ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

FORECASTING OF PRODUCED OUTPUT ELECTRICITY IN PHOTOVOLTAIC POWER PLANTS

Ph.D. THESIS

Taraneh SAADATI

Department of Energy Science and Technology

Energy Science and Technology Programme

MAY 2025

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

FORECASTING OF PRODUCED OUTPUT ELECTRICITY IN PHOTOVOLTAIC POWER PLANTS

Ph.D. THESIS

Taraneh SAADATI
(301182013)

Department of Energy Science and Technology

Energy Science and Technology Programme

Thesis Advisor: Assoc. Prof. Dr. Burak BARUTÇU

MAY 2025

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

FOTO-VOLTAİK GÜÇ SANTRALLARINDA ELEKTRİK ÜRETİM TAHMİNİ

DOKTORA TEZİ

Taraneh SAADATI
(301182013)

Enerji Bilim ve Teknoloji Anabilim Dalı

Enerji Bilim ve Teknoloji Programı

Tez Danışmanı: Doç. Dr. Burak BARUTÇU

MAYIS 2025

Taraneh SAADATI, a Ph.D. student of ITU Graduate School student ID 301182013, successfully defended the thesis entitled “FORECASTING OF PRODUCED OUTPUT ELECTRICITY IN PHOTOVOLTAIC POWER PLANTS”, which she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Assoc. Prof. Dr. Burak BARUTÇU** Istanbul Technical University	.....
Jury Members :	**Prof. Dr. Gülgün KAYAKUTLU** Istanbul Technical University	.....
	Asst. Prof. Dr. Emre ÇELEBİ Yeditepe University	.....
	Asst. Prof. Dr. Mustafa Berker YURTSEVEN Istanbul Technical University	.....
	Asst. Prof. Dr. Zeynep BEKTAŞ Kadir Has University	.....

Date of Submission : 8 March 2025
Date of Defense : 16 May 2025

To my beloved family,

FOREWORD

As I conclude this journey of academic exploration, I am deeply humbled and grateful to reflect on the many individuals and institutions whose support and guidance have been instrumental in bringing this thesis to fruition.

First and foremost, I wish to extend my heartfelt gratitude to my esteemed supervisor, Assoc. Prof. Dr. Burak BARUTÇU, for his unwavering support and significant contributions to every stage of this thesis. His insightful feedback and continuous encouragement have been invaluable in ensuring the meticulous progression of my research. I am also grateful to Professor Gülgün KAYAKUTLU for her dedicated guidance and mentorship during my time at Istanbul Technical University. Her wisdom and commitment have left an indelible mark on my academic journey. Additionally, I sincerely thank Asst. Prof. Dr. Emre ÇELEBİ, a member of my thesis monitoring committee, for his invaluable insights and constructive contributions throughout this process.

This work has been generously supported by the Scientific Research Projects (BAP) Coordination Unit at Istanbul Technical University [Project ID: 44133, Project code: MDK-2022-44133], to whom I extend my deepest gratitude. Their research grant has been pivotal in facilitating this study. I would also like to acknowledge “Tegnatia Enerji” for their generosity in providing the critical data essential for this research. Their support has been a cornerstone of this study’s success.

I am profoundly thankful to Professor Saeed HERAVI from Cardiff Business School for granting me the opportunity to collaborate as a visiting researcher. Working alongside him and his esteemed colleagues, such as Professor Bahman ROSTAMI-TABAR, has significantly enriched my knowledge and research skills in the fields of hierarchical time series forecasting and data science.

Furthermore, I wish to express my heartfelt appreciation to Assoc. Prof. Ali NAZEMİ, my supervisor during my Master’s studies, for his guidance and encouragement in steering me toward the field of data science. His mentorship played a critical role in shaping my academic interests and pursuits.

Finally, and most importantly, I extend my boundless love and deepest gratitude to my beloved family. I am especially grateful to my mother, Taj FALLAH, whose endless love, resilience, and quiet sacrifices have been the soul of my journey—your grace inspires me every day. To my father, Hossein SAADATI, whose strength, wisdom, and unwavering belief in my potential have shaped the foundation of who I am. This achievement would not have been possible without their faith in me and their unconditional love. I also wish to thank my dear husband, Mesut HATEMI, for his steadfast love, patience, and constant encouragement, which have illuminated even the most challenging moments of this path.

May 2025 Taraneh SAADATI

TABLE OF CONTENTS

Page

ABBREVIATIONS

AI : Artificial Intelligence
ANFIS : Adaptive Neuro Fuzzy Inference Systems
ANN : Artificial Neural Network
AR : Auto Regressive
BPNN : Back Propagation Neural Networks
BRT : Boosted Regression Tree
CNN : Convolutional Neural Network
DL : Deep Learning
EL : Ensemble Learning
ELM : Extreme Learning Machine
EPC : Engineering, Procurement, and Construction
ESN : Echo State Network
FFNN : Feed-forward Neural Networks
FNN : False Nearest Neighbors
GA : Genetic Algorithm
GB : Gradient Boosting
GP : Gaussian Process
GRNN : General Regression Neural Networks
GRU : Gated Recurrent Unit
KNN : K-Nearest Neighbors
LSR : Least Square Regression
LSTM : Long-Short-Term-Memory
ML : Machine Learning
MP : Multi-layer Perceptron
NN : Neural Networks
PV : Photovoltaic
RBF : Radial Basis Function
RES : Renewable Energy Sources
RF : Random Forest
RNN : Recurrent Neural Networks

SPF : Solar Power Forecasting

SPP : Solar Power Plant

SVM : Support Vector Machine

SVR : Support Vector Regression

UN SDGs : United Nations Sustainable Development Goals

XGBoost : eXtreme Gradient Boosting

LIST OF TABLES

Page

LIST OF FIGURES

Page

FORECASTING OF PRODUCED OUTPUT ELECTRICITY IN PHOTOVOLTAIC POWER PLANTS

SUMMARY

The global energy market is pivotal to modern economic growth, with renewable energy, particularly solar power, emerging as a key strategy for mitigating climate change and fostering sustainable development. Solar energy, a clean and abundant resource, directly supports UN Sustainable Development Goals (SDGs) such as Goal 7 (Affordable and Clean Energy) and Goal 13 (Climate Action) while indirectly contributing to Goals 8 (Decent Work and Economic Growth), 9 (Industry, Innovation, and Infrastructure), and 11 (Sustainable Cities and Communities). However, the variability of solar energy generation—shaped by factors like solar radiation, temperature, and technological efficiency—poses challenges for grid stability, resource planning, and the broader integration of renewables. Accurate forecasting is critical to addressing these issues, enabling better grid management, informed policymaking, and strategic investments in renewable energy systems.

This thesis tackles the challenges associated with solar energy forecasting by utilizing cutting-edge developments in artificial intelligence (AI) and machine learning (ML) to enhance prediction accuracy and reliability. It focuses on under-explored areas, including the integration of endogenous and exogenous variables, chaotic modeling, and hybrid model optimization. Through systematic investigation and evaluations, the research proposes innovative techniques validated with real-world data, providing practical insights for managing solar power generation and advancing low-carbon energy systems. By tackling critical gaps in forecasting methodologies, this work supports global sustainability goals, promotes energy security, and accelerates the transition to cleaner and more efficient power networks.

This thesis builds upon the extensive literature review to identify and address critical gaps in solar energy forecasting. The research focuses on four key dimensions: (1) integrating both endogenous variables and exogenous variables for a holistic understanding of solar energy generation, (2) employing robust data preprocessing techniques like dimensionality optimization and feature extraction to enhance data quality and model reliability, (3) exploring advanced deep learning architectures and ensemble frameworks to improve predictive performance, and (4) optimizing metaheuristic algorithms for efficient and scalable AI-based forecasting solutions. These four methodological pillars aim to address challenges such as nonlinearity, temporal dependencies, and chaotic behavior in solar power datasets.

This research utilizes five categories of predictive models—linear methods, algorithms based on machine and deep learning, ensemble approaches, and models rooted in chaos theory—evaluated through empirical data collected from the Konya Eregli Solar Power Plant. Among the models, Feedforward Neural Networks (FFNN), Extreme Learning Machines (ELM), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest (RF), eXtreme Gradient Boosting (XGBoost), and Echo State Networks (ESN) emerged as top performers based on standard evaluation metrics.

These seven models were tested under scenarios combining endogenous and exogenous inputs and underwent optimization of hyperparameters using the Tree-structured Parzen Estimator (TPE) to maximize accuracy. Together, these advancements contribute to more reliable and precise solar energy forecasting, supporting the integration of renewable energy into power grids.

Prior to training the models, the dataset undergoes a thorough cleaning process to ensure the quality and reliability of the data. Outliers are identified and removed using the Local Outlier Factor (LOF) method, followed by a preliminary statistical exploration aimed at uncovering underlying patterns in the time series. Once the data is cleaned and understood, feature engineering strategies are applied to extract and transform relevant features. These strategies are specifically tailored to meet the unique requirements of both individual base models and ensemble stacking meta-models. To ensure robust model evaluation and prevent data leakage, the processed dataset is split into training, validation, and test sets, forming a reliable foundation for both base and meta-model development.

This study evaluates the performance of all implemented forecasting models using five well-established evaluation metrics: R-squared (R^2), Root Mean Square Error (RMSE), Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Together, these metrics offer a thorough analysis of the models' accuracy and reliability.

Among the fine-tuned base models, the LSTM and GRU models demonstrate the highest levels of precision and reliability. These deep learning frameworks are capable of modeling the intricate temporal relationships embedded within solar energy data. Following LSTM and GRU, other models such as FFNN, XGBoost, RF, ESN, and ELM also perform well, with their efficacy varying based on specific scenarios. These results underscore the strengths of deep learning models while validating the applicability of traditional machine learning and ensemble methods in addressing particular forecasting challenges.

To address chaotic behavior in the time series, the False Nearest Neighbors (FNN) method is incorporated as a preprocessing technique for reconstructing the phase space. This step determines an optimized embedding dimension intended to maintain the structural consistency of the original time series. The ESN algorithm is then employed to forecast solar energy production using both the original and embedded data. Results show that optimizing the embedding dimension via the FNN technique significantly improves the model’s capability to capture the nonlinear dynamics underlying solar power generation, leading to more accurate forecasts.

Enhanced ESN results on the embedded dataset emphasize the significance of preprocessing—particularly embedding dimension selection—in chaotic modeling for solar energy systems. Overall, this approach demonstrates how combining proper data embedding with an optimized model architecture leads to superior predictive capabilities, further contributing to efficient and reliable solar energy forecasting.

As the final stage of this thesis, stacking ensemble machine learning techniques are applied to enhance forecasting performance even further. The primary aim is to develop a novel metamodel that surpasses the prediction capabilities of the individual base models. Two categories of stacking ensemble models are investigated: regression-based stacking and neural network-based stacking. Each approach is critically evaluated and compared, leading to the identification of the most accurate model—an essential contribution of this research.

In the first category, traditional regression algorithms—Multi-Linear Regression, Ridge Regression, and Lasso Regression—function as the meta-model to combine the predictions of the seven base models. This approach leverages the strengths of diverse base predictors through a linear combination, aiming to minimize overall forecast error. However, it ultimately does not surpass the established accuracy benchmark.

In the second category, neural networks (NN) function as the meta-model, exploiting their capacity to capture complex nonlinear relationships within the data. Three distinct neural network-based stacking strategies are developed, including the proposed deep learning-based stacking model. This meta-model is deliberately constructed using only the two best-performing base models, selected based on their individual accuracy metrics. To further enhance predictive performance, advanced techniques and optimized hyperparameters are incorporated. As a result, this final stacking model delivers the highest forecast accuracy among all models investigated.

The meta-models' designs undergo several refinements, including the integration of Model Output Statistics (MOS) as a post-processing technique. These adjustments aim to determine whether further tuning can enhance forecasting accuracy. The outcome of these efforts is the Deep Learning-Based Stacking Meta-Model, proposed by this thesis, which predicts solar energy output for the Konya Eregli Solar Power Plant with an impressive precision exceeding 98%. This model not only outperforms the most accurate base model (LSTM) but also achieves the primary objective of this research: setting a new standard for high-accuracy solar energy forecasting.

A key strength of the proposed deep learning-based stacking meta-model lies in its simplified architecture, which distinguishes it from the remaining meta-models introduced throughout the thesis. By focusing exclusively on the two most accurate base models, the meta-model balances exceptional forecasting accuracy with reduced computational demands. This minimalistic design speeds up both the training and deployment processes, aligning with the research objective of optimizing metaheuristic model architectures to create more efficient and scalable AI-driven forecasting solutions.

This level of predictive precision underscores the transformative potential of advanced ensemble techniques in solar energy forecasting. By achieving improved accuracy, these methods enhance energy management systems, enable smoother grid integration, and support strategic long-term planning in renewable energy sectors. As a result, the proposed deep learning-based stacking meta-model proves to be an essential tool for advancing sustainable energy infrastructures and ensuring a reliable transition to renewable energy systems.

FOTO-VOLTAİK GÜÇ SANTRALLARINDA ELEKTRİK ÜRETİM TAHMİNİ

ÖZET

Küresel enerji piyasası, modern ekonomik büyümenin temel taşlarından biri olup, özellikle güneş enerjisi gibi yenilenebilir enerji kaynakları, iklim değişikliğini hafifletmek ve sürdürülebilir kalkınmayı teşvik etmek için önemli bir strateji olarak öne çıkmaktadır. Temiz ve bol bir kaynak olan güneş enerjisi, Birleşmiş Milletler Sürdürülebilir Kalkınma Amaçları'ndan (SKA) 7. Amaç (Erişilebilir ve Temiz Enerji) ve 13. Amaç (İklim Eylemi) gibi hedefleri doğrudan desteklerken, dolaylı olarak 8. Amaç (İnsana Yakışır İş ve Ekonomik Büyüme), 9. Amaç (Sanayi, Yenilikçilik ve Altyapı) ve 11. Amaç (Sürdürülebilir Şehirler ve Toplumlar) gibi hedeflere de katkı sağlamaktadır. Ancak, güneş enerjisi üretimindeki değişkenlik—güneş radyasyonu, sıcaklık ve teknolojik verimlilik gibi faktörlerden etkilenerek—şebeke istikrarı, kaynak planlaması ve yenilenebilir enerji kaynaklarının daha geniş entegrasyonu açısından zorluklar yaratmaktadır. Bu sorunların çözülmesinde doğru tahmin modelleri kritik bir rol oynamaktadır ve bu modeller, şebeke yönetiminin iyileştirilmesini, bilinçli politika oluşturulmasını ve yenilenebilir enerji sistemlerine yönelik stratejik yatırımların gerçekleştirilmesini mümkün kılmaktadır.

Bu tez, güneş enerjisi tahminindeki zorlukları ele alarak, yapay zeka (AI) ve makine öğrenimi (ML) alanlarındaki gelişmelerden yararlanmak suretiyle tahminlerin doğruluğunu ve güvenilirliğini artırmayı amaçlamaktadır. Çalışma, endojen ve eksojen değişkenlerin entegrasyonu, kaotik modelleme ve hibrit model optimizasyonu gibi yeterince araştırılmamış alanlara odaklanmaktadır. Sistematik bir inceleme ve deneysel yaklaşımlar aracılığıyla, gerçek dünya verileriyle doğrulanan yenilikçi teknikler önerilmekte ve güneş enerjisi üretiminin yönetimi ile düşük karbonlu enerji sistemlerinin geliştirilmesi için pratik içgörüler sunulmaktadır. Bu çalışma, tahmin metodolojilerindeki kritik boşlukları ele alarak küresel sürdürülebilirlik hedeflerini desteklemekte, enerji güvenliğini teşvik etmekte ve daha temiz ve daha verimli enerji ağlarına geçişi hızlandırmaktadır.

Bu araştırma, güneş enerjisi tahmini alanındaki kritik boşlukları belirlemek ve ele almak amacıyla kapsamlı bir literatür taramasına dayanmaktadır. Araştırma, dört ana boyuta odaklanmıştır: (1) güneş enerjisi üretiminin bütüncül bir şekilde anlaşılması için hem endojen hem de eksojen değişkenlerin entegrasyonu, (2) veri kalitesini ve model güvenilirliğini artırmak için boyut optimizasyonu ve özellik çıkarımı gibi sağlam veri ön işleme tekniklerinin kullanılması, (3) tahmin performansını iyileştirmek amacıyla ileri derin öğrenme mimarileri ve topluluk çerçevelerinin araştırılması ve (4) verimli ve ölçeklenebilir yapay zeka tabanlı tahmin çözümleri geliştirmek için meta-sezgisel algoritmaların optimize edilmesi. Bu dört metodolojik temel, güneş enerjisi veri setlerinde görülen doğrusal olmayan ilişkiler, zamansal bağımlılıklar ve kaotik davranış gibi zorlukları ele almayı hedeflemektedir.

Çalışmada, Konya Ereğli Güneş Enerjisi Santrali'nden elde edilen gerçek verileriyle doğrulanan beş farklı tahmin modeli grubu—doğrusal, makine öğrenimi, derin

öğrenme, topluluk öğrenimi ve kaotik modeller—kullanılmaktadır. FFNN, ELM, LSTM, GRU, RF, XGBoost ve ESN modelleri, standart değerlendirme metriklerine göre en iyi performans gösteren modeller olarak öne çıkmaktadır. Bu yedi model, endojen ve eksojen girdilerin birleştirildiği senaryolar altında test edilmiş ve doğruluk oranını en üst düzeye çıkarmak için Ağaç Yapılı Parzen Tahmincisi (Tree-structured Parzen Estimator, TPE) kullanılarak hiperparametre optimizasyonuna tabi tutulmuştur. Bu ilerlemeler, yenilenebilir enerjinin enerji şebekelerine entegrasyonunu destekleyerek daha güvenilir ve hassas güneş enerjisi tahminleri yapılmasına katkı sağlamaktadır.

Modellerin eğitilmesinden önce, veri seti, kalite ve güvenilirliği sağlamak amacıyla kapsamlı bir temizlik sürecinden geçirilmiştir. Anormal değerler "Local Outlier Factor" (LOF) yöntemi ile belirlenip çıkarılmış ve zaman serisi özellikleri keşifsel veri analiziyle incelenmiştir. Veriler temizlenip analiz edildikten sonra, ilgili özelliklerin çıkarılması ve dönüştürülmesi için özellik mühendisliği stratejileri uygulanmış. Bu stratejiler hem bireysel temel modellerin hem de topluluk yığınlama meta-modellerinin özel gereksinimlerine uygun şekilde tasarlanmıştır. Veri sızıntısını önlemek ve model değerlendirmesinin sağlamlanması amacıyla, işlenmiş veri seti eğitim, doğrulama ve test alt gruplarına ayrılmakta ve hem temel modellerin hem de meta-modellerin geliştirilmesi için sağlam bir temel oluşturmaktadır. Bu çalışma, uygulanan tüm tahmin modellerinin performansını beş iyi bilinen değerlendirme metriği aracılığıyla analiz etmektedir: R-kare (R^2), Kök Ortalama Kare Hatası (Root Mean Square Error, RMSE), Ortalama Kare Hatası (Mean Squared Error, MSE), Ortalama Mutlak Hata (Mean Absolute Error, MAE) ve Ortalama Mutlak Yüzde Hatası (Mean Absolute Percentage Error, MAPE). Bu metrikler, modellerin doğruluk ve güvenilirlik düzeylerini kapsamlı bir şekilde değerlendirmek için kullanılmaktadır. İncelikle optimize edilen temel modellerden LSTM ve GRU, en yüksek doğruluk ve güvenilirlik seviyelerini ortaya koymaktadır. Bu derin öğrenme mimarileri, güneş enerjisi verilerindeki karmaşık zamansal bağımlılıkları etkili bir şekilde yakalama becerisine sahiptir. LSTM ve GRU modellerinin ardından, FFNN, XGBoost, RF, ESN ve ELM gibi diğer modeller de tatmin edici performans sergilemiş olup, bu modellerin başarımı belirli senaryolara göre farklılık göstermektedir. Elde edilen sonuçlar, derin öğrenme modellerinin güçlü yanlarını öne çıkarırken, geleneksel makine öğrenimi ve topluluk yöntemlerinin belirli tahmin zorluklarının üstesinden gelme potansiyelini de desteklemektedir.

Zaman serisindeki kaotik davranışları analiz etmek için faz uzayı, bir ön işleme yöntemi olan "False Nearest Neighbors" (FNN) ile yeniden yapılandırılmıştır. Bu yöntem, orijinal zaman serisinin yapısal bütünlüğünü koruyarak optimize edilmiş bir gömülü boyut belirler. Ardından, hem orijinal hem de gömülü veriler kullanılarak güneş enerjisi üretimi tahmini için ESN algoritması uygulanmıştır. Elde edilen sonuçlar, FNN yöntemiyle optimize edilen gömülü boyutun, güneş enerjisi üretimindeki doğrusal olmayan dinamikleri daha iyi yakalayarak tahmin doğruluğunu önemli ölçüde artırdığını ortaya koymaktadır. Gömülü veri üzerinde ESN modelinin gösterdiği üstün performans, kaotik modelleme sürecinde—özellikle gömülü boyut seçiminde—veri ön işlemenin kritik önemini vurgulamaktadır. Bu yaklaşım, optimize edilmiş veri gömme işlemi ile etkili bir model mimarisinin birleştirilmesinin üstün tahmin yeteneklerine nasıl katkı sağladığını göstermekte ve güneş enerjisi tahmini için daha verimli ve güvenilir çözümler sunmaktadır.

Tezin son aşamasında, tahmin performansını daha da geliştirmek için yığınlama (stacking) topluluk makine öğrenimi teknikleri uygulanmıştır. Bu süreçteki temel amaç, bireysel temel modellerin tahmin gücünü aşan yenilikçi bir metamodel tasarlamaktır. Bu doğrultuda, yığınlama topluluk modelleri iki ana kategoride ele alınmıştır: regresyon tabanlı yığınlama ve sinir ağı tabanlı yığınlama. Her bir yaklaşım eleştirel bir şekilde değerlendirilmesi ve karşılaştırılması sayesinde en doğru modelin belirlenmesi sağlanmıştır. Elde edilen sonuçlar, bu araştırmanın önemli katkılarından biri olarak öne çıkmaktadır.

Birinci kategoride, yedi temel modelin tahminlerini bir araya getirmek için meta-model olarak geleneksel regresyon algoritmaları (Çoklu Doğrusal Regresyon, Ridge Regresyon ve Lasso Regresyon) kullanılmıştır. Bu yaklaşım, çeşitli temel tahmincilerin güçlü yönlerinden yararlanarak doğrusal bir kombinasyon yoluyla genel tahmin hatasını en aza indirmeyi hedeflemektedir. Ancak, bu yöntem, belirlenmiş doğruluk kriterlerini aşmayı başaramamıştır.

İkinci kategoride, meta-model olarak sinir ağları (NN) kullanılmış ve bu ağların verilerdeki karmaşık doğrusal olmayan ilişkileri yakalama kapasitesinden yararlanılmıştır. Bu bağlamda, üç farklı sinir ağı tabanlı yığınlama stratejisi geliştirilmiş, bunlar arasında önerilen derin öğrenme tabanlı yığınlama modeli yer almıştır. Bu meta-model, yalnızca bireysel doğruluk metriklerine göre en iyi performans gösteren iki temel model kullanılarak kasıtlı olarak tasarlanmış. Tahmin performansını daha da artırmak için ileri düzey teknikler ve optimize edilmiş hiperparametreler uygulanmıştır. Sonuç olarak, bu nihai yığınlama modeli, incelenen tüm modeller arasında en yüksek tahmin doğruluğunu sağlamıştır.

Meta-modellerin tasarımları, Model Çıkış İstatistikleri (Model Output Statistics, MOS) gibi bir son işleme tekniğinin entegrasyonunu içeren çeşitli iyileştirmelerden geçirilmiştir. Bu düzenlemeler, ek ayarlamaların tahmin doğruluğunu artırıp artırmayacağını belirlemeyi amaçlamaktadır. Bu çalışmaların sonucu olarak, bu tezde önerilen Derin Öğrenme Tabanlı Yığınlama Meta-Modeli, Konya Ereğli Güneş Enerjisi Santrali'nin enerji çıktısını %98'in üzerinde etkileyici bir hassasiyetle tahmin etmektedir. Bu model, en doğru temel model olan LSTM'yi aşmakla kalmayıp, aynı zamanda bu araştırmanın temel hedefi olan yüksek doğruluklu güneş enerjisi tahmini için yeni bir standart belirlemeyi de başarmaktadır.

