when feeding the model with the hours of the day, the 0-hour value and the 23-hour value are far from each other, so that could provide wrong information to the model, in fact saying that those values have a -22 hour difference, instead of just 1-hour difference, as known. To resolve that issue, cyclical variables were created. As the name suggests, this type of variable is specific for features that repeat cyclically. With that being said, the following expressions were applied to all the chronological variables used, day of the week, hour of the day, and month of the year, to transform them into cyclical variables:

$$x = sin(\frac{f * 2\pi}{max(f)}) \tag{3.8}$$

$$y = cos(\frac{f * 2\pi}{max(f)}) \tag{3.9}$$

Which, for instance, for the hour of the day translates into:

$$hour_{sin} = sin(\frac{f * 2\pi}{24}) \tag{3.10}$$

$$hour_{cos} = cos(\frac{f * 2\pi}{24}) \tag{3.11}$$

The evolution of both types of variables and the differences between them can be seen in Figure 3.5

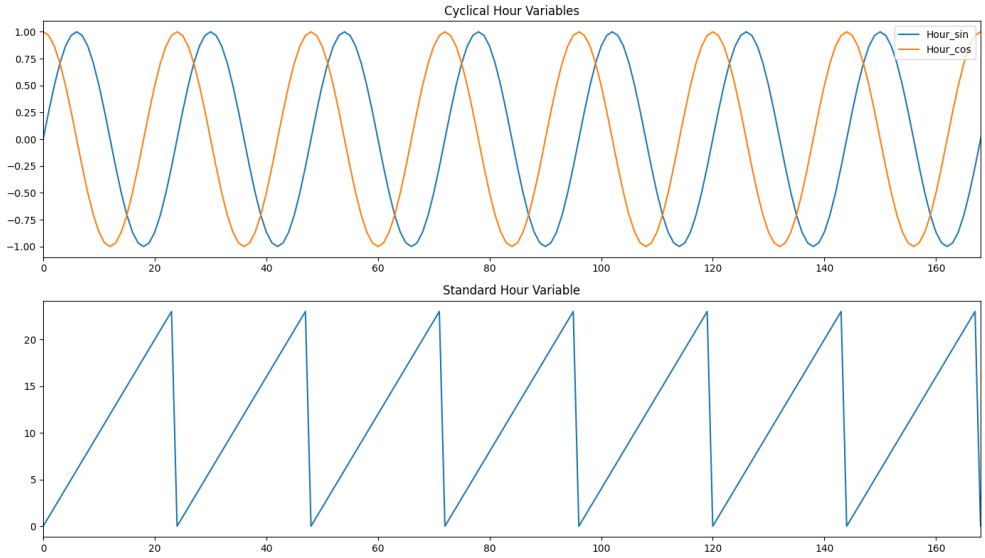

Figure 3.5: Difference between cyclical and non-cyclical variables

It was also necessary to do a general normalization of the features, to ensure they are all on the same scale, especially because of the way *Pytorch* works, otherwise the network would produce very poor results. Two normalization methods were tried, min-max scaling and standardization (also called z-score standardization), with the standardization method presenting better results. This method, essentially, makes sure the mean of the feature is zero and scales it to unit standard deviation. For this, the mean, μ, is subtracted from each element, x, and the results get divided by the standard deviation, σ, resulting in z represented in 3.12 [32].

$$z = \frac{x - \mu}{\sigma} \tag{3.12}$$

To recap, the variables used in the model were: the sine component of cyclical variables relating to the hour/day of the week/month, the cosine component of cyclical variables relating to the hour/day of the week/month, "A,B,C,D,E" binary variables, temperature and the consumptions with lags of 48/96/168/336/504 hours.

The last thing to mention is the division of the dataset into other datasets with different purposes. From the original dataset were created three other datasets: the training dataset, the validation dataset, and the test dataset. The training dataset was reserved for the training of the model, the validation dataset served as a tool to better tune the hyperparameters of the models, and the test dataset was where the actual predictions of the models were done. Regarding the split of the datasets, 70% was reserved for training, leaving the remaining 30% for the other two datasets, which were divided evenly, so 15% of the entire dataset for each of the validation and test datasets (Figure 3.6). This proportion is common for ML problems and ensures each dataset does what it has to do.

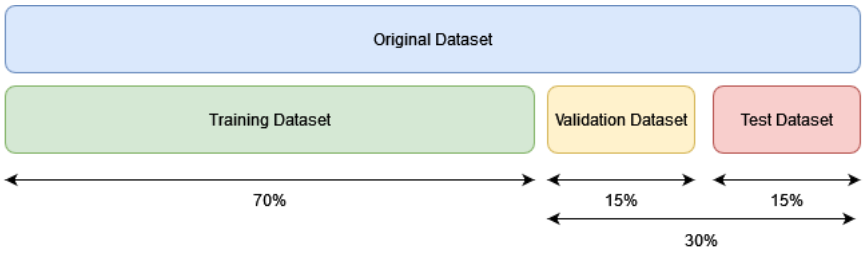

Figure 3.6: Split of the training, validation and test datasets

In Figure 3.7 there is a representation of the evolution of the consumption across the three datasets. Due to the split previously explained, the training dataset contains data for the first three years, approximately, the validation dataset contains the data corresponding to around the following eight months and the test dataset contains roughly the next eight months. This is, probably, not ideal, since the last two datasets do not take in a full year but, as seen in the Figure, they do incorporate substantial information, including vacation periods of lower consumption, namely holidays (August) and the transition from one year to another (end of December and beginning of January).

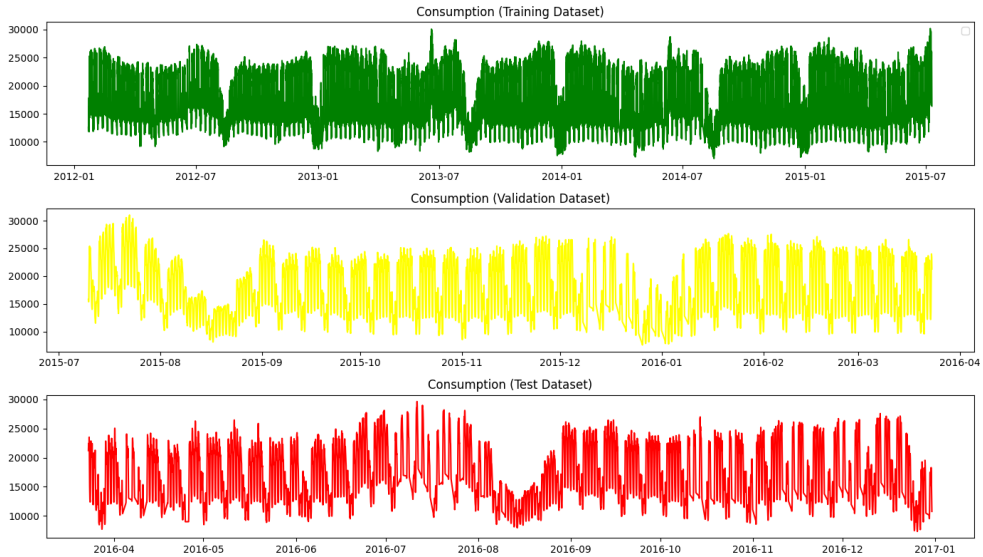

Figure 3.7: Evolution of the consumption across the three datasets

3.5 Overview of main studies

The main studies to be carried out in the next section are:

- Set a load forecasting model with attention mechanisms;

- Set a load forecasting model without attention mechanisms, to confirm if the attention mechanisms deteriorate the performance. Compare them with other architectures;
- Compute the global attention, to estimate the significance of the inputs. Compare it with other methods of calculating global importance;
- Compute the attention weight matrix, aiming to find the periods in which the attention is higher and, that way, inferring the effect of temperature on the load.

Chapter 4

Results

In this chapter are presented the main results obtained from the techniques listed in the previous chapter, regarding the predictions, and the analysis of the attention weights to interpret and understand the contribution and significance of various features, with a particular emphasis on temperature.

This section is divided into two main parts. In the first part, an analysis and comparison on the performances of all models, mainly focusing on their prediction errors, namely the RMSE and the MAPE, will be made.