Önerilen derin öğrenme tabanlı yığınlama meta-modelinin önemli bir gücü, bu çalışmada geliştirilen diğer meta-modellerle karşılaştırıldığında öne çıkan sadeleştirilmiş mimarisinde yatmaktadır. Yalnızca en doğru iki temel modele odaklanarak, meta-model üstün tahmin doğruluğu ile düşük hesaplama maliyetlerini dengeler. Bu minimalist tasarım, hem eğitim hem de dağıtım süreçlerini hızlandırarak, araştırmanın meta-sezgisel model mimarilerini optimize etme ve daha verimli, ölçeklenebilir yapay zeka tabanlı tahmin çözümleri oluşturma hedefine katkıda bulunmaktadır. Bu düzeydeki tahmin hassasiyeti, güneş enerjisi tahmininde ileri düzey topluluk tekniklerinin dönüştürücü potansiyelini ortaya koymaktadır. Artan doğruluk seviyelerine ulaşarak, bu yöntemler enerji yönetim sistemlerini geliştirmekte, şebeke entegrasyonunu kolaylaştırmakta ve yenilenebilir enerji sektörlerinde stratejik uzun vadeli planlamayı desteklemektedir. Sonuç olarak, önerilen derin öğrenme tabanlı yığınlama meta-modeli, sürdürülebilir enerji altyapılarının geliştirilmesi ve yenilenebilir enerji sistemlerine güvenilir bir geçişin sağlanması için önemli bir araç olarak öne çıkmaktadır.

1. INTRODUCTION

The global energy market is a key driver of economic growth and development, making it a cornerstone of modern society (Karim et al, 2023; Rao et al, 2023). The transition toward renewable energy sources, particularly solar energy, has become increasingly critical in addressing climate change and promoting sustainable economic practices. Solar energy, a clean and abundant resource, is a key driver of the green economy, supporting global efforts to reduce greenhouse gas emissions and mitigate the adverse impacts of climate change (Karim et al, 2022). In alignment with the United Nations Sustainable Development Goals (UN SDGs), solar energy directly contributes to Goal 7 (Affordable and Clean Energy) and Goal 13 (Climate Action), while indirectly supporting Goals 8 (Decent Work and Economic Growth), 9 (Industry, Innovation, and Infrastructure), and 11 (Sustainable Cities and Communities). These contributions underscore the transformative potential of solar energy in fostering global sustainability and resilience.

Despite its immense potential, solar energy generation is inherently variable and influenced by complex, interrelated factors such as solar radiation, ambient temperature, and system efficiencies. This variability poses challenges for energy grid stability and optimal resource utilization. Precise forecasting of solar energy output is crucial for overcoming these challenges, allowing grid operators to efficiently manage supply and demand, improve energy system reliability, and facilitate the integration of larger amounts of renewable energy into existing grids. Forecasting also plays a strategic role in planning renewable energy adoption and setting realistic electricity generation targets, thereby supporting the broader transition to low-carbon energy systems (Tiba and Belaid, 2021).

Over the years, forecasting approaches have progressed considerably, driven by advancements in artificial intelligence (AI) and machine learning (ML) (Vrettos and Gehbauer, 2019). Traditional statistical models, while effective for simpler time series, often fall short in capturing the nonlinear and chaotic dynamics of solar power generation. Modern approaches, including deep learning (DL), ensemble learning

(EL), and chaotic analysis techniques, have shown superior performance in handling the complexities of solar energy forecasting (Saadati and Barutcu, 2024b). AI-based models leverage the strengths of various techniques, utilizing real-time inputs along with historical and weather forecast data to enhance predictive accuracy and robustness. These advanced methods ensure dynamic, adaptive forecasting capabilities that are essential for the efficient management of renewable energy resources.

Two primary research directions have emerged in solar energy forecasting. The first focuses on predicting solar irradiance using meteorological data, while the second, referred to as direct forecasting, aims to predict power output using historical time series data of energy generation. The latter approach, which integrates both endogenous (e.g., historical power output) and exogenous variables (e.g., ambient temperature), offers a more comprehensive framework for forecasting. However, direct forecasting remains underexplored compared to irradiance-based methods (Saadati and Barutcu, 2024a). Moreover, chaotic analysis and neural network models, such as the Echo State Network (ESN), present promising yet underutilized avenues for improving forecast accuracy, especially when paired with data preprocessing techniques like dimensionality optimization (Saadati and Barutcu, 2024c).

Current thesis contributes to this field by addressing critical gaps and advancing the state of solar energy forecasting. First, a comprehensive review of AI and ML models for photovoltaic forecasting was conducted, highlighting methodological trends and identifying underexplored areas in the literature. This includes analyzing the characteristics of solar radiation and power output, which are crucial for improving model accuracy and guiding future research. Second, various forecasting approaches were implemented to allow for a wide comparison with respect to the literature. These approaches were categorized into five main groups of forecasting methodologies: linear approaches, machine learning techniques, deep learning architectures, ensemble strategies, and chaos-based methods. Empirical validation using real-world measurements collected at the Konya Eregli photovoltaic plant demonstrated the robustness of this approach. Third, the integration of chaotic analysis and ESN models further advanced direct forecasting methods, revealing the importance of preprocessing and dimensionality reduction in learning the complex, nonlinear behavior of solar energy production. This phase also leveraged the False Nearest Neighbors (FNN) technique to optimize embedding dimensions of time series data and

enhance model performance. Lastly, a novel forecasting methodology was developed by hybridizing the most successful implemented forecasting models, based on their superior performance in terms of evaluation metrics, and creating a stacking forecasting meta-model capable of surpassing the forecast accuracy of all the previously implemented models.

This research not only enhances forecasting accuracy but also supports global sustainability initiatives by enabling better management of renewable energy resources. Improved solar energy forecasting directly contributes to achieving the UN SDGs by facilitating the transition to a low-carbon economy, reducing resource uncertainty, and promoting energy system resilience. By addressing these critical challenges, this thesis lays the foundation for advancing solar energy forecasting methodologies and their application in real-world energy systems.

1.1 Purpose of the Thesis

This thesis aims to advance the state of solar energy forecasting by addressing key challenges and gaps in the field. Solar energy, a cornerstone of the green economy, offers immense potential for mitigating climate change and supporting global sustainability goals. However, the inherent variability of solar power generation poses significant challenges for energy grid stability, resource allocation, and renewable energy integration. Reliable forecasting is essential for addressing these challenges, allowing grid operators and policymakers to manage supply and demand, strengthen energy system reliability, and establish realistic goals for renewable energy integration.

This thesis contributes to this endeavor by exploring advanced artificial intelligence (AI) and machine learning (ML) methodologies to enhance precision and robustness of solar energy forecasting. Specifically, the research aims to address underexplored areas such as direct forecasting methods that integrate endogenous and exogenous variables, the application of chaotic models, and the optimization of forecasting models through hybridization. By developing innovative approaches and validating them using real-world data, this work enhances our understanding of solar power generation dynamics and provides actionable insights for the efficient management of renewable energy resources.

Ultimately, the thesis aligns with global efforts to transition to low-carbon energy systems and achieve the United Nations Sustainable Development Goals (UN SDGs), specifically Goals 7 (Affordable and Clean Energy) and 13 (Climate Action). Enhancing forecasting techniques for solar energy, this research contributes to the wider objectives of sustainability, resilience, and economic growth, laying the foundation for more effective integration of renewable energy into modern power systems.

1.2 Research Questions

This research is guided by several critical questions aimed at addressing the complexities and challenges of solar energy forecasting:

1. How can the precision and reliability of models used for forecasting solar energy be improved using advanced AI and ML methodologies?

This question explores the potential of modern forecasting techniques, including deep learning, ensemble learning, and chaotic models, in capturing the nonlinear and dynamic nature of solar power generation.

2. What is the role of integrating endogenous and exogenous variables in enhancing the efficiency of solar energy forecasting models?

This inquiry focuses on the significance of combining historical power output (endogenous) with external factors such as solar cell temperature and ambient temperature (exogenous) to create comprehensive forecasting frameworks.

3. How can preprocessing techniques, such as dimensionality optimization using the False Nearest Neighbors (FNN) method, improve model accuracy and efficiency in prediction?

This question examines the effect of data preprocessing on the accuracy and reliability of forecasting models, particularly in chaotic systems.

4. Can hybridization and stacking methodologies outperform individual forecasting models in terms of predictive accuracy?

This question investigates the potential of combining the strengths of multiple models to create a meta-model with superior performance.

By exploring these questions, this thesis seeks to provide new insights and practical solutions for solar energy forecasting, aiding its implementation in real-world energy systems.

1.3 Research Scope

The focus of this research includes the exploration and implementation of advanced forecasting methodologies to improve solar energy prediction accuracy. The research focuses on direct forecasting approaches, which predict power output based on historical energy generation data and integrate both endogenous and exogenous variables. Unlike irradiance-based methods, direct forecasting offers a more practical framework for electricity prediction by incorporating actual energy production values into the forecasting process.

This thesis utilizes a variety of forecasting approaches, classified into five key categories: linear models, machine learning models, deep learning models, ensemble learning models, and chaotic models. The study includes an empirical evaluation of these methods using real-world data collected at the Konya Eregli solar energy site, Turkey. This dataset provides a robust foundation for testing and validating the proposed methodologies under practical conditions.

In addition to model implementation, the research explores preprocessing techniques, such as dimensionality optimization using the False Nearest Neighbors (FNN) method, to enhance model performance. The integration of chaotic analysis and Echo State Network (ESN) models is also investigated to address the nonlinear dynamics of solar power generation. Finally, the development of a stacking meta-model represents an innovative step toward hybridizing the most effective forecasting models to achieve superior accuracy.

While the study makes significant contributions to solar energy forecasting, it does not delve into hardware optimization or economic feasibility studies. The main emphasis is on enhancing forecasting techniques and their practical implementation in real-world energy systems, ultimately aiding the global shift toward sustainable energy.

1.4 Research Method

The research methodology adopted in this thesis combines a comprehensive literature review with the implementation and evaluation of advanced forecasting models. The study begins with a detailed review of existing AI and ML methodologies for solar energy forecasting, identifying trends, gaps, and opportunities for innovation. This review forms the foundation for the development and implementation of the proposed methodologies.

The research employs five groups of forecasting models: linear models, machine learning models, deep learning models, ensemble learning models, and chaotic models. These groups were chosen to address the diverse characteristics of solar power data, including linear trends, nonlinearity, temporal dependencies, and chaotic behavior. By categorizing the models into these groups, the research aims to systematically evaluate their effectiveness in capturing different aspects of solar energy generation dynamics, ensuring a comprehensive and balanced analysis of forecasting methodologies. Each model is implemented and tested using real-world data from the Konya Eregli Solar Power Plant. This dataset includes both endogenous variables (e.g., historical energy output) and exogenous variables (e.g., ambient temperature, solar cell temperature), providing a comprehensive basis for analysis. In the initial phases of this study, ten different forecasting algorithms are implemented and evaluated across more than twenty distinct scenarios. These algorithms included Auto Regressive Integrated Moving Average (ARIMA), Exponential Smoothing (ETS), Feedforward Neural Network (FFNN), Extreme Learning Machine (ELM), Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), Gated Recurrent Unit (GRU), eXtreme Gradient Boosting (XGBoost), Random Forest (RF), and Echo State Network (ESN). The algorithms demonstrating the highest effectiveness based on key performance indicators—namely, the coefficient of determination (R-squared), Root Mean Square Error, and Mean Absolute Error— are selected for the next stages of the study. These models are subsequently incorporated for the combination of endogenous and exogenous variables and underwent further hyperparameter optimization.

To enhance model performance, the study incorporates preprocessing techniques such as dimensionality optimization using the False Nearest Neighbors (FNN) method. This

step ensures that the models effectively capture the nonlinear and chaotic dynamics of solar power generation. Additionally, the research investigates the application of chaotic models and Echo State Networks (ESNs) to manage the challenges associated with forecasting solar energy. Chaotic models are particularly well-suited for capturing the highly nonlinear and dynamic behavior inherent in solar energy generation, which conventional methods may struggle to represent effectively. ESNs, a type of reservoir computing model, excel in learning temporal dependencies in time series data while being computationally efficient, making them efficient for working with large datasets and intricate patterns associated with solar power forecasting (Shahi et al, 2022).

The study also employs the Tree-structured Parzen Estimator (TPE) method to fine-tune the forecasting models. The TPE method was selected for its effectiveness and capability to manage high-dimensional and complex search spaces, making it particularly well-suited for optimizing hyperparameters in computationally intensive forecasting models (Watanabe, 2023). By leveraging its probabilistic approach to optimization, TPE enhances model performance while maintaining computational feasibility, ensuring robust and reliable results. Each model's hyperparameters are optimized using the TPE method, and the resulting optimized hyperparameters are reported as part of the empirical study.

The final phase of the research focuses on creating a novel hybrid forecasting methodology. By combining the strengths of the most successful models through a stacking meta-model, the research demonstrates the potential for achieving superior forecasting accuracy. The proposed meta-model is empirically validated, showcasing its ability to surpass the performance of individual models.

This methodological framework not only advances the state of solar energy forecasting but also provides a scalable and adaptable approach for future studies and real-world implementations in renewable energy management.

2. LITERATURE STUDY

This chapter draws extensively from the author's literature survey paper titled "Forecasting Solar Power: Leveraging Artificial Intelligence and Machine Learning for Sustainable Energy Solutions," which was accepted for publication in the Journal of Economic Surveys in December 2024.

Accurate forecasting is essential for managing energy resources efficiently and optimizing the operation of energy production units. In grid-connected solar power systems, the need for precise predictions becomes even more significant. Such forecasts ensure that backup generators can effectively address the gap between predicted and actual energy production, thereby maintaining a continuous energy supply. Accurate predictions not only reduce the number of required backup units but also lower implementation costs. Moreover, some factors like solar radiation and temperature significantly influence the ramp rate, which measures the rate of change in power production on a per-minute basis. By accurately predicting energy production, operators can manage the ramp rate more effectively and optimize the deployment timing of backup systems, ensuring uninterrupted electricity generation.

This chapter provides a detailed review of significant studies in the field of solar energy forecasting. It highlights the necessity of PV power forecasting, the variables and parameters commonly used to enhance predictive models, and the diverse forecasting methods and approaches that have been employed. Furthermore, it discusses the conditions under which these methods have been applied and compares their results to offer insights into their relative efficacy.

Forecasting PV power is essential for maintaining balance in energy supply and demand, as well as for efficient energy network management. Accurate predictions enhance resource organization and facilitate more efficient functioning of production systems (David et al, 2016). With the increasing reliance on renewable energy sources such as solar, wind, and waves, forecasting the energy generated by these systems has

become more crucial than ever for optimal functionality (Haessig et al, 2015; Hanna et al, 2014; Hernández-Torres et al, 2015).

Prediction intervals play a key role in designing forecasting models, as they determine the most suitable data sources. Dambreville et al. (2014), along with Kühnert et al. (2013), demonstrated that satellite-based models perform well for hourly forecasting intervals. Diagne et al. (2013) recommended predicting load patterns two days ahead for short-term forecasts, enabling efficient scheduling of power plants and energy transactions in the market. Barbieri et al. (2017) identified "very short-term horizons" (a few seconds to 30 minutes) as critical for real-time decision-making, such as electricity market clearing and pricing. Similarly, Kostylev and Pavlovski (2011) categorized forecast horizons into day-ahead, intra-day, and intra-hour intervals, each suited for specific operational needs.

Barbieri et al. (2017) proposed a framework for forecasting PV power over very short time horizons. They emphasized that PV energy distribution is highly dependent on climatic conditions and is challenging without an appropriate energy storage system. The study concluded that accurate solar power forecasting is essential for two reasons: (1) ensuring continuous power generation and (2) enabling ramp rate control for the power grid. The authors outlined three steps in solar power forecasting: (1) defining the prediction time horizons based on grid functionality, (2) identifying suitable data sources, and (3) processing the data using statistical methods. They further highlighted that predictions based on cell temperature and solar irradiance yield the best results for modeling PV variability.

Forecast horizon selection is a foundational step in solar power forecasting, as it determines the suitable forecasting approach. In the context of ultra-short-term horizons—typically spanning from several minutes up to two hours, statistical methods based on time series are typically employed (Barbieri et al, 2017). However, recent studies emphasize the advantages of integrating diverse models. For instance, hybrid methods combining neural networks with regression models have shown superior performance by leveraging appropriate input data and optimization techniques (Amiri et al, 2021).

Several review studies, such as those by Antonanzas et al. (2016) and Das et al. (2018), have categorized forecasting methodologies into statistical, physical, and hybrid

models. Statistical approaches rely on historical data, while physical methods, including numerical weather predictions, satellite imagery, and sky imagery, incorporate atmospheric dynamics. Hybrid models, which combine elements from statistical and physical approaches, are particularly effective in enhancing accuracy and addressing the limitations of standalone models.

Advancements in machine learning (ML) and artificial intelligence (AI), especially artificial neural networks (ANNs) and support vector machines (SVMs), have significantly improved forecasting accuracy (Colak and Qahwaji, 2007; Sobri et al, 2018). Optimizing parameters using methods such as genetic algorithms and clustering has further enhanced ANN-based models (Aybar-Ruiz et al, 2016). For example, wavelet transforms are widely used in hybrid models to preprocess meteorological data by filtering noise, thereby improving model performance (Ahmed et al, 2020).

Hybrid models, especially those employing ensemble techniques, consistently outperform single models (Chen et al, 2013; Doorga et al, 2019; Basurto et al, 2019). They are particularly advantageous for day-ahead predictions across various climates (Blaga et al, 2019). Studies also highlight the critical role of selecting appropriate inputs, optimizing parameters, and tailoring forecasting methods to specific climatic conditions and prediction horizons (Guermoui et al, 2020; Voyant et al, 2017).

Emerging methods, such as recurrent neural networks (RNNs) and adaptive neuro-fuzzy inference systems (ANFIS), demonstrate strong potential for handling complex, nonlinear problems in solar power forecasting (Yona et al, 2013). However, challenges like overfitting and high data demands remain, highlighting the necessity for ongoing innovation and optimization in the field (Dolara et al, 2015).

Among the studies reviewed in this section of the survey, several have made significant contributions to advancing the understanding, methodologies, or theories within this field. These contributions are summarized in Table 2.1. Notably, many of these studies propose hybrid methodologies and highlight their advantages and positive impacts on forecasting performance. As shown in Table 2.1, the majority of these studies conclude that adopting a hybrid approach improves forecasting accuracy. Studies marked with

an asterisk (*) after the citation indicate the use of hybrid methodologies that integrate statistical, AI, and/or physical approaches.

Table 2.1 : Summary of the studied papers and their contributions (Saadati and Barutcu, 2024a).

Article	Methodology	Result
(Soman et al, 2010)	Review Article	The precision of the physical forecasting model increases notably during periods of stable weather conditions.
(Pedro and Coimbra, 2012)*	Assessment of solar power forecasting techniques with no exogenous inputs	- Prediction models based on ANN operate better than other prediction methods. - By optimizing the parameters of this model using the Generic Algorithm, a significant advance in results is achieved.
(Yona et al, 2013) *	Combination of the fuzzy inference model and Recurrent Neural Networks	Employing the fuzzy inference model aids in smoothing the meteorological data utilized for forecasting PV power generation.
(Russo et al, 2014)	GA Programming	The simple models with only two input variables act like the more complex models examined in the study.
(Aggarwal and Saini, 2014)*	- Least-Square Regression model (LSR) - Regularized Least-Square Regression model - Artificial Neural Networks	**The most efficient functionality among the introduced models:** Combined Neural Networks and Least-Squares Regression model
(Hu et al, 2014)	Support Vector Regression (SVR)	Time series regression can be performed using SVR, which is based on Support Vector Machine methodology.
(Dolara et al, 2015) *	Combination of AI, the Physical forecasting model, and Statistical models	Higher forecasting accuracy
(Z. Li et al, 2016)	Comparing the SVR and ANN to forecast three time- horizons	Improved forecasting results are achieved when the prediction model is individually applied to the inverters of the PV system.
(Antonanzas et al, 2016)	- Economic benefits of a precise prediction - Point/Regional forecasts -Probabilistic/ Deterministic forecasts	Emphasized on the importance of the Probabilistic forecast.
(Aybar-Ruiz et al, 2016)*	A new hybrid Grouping Genetic Algorithm-Extreme Learning Machine approach	More accurate prediction for global solar radiation
(Voyant et al, 2017)	Review Article	Highlights the important role of ANN on forecasting solar radiation.
(He Jiang et al, 2017) *	- Sparse Quadratic RBF Neural Networks - Eclat algorithm - Cuckoo Search Algorithm	Increase in the precision of the solar radiation forecast
(Barbieri et al, 2017)	**Three steps to forecast the Solar Power** 1-Defining Time Horizon 2-Determine adequate source of input data 3- Process data by statistical methods	**The best approaches to forecast the PV fluctuations:** - Forecasting the irradiance and the cell temperature. - Combining several sources of input to achieve a better forecasting result.

Table 2.1 (continued) : Summary of the studied papers and their contributions (Saadati and Barutcu, 2024a).

(Sobri et al, 2018)	Review Article	ANN and Support Vector Machine are intensively utilized due to their effectiveness in solving non-linear and complex forecasting models.
(Das et al, 2018) *	Review Article	- Hybrid models have achieved superior outcomes in addressing the PV power forecasting issue when compared to standalone methods. - RNNs excel in learning diverse complex relationships and information processing frameworks.
(de Freitas Viscondi and Alves-Souza, 2019)	Review Article	Enhancing the precision of the forecast: Raising the number of meteorological parameters in the training process.
(Basurto et al, 2019) *	- Hybrid Intelligent System (HIS) - Execute both supervised and unsupervised learning (ANN and Clustering)	Improved regression results
(Blaga et al, 2019) *	Comparing the forecasting models' performance based on the case-study's climate	Generally, hybrid models have the best performance, but to be specified, for day-ahead forecasts these models demonstrate remarkable performance across all climates.
(Guermoui et al, 2020) *	Review Article	- In all examined scenarios with varying inputs and outputs, all proposed hybrid models consistently outperform standalone models, demonstrating high performance across the board. - "Decomposition-Clustering based Ensemble Learning Approaches" can be selected as the top hybrid techniques for solar radiation forecasting.
(Amiri et al, 2021)	Clustering	New approaches in the field of ANN have appeared to reduce the problem of determining the activation function, the number of hidden neurons and the value of the weights.

The integration of various approaches in photovoltaic forecasting has significantly improved model accuracy, with a greater focus on combining methods rather than optimizing standalone algorithms. Numerous studies demonstrate that blending numerical weather models with machine learning or other algorithms produces more reliable outcomes. This achievement highlights both the potential for future research and the inherently interdisciplinary nature of challenges in photovoltaic forecasting.

2.1 Key Challenges and Emerging Themes

This section synthesizes various perspectives on solar energy forecasting from previous research, integrating findings from de Freitas Viscondi and Alves-Souza

(2019), who conducted a systematic literature review of 38 research papers. By focusing on the motivations behind the analyzed studies and literature surveys, it is evident that most research questions in this domain address the following key areas:

- Enhancing Accuracy: The implementation of diverse machine learning (ML) and statistical models to improve forecasting precision.
- Input Parameter Selection: Evaluating different combinations and numbers of input variables to achieve more accurate predictions.
- Statistical Foundations: Exploring the essential statistical steps necessary for constructing robust forecasting models.
- Hybrid Models: Investigating the potential of hybrid approaches that combine two or more statistical and/or ML models to enhance performance.

In general, machine learning methodologies used for solar energy output prediction rely on historical solar irradiance data and weather information as key input variables. Various algorithms have been utilized, with the most prominent ones in the reviewed literature being artificial neural networks (ANN), support vector machines (SVM), regression forests, multilayer perceptrons (MLP), and gradient boosting (GB).

Figure 2.1 illustrates the frequency of these algorithms' implementation, based on the aggregation of results from de Freitas Viscondi and Alves-Souza (2019) and the studies analyzed in this survey.

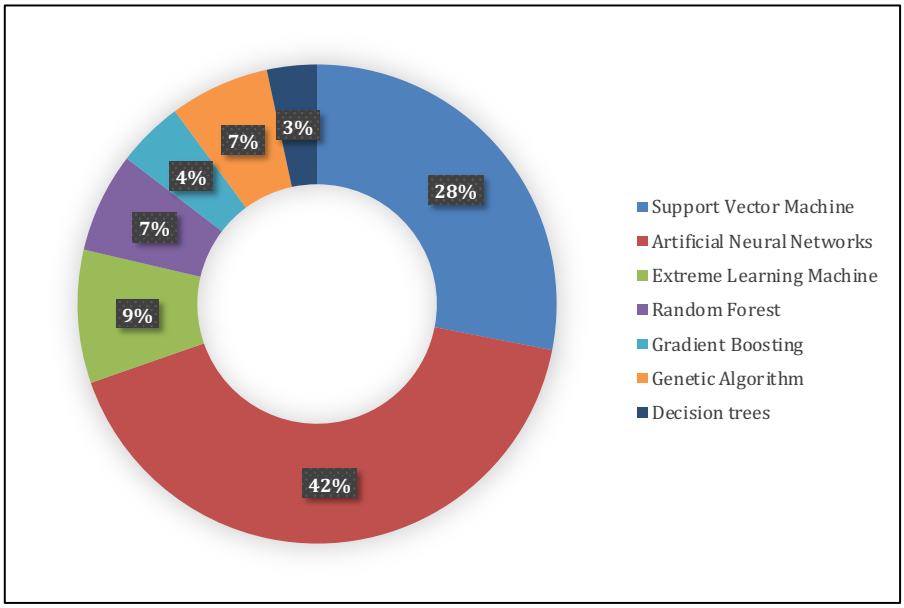

Figure 2.1 : Usage percentage of each method in the analyzed papers (Saadati and Barutcu, 2024a).

Among the various algorithms reviewed, artificial neural networks (ANN) and support vector machines (SVM) are the most frequently implemented methods in solar photovoltaic forecasting studies. However, several studies have also explored ensemble learning and deep learning methods, achieving satisfactory levels of accuracy. This highlights the untapped potential of techniques such as Random Forest (RF), eXtreme Gradient Boosting (XGBoost), and Long Short-Term Memory (LSTM), which, despite their promise, have been investigated less frequently than ANN and SVM in this domain.

Different input parameters have been explored across the studies. To examine this aspect in greater depth and identify potential research gaps, a specific research question is defined and followed in this section: "What type of data is most commonly used in this research area?" This section of the study addresses this question by focusing on the empirical data used in forecasting studies, detailing its nature, scope, and application. The analysis emphasizes the contribution of power generation data—such as power (W), electrical energy generation (kWh), voltage (V), and current (A)—to forecasting accuracy, as this remains the central objective of most studies in this field.

Another significant challenge identified in relation to this research question is the issue of data quality (Francisco et al, 2017). Table 2.2 provides an overview of the nature of input data, data pre-processing techniques, and the machine learning approaches employed in the reviewed studies on solar energy forecasting. The goal is to shed light on how electricity-related data is utilized and how these practices influence the accuracy of photovoltaic electricity generation predictions.

Table 2.2 : Investigation of data nature and quality assessments (Saadati and Barutcu, 2024a).

Article	Electricity Generation Data	Reporting Data Quality Assessments	Machine Learning Method
(Sözen et al, 2004)	No	No	ANN
(Colak and Qahwaji, 2007)	No	Yes	Image classification
(Yona et al, 2008)	No	No	NN
(Reikard, 2009)	No	No	ANN
(Huang et al, 2010)	Yes	No	NN
(Kostylev and Pavlovski, 2011)	No	No	-
(Ding et al, 2011)	Yes	No	ANN
(Pedro and Coimbra, 2012)	Yes	Yes	ANN/KNN/GA

Table 2.2 (continued) : Investigation of data nature and quality assessments (Saadati and Barutcu, 2024a).

(Bizzarri et al, 2012)	No	No	-
(Yona et al, 2013)	No	No	NN + Fuzzy theory
(Kühnert et al, 2013)	No	No	Image analysis
(Dambreville et al, 2014)	No	Yes	-
(Hanna et al, 2014)	Yes	No	NN+LSR
(Aggarwal and Saini, 2014)	No	Yes	AR model
(Russo et al, 2014)	Yes	No	Genetic programming
(Gandelli et al, 2014)	Yes	Yes	Hybrid model based on ANN
(Jiménez-Pérez and Mora-López, 2014)	No	No	Clustering data mining
(Lauret et al, 2015)	No	Yes	NN, SVM, GP
(Dolara et al, 2015)	Yes	No	Hybrid ANN
(AlHakeem et al, 2015)	Yes	Yes	NN
(David et al, 2016)	No	No	-
(J. Li et al, 2016)	No	No	Hybrid SVM
(Gala et al, 2016)	No	No	SVR, RF, GB
(Z. Li et al, 2016)	Yes	No	SVR, ANN
(Lou et al, 2016)	No	Yes	BRT
(Aybar-Ruiz et al, 2016)	No	No	ELM
(Y. Li et al, 2016)	No	Yes	-
(Muntean et al, 2016)	Yes	Yes	Data mining
(He Jiang et al, 2017)	No	No	-
(Assouline et al, 2017)	No	Yes	SVR
(Doorga et al, 2019)	No	No	Hybrid NN
(Basurto et al, 2019)	No	Yes	Hybrid NN
(Amiri et al, 2021)	No	Yes	Hybrid NN

As reported by Saadati and Barutcu (2024a), 76% of the reviewed papers did not incorporate power generation data (endogenous inputs), and 67% did not report any data pre-processing evaluations. Figure 2.2 illustrates these findings, which highlight significant gaps in the use of electricity-related data and the reporting of data pre-processing practices within the field of solar energy forecasting.

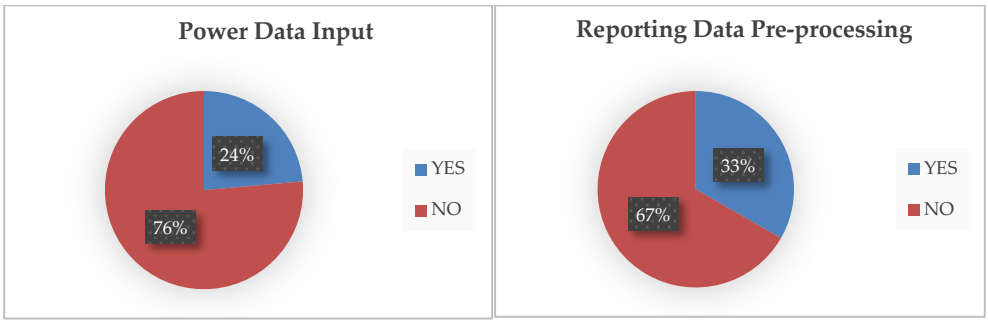

Figure 2.2 : Use of electricity related data and report of data quality (Saadati and Barutcu, 2024a).

The analysis reveals that more than half of the reviewed papers rely on meteorological data or other exogenous inputs, such as electricity market-related data, rather than focusing on electricity-specific variables. These studies predominantly emphasize the direct use of natural solar radiation resources, directing their efforts towards the meteorological aspects of energy conversion rather than the complexities of the electrical infrastructure.

Forecasting future power generation using endogenously generated power data involves modeling nonlinear and artificial parameters, such as panel temperature and operational factors like inverters, which are significantly more complex to predict compared to natural parameters. This complexity makes the direct prediction of electricity output more challenging, with a higher risk of error, and thus less frequently addressed in the literature.

As illustrated in Figure 2.2, only 33% of the analyzed papers reported data quality assessments. This indicates that the issue of data quality has been relatively overlooked by the majority of the reviewed studies. However, the importance of ensuring high data quality is undeniable, as it is essential in enhancing forecasting models' precision and reliability.

2.2 Discussing the Study Gaps

This section explores the underexplored areas and identifies less-focused aspects of solar power prediction from photovoltaics, drawing insights from the reviewed literature and recent studies.

- A significant challenge in solar forecasting lies in the dominance of black-box models, which limit the interpretability of mathematical relationships between input variables and output targets. Therefore, one critical research direction is to develop more transparent and comprehensive explanations of solar forecasting models, enabling a better understanding of their functionality.
- The accuracy of machine learning, deep learning, and ensemble learning models heavily depends on effective data preprocessing techniques. While methods such as filtering, normalization (L.-L. Li et al, 2020), wavelet packet transform (Fu and Wang, 2018), and variational mode decomposition (Zhou et

al, 2022) have been employed to enhance data quality, further advancements in preprocessing are essential to improve prediction accuracy.

- AI models, particularly those utilizing machine learning and deep learning, require extensive, high-quality datasets. However, obtaining comprehensive datasets for solar energy forecasting remains challenging due to the high cost of data collection equipment and limited availability of data for specific regions. Many studies rely on localized datasets, making it difficult to generalize findings across broader geographic areas. Expanding data collection efforts to encompass larger regions could significantly enhance AI models and allow for the application of big data techniques to improve renewable energy predictions.
- The pursuit of more accurate renewable energy forecasting has led to the development of complex AI models, which often result in higher computational costs and longer processing times. Integrating metaheuristic models with machine learning, deep learning, and ensemble learning further exacerbates these challenges. Future studies should aim at optimizing metaheuristic models to enhance computational efficiency, thereby reducing costs and simplifying the implementation of AI-based forecasting systems.
- Another key finding is that most studies prioritize very short-term and short-term forecasts, with limited attention given to medium- and long-term predictions. Greater emphasis on medium- and long-term forecasting is essential for better planning of maintenance schedules and power system operations.
- In selecting evaluation metrics, reliance on popular choices without considering the specific requirements of the forecasting system can lead to suboptimal results. Metrics should be chosen according to factors like forecast periods, geographic location, and the study's objectives (Zhang et al, 2015b).
- While deep learning algorithms show great potential for photovoltaic forecasting, their adoption remains limited. Future research should focus on exploring novel deep learning architectures and algorithms to enhance the precision and reliability of these models.

- Finally, with the growing reliance on renewable energy, probabilistic forecasting is becoming increasingly important for power system operations. Future efforts should prioritize the development of probabilistic forecasting models capable of accurately addressing uncertainties in photovoltaic power generation. The effective incorporation of renewable energy into the grid requires that uncertainty be considered in both decision-making and risk management processes.

3. MATERIALS AND METHODS

The chapter presents the forecasting strategies and analytical procedures employed throughout the thesis. It includes descriptions of the forecasting models, optimization techniques, preprocessing methods, Python libraries, and evaluation metrics employed.

3.1 Forecasting Methods

The forecasting methods outlined below were used to predict solar energy output in this thesis. These methods are categorized into three groups: Linear models, Non-linear models, and Chaotic models. The Non-linear models were selected to include three distinct categories—Machine Learning, Deep Learning, and Ensemble Learning—providing a comprehensive basis for comparison.

3.1.1 Linear models

Linear forecasting models assume a linear relationship between the input variables and the target variable, which makes them specifically efficient for time series data exhibiting clear patterns, such as trends or seasonality. These models are computationally efficient and interpretable, making them widely used in both academic and practical forecasting applications. ARIMA and Exponential Smoothing (ETS) are two of the most prominent linear models (Serana, 2020) that are used as the linear forecasting methods in this thesis study.

3.1.1.1 Auto regressive integrated moving average (ARIMA)

This method introduced by (Box and Jenkins, 1976), is a commonly employed technique for time series forecasting that integrates three key components: autoregression (AR), differencing (I), and moving average (MA).

Autoregressive (AR) part: This component models the current value of the time series as a linear combination of its previous values. The order of the autoregressive model is denoted by p, which represents the number of lagged observations used in the model.

Integrated (I) part: This refers to transforming the series, through differencing, to achieve stationarity. For many time series models, stationarity is critical, as it maintains the stability of statistical properties like mean and variance over time. The integration order d indicates how many times the data are differenced to achieve stationarity.

Moving Average (MA) part: This component models the dependency between the present value and previous residual errors. The moving average order, represented by q, defines the number of lagged forecast errors that are taken into account.

The notation ARIMA(p, d, q) is used, in which p denotes the AR component's order, d the level of differencing, and q the MA component's order. ARIMA is especially effective for time series data that shows trends but does not exhibit seasonal patterns (Hyndman and Athanasopoulos, 2018). The model requires a series to be stationary, and if it is not, differencing is applied to achieve this. In cases where seasonality is present, the seasonal ARIMA (SARIMA) model is used, which includes additional seasonal terms. For ARIMA, parameters can be selected using techniques like the Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC), which aid in identifying the best-fitting p, d, and q values.

3.1.1.2 Exponential smoothing (ETS)

The ETS method (Brown, 1959) is a commonly applied method for time series forecasting that models the level, trend, and seasonality of a time series data by using weighted averages of past observations. ETS models are especially beneficial for time series that show patterns over time, such as trends and seasonality. The key components of the ETS method include:

Level: This is the smoothed estimate of the current value of the series. It represents the baseline level of the time series.

Trend: This component models the underlying direction in the data (upward or downward), reflecting the long-term movement.

Seasonality: If the data exhibits seasonality (periodic fluctuations), this component models those seasonal variations.

The ETS model can be written as ETS (error, trend, seasonality), where each component (error, trend, and seasonality) can take one of three forms:

- Error (E): Additive (A) or multiplicative (M).
- Trend (T): None (N), additive (A), or multiplicative (M).
- Seasonality (S): None (N), additive (A), or multiplicative (M).

This method assigns decreasing weights to older observations, placing more focus on recent data. As a result, it can quickly adjust to shifts in the underlying patterns of the time series (Hyndman and Athanasopoulos, 2018).

3.1.2 Machine learning models

Machine Learning (ML), a subset of artificial intelligence (AI), is centered on creating algorithms that can identify patterns and relationships in data autonomously, without the need for explicit programming. By leveraging statistical methods, ML enables systems to improve their performance over time through experience, making it a powerful tool for tackling complex problems in various domains. ML models are typically categorized into traditional methods, including decision trees and ensemble techniques, and neural network-based approaches that replicate the structure and operation of the human brain. These models are capable of handling complex and nonlinear relationships, making them highly suitable for forecasting tasks. In this thesis, three branches of machine learning models—Shallow Neural Network Models, Deep Learning Models and Traditional ML Models—are employed, each offering unique strengths in capturing intricate patterns in time series data. The implemented shallow neural network models in this study include Feed Forward Neural Network, Extreme Learning Machine, and Echo State Network. Deep learning models include Gated Recurrent Unit and Long Short-Term Memory, and traditional ML models include the ensemble learning models of Random Forest and eXtreme Gradient Boosting.

3.1.2.1 Feed forward neural network (FFNN)

FFNN (Bebis and Georgiopoulos, 1994) is one of the most fundamental types of artificial neural networks, often used as the building block for more complex architectures in machine learning. FFNNs consist of three primary components: an input layer, one or more hidden layers, and an output layer. Information flows in one direction—from the input layer through the hidden layers to the output layer—without forming cycles or feedback loops, making it a non-recurrent network (Goodfellow et

al, 2016). Each layer comprises neurons, or nodes, which are interconnected by weights. These weights are adjusted during training to minimize the error between the predicted and actual outcomes (Rumelhart et al, 1986).

The process of learning in FFNNs relies on backpropagation, a supervised learning algorithm that uses gradient descent to iteratively update weights by calculating the gradient of the loss function (LeCun et al, 2015). Activation functions like ReLU, tanh, and sigmoid introduce non-linearity into the model, allowing it to capture complex and non-linear patterns in the data (Nair and Hinton, 2010). FFNNs are commonly applied to tasks such as regression, classification, and time series forecasting due to their versatility and ability to approximate continuous functions with arbitrary accuracy, as demonstrated by the universal approximation theorem (Hornik et al, 1989).

While FFNNs are powerful, they have limitations, such as difficulty in handling sequential data or temporal dependencies. These limitations have driven the creation of specialized neural network architectures, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). However, FFNNs remain a critical foundation in neural network research and applications across fields like energy forecasting, finance, and healthcare.

3.1.2.2 Extreme learning machine (ELM)

ELM is a learning algorithm designed for single-layer feedforward neural networks (SLFNs) that achieves high efficiency and fast training by randomly initializing weights and biases in the hidden layer. Unlike traditional neural networks, ELM does not iteratively adjust these parameters during training; instead, it focuses on solving a simple linear system to determine the output weights, making it computationally efficient (Huang et al, 2006). This unique approach eliminates the need for time-consuming iterative optimization processes such as backpropagation.

The core idea of ELM is to map the input data into a high-dimensional feature space using a set of non-adjustable hidden neurons and then solve for the output weights using a least-squares solution (Huang et al, 2012). This makes the training process extremely fast compared to conventional neural networks, while still achieving good generalization performance. ELMs can use various activation functions, such as sigmoid or radial basis functions, to enhance their ability to approximate non-linear mappings.

Despite its advantages in speed and simplicity, ELM has some limitations, such as sensitivity to the random initialization of weights and biases, which may affect its performance consistency. However, it has been successfully applied in a wide range of tasks, including classification, regression, and time series forecasting, particularly when computational resources or time are constrained (Zhang et al, 2015a). Its capability and scalability make it effective for applications involving large datasets.

3.1.2.3 Echo state network (ESN)

Neural networks inspired by chaos theory integrate dynamic properties from chaotic systems with learning capabilities of neural architectures, enabling them to model intricate patterns and behaviors. A widely recognized implementation of this approach is the Echo State Network (ESN), introduced by Jaeger (2001), which serves as a foundational model within the reservoir computing (RC) framework. As noted by Shahi et al. (2022), RC builds upon the structure of recurrent neural networks (RNNs) but distinguishes itself through its streamlined and computationally efficient training methodolog. RC optimizes only the output layer's weights, while the reservoir weights and biases are fixed after random initialization throughout training. Despite their simplified structure, ESNs have proven effective in capturing the behavior of chaotic systems and forecasting nonlinear time series, all while maintaining low computational requirement (Bianchi et al, 2017).

The reservoir in an ESN serves as a dynamic memory that projects input data into a high-dimensional space, capturing complex temporal dependencies (Jaeger, 2001). The learning process involves feeding input signals into the reservoir, where the dynamic states of the neurons evolve based on the reservoir's fixed internal connections. Next, a linear regression method adjusts the output weights to minimize the discrepancy between predicted and actual values (Lukoševičius and Jaeger, 2009). While ESNs are highly efficient and suitable for time series forecasting and chaotic system modeling, their performance depends on careful tuning of hyperparameters such as reservoir size and spectral radius.

3.1.3 Deep learning models

Deep learning, a specialized branch of machine learning, employs artificial neural networks with multiple layers to capture and interpret intricate patterns within data. Unlike traditional machine learning algorithms, deep learning models autonomously extract feature representations from raw data, removing the necessity for manual feature engineering (LeCun et al, 2015). These models are especially efficient for tasks that deal with extensive and unstructured datasets, such as images, text, and time series, due to their hierarchical structure that progressively extracts higher-level features (Goodfellow et al, 2016). Popular architectures like Convolutional Neural Networks (CNNs) excel in image recognition (Saadati and Taskin, 2024), while Recurrent Neural Networks (RNNs) and their extensions, such as Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM), are widely used in sequential data modeling (Schmidhuber, 2015). CNN, LSTM and GRU models are employed within this thesis study. The development of deep learning has been driven by advances in computational power, the availability of large datasets, and optimization techniques like backpropagation. It has found applications in diverse fields, including renewable energy forecasting, healthcare, natural language processing, and autonomous systems.

3.1.3.1 Long short-term memory (LSTM)

Long Short-Term Memory (LSTM) networks are a specialized form of recurrent neural networks (RNNs) designed to overcome the vanishing gradient issue found in traditional RNNs, enabling them to effectively learn long-term dependencies in sequential data. LSTMs are equipped with special units known as memory cells, which enable them to retain information over extended periods. Three gates regulate these memory cells: the input gate, forget gate, and output gate. The input gate controls the amount of new information that is incorporated into the memory cell, the forget gate decides what information should be discarded, and the output gate controls the information passed to the next layer (Hochreiter and Schmidhuber, 1997). This architecture allows LSTMs to capture long-range dependencies, making them particularly effective for tasks like time series forecasting, natural language processing, and speech recognition.

LSTMs have gained significant recognition for their capability to capture and retain long-term dependencies, making them a top choice for time series forecasting. As noted by Korstanje (2021), the LSTM cell significantly improves the ability to retain long-term dependencies through the learning of supplementary parameters, which is crucial for data with long-term trends. This capacity makes LSTMs the most powerful form of RNNs for forecasting. Their capability to model both short- and long-term dependencies has led to their widespread adoption in fields such as finance, healthcare, and renewable energy forecasting (Graves, 2013). Despite their advantages, LSTMs are computationally more expensive and complex to train compared to simpler models, requiring careful tuning of hyperparameters such as the number of layers, learning rate, and sequence length. Nevertheless, LSTMs remain state-of-the-art models for sequential data due to their capacity to model intricate temporal dynamics (Korstanje, 2021).

3.1.3.2 Gated recurrent unit (GRU)

Gated Recurrent Units are a variant of Recurrent Neural Networks (RNNs) designed to address the vanishing gradient problem and enhance the modeling of long-term dependencies in sequential data. Introduced by Cho et al. (2014), GRUs simplify the Long Short-Term Memory (LSTM) models' architecture by merging the forget and input gates into a unified update gate, simplifying computations while preserving similar performance. The GRU also features a reset gate, which controls how much of the past information should be forgotten, and an update gate, which determines how much of the new information should be incorporated. This streamlined structure allows GRUs to efficiently learn both short- and long-term dependencies without requiring as much computational power or memory as LSTMs.

GRUs are especially effective for handling temporal data tasks, including natural language processing, time series forecasting, and speech recognition, where understanding temporal relationships is essential (Cho et al, 2014). Compared to LSTMs, GRUs often achieve similar accuracy while being faster to train and requiring fewer resources, which renders them well-suited for scenarios where computational resources are constrained (Chung et al, 2014). However, like LSTMs, GRUs require careful hyperparameter tuning to achieve optimal performance. Despite their simplified architecture, GRUs remain powerful tools for sequence modeling and are

widely used across diverse domains, including renewable energy forecasting, financial modeling, and healthcare analytics.

3.1.3.3 Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are a type of deep learning model specifically designed for analyzing structured grid-like data, such as images or time series. They were introduced by LeCun et al. (1998) and have become a foundational tool in machine learning because of their ability to automatically extract spatial or temporal features from data. The core component of CNNs is the convolutional layer, which uses filters (kernels) to perform convolution operations, detecting patterns like edges, textures, or trends in the input data. Additional pooling layers reduce the spatial dimensions of the data, which helps in reducing computational complexity and preventing overfitting. By stacking convolutional and pooling layers, CNNs progressively learn hierarchical representations of the data, capturing both low-level and high-level features.

While originally designed for image recognition, CNNs have also been successfully adapted for other domains, including time series forecasting, natural language processing, and signal processing (Goodfellow et al, 2016). In time series analysis, 1D CNNs are often employed to extract temporal patterns, leveraging their ability to detect local dependencies in the data. CNNs are computationally efficient and capable of parallel processing, making them faster to train compared to sequential models like RNNs. However, they typically require larger datasets to perform well and may struggle with capturing long-term dependencies without additional architectures like recurrent layers. Despite these challenges, CNNs remain a powerful and versatile tool for various machine learning tasks due to their strong feature extraction capabilities and scalability.

3.1.4 Bagging and boosting ensemble learning models

Ensemble learning models are a powerful approach in machine learning that combine the predictions of multiple individual models to achieve better performance and robustness compared to a single model. The primary idea behind ensemble learning is that aggregating the predictions from diverse models can reduce variance, bias, or improve generalization, depending on the chosen method (Dietterich, 2000). There are

three main categories of ensemble methods: bagging, boosting, and stacking that are all implemented within this thesis study. Bagging, exemplified by algorithms like Random Forest, focuses on reducing variance by training multiple models on different subsets of the data and averaging their outputs (Breiman, 2001). Boosting, as used in models like XGBoost, reduces bias by sequentially training models and correcting errors made by previous ones (Friedman, 2001). Stacking combines predictions from multiple models using a meta-model to improve overall accuracy, which is further explained in section 3.2. Ensemble models are widely used in applications such as classification, regression, and time series forecasting due to their ability to enhance predictive performance and handle complex datasets with high variability.

3.1.4.1 Random forest (RF)

Random Forest is a bagging ensemble learning technique commonly employed for classification and regression tasks. It builds multiple decision trees during training and depending on the task, for regression or classification outputs the average prediction or the majority vote respectively. Each tree in the forest is trained using a random subset of the data, which helps to reduce overfitting and improve the model's capacity to generalize (Breiman, 2001). The random selection of features at each split in the decision trees further ensures diversity among the trees, which enhances the model's general performance level. One of the key strengths of RF is its ability to handle large datasets with high-dimensionality and its robustness to noisy data and outliers (Liaw and Wiener, 2002).

RF's effectiveness can be attributed to its ability to perform well without extensive hyperparameter tuning, making it accessible for practical applications. It also provides useful insights into feature importance, which can guide feature selection in predictive modeling. The model can handle both quantitative and qualitative data types and is extensively applied across various fields, including finance, bioinformatics, and energy forecasting (Cutler et al, 2007). However, RF models tend to be computationally intensive and less interpretable compared to simpler models like decision trees. Despite this, their high predictive power, ease of use, and flexibility have made RF a popular choice in machine learning tasks.

3.1.4.2 Extreme gradient boosting (XGBoost)

Extreme Gradient Boosting is an advanced ensemble learning method based on gradient boosting, designed to optimize both performance and computational efficiency. It constructs a series of decision trees in a sequential manner, where each subsequent tree attempts to correct the errors made by the previous one. XGBoost enhances the traditional gradient boosting method by introducing regularization terms to the loss function, which helps prevent overfitting and improves model generalization (Chen and Guestrin, 2016). The model employs a "boosting" technique where weak learners (usually shallow decision trees) are added iteratively, and the prediction accuracy is enhanced by focusing on the residual errors of prior models. This approach has made XGBoost one of the top-performing algorithms in various machine learning challenges and practical applications.