In the second part, the first analysis that will be made is a global examination of the impact of all features in the output, i.e., the predictions, of the model. The main focus will be, obviously, to understand the importance that is attributed by the attention weights to each feature. Then, as alternatives, some more popular techniques, such as SHAP, will be used to determine the importance of all features and to understand if, in any way, it corresponds to the ones attributed by the attention mechanisms. After that, a thorough investigation of the impact of a specific feature, temperature, on energy consumption will be done. The aim is to explore a potential correlation between the attention weights assigned to the temperature feature and the actual values of energy consumption. Ideally, this should provide insights into how closely the model associates temperature with energy consumption. Furthermore, the plan is also to visualize the changes in attention weights for the temperature feature over time and its relationship with the actual temperatures and if they align at all. This time frame could be hourly, daily, or even monthly. This visualization will provide a better understanding on how the model's importance on temperature fluctuates at different stages of the time series. For example, it could determine if the model pays more attention to temperature during certain hours, months, or under specific conditions such as high or low temperatures.

4.1 Comparison of the predictions of all models

The obtained results with the proposed models can be seen in Table 4.1. The first major conclusion that can be inferred is that the models including attention present better results in terms

of prediction of the test dataset because they have the lowest overall errors.

Table 4.1: Overall results of the models

Model	MAE	RMSE	MAPE
1	834.43	1173.3	5.113%
2	896.20	1235.3	5.474%
3	822.01	1163.3	5.080%
4	803,83	1148.0	5.009%
5	839.35	1189.6	5.121%

In Figure 4.1 it is possible to observe the validation and training errors (namely the MSE) of the model that obtained the best test results. As expected, the validation error ends up higher than the training error, and, after stabilizing, starts to improve a bit.

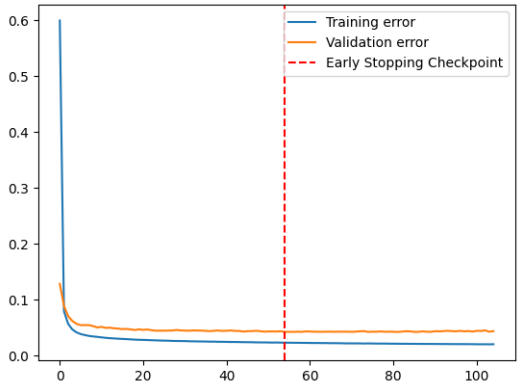

Figure 4.1: Training and validation errors along the epochs

The models with attention produce better results than the models without it, which most likely means that, as hoped, the inclusion of the attention mechanisms does not degrade the performance of the models and, in fact, seems like it is working relatively well on selecting/attending to the most important variables (or time steps). Model 2 is the worse performer, which is to be expected because it only consists of a simple neural network with two layers. Model 1, with the inclusion of attention, performed much better than model 2 and slightly better than the gradient-boosting trees, model 5, but still lost to the LSTM model, model 2. The better performer was model 4, which is an LSTM with the inclusion of the multi-head attention. This means, as talked about in previous sections of this work, the sequence-based models do work better for this type of problem, however, are not as easily interpretable as others.

A comparison between the real values of energy consumption of the test dataset and the obtained values with the prediction of model 1 can be seen in Figure 4.2. It's observable that the model captures relatively well seasonality and a lot of the patterns, although, as expected, misses some valleys and peaks.

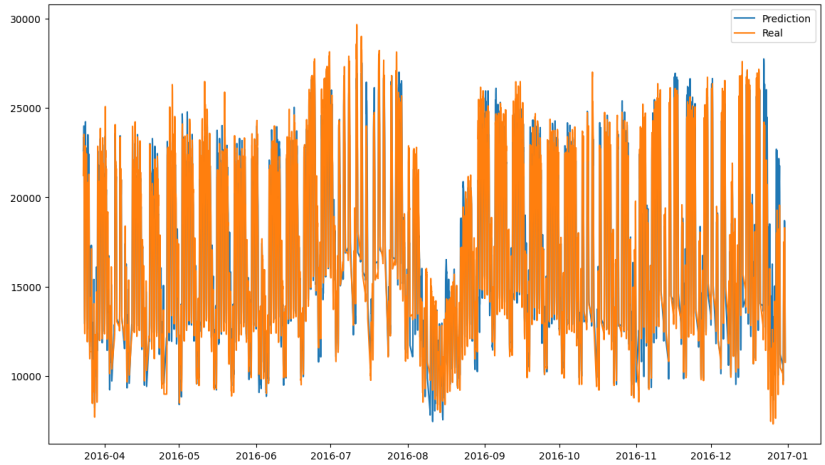

Figure 4.2: Predictions vs real values of consumption

4.2 Comparison of the global importance of all features

Regarding the global attention of all features, the model can return two types of results as described in the previous chapter. Starting with the global mean attention scores (3.4), it shows that the model does indeed attribute the biggest attention value to the temperature feature which is followed by the energy consumption 48 hours before, which makes total sense since the idea is to predict the consumption 48 hours in the future, and by the consumption variables of previous days/weeks. Next are the chronological variables which have similar importance, and then are the special binary variables corresponding to special days of the calendar. This is represented in Figure 4.3.

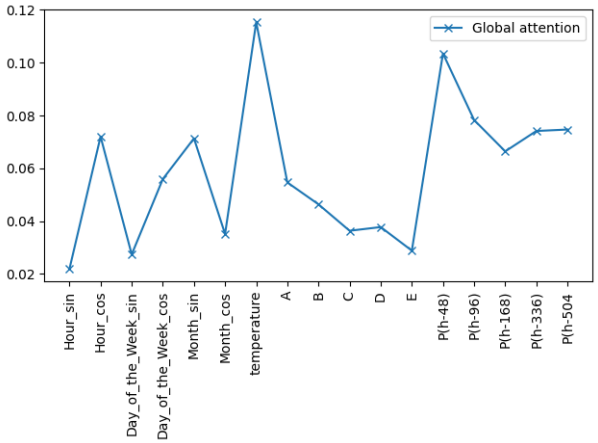

Figure 4.3: Global attention taken from 3.4

Regarding the other type of attention (3.3) the results are somewhat different. As seen in Figure 4.4, this time the most important feature, is considered to be the days of the week, which,

it appears that in this case this feature was considered very important for establishing the weekly patterns of consumption, and contrary to what happened with the previous analysis the other two chronological variables, months and hours, do not have the same importance as this one. However, this might not always be the case for every situation, especially because hours cannot be considered much less important. The second more important variable, however, is still the consumption forty-eight hours earlier, which, as formerly been said, is expected because the prediction is made for two days in the future; so, this is the consumption instance that is closer to the consumption to be forecasted. Moreover, the next variables in order of importance are the consumptions in previous days/weeks which is congruent with the previous attention analysis. The temperature is the next variable in terms of importance, in line with variables representing the more distant consumptions and some of the binary variables, depicted by “A,B,C,D,E".

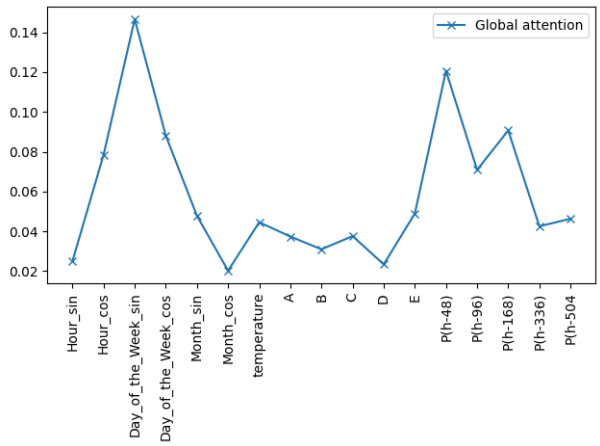

Figure 4.4: Global attention taken from 3.3

Comparing both results, the biggest difference is the importance attributed to the temperature feature, which is specially concerning since this is the specific variable that is meant to be studied in a more detailed way by taking advantage of the attention matrix returned by the model. This can be explained by the fact that the method depicted in Figure 4.3 calculates the attention globally only by looking at the weight matrices of the attention layer, and the method of Figure 4.4 actually takes in data and attributes the attention scores of each feature for every instance depending directly on the data that was fed into it.

Another factor that could be influencing the scores returned by the two methods is the fact that the considered variables are not entirely independent, in fact, it is quite the opposite, some variables are closely related to each other, as seen in the general correlation matrix of Figure 4.5, where 1 means extreme positive correlation and -1 means extreme negative correlation. Since the weights are randomly initialized the training process can attribute a bigger weight for P(h-168) in one test but in another test it can do it for P(h-336) instead, and so the attention could be influenced the same way. Other tests could be done by just considering variables that are close to being independent, selecting them by looking at the matrix, however that set of variables,

although potentially resulting in interesting results of the attention scores, may very well lead to worse performance results. This can mean that the use of attention can introduce a trade-off problem between performance and coherent interpretability attention scores.

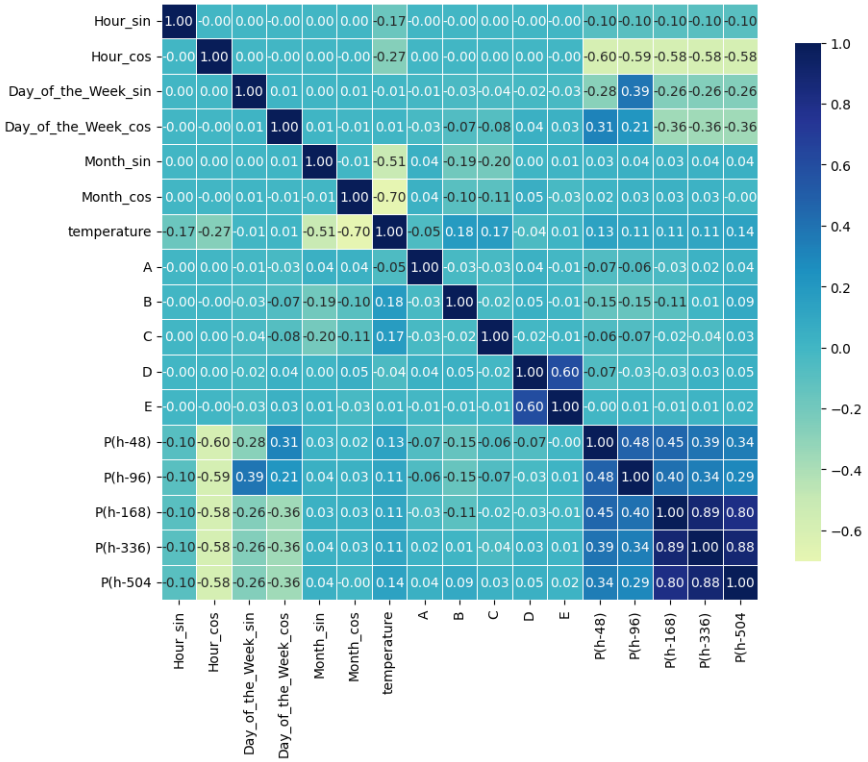

Figure 4.5: Global variable correlation

To try to get a better understanding of the global attention results and a certain validation of the results previously obtained, other variable importance methods were applied to this model, such as SHAP, and since model 5, the gradient boosted trees, also presented interesting results an analysis was also made on the scores attributed by that model by ways of permutation importance. The attained results for feature importance are represented in Figures 4.6 and 4.7.

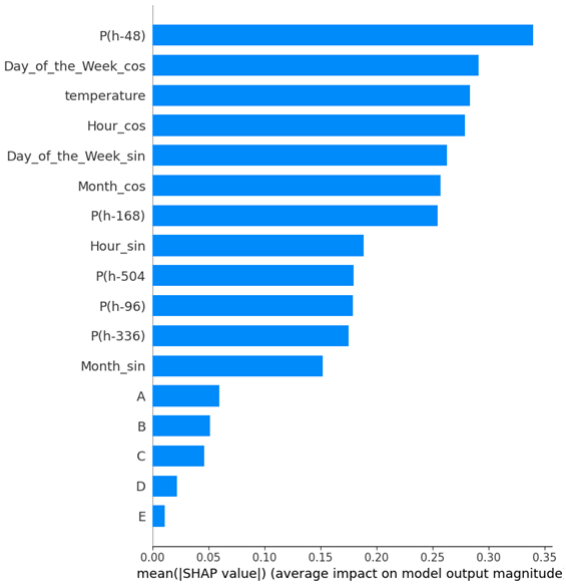

Figure 4.6: Global importances attributed by the shapley values

The Shapley value results indicate that several features play a significant role in the model. In particular, the P(h-48) feature has a high level of importance, followed closely by the 'day of the week' and 'temperature'. After these, a combination of chronological features and lagged consumptions are influential. The features with the least impact according to the Shapley values are the binary variables.

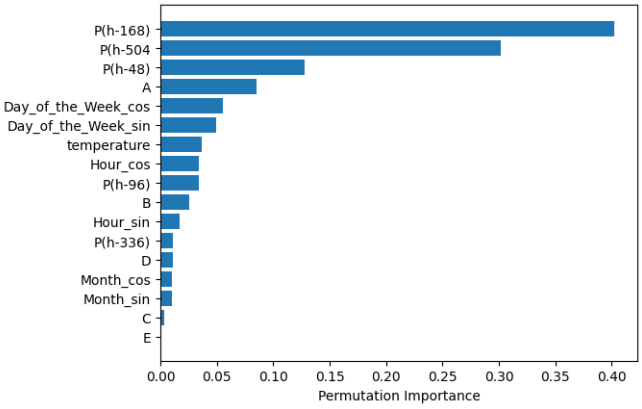

Figure 4.7: Global importances attributed by the permutation algorithm

Regarding the results attained from the permutation algorithm, P(h-48) was no longer the most important consumption variable, that title belonging now to P(h-168) and P(h-504), being the first one the most important overall. Surprisingly, one of the binary variables was the fourth most important variable, followed by the days of the week and only after those came the temperature, which was followed by the remaining consumption, chronological and binary values. It is worth

mentioning that the permutation importance algorithm works by selecting values for each feature based on the performance drop if one feature was shuffled, meaning the worse the performance degradation the more important that variable would be for the prediction of the model [33] [30].

Across the four results, the variable regarding consumption in the previous forty-eight hours is one of the most important, for the previously explained reasons. Hour and, specially, day of the week are also very important and the most important among the chronological variables. All the other time-lagged consumptions are moderately-to-very important and, as expected, the special "ABCDE" variables are the least important in the aggregate of the four methods. The most incoherent result across the four methods is the importance regarding temperature, which interestingly enough is exactly the variable that is of most interest for this work, because, although two of the methods attribute very high importance to the temperature, the other two methods consider it to be not so important, which is particularly upsetting because is based on the importance attained from one of those methods that the next more in-depth analysis about the attention weights regarding temperature will be made.

It should be noted that these metrics relate to global values spread out across the entire data, and are all calculated in different ways. In this dissertation, the main interest lies in the analysis of the relative attention given in certain circumstances. For example, it may be that temperature does not have a very relevant effect in general, but there could be specific conditions under which it does. It may, for example, be more important when the temperature has large variations. As mentioned, another point to note is that there are several methods of estimating attention, and even more of estimating importance, using different criteria, which can lead to different results. However, the large difference between results is not reassuring and leaves the impression that this concept of attention has not yet reached its maturity.

4.3 In-depth analysis of temperature using the attention matrix

In analyzing the attention weight matrix, the initial step involves seeking a correlation between the attention weights for temperature and the temperature itself. Figure 4.8 overlays the actual temperature with the attention weights for temperature, derived from the attention weight matrix. It is readily apparent that high real temperatures (in red) correspond to high attention weights, while lower temperatures result in attention weights close to 0.

However, this hourly analysis, where the x-axis represents time in hours, only captures a rudimentary correlation and falls short for a more comprehensive investigation. Therefore, it is helpful to visualize the same graph on a daily basis instead of hourly. This new representation is displayed in Figure 4.9.