A major benefit of XGBoost is its efficiency in handling large datasets due to its parallelization capabilities, which allow for faster training times compared to traditional gradient-boosted models (Chen et al, 2015). Additionally, XGBoost supports a wide range of hyperparameters, providing flexibility for fine-tuning the model to optimize performance for specific tasks. The algorithm is robust to noise and missing data, and its predictive power is particularly strong in classification, regression, and ranking tasks. It has become widely popular in fields such as finance, healthcare, and energy forecasting due to its accuracy and scalability (Caruana and Niculescu-Mizil, 2006). However, the model's complexity can make it more challenging to interpret compared to simpler models like decision trees.

3.2 Stacking Ensemble Machine Learning

Stacking is an advanced ensemble learning method that aggregates predictions from several base models to improve the overall predictive accuracy of a machine learning system. Unlike bagging and boosting, which rely on aggregating predictions or correcting errors in sequence, stacking focuses on integrating the strengths of diverse models by using a meta-model to learn from their outputs. This approach is particularly beneficial when individual models have strengths and weaknesses that complement each other, as the meta-model can effectively leverage these differences to make more accurate predictions (Wolpert, 1992).

The stacking process generally involves two main steps. In the first stage, multiple base models, such as neural networks, support vector machines, or decision trees, are trained independently on the training data. These models may vary significantly in architecture or hyperparameters to ensure diversity in their predictions. In the second stage, the predictions of these base models, often referred to as level-1 predictions, are combined and used as input features for a meta-model, also known as a level-2 model. The meta-model learns to predict the target variable based on these aggregated predictions, effectively capturing relationships that may not be apparent to any single base model (Ting & Witten, 1997).

One of the key advantages of stacking is its flexibility. Unlike bagging or boosting, which are limited to specific algorithms or strategies, stacking can integrate any type of base models, including traditional machine learning algorithms and deep learning architectures. For example, in time series forecasting, stacking might combine predictions from linear regression models, tree-based algorithms, and recurrent neural networks to capture both linear trends and nonlinear dependencies in the data (Zhou, 2012). However, this adaptability results in increased computational complexity, as stacking requires training multiple models and a meta-model.

To ensure the robustness of the stacking ensemble, train-test splitting techniques are very important to be employed during the first stage, helping to generate out-of-sample predictions for the training of the meta-model. These techniques help to prevent overfitting and ensure that the meta-model generalizes well to unseen data (Breiman, 1996). Despite these additional computational requirements, stacking has been widely adopted in real-world applications, including fraud detection, image recognition, and renewable energy forecasting, due to its ability to handle complex datasets and achieve state-of-the-art performance. The implementation of the stacking ensemble machine learning and the proposed meta-model by this thesis study is explained in Chapter 6.

3.3 Optimization and Fine-Tuning

Bayesian optimization is an advanced technique for optimizing objectives that are computationally expensive, lack a clear functional form, or are subject to noisy evaluations (Garnett, 2023). This method is particularly well-suited for scenarios such as hyperparameter tuning in machine learning models, engineering design optimization, or experimental sciences, where each function evaluation requires

significant time or resources. Among various Bayesian optimization methods, the Tree-structured Parzen Estimator (TPE) demonstrates outstanding effectiveness across a wide range of use cases (Watanabe, 2023).

The TPE algorithm, introduced by Bergstra et al. (2011), is a specialized form of Bayesian optimization that models the objective function using probabilistic distributions. Unlike traditional Gaussian Process (GP)-based Bayesian optimization methods, which assume a prior distribution over the entire objective function, TPE focuses on modeling two conditional distributions: The density of hyperparameter configurations that yield results better than a predefined threshold, and the density of all remaining configurations. This separation allows TPE to replace the global surrogate model used in GP-based optimization with more localized and flexible density estimators, such as Parzen windows or kernel density estimation (Bergstra et al, 2011).

A key characteristic of TPE is its capacity to manage tree-structured search spaces, where hyperparameter configurations depend on conditional choices. For example, in deep learning model optimization, some hyperparameters (e.g., the learning rate) might depend on the optimizer type, and others (e.g., dropout rate) might depend on the network architecture. TPE excels in such scenarios, making it a preferred choice for hyperparameter tuning tasks (Bergstra et al, 2013).

The method has achieved significant milestones in machine learning research and practice. Notably, In hyperparameter optimization for deep learning, TPE has been widely used, often leading to top-ranking outcomes in Kaggle contests (Watanabe and Hutter, 2022). Moreover, the TPE-based approach played a crucial role in securing the first place in the 2022 AutoML contest focused on multiobjective hyperparameter tuning for Transformer models, where it was employed to optimize large-scale deep learning models (Watanabe et al, 2022). Its effectiveness is due to its ability to balance exploration and exploitation, while efficiently managing computationally costly evaluations.

In addition to its performance benefits, TPE is computationally efficient and easy to implement, making it accessible for both academic research and industry applications. The approach is widely adopted in popular optimization libraries such as Hyperopt, which provides a practical implementation of TPE for various machine learning

frameworks (Bergstra et al, 2015). This accessibility, combined with its effectiveness, has established TPE as a cutting-edge approach to hyperparameter optimization and other challenging optimization tasks.

3.4 Preprocessing Methods

An essential step in every area of data analytics is Pre-processing, which plays a crucial role in ensuring the reliability and accuracy of subsequent modeling and analysis. This study employs two distinct pre-processing techniques to address key data challenges. First, the Local Outlier Factor (LOF) technique is applied during the data cleaning stage to identify and remove outliers, thereby preserving the dataset's integrity before any analysis or implementation of forecasting models. Second, the False Nearest Neighbors (FNN) method is utilized in the chaotic analysis section to effectively embed the time series data and determine the optimal embedding dimension. Detailed explanations of these methods are provided in the following sections.

3.4.1 Local outlier factor

The Local Outlier Factor (LOF) method, introduced by Breunig et al. (2000), is a density-based algorithm used for detecting outliers in datasets by analyzing the local neighborhood density of data points. Unlike global outlier detection methods, which may overlook local irregularities, LOF focuses on identifying deviations in density at a local scale. It calculates an outlier score for each data point by comparing its local reachability density to the average local reachability density of its nearest neighbors. The local reachability density is determined based on the distance between the point and its neighbors, considering the density of the surrounding region. Points with significantly lower densities than their neighbors are assigned higher LOF scores, indicating a greater likelihood of being outliers. This localized approach makes LOF particularly useful in datasets with varying density distributions, where global methods might fail to identify anomalies effectively.

A major advantage of the LOF method is its capability to identify both global and local outliers, making it versatile for various applications such as fraud detection, intrusion detection, and quality control in manufacturing (Aggarwal, 2017). This method is a powerful and widely used technique in anomaly detection, providing robust results in datasets with complex structures and varying densities (Chandola et al., 2009).

3.4.2 False nearest neighbors

False Nearest Neighbors (FNN), a method introduced by Kennel et al. (1992), serves to determine the best embedding dimension for rebuilding the time series' state space. In time series analysis, reconstructing the state space with the correct embedding dimension is crucial to uncover the underlying dynamics of the system. The FNN method is especially useful for evaluating if a reconstructed state space, using a given embedding dimension, faithfully captures the original time series structure. A low embedding dimension can cause points that look close together in the reconstructed space to be incorrectly identified as neighbors, even though they are distant in the original space. To detect false neighbors, the FNN method compares neighbor relationships in the reconstructed and original time series spaces, facilitating the identification of the smallest embedding dimension that accurately reflects the data's inherent structure (Kennel et al, 1992). Therefore, the FNN method is mainly employed to determine the optimal embedding dimension; however, it also involves the process of embedding the time series in a higher-dimensional space. As a result, FNN can be viewed as a technique for embedding time series data.

The FNN technique involves embedding the time series data into higher-dimensional spaces using different embedding dimensions and examining the distance between neighboring points in the reconstructed space. As the embedding dimension increases, the percentage of false nearest neighbors generally decreases, signifying a more accurate representation of the system's dynamics. By identifying the appropriate embedding dimension, FNN helps avoid the issue of under-embedding, where the dynamics of the system are not fully captured. This method is particularly useful in chaotic time series analysis and has applications in areas such as forecasting, anomaly detection, and system identification, where accurate data representation is crucial for effective modeling (Hegger et al, 1999).

3.5 Post-processing Technique (Model Output Statistics)

In the context of forecasting, post-processing involves applying additional steps to enhance the accuracy of forecasts after they have been initially generated. A simple example can illustrate this concept: if a forecasting system consistently produces predictions that are 10% higher than the observed values, the forecaster could adjust the output by reducing it by 10%. Although real-world situations are much more

complex, the core idea remains unchanged as forecasters adjust predictions based on prior knowledge or informed insights. These insights may come from analyzing historical data or relying on subjective judgment. In physical sciences, data-driven adjustments are predominantly used, whereas in social forecasting, subjective judgment often plays a significant role (Yang and Meer, 2021).

One of the first efforts to improve solar forecasts derived from Numerical Weather Prediction (NWP) models was conducted by Lorenz et al. (2009), who explored a widely recognized method known as Model Output Statistics (MOS). Glahn and Lowry (1972) were the first to introduce this term. The core idea behind MOS involves representing the bias in NWP forecasts using a regression model.

Several models can be utilized for MOS depending on the particular problem, the characteristics of the meta-features, and the dataset. These models range from simple techniques like linear regressions to more advanced non-linear methods such as Support Vector Regression (SVR), Neural Networks, and Gradient Boosting. Every approach has its distinct advantages and is best suited for specific scenarios. For instance, linear models like Ridge or Lasso are effective for managing multicollinearity and performing regularization, while non-linear models excel at capturing complex relationships among meta-features. The selection of the MOS model largely depends on factors such as the dataset's characteristics, the nature of the relationships between meta-features, and the desired balance between computational efficiency and model interpretability.

In this thesis, Ridge regression was employed as the MOS technique during the design refinement of the ensemble stacking meta-models implementation, as described in detail in Chapter 6. Ridge regression was chosen primarily due to its ability to handle multicollinearity among the predictions of base models effectively, which is a common issue in stacking ensembles. By introducing an L2 regularization penalty, Ridge regression minimizes the risk of overfitting and ensures that the meta-model remains robust and stable, even when dealing with potentially redundant or highly correlated base models. Furthermore, Ridge regression is computationally efficient and easily interpretable, making it a practical choice for optimizing the ensemble stacking

framework. This combination of theoretical robustness and computational simplicity aligns well with the objectives of the stacking meta-models designed in this study.

3.6 Evaluation Metrics

In this study, the performance of all implemented forecasting models was evaluated using five widely recognized evaluation metrics: R-squared (R^2), Root Mean Square Error (RMSE), Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). These metrics collectively provide a comprehensive assessment of the accuracy and reliability of the forecasting models. Below, each metric is briefly explained.

3.6.1 R-squared (R^2)

R-squared, also known as the coefficient of determination, quantifies the proportion of variance in the dependent variable that can be explained by the independent variables. It ranges from 0 to 1, where higher values indicate superior model performance. An R^2 value near 1 indicates that the model accounts for the majority of the variability in the observed data, while a value closer to 0 suggests poor predictive accuracy. This metric is defined in equation (3.1).

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.1}$$

3.6.2 Root mean square error (RMSE)

RMSE, a widely recognized evaluation metric, is determined by taking the square root of the average squared differences between predicted and observed values. RMSE assigns greater weight to larger errors over smaller ones, making it especially sensitive to outliers. A lower RMSE value indicates higher forecasting accuracy. This metric is determined by equation (3.2).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.2}$$

3.6.3 Mean squared error (MSE)

MSE is the mean of the squared differences between the predicted and observed values. Like RMSE, it emphasizes larger errors due to the squaring process. MSE is commonly used to compare different models, as it offers a comprehensive assessment of model performance. However, its values depend heavily on the data's scale, with larger values indicating worse performance. MSE is calculated using equation (3.3).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.3}$$

3.6.4 Mean absolute error (MAE)

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It is calculated as the mean of the absolute differences between predicted and actual values. MAE provides an easy-to-interpret measure of forecast accuracy, with smaller values indicating better performance. Unlike RMSE and MSE, MAE does not disproportionately penalize larger errors. This evaluation metric is calculated using equation (3.4).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.4}$$

3.6.5 Mean absolute percentage error (MAPE)

MAPE quantifies forecast errors relative to the observed values as a percentage, providing a scale-independent accuracy metri. It is calculated as the mean of the absolute percentage errors for each observation. MAPE is especially valuable for comparing model performance across datasets that have varying scales. However, it can be sensitive to very small actual values, which may result in disproportionately large percentage errors. MAPE is defined by equation (3.5).

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{3.5}$$

3.7. Python Libraries

The following Python libraries were utilized in this study, each serving specific purposes in the implementation and evaluation of forecasting models. This comprehensive suite of Python libraries ensured the effective implementation, evaluation, and optimization of the models and methods described in this thesis.

3.7.1 Core libraries

- NumPy

Used for numerical computations, including matrix operations, array manipulations, and generating random distributions for hyperparameter optimization. Example use: Efficient numerical operations in models like FFNN, ELM, and ensemble stacking.

- Pandas

Facilitated data manipulation, reading input files, preprocessing time-series data, and handling structured datasets. Example use: Loading datasets, creating lag features, and setting datetime indices.

- Matplotlib

Used for data visualization, enabling exploratory data analysis (EDA) and result presentation. Example use: Plotting trends and distributions across various models.

- Seaborn

Enhanced data visualization with statistical graphics and aesthetic themes. Example use: Creating heatmaps, scatterplots, and applying thematic styles.

- Math

Provided basic mathematical operations, such as calculating square roots for evaluation metrics like RMSE.

- OS

Assisted in file path management and system-level operations.

3.7.2 Machine learning and statistical libraries

- Scikit-learn

Supported preprocessing, evaluation, and model development tasks. Data preprocessing: Scaling and normalization (e.g., MinMaxScaler). Model evaluation: Metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), R^2 Score, and Mean Absolute Percentage Error (MAPE). Model implementation: Random Forest and stacking models (e.g., Ridge and Lasso regression).

- Hyperopt

Used for Bayesian optimization of hyperparameters using the Tree-structured Parzen Estimator (TPE). Example use: Efficient tuning of hyperparameters for FFNN, ELM, and ensemble models.

- Optuna

Performed hyperparameter optimization for neural network-based stacking models using TPE. Example use: Optimizing parameters like learning rates, layer sizes, and activation functions.

- SciPy

Provided advanced mathematical tools, such as pseudoinverse computations (e.g., in ELM). Example use: Utilizing the linalg.pinv() function for calculating output weights.

- XGBoost

Implemented and trained XGBoost regression models for high-performance ensemble learning.

- Scikit-neighbors

Used for calculating distances in phase space during False Nearest Neighbors (FNN) analysis. Example use: Assessing embedding dimensions in chaotic time-series analysis.

3.7.3 Deep learning libraries

- TensorFlow/Keras

Core framework for developing and training deep learning models (e.g.,LSTM, GRU, and stacking metamodels). Keras.layers: Added dense, dropout, and recurrent layers. Keras.optimizers: Provided optimizers like Adam and RMSprop. Callbacks: Included tools like EarlyStopping to prevent overfitting.

4. DATASET

This research utilizes data obtained from the Konya Ereğli Solar Power Plant, which has a capacity of 9.20 MWp. The facility was developed by the company "Tegnatia Enerji" and is located in the Ereğli district of Konya, Turkey. It has been producing energy and feeding the electricity grid continuously since its commissioning in April 2016.

The dataset consists of two main components. The first component contains endogenous data, specifically the energy production time series collected from eight inverters at the solar power plant. These values, logged every 10 minutes, cover the timeframe between June 6, 2021, and December 8, 2022, and are used as the target variable in all forecasting models applied throughout this research. These values are shown in figure 4.1. The second part consists of exogenous meteorological data, specifically the solar cell temperature and ambient temperature, also recorded at 10-minute intervals during 2021 and 2022. These measurements are shown in figure 4.3.

The main objective of this research is to predict energy production, designated as the target variable, by using different sets of endogenous and exogenous inputs in the forecasting models. The problem is structured as a multivariate input with a univariate output time series, where weather-related and time-based features act as exogenous variables. Utilizing this multivariate framework allows the study to capture valuable information from multiple related input series, thereby improving the comprehension of the inherent patterns and dynamics within the data (Brownlee, 2018).

4.1 Data Cleaning and Preprocessing

The energy generation data collected for this study was preprocessed using Python programming. The data preprocessing phase included addressing gaps in the dataset, standardizing data scales, and detecting as well as eliminating anomalous data points using the Local Outlier Factor method. These steps guaranteed that the dataset was properly prepared and ready for analysis, as illustrated in Figure 4.1. All data analysis and preprocessing tasks were performed using Python version 3.10.13. To simplify the

computations for this case study, data from only one of the eight inverters was chosen, specifically the energy production recorded by inverter 4, as illustrated in Figure 4.2.

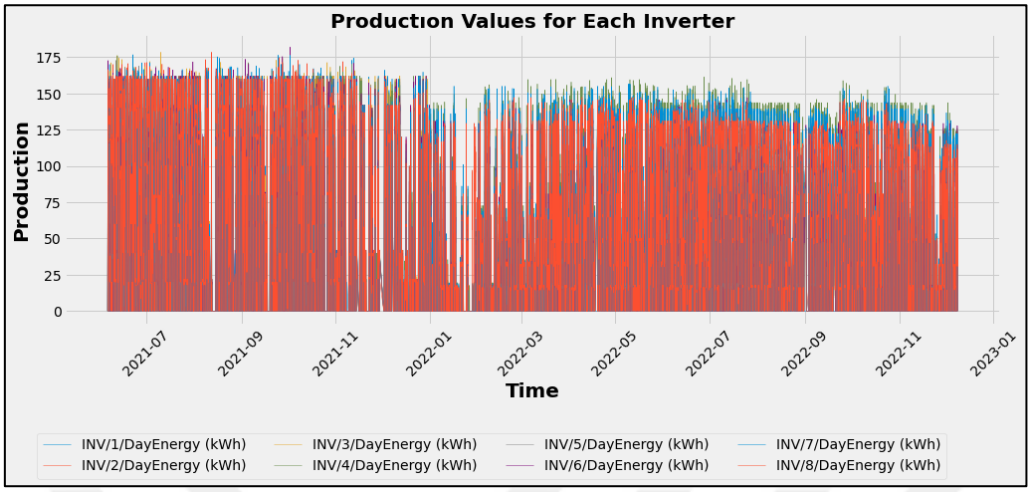

Figure 4.1 : Cleaned data.

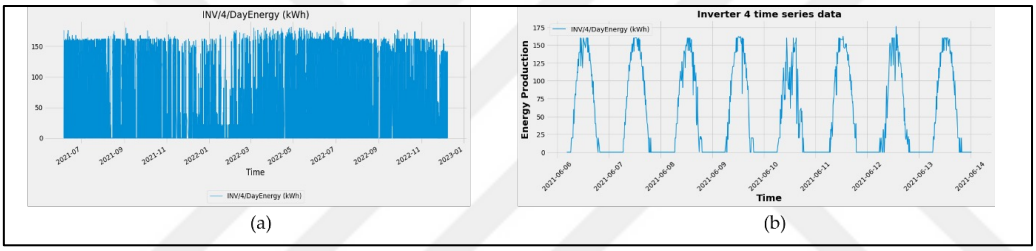

Figure 4.2 : Inverter 4 energy production data. (a) Complete dataset containing 70,973 data points. (b) A subset showing 1,000 data points.

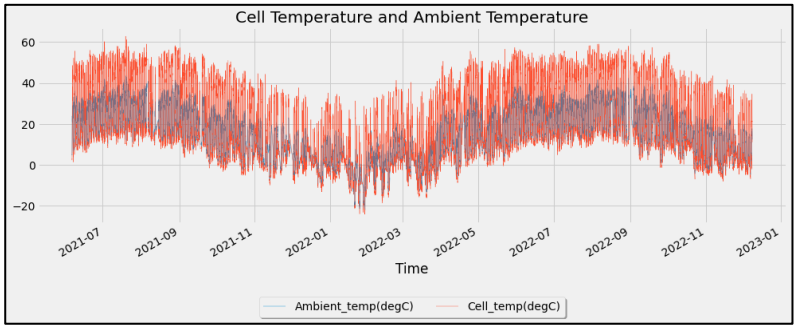

Figure 4.3 : Cell temperature and ambient temperature data.

4.2 Exploratory Data Analysis

The exploratory data analysis (EDA) focuses on uncovering the key characteristics of the time series data. This involves analyzing aspects such as autocorrelation,

seasonality, and stationarity. Insights gained from these analyses guide the selection of appropriate tests and modifications needed to prepare the data for modeling.

Visualization plays a crucial role in EDA, as plotting the data can reveal important features that may not be readily detected using purely numerical techniques. For instance, the seasonality of the data is evident in the visualizations, such as Figure 4.2, which clearly illustrates the daily seasonality exist in the energy production time series data.

The interval between successive peaks in the autocorrelation plot, as shown in Figure 4.4, indicates the seasonal period present in the dataset. This observation aligns with the production plots, which clearly demonstrate a daily seasonality. By measuring the distance between two consecutive peaks, the calculated interval is approximately 86,400 seconds, equivalent to 24 hours, confirming the seasonal length of the data.

Additionally, Figure 4.4 illustrates a gradual decrease in the amplitude of the autocorrelation plot over time. This trend reflects the variation in daylight hours throughout the year. While the seasonal length remains constant at 24 hours, the duration of sunlight changes as the seasons progress. As winter approaches, the days become shorter, and the shifts in sunrise and sunset times across consecutive days contribute to the observed shape of the autocorrelation plot.

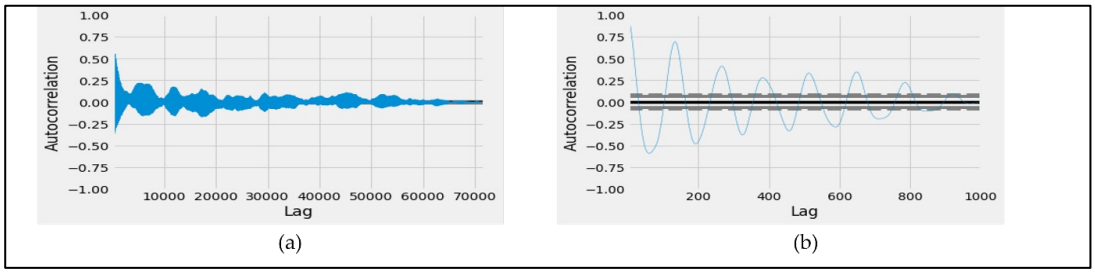

Figure 4.4 : Autocorrelation plot. (a) Entire 70973 datapoints. (b) 1000 datapoints.

Another critical aspect assessed during the Exploratory Data Analysis is stationarity. To evaluate this, Cheung and Lai (1995) presented the Augmented Dickey-Fuller approach was performed using the Statsmodels library in Python (Seabold & Perktold, 2010). The results of the test, which showed a highly negative test statistic and a very low p-value, provide strong evidence to reject the null hypothesis of a unit root. This implies the series is probably stationary, meaning differencing might be unnecessary.

4.3 Signal Processing

This section analyzes the energy production data in the frequency domain by utilizing Power Spectral Density (PSD), Cross Power Spectral Density (CPSD), Cross Correlation Function, and Coherence plots. These tools offer a deeper understanding of the periodic behaviors and correlations within the data.

The PSD and CPSD plots, shown in figure 4.5, are particularly useful for determining the signal's period. In the frequency domain, the period can be calculated as the reciprocal of the frequency component. The analysis confirms that the data exhibits one dominant periodic event—24 hours—indicating daily seasonality, as previously discussed. Furthermore, the harmonics observed in the PSD and CPSD plots reflect the variations in daylight hours throughout the year.

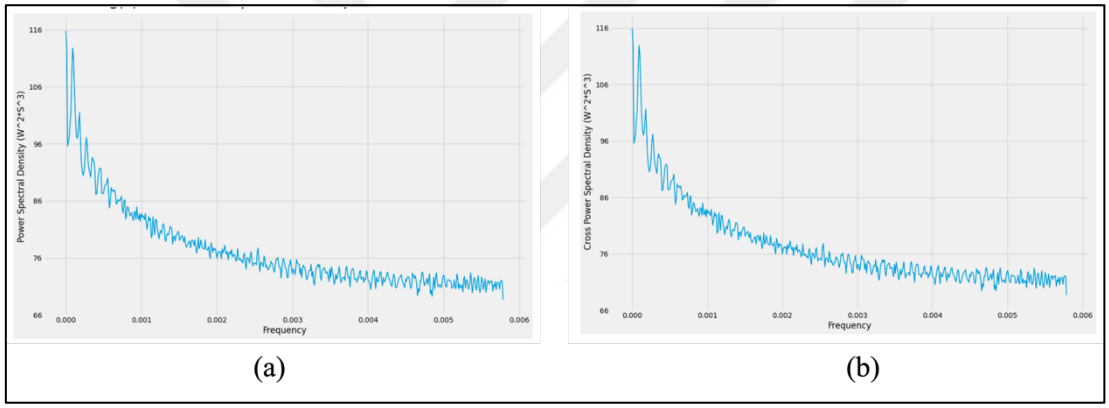

Figure 4.5 : PSD and CPSD. (a) Power spectral density. (b) Cross power spectral density.