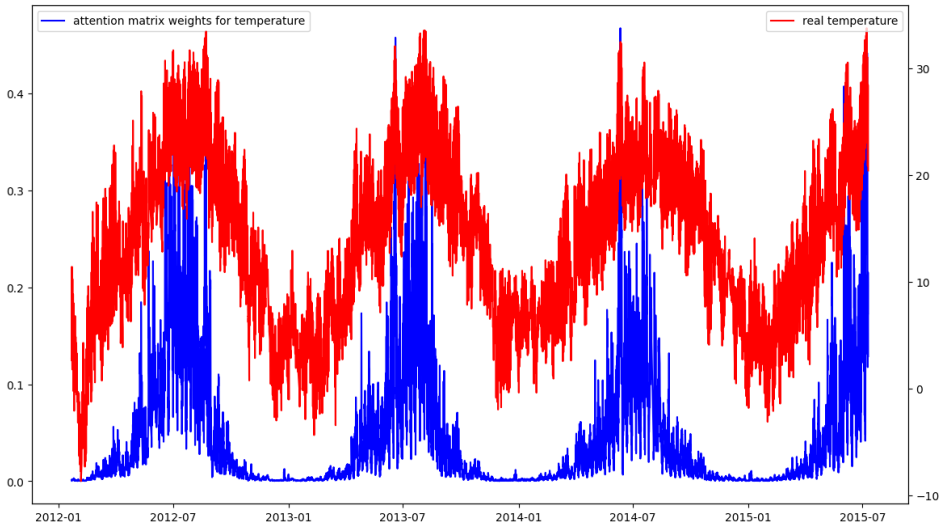

Figure 4.8: Actual temperature vs. Attention weights for temperature (hour by hour)

The graphs below provide a more detailed view of how attention weights change on a day-by-day basis. It continues to support the observation that the attention weights and temperature appear to be correlated – when the temperature increases, the attention weights also increase, and vice versa. From these new plots, the days in which the highest, and lowest, attention values occur are more explicit, for example, around the 19^{th} of June 2013 (Figure 4.10) depicted in the data is when the highest peak of attention takes place, and that peak makes up for more than one-third of the whole attention in that specific instance, meaning that the other sixteen features only get less than two-thirds of the whole attention, most likely meaning that temperature is one of the, if not the, most important feature in that occurrence. It was also expected an increase in attention in the initial part where there is a period of very low temperatures.

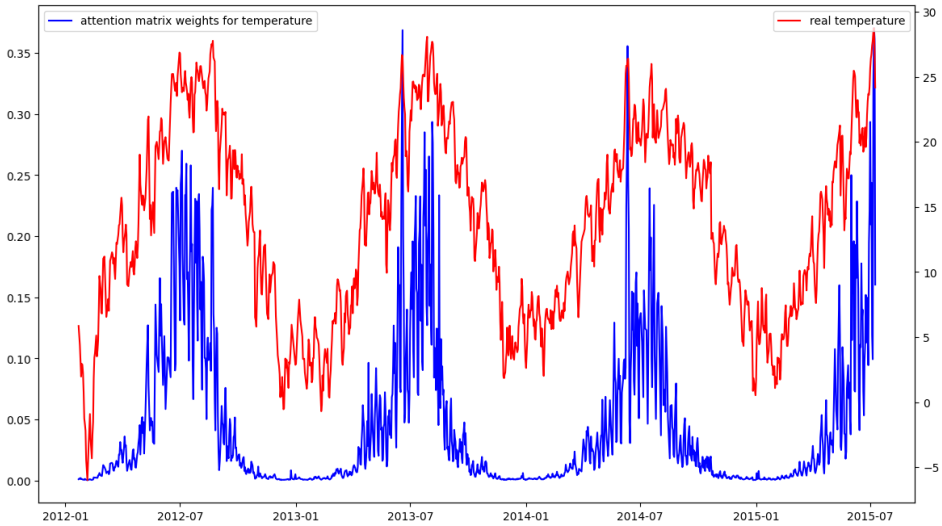

Figure 4.9: Daily temperature vs attention weights for temperature

In addition to the general tendency to have higher attention for higher temperatures, the graph also seems to show that attention is higher when there are large variations in temperature and the temperature is high.

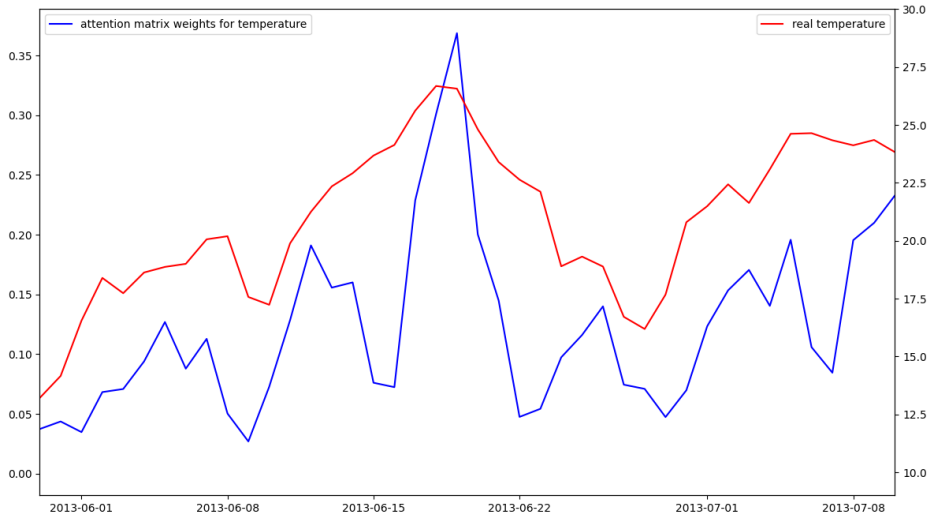

Figure 4.10: Zoom in on of one the highest peaks of attention

Figure 4.11 also shows that the attention peaks match the consumption peaks, which means that in those circumstances the temperature has a determining effect. The higher attention for large variations is also apparent. As previously stated, it was also expected an increase in attention in the initial part where there is a period of very low temperatures. Meaning that it be can concluded that attention was not able to detect this case, which may indicate a limitation. However, this can be justified by the fact that the heating system in Italy relies heavily on gas and, therefore, does not have the same impact as the air conditioning systems during periods of heat.

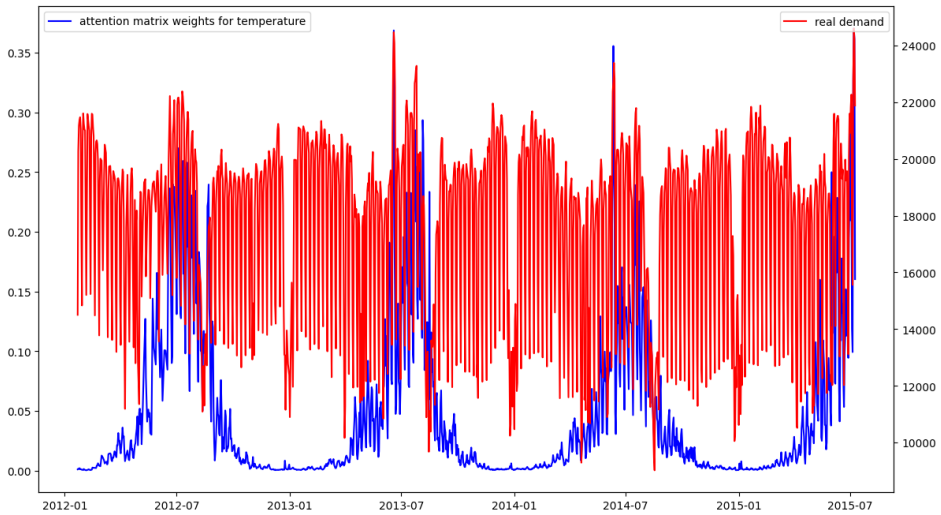

Figure 4.11: Daily consumption vs attention weights for temperature

However, it is worth noting that although the visual inspection of Figures 4.8 and 4.9 might suggest correlation, it does not establish a statistically significant relationship. For that purpose, a formal statistical test was done, Pearson's correlation. With the Pearson's correlation coefficient is it possible to obtain a numerical value that indicates the linear relationship between two sets. The calculation of the coefficient translates the strength and direct association that exists between two continuous variables. This coefficient ranges between -1 and 1 in which -1 means the data is perfectly negatively correlated, 0 means no correlation at all and 1 means perfectly positive correlation. Besides the coefficient, Pearson's correlation also has another characteristic value, known as p-value which can be used to determine the statistical significance of the correlation. A lower p-value (<0.05) means the correlation is statistically significant [34] [35].

A Pearson correlation of 0.624 was obtained from the analysis as well as a null p-value. The obtained p-value means that the observed correlation (i.e., the correlation coefficient) is statistically significant. The Pearson correlation coefficient value obtained suggests a moderate positive correlation between the temperature and the attention weights. This means that as the temperature values increase, the attention weights also tend to increase, and vice versa. This positive correlation is in line with the visual analysis done earlier, where it was observed that higher temperatures seemed to correspond with higher attention weights.

To further confirm the correlation between the two variables and the obtained Pearson coefficient, a scatter diagram based on the two variables was plotted (Figure 4.12). From the scatter plot, it looks like there is a general trend where attention weights increase as temperature increases, which aligns with the positive Pearson correlation coefficient of 0.624 obtained. This, as previously stated, indicates that there is a moderate positive linear relationship between temperature and attention weights. However, it can also be seen a considerable amount of spread in the data points, indicating that while temperature is an important factor, there are likely other factors at play influencing the attention weights.

The line of best fit visually confirms the positive correlation coefficient. It shows that as temperature increases, the model's attention to the temperature feature also tends to increase, which suggests that the model considers temperature more important when it's high. However, the scatter plot also shows a significant amount of variability around this line, indicating that while there's a general trend, there's also a lot of variability in the attention weights that is not explained by temperature alone.

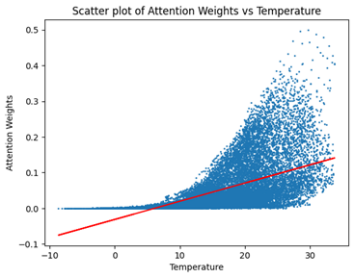

Figure 4.12: Scatter of attention weights vs Temperature

The scatter plot, along with the Pearson correlation coefficient, suggests that temperature is indeed a factor the model considers, but, as anticipated, it's not the only factor. Additionally, while this correlation is statistically significant (as indicated by the p-value), it does not necessarily imply causation.

Also, the strength of the correlation (0.624) while moderate, is not extremely high. This suggests that while there is a relationship between temperature and attention weights, there are likely other factors that the model is also considering when making predictions, being, of course, other features. Moreover, attention mechanisms are complex and can be influenced by various factors in the model and data.

To further analyze this relationship and try to expand on the correlation between those variables, a different temporal analysis was made, in order to understand how the importance of temperature changes over different time periods. For that, the next analysis made was for the variation of the weights over the different hours of the day, which is represented in Figure 4.13.

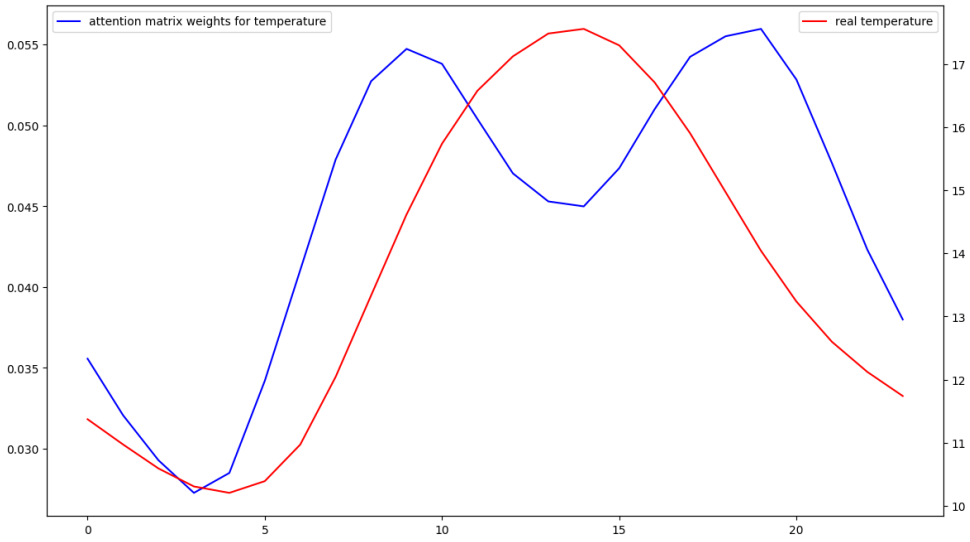

Figure 4.13: Hourly average across all days of the temperature vs attention weights for temperature

The graph displays the model's average attention weight assigned to the temperature feature for each hour of the day, juxtaposed with the actual temperature values. The results are intriguing. It's evident that the patterns for the attention weights and the temperature values differ. The temperature values seem to follow a sinusoidal-type pattern, peaking around midday. Conversely, the attention weights seem to peak twice, once in the early morning and again in the late afternoon.

At first glance, these results seem to contradict a prior analysis, which suggested a correlation between increasing temperature and increasing attention weights. Especially as the attention weights dip precisely when the temperature reaches its peak. Moreover, both the initial rise and final drop in temperature and attention weights appear to be delayed.

However, a deeper analysis reveals a nuanced relationship. While it's true that higher temperatures generally correspond to higher attention, as seen in summer, a closer examination of the

hourly data tells a different story. The attention given is more significant during the hours leading up to the warmest parts of the day. Because of this, it is of good interest to perform a similar analysis to the previous one, but this time inspecting the change in the attention weights juxtaposed to the actual consumption hourly distribution across the day.

The results, visible in Figure 4.14, display a stronger relationship than in the previous analysis. Now, the attention weights appear to follow a pattern similar to consumption, which peaks in the morning and does not drop much until late afternoon, with two significant variations. First, the attention weights exhibit a delay, and second, they show a more prolonged slope around the 10-hour mark. In contrast, consumption tends to remain constant after that, until it drops late in the afternoon, while attention weights present a somewhat big valley and only reach its second peak when the consumption is already on a descending curve.

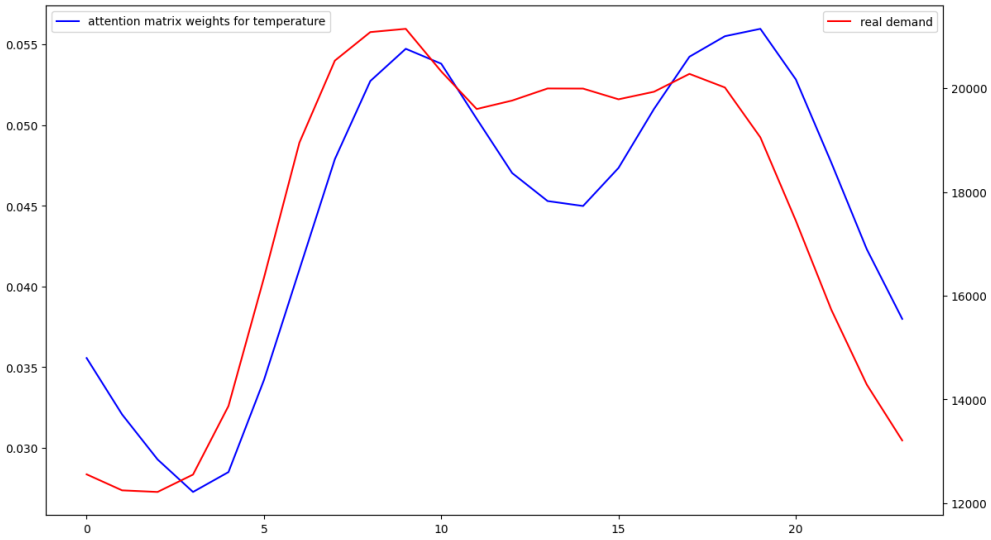

Figure 4.14: Hourly average across all days of the demand vs attention weights for temperature

A deeper and more insightful analysis can be performed by examining each month of the year to identify patterns or unique relationships. To further investigate the relationship between the temperature attention weights and the actual temperature, the plots of the daily values for specific months were done (Figure 4.16) as well as the monthly averages (Figure 4.15). This approach allows for a better understanding of the monthly evolution of both variables.

As confirmed by the previous overall analysis, the attention weights for temperature are generally higher when the temperature is high. Thus, during warmer months like June, July, and August, the weights are relatively high. Conversely, during colder months like November, December, and January, the weights are lower.

As for the shapes of the plots, the weights tend to fluctuate more, displaying peaks and valleys, compared to the more stable temperature data. This fluctuation aligns with the expected inconstant nature of attention mechanisms. An excellent illustration of this is July, from days 5 to 15. Despite only subtle changes in temperature during this period, the weights undergo much more drastic

shifts. This variation may indicate a shift in the model's attention from temperature to another feature it deems more important at that specific time.

That said, despite significant changes in weights in some months, the overall trends and patterns of those months' temperatures are well-captured by the model. For instance, this is evident in March.