In the Cross Correlation plot in figure 4.6, Inverter#5 is taken as the variable being shifted (X), while another inverter (Y) remains stationary. The coefficients to the left of zero indicate instances where X leads and Y lags, while those to the right show when Y leads and X lags. The plot reveals that the highest positive correlation coefficient, +1, occurs at zero lag, meaning there is perfect positive correlation between Inverter#5 and Inverter#6 at each time step. This result aligns with earlier observations from the production plots, which indicated that the inverters exhibit

nearly identical behaviors, suggesting a strong correlation between their data. Coherence plot is demonstrated in figure 4.7.

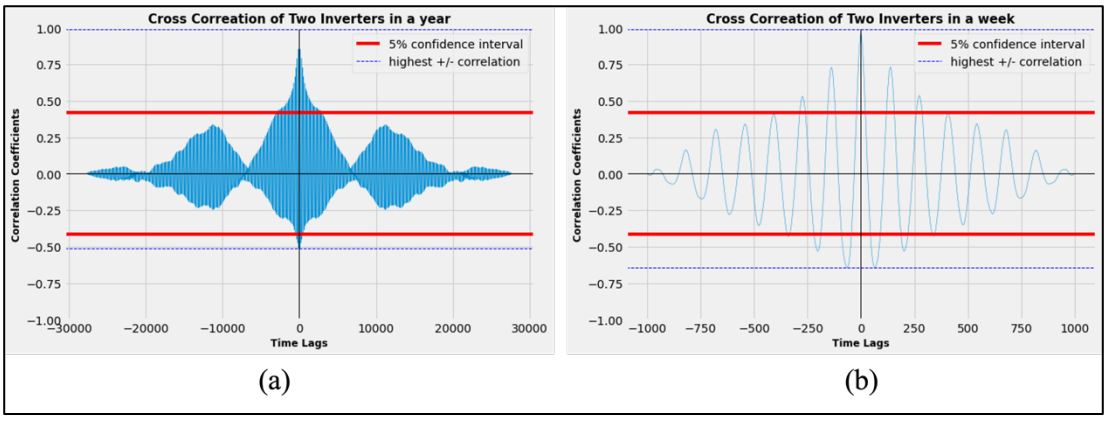

Figure 4.6 : Cross correlation function. (a) The entire dataset. (b) For 1000 data points.

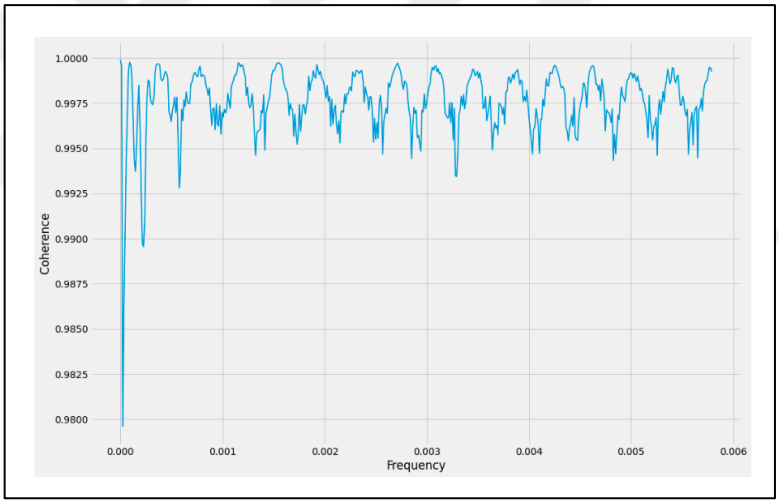

Figure 4.7 : Coherence plot.

4.4 Feature Engineering

Feature engineering refers to the process of transforming unprocessed data into meaningful input features that enhance the predictive power and performance of machine learning models. It involves creating new features, selecting the most relevant ones, or modifying existing ones based on domain knowledge and data characteristics. Effective feature engineering can significantly improve the model's ability to capture underlying patterns in the data, leading to better results. In time series data analysis, feature engineering often involves creating new features that capture temporal patterns

and trends. These techniques enhance the model's ability to interpret time-based dependencies and improve forecasting accuracy.

In this study, feature engineering has an essential role in preparing the time series data for various forecasting models, ensuring that the models could effectively capture temporal dependencies and external influences. The key feature engineering steps across the different models are listed below. It is important to note that steps 1 to 6 are dedicated to the implementation of the base models, while steps 7 and 8 pertain to the implementation of the meta-models.

1- Lag Features: Lagged versions of the target variable (energy production) are created to capture temporal dependencies. For most models, multiple lag features (ranging from lag1 to lag9) are generated to account for past values and enhance predictive performance.

2- Exogenous Variables: Ambient temperature and cell temperature are consistently integrated as additional input variables. These exogenous factors are critical for improving model accuracy by incorporating external influences on the target variable, such as environmental conditions affecting solar energy production.

3- Data Rescaling and Normalization: Several models apply rescaling techniques to normalize the input data, ensuring that features are on a comparable scale and improving model convergence.

4- Windowing and Sliding Window Techniques: In models like LSTM and GRU, a sliding window technique is applied to convert the time series data into a supervised learning format, where each input sequence contain a fixed number of time steps. This helps capture temporal patterns and dependencies over time.

5- Time-Based Features: Time-related features (e.g., hour of the day, day of the week, month, year) are extracted from the dataset index to reflect the seasonal and cyclical trends present in the time series. These features are demonstrated in Figures 4.8, 4.9, and 4.10.

6- Data Splitting: A consistent data splitting strategy is employed across all models, typically dividing the data into training, validation, and test sets. This

ensures robust model evaluation and prevents overfitting and is explained in detail in the Section 4.5 of this chapter.

7- Stacking Predictions from Base Models: The outputs from the base models are combined into a single feature matrix, aggregating diverse predictions to form the input for the meta-models.

8- Weighting Base Model Outputs: A weight vector is applied to adjust the influence of each base model's prediction, based on its R^2 score ranking.

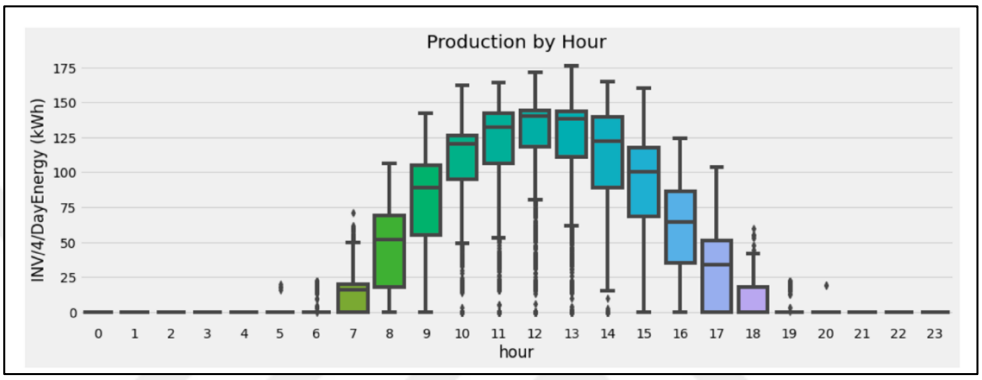

Figure 4.8 : Changes in energy production data based on the Hour feature.

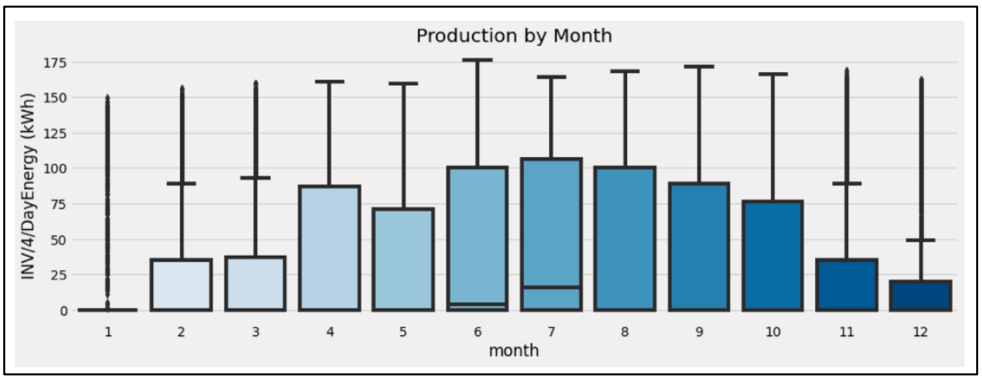

Figure 4.9 : Changes in energy production data based on Month feature.

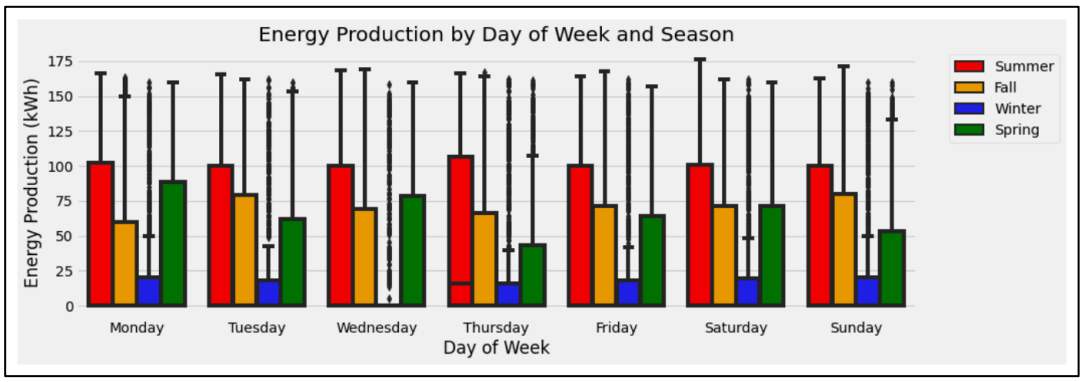

Figure 4.10 : Changes in energy production data based on the Day of Week feature.

For all the base models implemented in this study, the same feature engineering steps were applied during the initial stages of modeling. The forecasting accuracy of the models was evaluated based on these features, and the features that provided the best accuracy for each model were selected accordingly. One example of this is the structure of exogenous variables in the input features. The inclusion of time-based features initially reduced the forecasting accuracy for the neural network and deep learning base models. However, incorporating these features as exogenous variables improved the accuracy of the RF and XGBoost models. Based on these initial evaluations and results, feature selection was performed for each base model in the early stages, and the optimal feature structures are shown in Figures 4.11 and 4.12. Consequently, the selected feature structures were used for each base model in the remainder of the analysis and study.

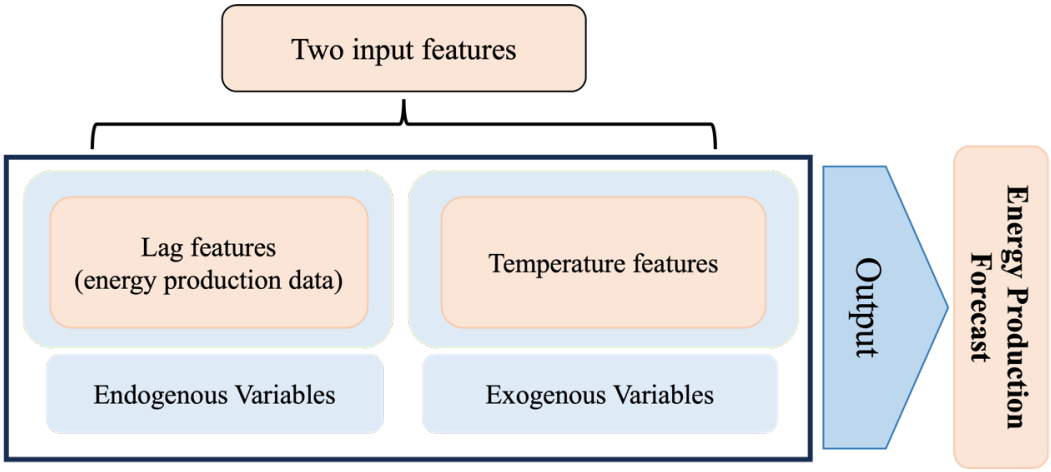

Figure 4.11 : Input features structure for NN and DL base models.

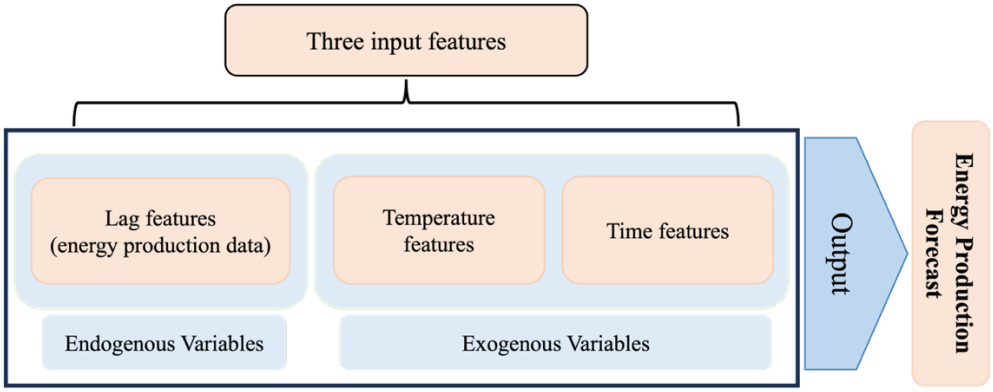

Figure 4.12 : Input features structure for bagging and boosting base models.

For the ensemble stacking meta-models, the outputs from the seven base models are combined into a single feature matrix. This aggregation of diverse predictions forms the input for the meta-model, effectively performing feature engineering by stacking the predictions from the base models to create a new, enhanced feature set. This structure is depicted in Figure 4.13. In certain implementations of the meta-models, weights are applied to the stacked predictions to modify the influence of each base model, based on its R^2 score ranking. The structures of the meta-models implemented in this study are presented and explained in detail in Chapter 6.

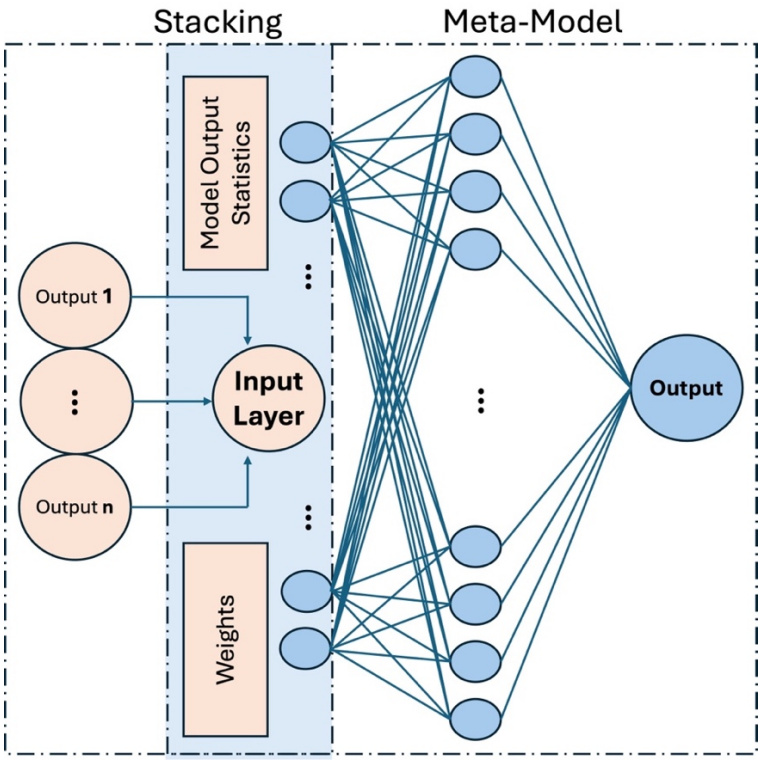

Figure 4.13 : Features structure for meta-models.

4.5 Train-Test Split

In this section, the methodology for segmenting the dataset into training, validation, and test sets is described, ensuring a robust and leakage-free approach to model evaluation. The process involved a careful division of the data to cater to the requirements of both the base models and the meta-model used in the stacking ensemble framework.

For the base models, 70% of the total dataset was allocated for training, while 5% of the remaining data was used for validation. The remaining 25% was divided into three sections: the test set for base models, the validation set for the meta-model, and the test set for the meta-model. This segmentation, presented in figure 4.14, was designed with strict attention to avoiding data leakage, ensuring that no information from the test or validation sets leaked into the training phase.

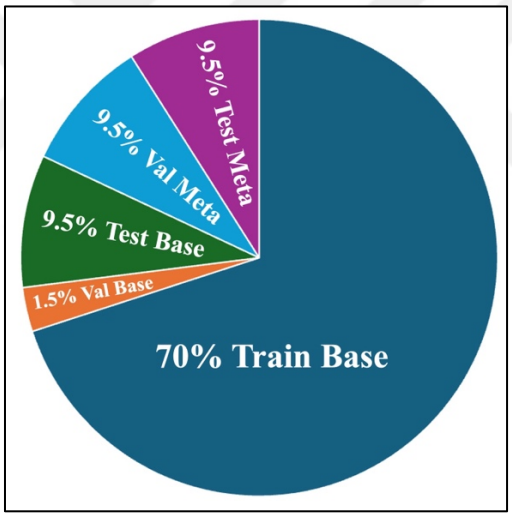

Figure 4.14 : Train-test split for base models and meta-model.

When working with time series data, maintaining the chronological order of data during segmentation is critical. The temporal sequence of the data must be preserved to reflect real-world scenarios where future data cannot influence past predictions. For instance, the test set must always follow the validation set in time, and the validation set must follow the training set. To address this, a Time-Based Split was employed, where older data was used for training and newer data for testing. This method ensures that the order of data appearance is respected throughout the segmentation process. An additional consideration in this study arises from the use of a stacking ensemble and the training of a meta-model. Since the outputs of the base models are the inputs

to the meta-model, the chronological order of data is especially crucial in this context. Consequently, for the meta-model, the validation and test data sets' time indices must align with those of the test sets for the base models, as illustrated in Figure 4.16. This setup ensures that the time index of the meta-model's inputs precedes the time index of its outputs, thereby preventing any form of temporal data leakage. This Time-Based Split is demonstrated in figure 4.15.

The specific segmentation of the data is as follows:

- Training set for base models: June 6, 2021, to June 27, 2022
- Validation set for base models: June 27, 2022, to July 5, 2022
- Test set for base models: July 5, 2022, to August 25, 2022

Since the inputs of the meta-model are the prediction outputs of the base models, the training set for the meta-model corresponds to the test set of the base models:

- Training set for meta-model: July 5, 2022, to August 25, 2022
- Validation set for meta-model: August 25, 2022, to October 18, 2022
- Test set for meta-model: October 18, 2022, to December 8, 2022

This segmentation strategy guarantees the integrity of the data and the reliability of the results by adhering to the temporal structure of the time series. The clear separation between training, validation, and testing phases for both the base models and the meta-model provides a robust framework for evaluating the performance of the proposed approach.

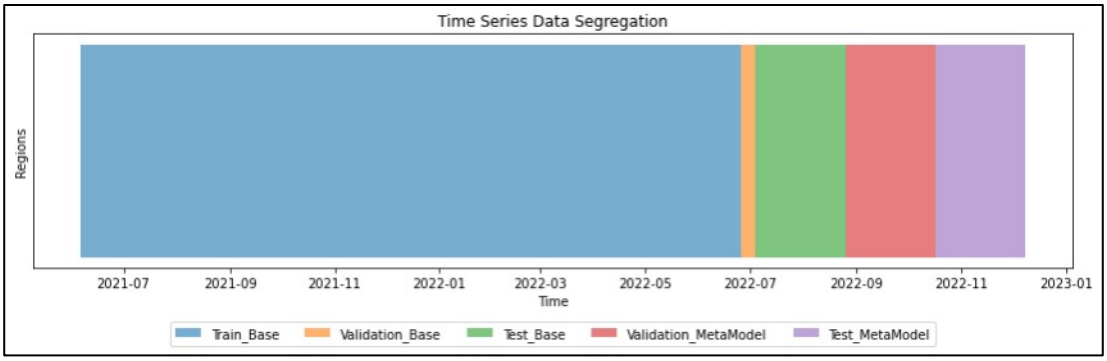

Figure 4.15 : Time series data segregation considering stacking ensemble.

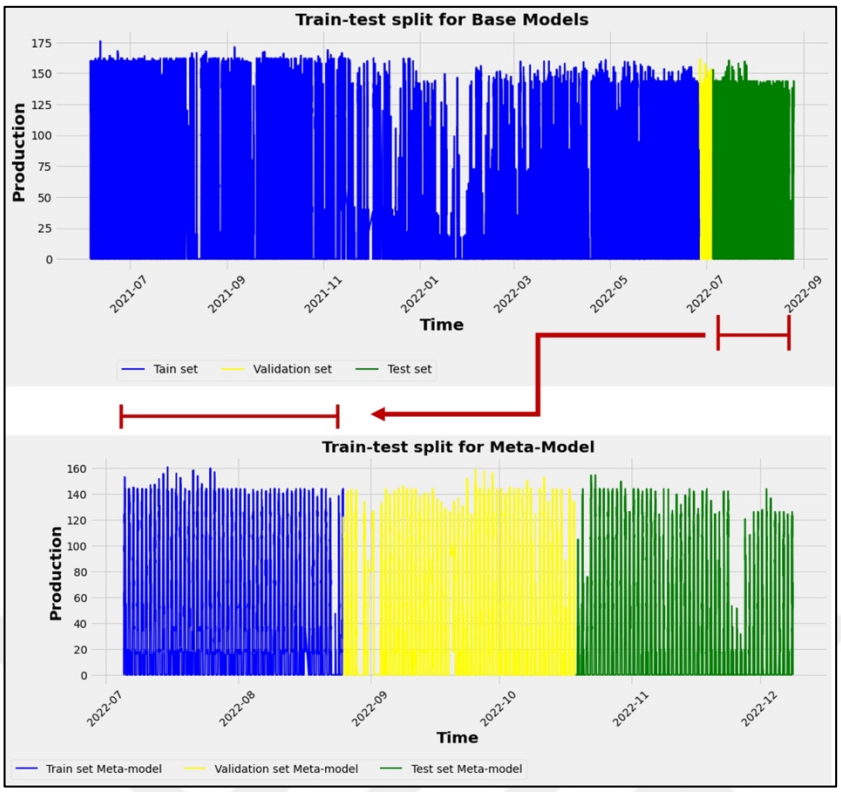

Figure 4.16 : Time-based split for base models and meta-model.

5. EMPIRICAL EVALUATION OF BASE MODELS

In the initial phases of this study, ten different forecasting algorithms were implemented and evaluated across more than twenty distinct scenarios. These algorithms included Auto Regressive Integrated Moving Average (ARIMA), Exponential Smoothing (ETS), Feedforward Neural Network (FFNN), Extreme Learning Machine (ELM), Echo State Network (ESN), Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), Gated Recurrent Unit (GRU), eXtreme Gradient Boosting (XGBoost), and Random Forest (RF). Among these, seven algorithms demonstrated superior performance based on key evaluation metrics, such as Root Mean Square Error (RMSE) and the coefficient of determination (R-squared). These top-performing methods—FFNN, ELM, LSTM, GRU, RF, XGBoost, and ESN—were selected as the base models for further analysis. These models were then applied to scenarios that combined endogenous and exogenous variables and subjected to hyperparameter optimization to enhance their forecasting accuracy. All implementations in this study were carried out using Python version 3.10.13.

It is important to note that linear models performed significantly worse compared to nonlinear approaches and are therefore excluded from the results presented in this chapter. However, for the sake of completeness, the implementation details and outcomes of the less successful methods are summarized in the appendices.

This chapter presents the perfromance assessments of the seven base models, evaluated across three distinct scenarios: using only endogenous variables, incorporating exogenous variables, and optimizing the models for enhanced performance. The chapter is structured to systematically report the findings for each category of forecasting approaches, including machine learning and deep learning neural network models, bagging and boosting ensemble learning models, and chaotic analysis models.

The implementation results of the neural network machine learning models are detailed in Sections 5.1 and 5.2, covering the Feedforward Neural Network (FFNN) and Extreme Learning Machine (ELM), respectively. The outcomes of the deep learning models, Long Short-Term Memory (LSTM) and Gated Recurrent Unit

(GRU), are presented in Sections 5.3 and 5.4. Sections 5.5 and 5.6 focus on the results of the bagging and boosting ensemble learning models, namely Random Forest (RF) and XGBoost. Finally, Section 5.7 reports the findings from the chaotic analysis models.

This chapter aims to provide a comprehensive comparison of the forecasting capabilities of these diverse approaches, highlighting their strengths and limitations under different modeling scenarios.