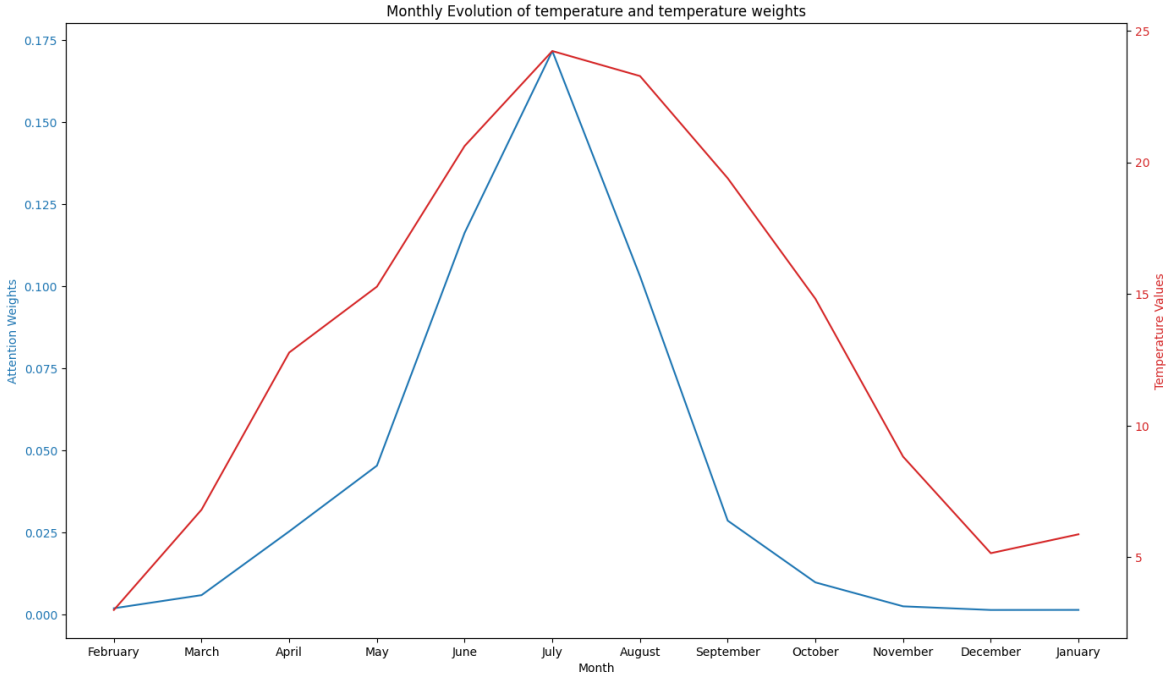

Figure 4.15: Monthly averages of the temperature vs attention weights for temperature

To sum up, this overall monthly analysis confirms the previous results regarding the attention levels. On the one hand, the months with higher temperatures have a higher level of attention, which coincide with periods of higher consumption, and the variations in the various days, in a way, also impact the level of attention. It can probably be inferred that Italy is more sensitive to heat than cold, meaning more sensitive in the summer, and is more influenced by the usage of air conditioning than the usage of heaters.

Finally, one last simple experiment that could be made is, with the knowledge acquired from the previous analysis regarding the relative attention given to temperature in the studied circumstances, to introduce one more or more variables that could represent those cases that corresponded to higher values of attention.

A new variable that could be introduced is one that took the temperature value only if the variation of temperature in the previous day would be higher than a certain threshold ($k1$). Similarly, another new feature could be one that takes the value of the temperature only if it is above a specific threshold ($k2$). It should be mentioned that, initially, the values of the thresholds would be established by observation of the graphs (attention and temperatures) and so it would just be an approximation.

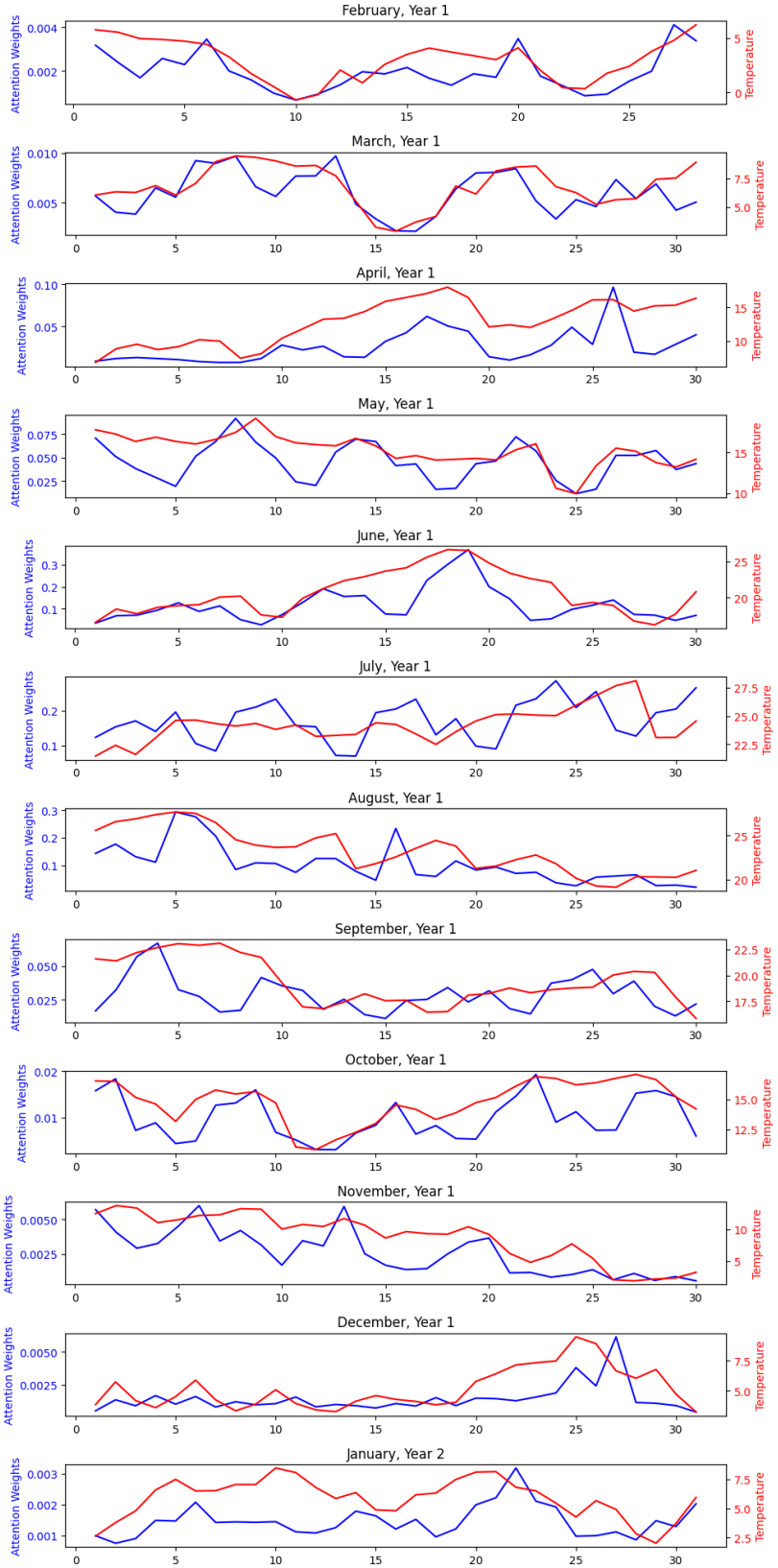

Figure 4.16: Monthly plots of the temperature vs attention weights for temperature

The approach addressed in the previous paragraph was tried using $k1$ equal to 4 and $k2$ equal to 15, resulting in Figures 4.17 and 4.18, respectively.