5.1 Feed Forward Neural Network Forecasting Results

The Feedforward Neural Network (FFNN) implemented in this study follows a dense fully connected architecture. The model's structure was designed to include the following components:

- Input Layer: The input layer corresponds to the selected features, including lag variables (lag1 to lag5) and two exogenous variables (ambient temperature and cell temperature). The input dimension is defined as the number of these features, set to 7.
- Hidden Layers: The optimal model includes two hidden layers with each layer consisting of 32 neurons, as determined through Bayesian optimization. The ReLU activation function was used to introduce non-linearity and enhance the learning capacity of the model.
- Dropout Regularization: A dropout layer with a 2.23% rate was incorporated following each hidden layer to prevent overfitting.
- Output Layer: A single neuron in the output layer predicts the target variable, which is the energy output.

The optimization algorithm used for training the FFNN was Adam with the value of 0.00143 as the learning rate, chosen for its computational efficiency and effectiveness in handling sparse gradients. The model was compiled with the loss function of MSE and assessed utilizing metrics like R-squared and RMSE.

The FFNN model underwent hyperparameter optimization using the Tree-structured Parzen Estimator (TPE) algorithm implemented through the hyperopt

library in Python. The final set of optimized hyperparameters is reported in Table 5.1.

Table 5.1 : Optimized hyperparameters of FFNN model.

Hyperparameters	Optimized value
Number of Hidden Layers	2
Number of Neurons per Layer	32
Optimizer	Adam
Activation Function	ReLU
Learning Rate	0.00143
Dropout Rate	2.23%
Epochs	150
Batch Size	64

During the implementation of each base model, three scenarios were analyzed: (1) applying the model solely to the endogenous variables, (2) incorporating the exogenous variables into the model, and (3) optimizing the model with both endogenous and exogenous variables integrated. The FFNN model forecasting results for each scenario, evaluated using the selected metrics, are summarized in Table 5.2. The results demonstrate a progressive improvement in forecast accuracy with each successive scenario. The final forecasting performance of the optimized FFNN model is illustrated in Figure 5.1, which compares the actual values with the forecasted values on the test set for this base model.

Table 5.2 : Evaluation metrics for FFNN model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
Endogenous variables	0.9611	0.1520	9.68	93.6720	4.7450
Exogenous variables	0.9712	0.1458	7.95	63.2432	4.2550
Optimized model	**0.9727**	**0.1456**	**7.74**	**59.8672**	**4.3650**

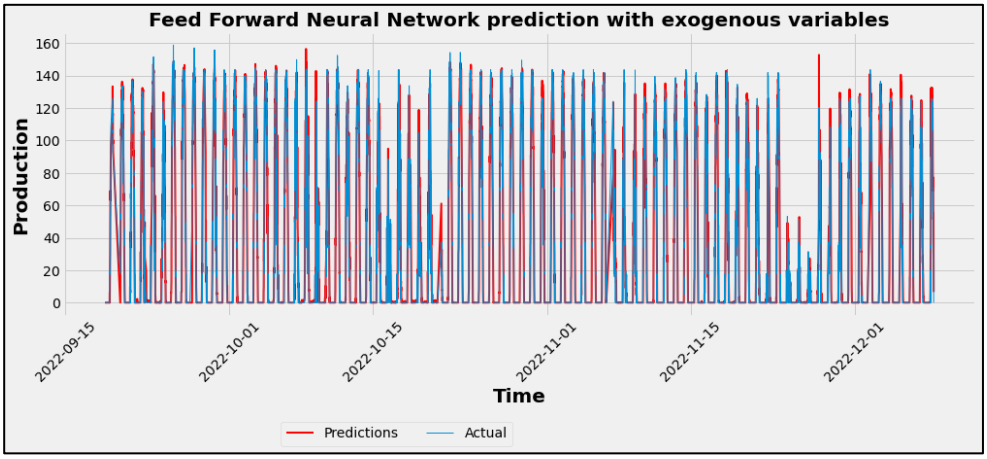

Figure 5.1 : FFNN model forecasting result after optimization.

5.2 Extreme Learning Machine Forecasting Results

The Extreme Learning Machine (ELM) model implemented in this script is structured as a feedforward neural network with a single hidden layer. Unlike traditional neural networks, ELMs feature a unique training mechanism where the input weights and biases are randomly assigned and remain fixed, while only the output weights are computed during training. This results in a much faster training process. The fine-tuned model's architecture is as the following:

- Input Layer: The input layer takes the lagged features and exogenous variables.
- Hidden Layer: The number of hidden nodes in the single hidden layer is a hyperparameter optimized through Bayesian optimization. In the optimized model 200 neurons are included in the hidden layer.
- Activation Functions: The hidden layer uses one of three activation functions—ReLU, Sigmoid, or Tanh—selected as part of the hyperparameter optimization process.
- Output Layer: The output layer generates the forecasted energy production values.

Tree-structured Parzen Estimator is employed to fine-tune the model's hyperparameters. The optimization minimizes the RMSE on the validation set. The optimized parameters are reported in Table 5.3.

Table 5.3 : Optimized hyperparameters of ELM model.

Hyperparameters	Optimized value
Hidden Size	200
Inputs Weights Range	[-0.700, 0.993]
Biases Range	[-0.681, 0.998]
Regularization Parameter	0.00034
Activation Function	ReLU

During the implementation of the ELM model, the three scenarios were analyzed as well: (1) applying the model exclusively to the endogenous variables, (2) integrating the exogenous variables into the model, and (3) optimizing the model with both exogenous and endogenous variables combined. The forecasting results for the ELM model under each scenario, evaluated using the selected metrics, are summarized in Table 5.4. The results indicate a consistent improvement in forecasting accuracy across the scenarios. The output generated by the optimized ELM model is visualized in Figure 5.2, showcasing a comparison between the actual values and the forecasted values on the test set for this base model.

Table 5.4 : Evaluation metrics for ELM model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
Endogenous variables	0.9601	0.1520	9.71	94.3120	4.6910
Exogenous variables	0.9632	0.1511	9.1	82.7442	4.6612
Optimized model	**0.9651**	**0.1444**	**8.76**	**76.6796**	**4.3304**

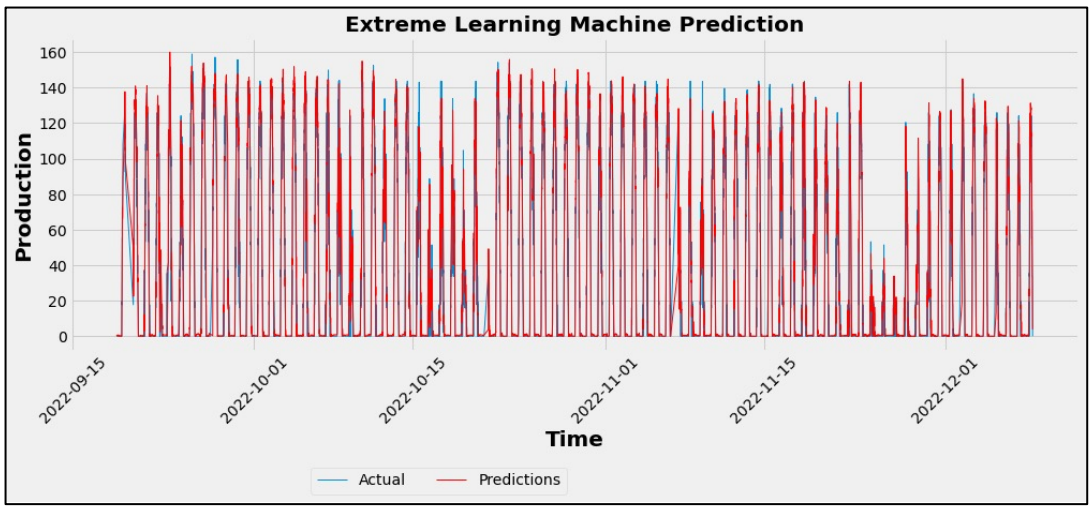

Figure 5.2 : ELM model forecasting result after optimization.

5.3 Long Short-Term Memory Forecasting Results

The LSTM model was developed to forecast energy output using both endogenous and exogenous variables. The model architecture is a hybrid structure designed to capture sequential dependencies and external influences. It includes an LSTM layer to process time-series data, a Dense layer for exogenous variables, and a combined fully connected layer to refine the integrated inputs. The final output layer uses a linear activation function to predict energy output.

Bayesian optimization was employed to fine-tune the model's hyperparameters. Using the Tree-structured Parzen Estimator (TPE) method, the optimization process minimized the validation loss (mean squared error) over 20 iterations. The hyperparameters alongwith their optimized values are reported in Table 5.5.

Table 5.5 : Optimized hyperparameters of LSTM model.

Hyperparameters	Optimized value
Number of LSTM Units	64
Number of Dense Units	64
Activation Function	tanh
Learning Rate	0.000212
Optimizer	RMSprop
Dropout Rate	0.2018
Batch Size	32
Number of Epochs	50

Same as the other base models, the three scenarios were analyzed for LSTM model as well. The forecasting results for the LSTM model in each scenario, evaluated using the chosen metrics, are summarized in Table 5.6. These results demonstrate a consistent improvement in forecasting accuracy across all scenarios. Figure 5.3 displays the final forecasting performance of the optimized LSTM model and compares the actual and forecasted values on the test set for this base model.

Table 5.6 : Evaluation metrics for LSTM model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
Endogenous variables	0.9380	0.2010	11.60	131.6430	6.0921
Exogenous variables	0.9760	0.1311	7.04	49.5701	3.8120
Optimized model	**0.9784**	**0.1230**	**6.80**	**46.2591**	**3.5959**

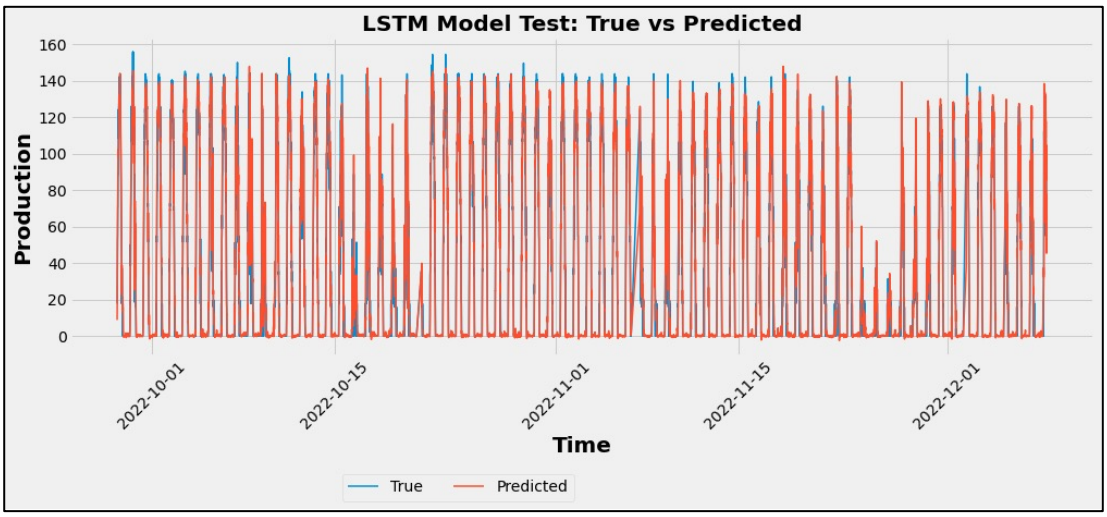

Figure 5.3 : LSTM model forecasting result after optimization.

5.4 Gated Recurrent Unit Forecasting Results

The implemented Gated Recurrent Unit (GRU) model leverages a sequential architecture designed to handle time series forecasting with both endogenous and exogenous variables. The model is constructed using the TensorFlow Keras framework. Below is a detailed description of its structure:

- Input Layers: The model consists of two input branches. The first branch processes the sequence of endogenous variables using the GRU layer. The second branch handles the exogenous variables. These inputs are passed through a dense layer to align their dimensions for concatenation.
- GRU Layer: The GRU layer includes 128 units, as determined by the hyperparameter optimization process. GRUs are well-suited for identifying temporal relationships in sequential data and are computationally efficient compared to LSTMs.
- Dense Layers: After the GRU output and exogenous variable representations are concatenated, the model applies a dense layer with 64 neurons along with the ReLU activation function. To prevent overfitting, a dropout layer with a rate of 0.2478 is incorporated.
- The final output layer is a single neuron with a linear activation function, which generates the predicted value for the target variable (solar energy output).

The optimizer of the model is RMSprop, using the Mean Squared Error (MSE) loss function, with a learning rate of 0.0002, which was selected during hyperparameter tuning. RMSprop works well for RNN-based architectures like GRUs.

The hyperparameter tuning process utilized TPE algorithm carried out via the hyperopt library. The optimized parameters are reported in Table 5.7.

Table 5.7 : Optimized hyperparameters of GRU model.

Hyperparameters	Optimized value
GRU Units	128
Dense Units	64
Activation Function	ReLU
Learning Rate	0.0002
Optimizer	RMSprop
Dropout Rate	0.2478
Batch Size	32
Number of Epochs	40

Similar to the other base models, the GRU model was analyzed under three scenarios. The forecasting results for the GRU model, evaluated using the selected metrics, are summarized in Table 5.8. These results show a consistent improvement in forecasting precision across all scenarios. Figure 5.4 shows the final forecasting performance of the optimized GRU model, presenting a comparison between the actual values and the forecasted values on the test set for this base model.

Table 5.8 : Evaluation metrics for GRU model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
Endogenous variables	0.9530	0.1610	10.05	101.0911	4.6321
Exogenous variables	0.9760	0.1220	7.13	50.8720	3.6804
Optimized model	**0.9776**	**0.1197**	**6.93**	**48.0372**	**3.50**

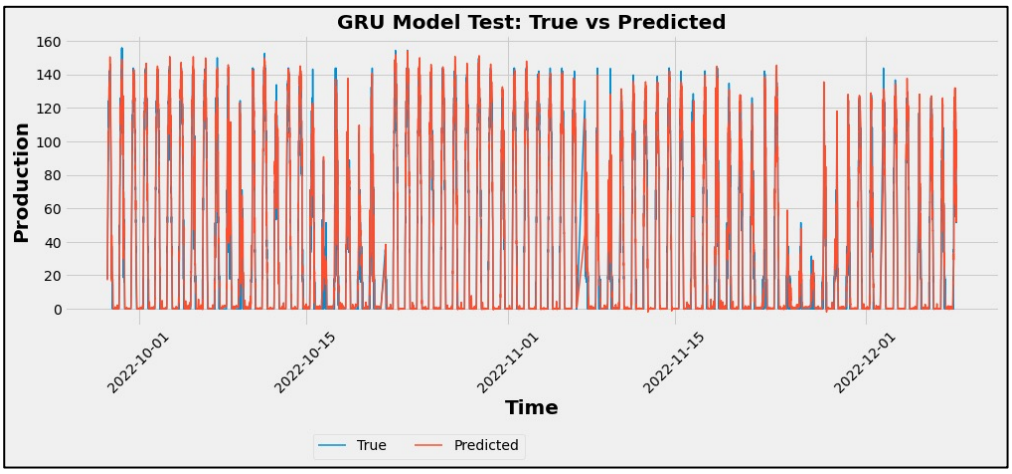

Figure 5.4 : GRU model forecasting result after optimization.

5.5 Random Forest Forecasting Results

The Random Forest (RF) model, an ensemble learning technique suited for regression tasks, enhances predictive accuracy by training multiple decision trees and averaging their individual outputs. This approach not only improves overall model performance but also mitigates the risk of overfitting. In the current implementation, the RF model was trained using a feature set that included lagged values of energy production, temporal features (such as month, hour, and day of the year), and exogenous variables including ambient and cell temperatures.

The Tree-structured Parzen Estimator (TPE) algorithm is used to efficiently search for the best hyperparameter combination. The search space for the RF hyperparameters included ranges for number of estimators, minimum samples split, minimum samples leaf, maximum depth, maximum number of features, and bootstrap. The optimization process minimizes the negative mean squared error using 5-fold cross-validation. The objective function, guided by the TPE algorithm, evaluates 100 hyperparameter combinations to identify the best-performing set. The optimized parameters enhance the model's accuracy while avoiding overfitting. The final optimized RF model parameters are reported in Table 5.9.

Table 5.9 : Optimized hyperparameters of RF model.

Hyperparameters	Optimized value
Number of estimators	298
Maximum features	10
Maximum depth	14
Minimum samples split	13
Minimum samples leaf	4
Bootstrap	True
Random state	1

Similar to the other base models, the RF model was evaluated under three scenarios. The forecasting results, assessed using the selected metrics, are summarized in Table 5.10. These results demonstrate a consistent improvement in forecasting accuracy across all scenarios. Figure 5.5 depicts the final optimized RF model forecasting performance that compares the actual and forecasted values on the test set for this base model.

Table 5.10 : Evaluation metrics for RF model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
Endogenous variables	0.9530	0.1510	10.2802	105.6901	4.8701
Exogenous variables	0.9710	0.1210	8.06	65.0001	3.9121
Optimized model	**0.9717**	**0.1200**	**8.06**	**64.9560**	**3.8885**

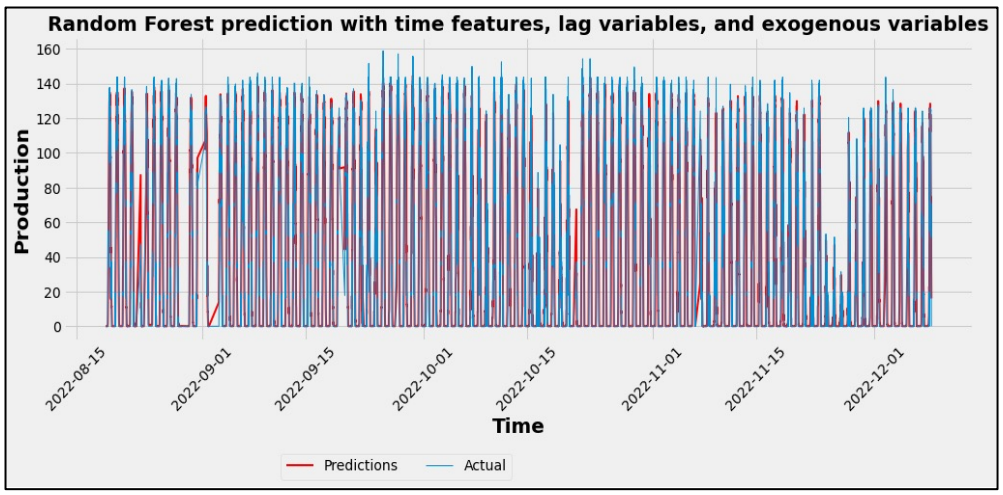

Figure 5.5 : RF model forecasting result after optimization.

5.6 Extreme Gradient Boosting Forecasting Results

A powerful gradient-boosting framework is the Extreme Gradient Boosting (XGBoost) algorithm designed for efficiency and scalability in supervised learning tasks. It uses an ensemble approach, combining multiple decision trees to minimize prediction error. The primary architecture includes:

- Boosting Framework: Sequentially trains decision trees, with each subsequent tree aiming to correct the errors made by its predecessor.
- Regularization: Includes L1 (LASSO) and L2 (Ridge) regularization to prevent overfitting.
- Split Finding: Optimized split search using histogram-based approaches.
- Early Stopping: Halts training when no further improvement is observed, reducing computation time.

After running the Tree-structured Parzen Estimator algorithm with 100 evaluations, the optimal hyperparameters for the XGBoost model were found and reported in Table 5.11.

Table 5.11 : Optimized hyperparameters of XGBoost model.

Hyperparameters	Optimized value
Number of estimators	398
Learning Rate	0.21287
Maximum depth	5
Gamma	0.26
Subsample	0.73754
reg_alpha	0.61491
reg_lambda	0.41575
colsample_bytree	0.982
min_child_weight	3.5568
early_stopping_rounds	52

As with the other base models, the XGBoost model was evaluated across three distinct scenarios. The forecasting results, measured using the selected performance metrics, are presented in Table 5.12. These results indicate a consistent enhancement in forecasting accuracy across all scenarios. Figure 5.6 depicts the final forecasting performance of the optimized XGBoost model, showcasing a comparison between the actual values and the predicted values on the test set.

Table 5.12 : Evaluation metrics for XGBoost model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
Endogenous variables	0.9550	0.1520	10.00	99.9720	4.8211
Exogenous variables	0.9610	0.1401	9.16	83.8422	4.3710
Optimized model	**0.9722**	**0.1313**	**7.81**	**60.9800**	**3.9362**

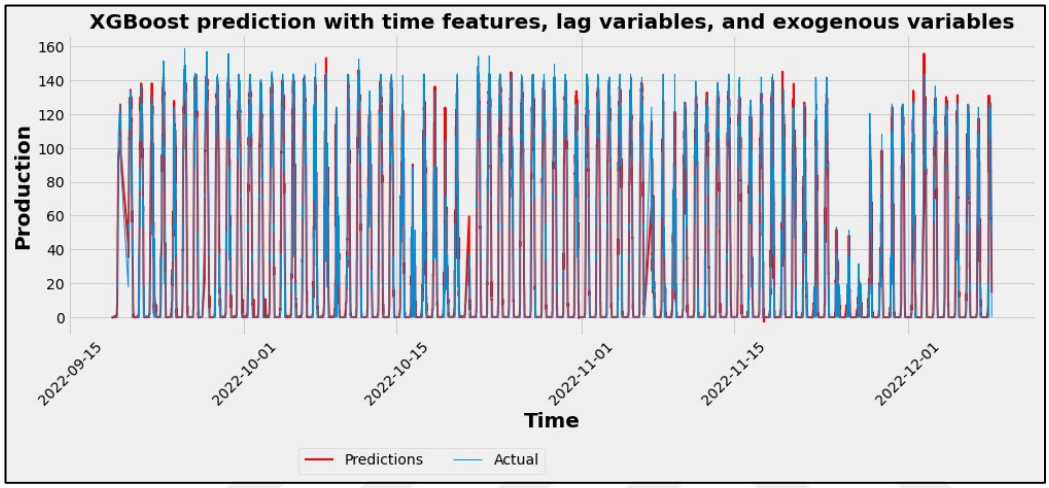

Figure 5.6 : XGBoost model forecasting result after optimization.

5.7 Chaotic Analysis Results

In this study's chaotic analysis, the False Nearest Neighbors (FNN) method is applied as a preprocessing technique to embed the time series data. The Echo State Network (ESN) algorithm is then utilized to forecast energy production using both the original and embedded datasets. Additionally, the Tree-structured Parzen Estimator (TPE) method is employed to fine-tune the ESN algorithm.

Considering that the ESN algorithm is less commonly used compared to popular methods such as support vector machines or artificial neural networks, this research evaluates its performance on both the original and embedded datasets. By doing so, this research takes a step towards covering the existing study gaps discussed in the literature study in Chapter 2, regarding the choice of forecasting models. It also provides a comparative analysis of the benefits of an optimized, embedded data approach versus a non-embedded approach. This underscores the importance of effective data preprocessing and dimensionality optimization in improving forecasting

accuracy, which is another important important study gap discussed in the literature review in section 2.

The application of the FNN method for data preprocessing and dimensionality optimization is detailed in Section 5.7.1. The results of the ESN algorithm implementation, along with its optimization, are presented in Section 5.7.2.

5.7.1 False nearest neighbors analysis results

Within the domain of solar power prediction, the False Nearest Neighbors (FNN) algorithm is utilized to capture the complex, nonlinear dynamics of solar power production systems by reconstructing the phase space (Saadati and Barutcu, 2024b). By determining the optimal embedding dimension, the algorithm enhances the accuracy of models trained on this data, such as Echo State Networks (ESNs), as demonstrated in this study. Specifically, the FNN algorithm was utilized to optimize the embedding dimension for time series data collected from a solar power plant. By incorporating both endogenous and exogenous variables, this approach provides a more comprehensive representation of the system's dynamics. This study highlights the significance of integrating advanced dimensionality optimization methods with domain-specific knowledge to improve the machine learning models' predictive performance in renewable energy forecasting.

The implementation of the FNN method for time series analysis in Python involves a series of critical steps, detailed as follows:

I. Data Preparation: The time series data is organized into an appropriate format for analysis, typically as a one-dimensional array or a Pandas Series.

II. Time-Delay Embedding: The data is embedded into a higher-dimensional phase space using time-delay embedding, with careful consideration given to selecting embedding parameters such as the embedding dimension (m).

III. Nearest Neighbor Identification: A nearest neighbors algorithm, such as a brute-force approach, is employed to determine the closest neighbors of each data sample in the reconstructed phase space.

IV. Detecting False Nearest Neighbors: Each point in the embedded space is perturbed slightly, and the distance to its nearest neighbor is recalculated. If the ratio of the

perturbed distance to the original distance surpasses a specified threshold, the point is classified as a false nearest neighbor.

V. Calculating the FNN Ratio: The FNN ratio provides insights into the effectiveness of the chosen embedding parameters in capturing the system's dynamics. It measures the fraction of points in the embedded phase space incorrectly identified as nearest neighbors following perturbation.

VI. Results Visualization: The outcomes are visualized to highlight regions in the phase space where false nearest neighbors occur, offering a clearer understanding of the dynamics of the time series data.