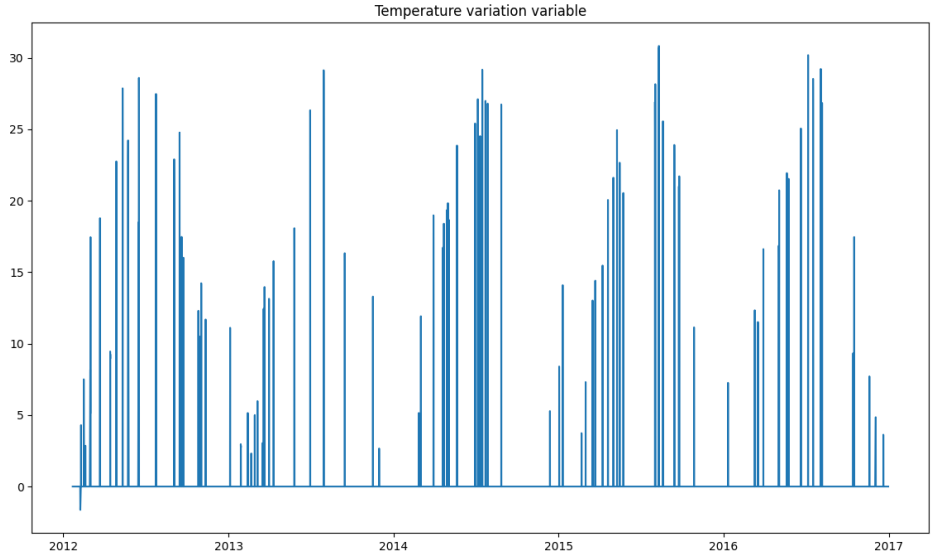

Figure 4.17: Temperature-variation feature

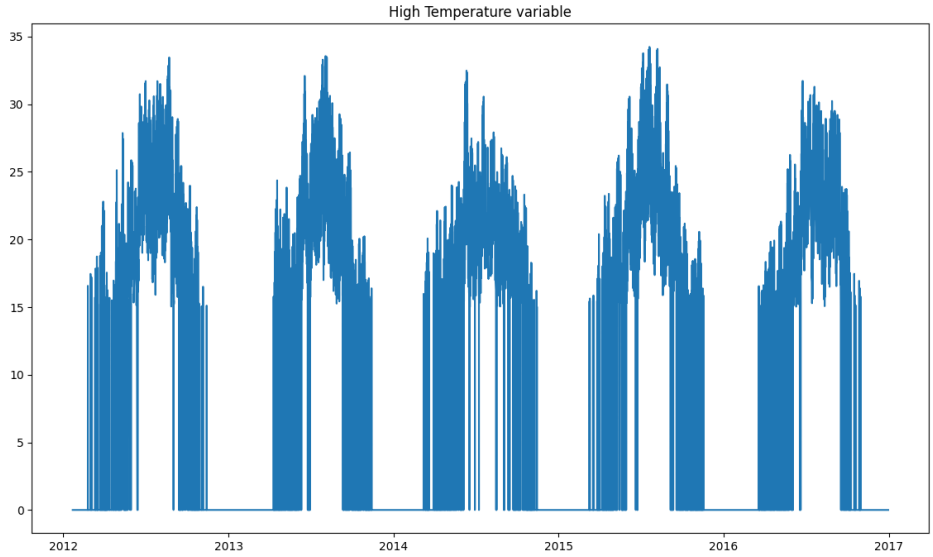

Figure 4.18: High-temperature feature

The results obtained for the predictions with each separate variable and with both are represented in Table 4.2. The new results show that the model does not improve, performance-wise, with the addition of the new variables. It is possible that the model finds the new information to be redundant or even that the threshold values were not the best and should be adjusted.

Table 4.2: Results with the addition of the new variables

Variable	MAE	RMSE	MAPE
Temperature variation variable	853.29	1188.2	5.219%
High Temperature variable	854.97	1188.6	5.290%
Both	857.35	1195.4	5.245%

To see how the attention changes with the inclusion of these two variables, in Figure 4.19, 4.20 and 4.21 are the plots of the general attention of each feature, obtained from the attention matrix, in which the first one corresponds to just the addition of the variation variable, the second one corresponds to the addition of the high-temperature variable and the third one corresponds to the addition of both variables to the inputs of the model.

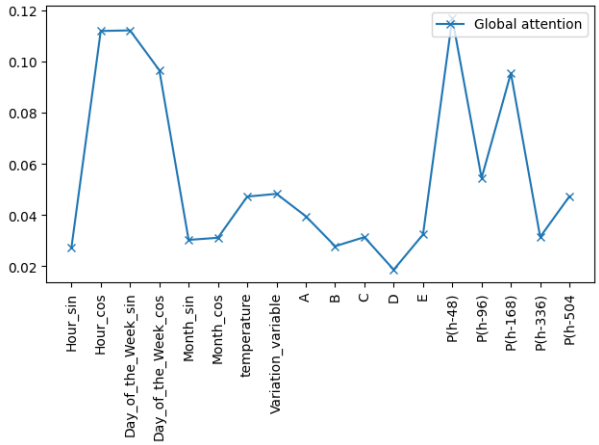

Figure 4.19: New attentions with the variation variable

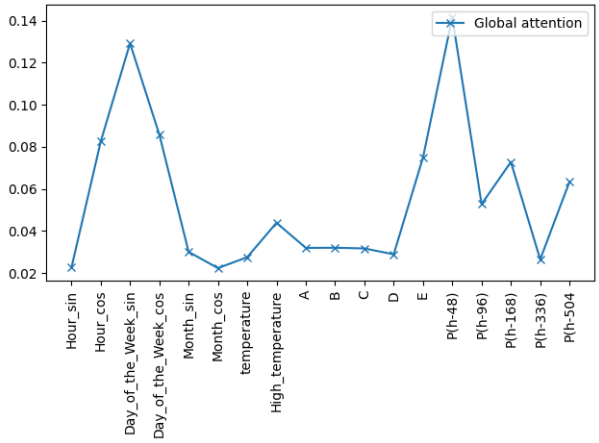

Figure 4.20: New attentions with the high-temperature variable

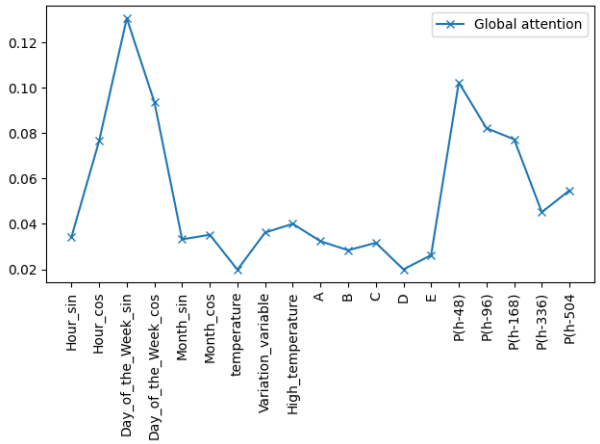

Figure 4.21: New attentions with both variables

Through the analysis of the three figures, first and foremost, is possible to observe the volatile nature of this attention mechanism, with the values regarding attention scores of some of the features fluctuating a little, which could be explained by an analysis done earlier in this work about the correlation between the variables. Regarding the temperature, in Figure 4.19 the new feature has a slightly bigger score than the temperature and both are a bit higher than the previous general analysis. In Figure 4.20 the new feature has a somewhat bigger score than the temperature itself. Finally, with the addition of both variables, in Figure 4.21, the score of the temperature itself drops very close to zero and the two new ones are almost on par, with the high-temperature variable just marginally higher.

The main conclusion drawn from this is that the two new variables can potentially constitute an advantage in terms of the analysis of the attention weights since they generally have higher scores than the temperature itself which. In terms of performance, the addition of these new variables, at first glance, does not seem to improve the results of the model, however, a better adjustment of the thresholds could help the model in performance (and/or analysis). It is worth noting that, the two new variables are the most obvious ones inferred from the previously done studies and, therefore, the integration of different, more sophisticated, variables could help in deeper ways. However, it is necessary to always keep in mind the nature of these scores (the sum of all importances has to be equal to 1) and the correlation between all the variables and how it can impact those scores.

Chapter 5

Conclusions and future work

5.1 Conclusions

This work had two main goals, to examine if the inclusion of attention mechanisms in load forecasting problems brought benefits for predictions, and to employ those mechanisms in the study of the impact of temperature on consumption.

After researching the current state of the art for attention mechanisms, time series forecasting, and model interpretability, a choice was made to slightly sacrifice the quality of the predictions in favor of a simpler, lighter, and easier-to-explain model with also much more direct interpretability. It was indeed concluded that, although the model performed better with than without the attention, it still lost performance-wise to sequence-based models.

Regarding interpretability, the model calculated the global attention in two ways, by averaging all features of the attention matrix and through a novel way based on the direct inspection of the weights of the model. Both methods resulted in somewhat different results, especially regarding temperature, where one gave it the biggest importance across all features while the other gave it small importance. Other global comparisons were made, utilizing SHAP and permutation feature importance, which, again, resulted in different scores for some features, namely temperature. It is important to state, however, that all methods calculate importance using different criteria and different approaches, and the permutation one was applied to a completely different model, which could explain the distinct results, as well as the strong correlation between the features. However, the large difference in some of the results could indicate that this concept of attention has not yet reached its maturity. Nevertheless, the core of this dissertation is focused on the attention variation over time and not particularly on its value compared to the other inputs' attention. In fact, the goal was to find particular conditions for which the attention boosts and try to interpret these variations.