Applying the FNN method to the energy production time series reveals an FNN ratio of 0.77, indicating that a substantial fraction of points in the embedded phase space are incorrectly labeled as nearest neighbors post-perturbation. The implications of this ratio are explored below:

- Complex Dynamics:

  A high FNN ratio suggests that the time series exhibits complex and potentially nonlinear dynamics. The presence of false nearest neighbors implies that points appearing close in the embedded phase space are not genuinely proximate in the original phase space, potentially indicating chaotic or nonlinear behavior within the system.

- Dimensionality Estimation:

  The FNN ratio is a critical metric for determining an appropriate embedding dimension (m) for time series analysis. A high ratio signifies that the current embedding dimension is insufficient to capture the system's underlying dynamics accurately. In such cases, increasing the embedding dimension can help reduce the FNN ratio, thereby delivering a clearer reflection of the system's behavior in the embedded phase space and enhancing the understanding of its true dynamics.

The previously implemented FNN function is reapplied, this time integrating two exogenous variables—ambient temperature and solar cell temperature—into the input data structure. These exogenous variables are combined with the target (endogenous) variable to form a unified DataFrame, which is subsequently used as input for the FNN

analysis. Under this setup, the FNN ratio significantly decreases to 0.277. This notable reduction in the FNN ratio, compared to the scenario where only the target variable is used, provides meaningful insights into the interdependencies and dynamics within the dataset, highlighting the added value of incorporating exogenous variables in the analysis.

In the earlier results, the embedding dimension was manually selected to transform the time series data into a higher-dimensional phase space. Identifying the optimal embedding dimension for time series data generally involves employing techniques such as the FNN itself.

In this section, the FNN algorithm is employed to optimize the embedding dimension. Initially, the maximum embedding dimension is set to 100. Subsequently, the embedding dimension is systematically varied, and the FNN ratio is calculated for each dimension. The optimal embedding dimension corresponds to the minimum value of m where the FNN ratio stabilizes or hits its lowest value. Figure 5.7 illustrates the relationship between the FNN ratio and the embedding dimension, providing a clear view of how the FNN ratio evolves with changes in embedding parameters. Following the optimization of the embedding dimension, the FNN ratio for the merged dataset containing both the target variable and exogenous variables, decreases from 0.277 to 0.007. The FNN ratio changes during these studied scenarios are summarized in Table 5.13.

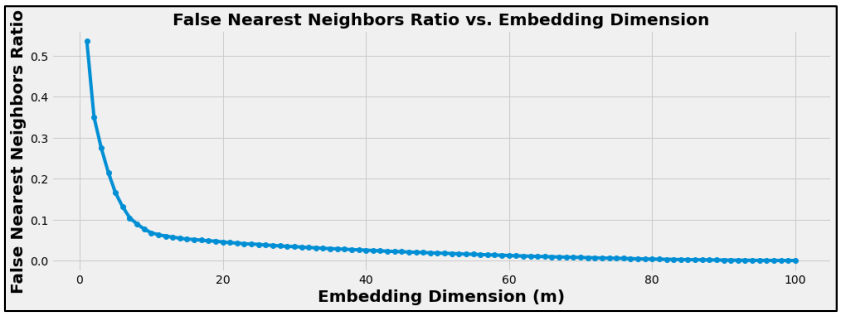

Figure 5.7 : Relation between FNN ratio and embedding dimension.

Table 5.13 : FNN ratio values in studied scenarios.

Scenario	FNN ratio
Endogenous Inputs	0.77
Adding Exogenous Inputs	0.277
After Optimization	0.007

The optimized FNN ratio provides important insights into how well the selected embedding parameters capture the input data's patterns and relations, which is the time series data's dynamics. A very low FNN ratio, such as 0.007 in our case, indicates that the chosen embedding dimension and time delay are well-suited to the data, with minimal false nearest neighbors. This suggests that the time series dynamics are effectively represented in the reconstructed space. The low value of the FNN ratio signifies effective embedding of the time series into a space of higher dimension where nearby points reflect meaningful relationships. This ensures a reliable representation of the dynamics with the optimal embedding dimension of 70. This result is robust, suggesting that the analysis is resilient to noise and perturbations, and confirms that the structure observed in the embedded space accurately represents the true underlying dynamics.

To explore the data's structure and highlight zones vulnerable to false nearest neighbors, the embedded phase space is visualized in Figure 5.8. Given the high inherent dimensionality of the data, the principal component analysis (PCA) dimensionality reduction technique (Jolliffe, 2011) is employed. PCA facilitates projecting the data into lower-dimensional spaces, enabling easier visualization and offering clearer insights into the relationships among data points within the embedded phase space.

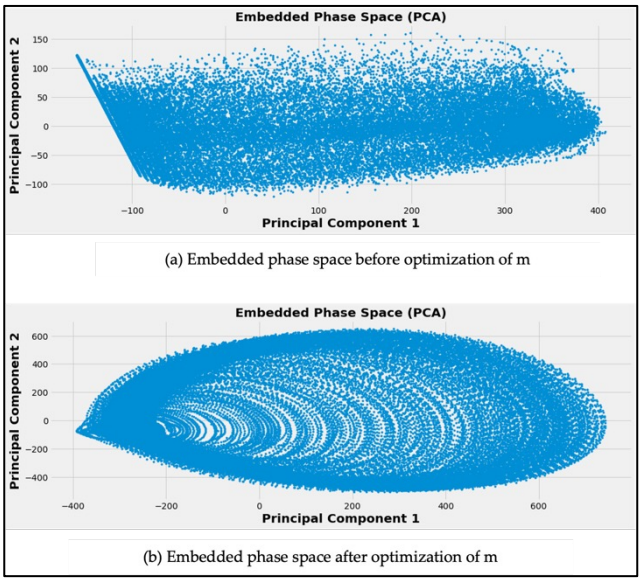

Figure 5.8 : Embedded phase space. Before optimization of m. (b) After optimization of m.

This visualization offers a clearer view of the embedded phase space, making it easier to detect patterns, clusters, and anomalies. Densely packed points suggest regions with similar dynamic behavior, whereas scattered points may reveal areas vulnerable to false nearest neighbors. As discussed earlier, reconstructing the phase space with an optimized embedding dimension ensures that the embedded data retains the structural characteristics of the original time series. This enhanced structure provides a robust foundation for developing accurate predictive models, as further illustrated in the subsequent analysis.

5.7.2 Echo state network model results

The Echo State Network (ESN) model is implemented under two distinct scenarios: (1) using the original (non-embedded) solar energy production data alongside weather-related exogenous variables, and (2) utilizing the embedded data, which includes both exogenous and endogenous variables. Hyperparameter tuning for the ESN model is carried out using the Tree-structured Parzen Estimator (TPE) method. The optimized hyperparameters and their corresponding values are detailed in Table 5.14. The model's forecasting performance for both the original and embedded data, before and after optimization, is summarized in Table 5.15. Figures 5.9 and 5.10 illustrate the forecasts generated by the optimized ESN model using the original time series data and the embedded data, respectively.

The ESN model applied to the original time series data consists of the following architectural components:

- Input Layer: Receives input features, which include lagged energy production values and exogenous variables.
- Reservoir: Composed of a fixed, randomly generated recurrent neural network with a large number of neurons. This reservoir transforms the input into a higher-dimensional feature space. The hyperparameters optimized for the reservoir include:
    - Number of Reservoir Neurons (n_reservoir): Defines the reservoir's capacity.
    - Spectral Radius: Controls the stability of the reservoir dynamics and ensures the network's echo state property.

- Sparsity: Specifies the percentage of zero connections in the reservoir, promoting computational efficiency.
- Noise: Adds randomness to the reservoir to enhance generalization.
- Input Scaling: Adjusts the magnitude of input features before feeding them into the reservoir.

- Output Layer: A linear regression layer maps the reservoir states to the final prediction target, in this case, the solar energy output.

The training involves adjusting only the output weights, while the reservoir and input weights remain fixed. This design significantly reduces computational overhead compared to traditional recurrent networks.

Table 5.14 : Optimized hyperparameters of ESN model.

Hyperparameters	Optimized value (Original data)	Optimized value (Embedded data)
Reservoir units Number	1105	778
Reservoir Connections Sparsity	0.829	0.021
Reservoir Weights Spectral Radius	1.357	0.357
Magnitude of Noise (to be added to the reservoir activations)	0.003	0.054

Table 5.15 : Evaluation metrics for ESN model forecasting results.

Scenario/Metric	R-squared	MAPE	RMSE	MSE	MAE
ESN model – Original data					
Before optimization	0.9610	0.1711	9.26	85.7121	5.1320
After optimization	0.9703	0.1545	8.91	85.0831	5.0130
ESN model – Embedded data					
Before optimization	0.9721	0.1301	8.80	77.48	5.02
After optimization	**0.9724**	**0.1328**	**8.75**	**76.6108**	**4.9467**

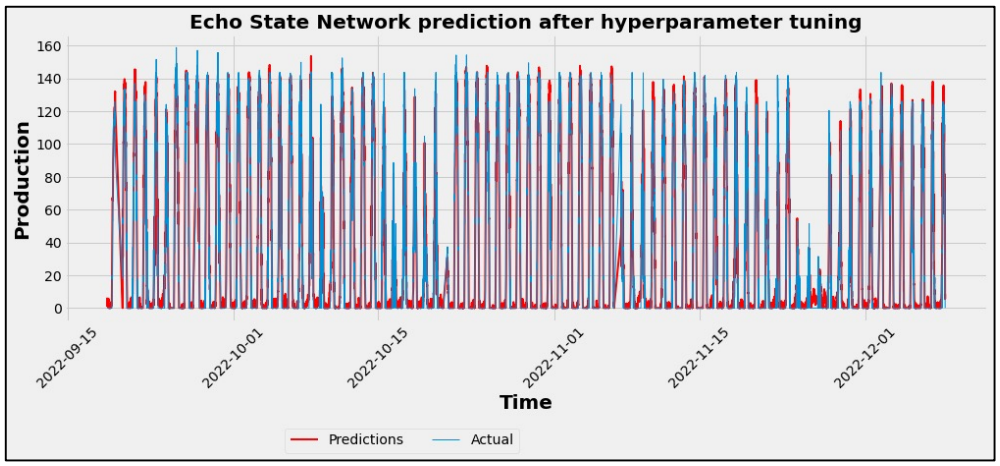

Figure 5.9 : ESN model forecasts using the original time series data.

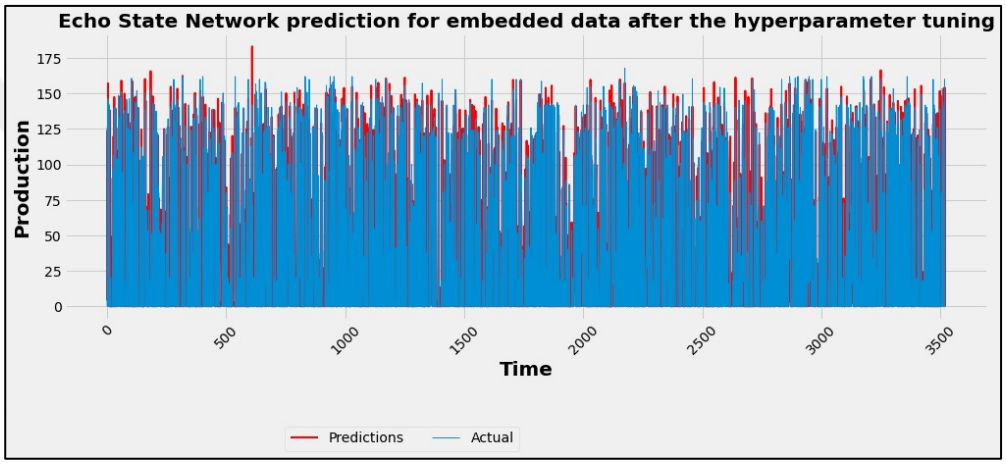

Figure 5.10 : ESN model forecasts using the embedded data.

Using the FNN method to identify the optimal embedding dimension significantly improved the model's capability to learn and represent the complex, nonlinear patterns in solar energy generation, resulting in more precise forecasts. The optimized ESN model demonstrated superior performance when applied to the embedded data compared to the original non-embedded data. This emphasizes the critical role of selecting and fine-tuning the embedding dimension, and generally data preprocessing, in chaotic modeling of solar energy systems to enhance forecasting accuracy.

The improved prediction accuracy achieved with the optimized model highlights its potential to advance solar energy forecasting, a key factor for efficient energy management, seamless grid integration, and strategic long-term planning in renewable energy systems.

The possibility of generalizing the embedding data technique to other machine learning algorithms implemented in this study was explored, with the results

summarized in Appendix B. However, the findings indicated that compared to the embedded data, these algorithms showed improved performance on the original time series data. Considering this, and given that the objective of the next chapter is to construct a metamodel requiring all base models to be trained and tested on the same dataset, all seven base models, including the ESN model, were trained using the original time series data. Therefore, the application of embedded data is confined to the chaotic analysis performed in this study, which concludes with this section.

6. STACKING ENSEMBLE MODELS

This chapter focuses on the stacking ensemble machine learning approaches implemented in this thesis to enhance forecasting performance. The main objective of this chapter is to propose a novel metamodel that outperforms the base models in predicting solar energy production. Two categories of stacking ensemble models are explored: regression-based stacking and neural network-based stacking. Each approach is evaluated and compared, culminating in the identification of the most accurate model, which represents a significant contribution of this research.

The regression-based stacking models combine the outputs of the base predictors (level-0 models) by utilizing traditional regression algorithms as the meta-model. The following regression stacking models were implemented: Multi-Linear Regression, Ridge Regression, and Lasso Regression stacking model presented respectively in sections 6.1, 6.2, and 6.3.

In the second category, neural networks (NN) serve as the meta-model, leveraging their ability to model complex nonlinear relationships. Three neural network-based stacking approaches are created: NN stacking model with 7 base models, presented in section 6.4, NN stacking model with 5 base models, presented in section 6.5, and the proposed deep learning-based stacking model that is a NN stacking model with 2 base models, presented in section 6.6. This novel stacking model uses only two base models, strategically selected based on their individual performance. It incorporates optimized hyperparameters and advanced techniques, resulting in the highest forecasting accuracy among all models studied.

6.1 Multi-Linear Regression Stacking Model

The stacking model is implemented using a two-layer architecture structured as follows.

- Base Models (First Layer):
    - Predictions are generated from seven base models: LSTM, GRU, FFNN, ELM, RF, XGBoost, and ESN.
    - These base models' predictions are loaded from CSV files, aligned to have the same index, and stacked as features.
    - The base models' outputs are treated as inputs for the meta-model.
- Meta-Model (Second Layer):
    - A Linear Regression model is used as the meta-model.
    - It is trained on the outputs of the base models. Its X (features) values are the outputs matrix of the base models and its Y (true) values are the corresponding target values.

The meta-model's performance, after training, is evaluated using unseen test data and their actual values. This test dataset, referred to as test_MetaModel, was set aside during the data segmentation phase of the study to prevent data leakage and ensure that a completely unseen dataset is used for the meta-model to be assessed on. A detailed explanation of this process is provided in Section 4.5 of this thesis. Evaluation results for this meta-model is presented in Table 6.1.

Table 6.1 : Evaluation metrics for multi linear regression stacking model forecasting results.

R-squared	MAPE	RMSE	MSE	MAE
0.9721	0.1615	7.56	57.1526	4.4711

6.2 Ridge Regression Stacking Model

The architecture of this stacking ensemble is the same with the multi linear regression model. It integrates predictions from seven base models: LSTM, GRU, FFNN, ELM, RF, XGBoost, and ESN. Input features of the Ridge regression meta-model are the outputs of these models.

Ridge regression is a linear regression model that includes regularization and minimizes the sum of squared residuals with an L2 penalty term (controlled by the alpha hyperparameter). This helps reduce overfitting by shrinking model coefficients.

The predictions from the base models are combined with predefined weights (weights array). Higher weights are assigned to base models with better performance, as indicated by R^2 scores. The weighted predictions are used as input to the Ridge and Lasso models.

Evaluation results for this meta-model is presented in Table 6.2.

Table 6.2 : Evaluation metrics for Ridge regression stacking model forecasting results.

R-squared	MAPE	RMSE	MSE	MAE
0.9721	0.1614	7.56	57.0944	4.4679

6.3 Lasso Regression Stacking Model

The architecture of this stacking ensemble is also the same with multi linear and Ridge regression models. It integrates predictions from seven base models and the outputs of these models serve as input features for the Lasso regression meta-model.

Lasso regression is a linear regression model that includes regularization and minimizes the sum of absolute values of the residuals with an L1 penalty term (controlled by the alpha hyperparameter). It performs both regularization and feature selection by shrinking some coefficients to zero.

The primary difference between Lasso and Ridge regression lies in the type of regularization penalty applied and their effects on the model. Ridge regression introduces a penalty term, which is proportional to the sum of the squared coefficients, to the loss function, while Lasso regression adds a penalty to the loss function based on the sum of the absolute values of the coefficients.

The base models predictions are combined with predefined weights (weights array) using as input to the Ridge and Lasso models. Higher weights are assigned to base models with better performance, as indicated by R^2 scores.

Evaluation results for this meta-model is presented in Table 6.3.

Table 6.3 : Evaluation metrics for Lasso regression stacking model forecasting results.

R-squared	MAPE	RMSE	MSE	MAE
0.9740	0.1532	7.29	53.1462	4.2398

6.4 NN Stacking Model with Seven Base Models

This ensemble stacking meta-model is a feedforward neural network designed and trained with the best hyperparameters determined by the Tree-structured Parzen Estimator (TPE) algorithm. Its inputs are predictions from the seven base models and its output is the forecasted value of the solar energy generation. The inputs of the meta-model in the stacking ensemble approach are all the base (level-0) models' outputs, as it is also explained in the earlier sections of this study. So, with respect to the data segmentation that is explained in section 4.5 of this thesis, the inputs for the meta-model's training, validation, and test sets consist of the outputs generated by the base models for their respective datasets.

The structure of this meta-model is described below, and Table 6.4 presents the optimized hyperparameter values.

The input layer consists of features from the base models' predictions. The model contains 3 fully connected (dense) layers. Each hidden layer has 96 units with the ReLU activation function which helps the network learn nonlinear relationships effectively. The output layer contains a single neuron utilizing a linear activation function. The model is trained using the Adam optimizer with a tuned learning rate, which balances convergence speed and stability during training. The model minimizes the MSE loss, which is a standard choice for regression problems. The model is trained for 60 epochs with a batch size of 128. These settings aim to provide sufficient training iterations for the model to converge while balancing computational efficiency. During training, the model uses validation data to monitor performance and prevent overfitting. Figure 6.1 illustrates the structure of this meta-model.

Table 6.4 : Optimized hyperparameters of the NN meta-model (seven bases).

Hyperparameters	Optimized value
Number of Hidden Layers	3
Number of Neurons per Layer	96
Optimizer	Adam
Learning Rate	0.00003
Activation Function	ReLU
Batch Size	128
Epochs	60

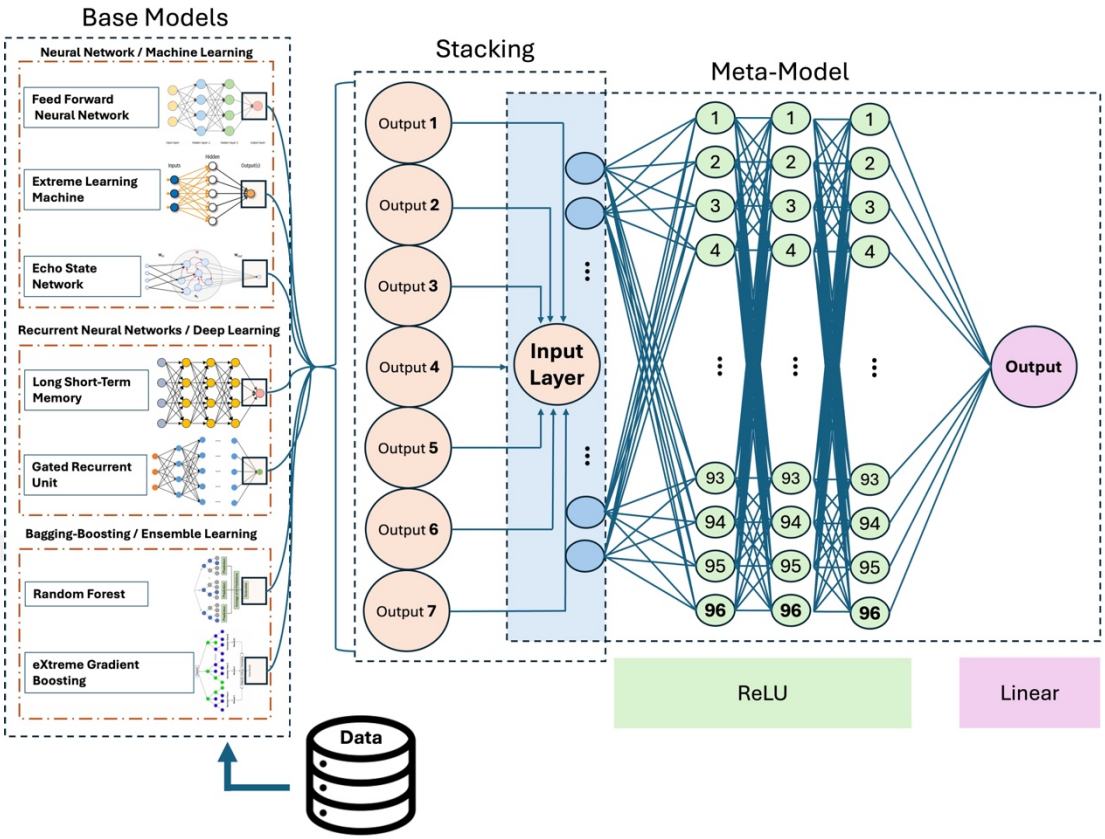

Figure 6.1 : Neural network stacking meta-model with seven base models.

Table 6.5 presents the forecasting results of this meta-model. While this ensemble meta-model performs well—achieving a forecasting precision above 0.97, which is considered high—the ultimate goal of this study is to leverage multiple successful base learners, each capable of strong individual predictions, to develop a stacking meta-model with superior forecasting accuracy. Among the base models, the LSTM achieved the highest performance, with an R-squared value of 0.9784. The aim is to surpass this benchmark.

However, the current meta-model, which uses seven base models as level-0 learners, failed to achieve this target. Several factors may contribute to this limitation, including overfitting of the meta-model, redundant features, negative contributions from underperforming base models, and a mismatch between the size of the training data and meta-model's complexity.

To address these issues, several strategies were implemented, such as reducing the meta-model's complexity, incorporating regularization, performing feature

engineering, and weighting the contributions of base models. These approaches led to the development of a revised stacking meta-model (the NN stacking model with five base models), which is discussed in detail in the upcoming Section 6.5.

Table 6.5 : Evaluation metrics for the NN meta-model (seven bases) forecasting results.

R-squared	MAPE	RMSE	MSE	MAE
0.9740	0.1403	7.30	53.2841	3.8830

6.5 NN Stacking Model with Five Base Models

This meta-model is derived from the previous meta-model discussed in Section 6.4, incorporating several refinements in its design and post-processing techniques. Regularization was introduced by adding dropout layers, and two less successful base models—ELM and ESN, which exhibited the lowest forecasting precision among the seven base models—were removed. The updated meta-model now utilizes the remaining five base models. Additionally, as part of the post-processing, Model Output Statistics (MOS) was applied using a Ridge regression model to achieve better weighting of the base models.

This meta-model's design is refined by adding regularization using dropout layers and also optimizing hyperparameters using Tree-structured Parzen Estimator (TPE) method. Dropout layers are added to the meta-model neural network to randomly deactivate a subset of neurons during training, helping to reduce overfitting. The dropout rate is optimized using TPE. The list of hyperparameters and their optimized values are reported in Table 6.6. The optimized neural network meta-model's architecture is illustrated in Figure 6.2.

Table 6.6 : Optimized hyperparameters of the refined NN meta-model.