The computation of the attention weight matrix, also made it possible to find out the periods in which the attention was higher and, that way, infer the effect of temperature on the load. Those periods were around the warmer occasions of the year, namely summer. The attention was, however, also particularly high when those hot periods also had big variations, which was also confirmed

when analyzing the evolution of the consumption values. The main conclusion of this is that consumption in Italy is more sensible to warmer periods, probably due to the heavy air conditioning usage, and not as much to colder periods when heaters are used the most. Maybe in this country heaters are not so temperature-responsive as air conditioners. However, it was also expected an increase in attention in the initial part of the data where there was a period of very low temperatures, which is further confirmed by the increase in consumption in this period. Meaning that it be can concluded that attention was not able to detect this case, which may indicate a limitation. However, this can be explained by the substantial reliance on gas in Italy's heating systems during cold periods, which, therefore, does not yield the same effects as air conditioning systems during hot periods that do not work on gas.

Finally, inspired by the previous analysis, two other variables were introduced in the model, one relating to big variations in temperature and the other concerning high-temperature values. Both seem to validate the previous analysis since the score of these two new variables surpassed the score of the temperature itself. Moreover, it seemed that the high-temperature variable assumed more importance than the other two, confirming the previous suspicions regarding the Italy example carried out in this work.

5.2 Future work

Due to some limitations, like inexperience in the field, computational restrictions and time constraints, some interesting analysis could not be carried out and should be considered in further similar studies. Here are the main ones:

- in-depth simulations with independent variables, i.e. not correlated, in which the attention results would, most likely, be more consistent;
- better adjustment of the thresholds in the proposed new variables, as well as, for example, instead of the high-temperature variable taking the values of the temperature above the threshold it could just take value 1, becoming a binary variable
- conception of other, if possible more sophisticated, variables similar to the two new ones relating to temperature made in this work;
- use of other types of data, namely from other countries with different characteristics found in the Italian case;
- use of different models, with different architectures and other types of attention, that could function with sequences, which could possibly capture better temporal dynamics.

References

[1] Naqash Ahmad, Yazeed Ghadi, Muhammad Adnan, and Mansoor Ali. Load forecasting techniques for power system: Research challenges and survey. *IEEE Access*, 10:71054–71090, July 2022. doi:10.1109/ACCESS.2022.3187839.

[2] John McCarthy. What is AI? / basic questions, 2007. Last accessed 2 January 2023. URL: http://jmc.stanford.edu/artificial-intelligence/what-is-ai/index.html.

[3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[4] Sergios Theodoridis. *Machine Learning: A Bayesian and Optimization Perspective*. Academic Press, Inc., USA, 1st edition, 2015.

[5] Crina Grosan and Ajith Abraham. *Machine Learning*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[6] Admin M2 IESC. Artificial intelligence, machine learning, and deep learning: Same context, different concepts, Apr 2018. Last accessed 2 January 2023. URL: https://master-iesc-angers.com/artificial-intelligence-machine-learning-and-deep-learning-same-context-dif

[7] Darniton Amorim Viana. Using attention networks to learn representations for house price prediction, 2019.

[8] Facundo Bre, Juan Gimenez, and Víctor Fachinotti. Prediction of wind pressure coefficients on building surfaces using artificial neural networks. *Energy and Buildings*, 158, 11 2017. doi:10.1016/j.enbuild.2017.11.045.

[9] Mingfei Zhang, Zhoutao Yu, and Zhenghua Xu. Short-term load forecasting using recurrent neural networks with input attention mechanism and hidden connection mechanism. *IEEE Access*, pages 186514–186529, October 2020.

[10] Avijeet Biswal. Recurrent neural network (rnn) tutorial: Types and examples [updated]: Simplilearn, Nov 2022. URL: https://www.simplilearn.com/tutorials/deep-learning-tutorial/rnn.

[11] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6:107–116, 04 1998. doi:10.1142/S0218488598000094.

[12] Dzmitry Bahdanau, Kyunghyun Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *ArXiv*, 1409, 09 2014.

[13] Matthew Riemer, Aditya Vempaty, Flavio Calmon, Fenno Heath, Richard Hull, and Elham Khabiri. Correcting forecasts with multifactor neural attention. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 3010–3019, New York, New York, USA, 20–22 Jun 2016. PMLR. URL: https://proceedings.mlr.press/v48/riemer16.html.

[14] Diego F. Godoy-Rojas, Jersson X. Leon-Medina, Bernardo Rueda, Whilmar Vargas, Juan Romero, Cesar Pedraza, Francesc Pozo, and Diego A. Tibaduiza. Attention-based deep recurrent neural network to forecast the temperature behavior of an electric arc furnace sidewall. *Sensors*, 22(4), 2022. URL: https://www.mdpi.com/1424-8220/22/4/1418, doi:10.3390/s22041418.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[16] Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. Multi-head attention: Collaborate instead of concatenate, 06 2020.

[17] Bryan Lim, Nicolas Loeff, Sercan Arik, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. 2021.

[18] Chenyou Fan, Yuze Zhang, Yi Pan, Xiaoyue Li, Chi Zhang, Rong Yuan, Di Wu, Wensheng Wang, Jian Pei, and Heng Huang. Multi-horizon time series forecasting with temporal attention learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, KDD '19, page 2527–2535, New York, NY, USA, 2019. Association for Computing Machinery. URL: https://doi.org/10.1145/3292500.3330662, doi:10.1145/3292500.3330662.

[19] Edward Choi, Mohammad Taha Bahadori, Joshua A. Kulas, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 3512–3520, Red Hook, NY, USA, 2016. Curran Associates Inc.

[20] Huan-Zhi Song, Deepta Rajan, Jayaraman J. Thiagarajan, and Andreas Spanias. Attend and diagnose: Clinical time series analysis using attention models. In *AAAI Conference on Artificial Intelligence*, 2017. URL: https://api.semanticscholar.org/CorpusID:5985448.

[21] Blaz Skrlj, Saso Dzeroski, Nada Lavrac, and Matej Petkovic. Feature importance estimation with self-attention networks. In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang, editors, *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 1491–1498. IOS Press, 2020. URL: https://doi.org/10.3233/FAIA200256, doi:10.3233/FAIA200256.

[22] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc.

[23] Shapley values. Accessed: 05/06/2023. URL: https://christophm.github.io/interpretable-ml-book/shapley.html.

[24] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3145–3153. JMLR.org, 2017.

[25] 10 ways to estimate shap. Accessed: 05/06/2023. URL: https://mindfulmodeler.substack.com/p/10-ways-to-estimate-shap.

[26] Deep explainer. Accessed: 05/06/2023. URL: https://arize.com/glossary/deep-explainer-deep-shap/.

[27] Shap. Accessed: 05/06/2023. URL: https://christophm.github.io/interpretable-ml-book/shap.html.

[28] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions, 2017. arXiv:1710.05941.

[29] Introduction to boosted trees -xgboost1.7.6documentation. Accessed: 07/07/2023. URL: https://xgboost.readthedocs.io/en/stable/tutorials/model.html.

[30] L Breiman. Random forests. *Machine Learning*, 45:5–32, 10 2001. doi:10.1023/A:1010950718922.

[31] Learning rate finder - pytorchlightning2.1.0devdocumentation. Accessed: 23/06/2023. URL: https://lightning.ai/docs/pytorch/latest/api/lightning.pytorch.callbacks.LearningRateFinder.html#lightning.pytorch.callbacks.LearningRateFinder.

[32] sklearn preprocessing standardscaler. Accessed: 13/06/2023. URL: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html.

[33] Permutation importance. Accessed: 07/07/2023. URL: https://scikit-learn.org/stable/modules/permutation_importance.html#id2.

[34] David Nettleton. Chapter 6 - selection of variables and factor derivation. In David Nettleton, editor, *Commercial Data Mining*, pages 79–104. Morgan Kaufmann, Boston, 2014. URL: https://www.sciencedirect.com/science/article/pii/B9780124166028000066, doi:https://doi.org/10.1016/B978-0-12-416602-8.00006-6.

[35] Pearson's correlation. Accessed: 09/07/2023. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html.