Hyperparameters	Optimized value
Number of Hidden Layers	1
Number of Neurons per Layer	224
Optimizer	RMSprop
Learning Rate	0.000018
Activation Function	ReLU
Batch Size	80
Epochs	60
Dropout Rate	0.4330
Alpha (MOS)	100

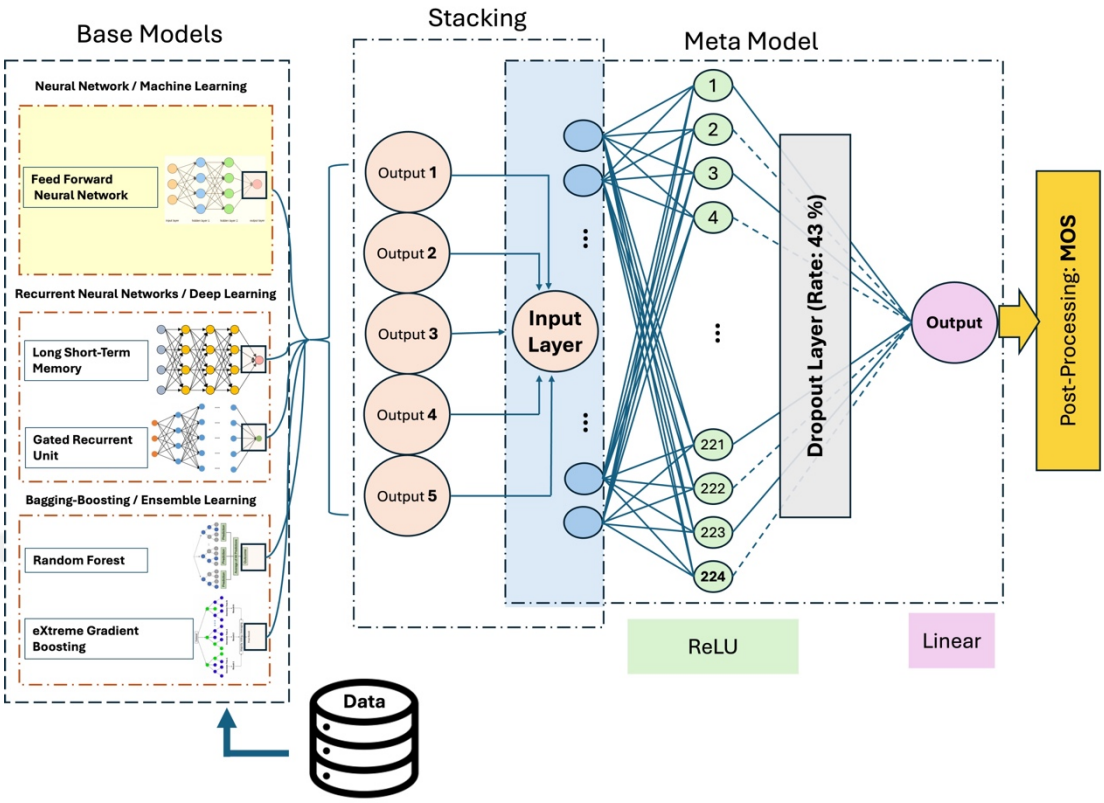

Figure 6.2 : Neural network stacking meta-model with five base models.

This meta-model employs MOS to improve the final meta-model's performance by adjusting the weighting of the individual base predictions. Here's how MOS is integrated into the implementation:

1- Stacked Base Model Predictions as Input Features:

- After generating predictions from five different base models (LSTM, GRU, FFNN, RF, XGBoost), these predictions are stacked together to create input features (X_train_MetaModel, X_val_MetaModel, and X_test_MetaModel).
- This stacked data represents each individual model's predictions for the training, validation, and test sets, and forms the basis for applying MOS.

2- MOS Implementation Using Ridge Regression:

- A Ridge regression model is used as the MOS model to assign optimal weights to the predictions from the individual base models. The Ridge

model is chosen because it helps to reduce multicollinearity (the correlations between the base model predictions) through L2 regularization, which is especially helpful in stacking scenarios where predictions from different models are often highly correlated.

- The Ridge model is fitted using the stacked predictions (X_train_MetaModel) as input features and the true target values (y_train_MetaModel). This fitting process helps determine the optimal combination (or weighting) of the base model predictions to best match the actual target values.

3- Generating Final Predictions Using MOS:

- Once the Ridge model is trained, it is used on the test set (X_test_MetaModel) to make predictions. This prediction is essentially a weighted combination of the base model predictions.

The Ridge regression parameter "alpha" controls the degree of regularization. A higher alpha increases the penalty for large coefficients, leading to a smoother (less complex) model. Finding the optimal alpha value that minimizes prediction error is the goal here. For finding the optimal value of alpha, GridSearchCV method is used. This method is included in scikit-learn library and performs an exhaustive search over a specified hyperparameter grid to find the optimal combination of hyperparameters for a specific model. This method is widely used for hyperparameter tuning to optimize the performance of machine learning models. This algorithm is implemented with minimizing the mean squared error to reach the optimum. 5-fold cross-validation is performed to evaluate each alpha value. This guarantees that the results remain robust and are not influenced by a single train-test split. The final Ridge regression model (with the optimal alpha equal to 100) is used to predict the test data (X_test_MetaModel). The output represents the smoothed and weighted predictions of the stacking ensemble, as adjusted by the MOS technique.

Table 6.7 presents the forecasting results of this meta-model.

Table 6.7 : Evaluation metrics for the refined NN meta-model (5 bases) forecasting results.

R-squared	MAPE	RMSE	MSE	MAE
0.9785	0.1200	6.63	43.9165	3.3225

This represents the first instance of exceeding the LSTM model's forecasting R-squared value of 0.9784.. Although the improvement is minimal (0.0001), it is noteworthy, as even slight advancements in accuracy are highly valued in this field. However, to further explore whether this study can propose a forecasting meta-model with a significantly higher accuracy than the LSTM model, the current meta-model will undergo an additional design refinement process. This refinement aims to reduce the model's complexity and develop a more simplified structure, with the expectation of achieving improved forecasting accuracy. The refined meta-model, which represents the extract of all the models implemented in this thesis and stands as the most accurate forecasting model proposed by this research, will be presented and discussed in detail in Section 6.6.

6.6 Deep Learning-Based Stacking Model

The proposed meta-model presented in this section is the result of several critical modifications to the design and architecture of previously implemented ensemble meta-models. These modifications pertain to base model selection, meta-model architecture, regularization strategies, and, more broadly, align with the goals of reducing the complexity of the meta-models while enhancing their efficiency. These refinements are discussed in detail in the following.

1- Base model selection: The propesed meta-model strategically reduces the input dimensionality by limiting the base models to only two models, LSTM and GRU models. These two models are the only deep learning algorithms among the previously implemented base models, distinguished by their highest precision values. Both are sequential models specially designed to capture time-related dependencies effectively in time series data. By focusing on these two complementary models, our proposed deep learning-based meta-model reduces potential noise and achieves a more robust integration of predictions. This reduction in input complexity enhances the meta-model's ability to generalize.

2- Meta-model architecture: Comparing to the previous implemented scenarios of ensemble meta-models, our new meta-model in this section adopts a deeper architecture with three layers, each containing 32 units. This deeper yet more compact design allows our proposed meta-model to capture hierarchical

patterns in the input features more effectively while maintaining a balance between model complexity and generalization capability.

3- Optimization strategy: The proposed meta-model omits dropout, which is appropriate given its smaller network size and shorter training duration. This decision reflects a more tailored regularization approach aligned with the model's design and dataset characteristics. This meta-model does not explicitly use dropout or another regularization mechanism like L2 regularization. Instead, it indirectly achieves regularization through its simpler design choices:

    - Compact Model Architecture: The use of only 32 units per layer and three layers helps limit the model's capacity, naturally reducing the risk of overfitting.
    - Smaller Batch Size: Training with a batch size of 32 (compared to 80 in previous model) introduces more noise in weight updates during training, which can act as a form of implicit regularization.
    - Shorter Training Duration: By training for only 20 epochs (compared to 60 epochs in previous model), the model avoids overfitting to the training data.

While these are not explicit regularization techniques in the traditional sense, they collectively contribute to controlling overfitting and improving generalization. The simplicity and balance of these design choices align with the smaller feature space and the characteristics of the dataset, allowing the model to learn effectively without requiring additional regularization layers like dropout. Figure 6.3 displays the proposed deep learning-based meta-model. This model is fine-tuned using TPE approach and the optimal hyperparameters are reported in Table 6.8. Table 6.9 presents the forecasting results of this meta-model.

Table 6.8 : Optimized hyperparameters of the proposed deep meta-model.

Hyperparameters	Optimized value
Number of Hidden Layers	3
Number of Neurons per Layer	32
Learning Rate	0.000017
Optimizer	RMSprop
Activation Function	ReLU
Batch Size	32
Epochs	20

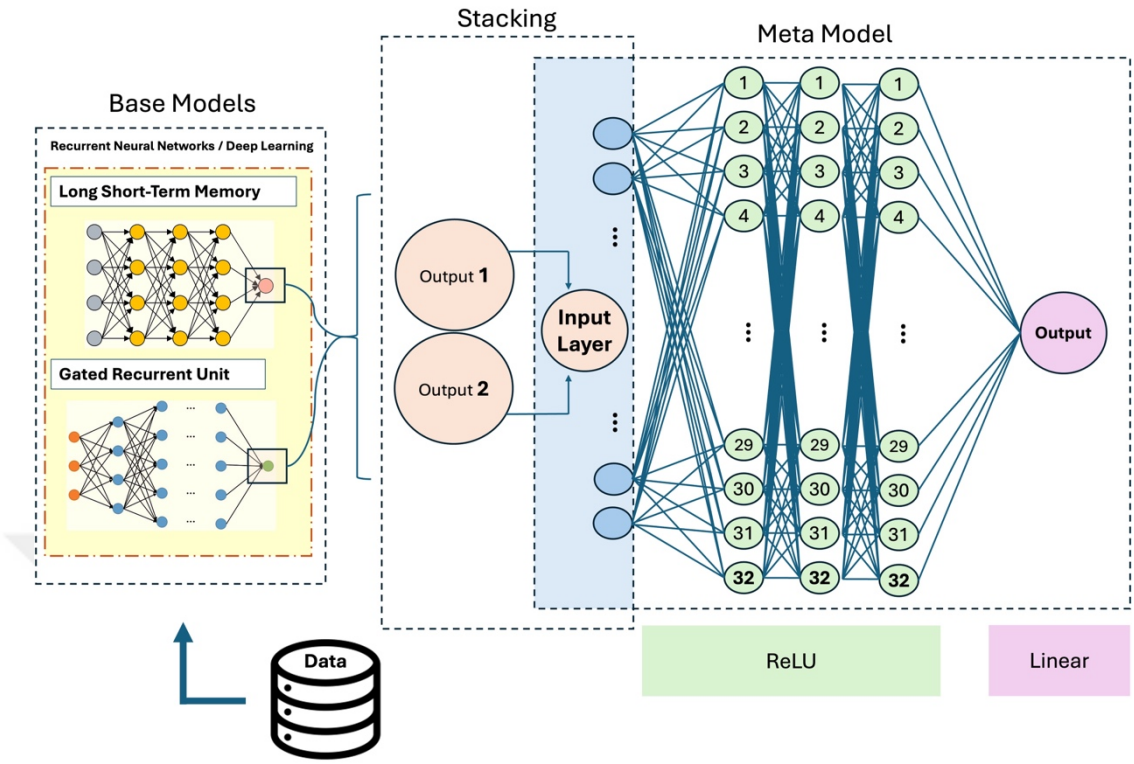

Figure 6.3 : Deep stacking meta-model with two base models.

Table 6.9 : Evaluation metrics for the proposed deep meta-model (2 bases) forecasting results.

R-squared	MAPE	RMSE	MSE	MAE
0.9805	0.1146	6.32	39.9949	3.1709

This is the first model developed in this thesis that demonstrates the capability to forecast the empirical data of solar energy generation from the Konya Eregli SPP, with a precision exceeding 98%. The proposed deep meta-model is designed to achieve superior forecasting accuracy and is poised to ultimately fulfill the ultimate objective of this study: surpassing the forecasting accuracy benchmark set by the LSTM model. Figure 6.4 shows this outcome.

The proposed deep-based meta-model achieves higher forecasting precision due to its:

- Focus on fewer, more complementary base models, leading to a cleaner input feature space.
- Deeper yet simpler meta-model architecture with reduced risk of overfitting.

- Efficient training configuration with fewer epochs and smaller batch sizes.
- Avoidance of redundant steps such as MOS, ensuring a streamlined stacking process.

These design choices enable our proposed meta-model to better identify the fundamental patterns in the data while maintaining generalization capabilities, resulting in superior performance compared to all forecasting models implemented in this thesis study.

Metrics / Methods	R^2	MAPE	RMSE	MSE	MAE
1 **Base Model** Long Short Term Memory	**0.9784**	0.1230	**6.80**	**46.2591**	3.5959
2 **Base Model** Gated Recurrent Unit	0.9776	**0.1197**	6.93	48.0372	**3.50**
3 **Base Model** Feed Forward Neural Network	0.9727	0.1456	7.74	59.8672	4.3650
4 **Base Model** Extreme Gradient Boosting	0.9722	0.1313	7.81	60.98	3.9362
5 **Base Model** Random Forest	0.9717	0.12	8.06	64.9560	3.8885
6 **Base Model** Echo State Network	0.9703	0.1545	8.07	65.0831	4.6330
7 **Base Model** Extreme Learning Machine	0.9651	0.1444	8.757	76.6796	4.3304
Stacking Model Multi Linear Regression	0.9721	0.1615	7.56	57.1526	4.4711
Stacking Model Ridge Regression	0.9721	0.1614	7.56	57.0944	4.4679
Stacking Model Lasso Regression	0.9740	0.1532	7.29	53.1462	4.2398
Stacking Model Neural Network 7 bases	0.9740	0.1403	7.30	53.2841	3.8830
Stacking Model Neural Network 5 bases	0.9785	0.1200	6.63	43.9165	3.3225
Stacking Model Neural Network 2 bases LSTM + GRU	**0.9805**	**0.1146**	**6.32**	**39.9949**	**3.1709**

Figure 6.4 : Performance tableau of all forecasting models implemented in the thesis study.

7. CONCLUSIONS AND RECOMMENDATIONS

The global energy market underpins modern economic growth and development, positioning energy as a central component in shaping societal progress. Within this market, the transition toward renewable energy—particularly solar power—has emerged as a critical strategy for fostering sustainable development and mitigating climate change. As a clean and abundant resource, solar energy aligns directly with multiple UN SDGs, notably Goal 7 and Goal 13. Moreover, it indirectly supports "Decent Work and Economic Growth" (Goal 8), "Industry, Innovation, and Infrastructure" (Goal 9), and "Sustainable Cities and Communities" (Goal 11). These collective contributions underscore solar energy's transformative potential in building more resilient and low-carbon economies worldwide.

Despite its promise, solar energy generation is inherently variable due to complex interactions among parameters like solar radiation, temperature of ambient, and technological efficiency. This variability presents challenges for grid stability, long-term resource planning, and the seamless integration of higher shares of renewable energy. Accurate forecasting of solar power output is therefore essential. It not only allows energy grid operators to balance supply and demand more effectively but also supports policy formulation, investment planning, and the achievement of renewable energy goals. Driven by rapid advancements in AI and ML, forecasting methodologies have evolved significantly, opening new avenues to enhance predictability and reliability in solar energy systems.

In response to these challenges, this thesis contributes to the advancement of solar energy forecasting by addressing critical gaps in existing methods and leveraging cutting-edge AI and ML approaches. Through systematic investigation, rigorous evaluations, and the proposal of novel techniques, the research offers practical insights that support the broader transition to low-carbon energy systems. This contribution is integral to meeting global sustainability goals, ensuring energy security, and accelerating the shift toward cleaner, more efficient power networks.

The purpose of this thesis is to advance the state of solar energy forecasting by addressing key challenges and gaps in the field. Although solar energy holds immense promise for driving the green economy and mitigating the impacts of climate change, its inherent variability introduces considerable hurdles for grid stability, resource allocation, and broader integration of renewables. Recognizing accurate forecasting as central to tackling these challenges, this research explores advanced artificial AI and ML approaches to enhance both the accuracy and the robustness of solar energy forecasts.

This work specifically targets several under-explored areas, including the incorporation of exogenous and endogenous variables, the role of chaotic models, and the optimization of forecasting models through hybridization. By developing innovative approaches and validating them using real-world data, the thesis seeks to deepen our understanding of solar power generation dynamics and deliver practical insights for more efficient renewable energy management.

Building on the extensive literature review presented in Chapter 2, this thesis identifies several underexplored areas and research gaps in solar energy forecasting. To bridge these gaps and advance the field, the methodology is specifically designed to address the following four key dimensions. The research questions of this thesis are formulated in order to systematically address these four methodological pillars:

1- Integration of Electricity-Related Data

Instead of relying solely on exogenous input variables, the proposed framework incorporates endogenous variables (e.g., historical power output) alongside exogenous factors (e.g., ambient temperature, solar cell temperature). This integrated approach allows for a more complete insight into the drivers of solar energy production.

2- Data Preprocessing Techniques

Considering the critical role of data quality in predictive accuracy, the research emphasizes robust preprocessing methods. Dimensionality optimization and feature extraction techniques are employed to refine input data, thereby improving the reliability and precision of forecasting models.

3- Advanced Deep Learning Algorithms

Rather than depending only on traditional methods like ANN and SVM, this thesis explores sophisticated deep learning architectures. By harnessing the power of models such as LSTM, GRU, and ensemble frameworks, the study seeks to enhance predictive performance and capture the inherent complexity of solar energy systems.

4- Optimizing Metaheuristic Model Design

To further improve computational efficiency and simplify real-world implementation, the research focuses on optimizing metaheuristic algorithms. Streamlined design of these algorithms contributes to faster training and deployment times, opening the door to more effective and scalable AI-based forecasting solutions.

In this study, five groups of forecasting models—linear models, machine learning models, deep learning models, ensemble learning models, and chaotic models—are employed to capture the diverse characteristics of solar power data. These models were chosen to address linear trends, nonlinearity, temporal dependencies, and chaotic behavior in the dataset. Each model is rigorously implemented and validated using real-world data from the Konya Eregli SPP, which includes both endogenous variables and exogenous variables.

Among the implemented forecasting algorithms, seven emerged as the top performers based on standard evaluation metrics: FFNN, ELM, LSTM, GRU, RF, XGBoost, and ESN. These algorithms were subsequently adopted as the base models for further investigation. Each of the seven models was tested under scenarios that combined endogenous and exogenous inputs, and all were fine-tuned using the TPE hyperparameter optimization method to enhance forecasting accuracy.

Prior to model training, the dataset underwent a comprehensive cleaning procedure. Outliers were detected and removed using the Local Outlier Factor (LOF) method, followed by an exploratory data analysis phase. Once the data was cleaned and the time series characteristics were better understood, feature engineering strategies were applied to extract and transform relevant features. These strategies were tailored to the requirements of individual base models as well as the ensemble stacking meta-models. To ensure robust and leakage-free model evaluation, the processed dataset was divided

into training, validation, and test sets, establishing a strong foundation for both the base models and the meta-models.

Among the fine-tuned base models, LSTM and GRU demonstrated the highest levels of accuracy and reliability, followed by FFNN, XGBoost, RF, ESN, and ELM in that order. These results highlight the effectiveness of deep learning architectures (LSTM and GRU) in modeling the intricate time-based relationships present in solar power data, while also confirming the efficacy of more traditional machine learning and ensemble methods in particular scenarios.

To address chaotic behavior in the time series, the False Nearest Neighbors (FNN) method was incorporated as a preprocessing technique for reconstructing the phase space. This step determined an optimized embedding dimension intended to preserve the structural consistency of the original time series. The Echo State Network (ESN) algorithm was then employed to forecast solar energy production using both the original and embedded data. Results showed that optimizing the embedding dimension via the FNN technique significantly improved the model's capacity to learn and represent nonlinear dynamics underlying solar power generation, leading to more accurate forecasts.

The importance of data preprocessing—especially the selection of embedding dimensions—in chaotic modeling for solar energy systems is highlighted by the improved performance of the ESN model when applied to the embedded data. Overall, this approach demonstrated how combining proper data embedding with an optimized model architecture can lead to superior predictive capabilities, further contributing to efficient and reliable solar energy forecasting.

As the culminating step in this thesis, forecasting performance was further improved through employment of stacking ensemble ML techniques. The primary aim was to develop a novel metamodel that surpasses the prediction capabilities of the individual base models. Two categories of stacking ensemble models were investigated: regression-based stacking and neural network-based stacking. Each approach was critically evaluated and compared, leading to the identification of the most accurate model—an essential contribution of this research.

In the first category, traditional regression algorithms (Multi-Linear Regression, Ridge Regression, and Lasso Regression) functioned as the meta-model to combine the

predictions of the seven base models. This approach leverages the strengths of diverse base predictors through a linear combination, aiming to minimize overall forecast error. However, it ultimately did not surpass the established accuracy benchmark.

In the second category, neural networks (NN) functioned as the meta-model, exploiting their capacity to capture complex nonlinear relationships within the data. Three distinct neural network-based stacking strategies were developed:

1- NN Stacking with 7 Base Models

2- NN Stacking with 5 Base Models

3- Proposed Deep Learning-Based Stacking Model with 2 Base Models

The proposed deep learning-based stacking model was deliberately constructed using only the two best-performing base models, selected based on their individual accuracy metrics. To further enhance predictive performance, advanced techniques and optimized hyperparameters were incorporated. As a result, this final stacking model delivered the highest forecast accuracy among all models investigated. The thesis study framework is presented in Figure 7.1.

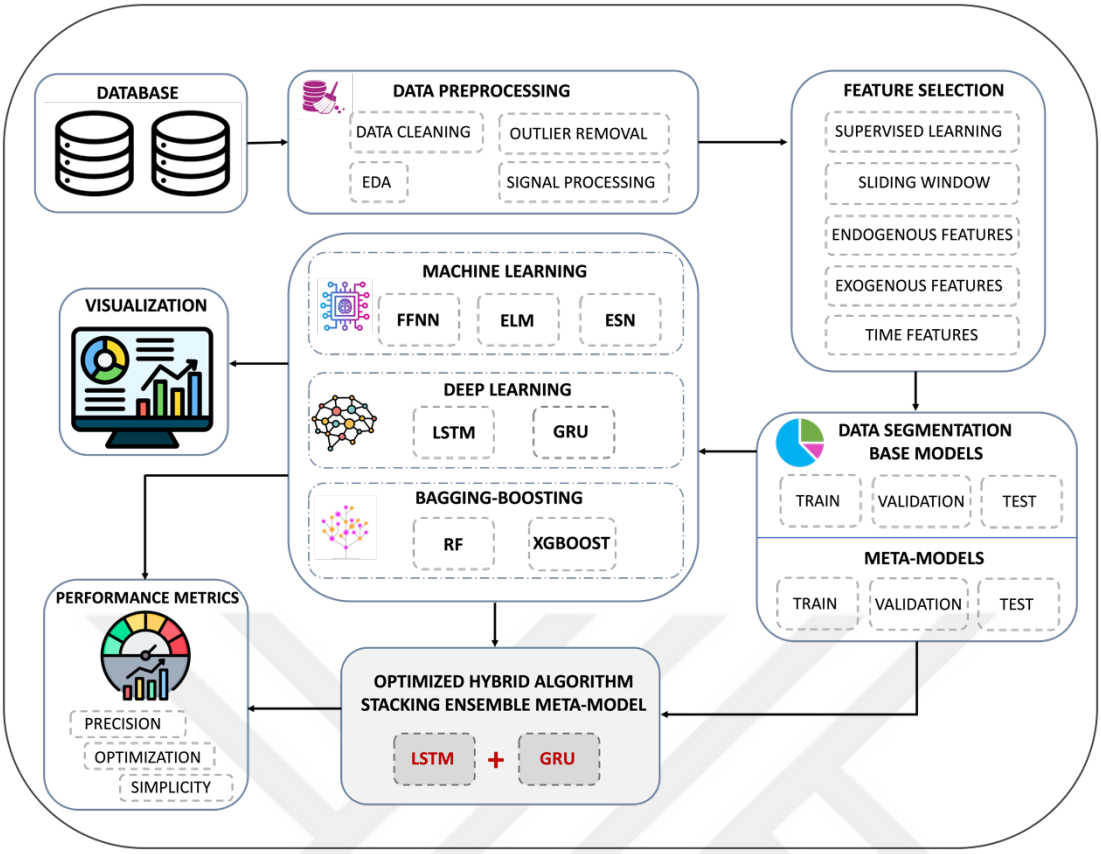

Figure 7.1 : Thesis study framework.

Several refinements were applied to the meta-models' designs, including the application of Model Output Statistics (MOS) as a post-processing technique. These refinements aimed to validate whether additional tuning could further elevate forecasting accuracy. The conclusive outcome of these enhancements is the Deep Learning-Based Stacking Meta-Model, introduced in Section 6.6, which successfully predicts solar energy output for the Konya Eregli Solar Power Plant with a precision exceeding 98%. This model not only surpasses the performance of the most accurate base model (LSTM) but also demonstrates the potential to meet the overarching goal of this study—establishing a new benchmark for high-accuracy solar energy forecasting.

Another notable contribution of the proposed deep learning-based stacking meta-model is its simpler architectural design compared to the other meta-models introduced in this thesis. By strategically focusing on only the two most accurate base models, the meta-model achieves high forecasting performance while minimizing computational overhead. This streamlined approach not only accelerates training and deployment but

also aligns with the broader research aim of optimizing metaheuristic model design for more efficient and scalable AI-based forecasting solutions.

Achieving this level of predictive precision highlights the transformative potential of optimized ensemble methods in solar energy forecasting. By enhancing accuracy, these advanced techniques can improve energy management, streamline grid integration, and guide strategic long-term planning in renewable energy systems. As a result, the proposed deep learning-based stacking metamodel emerges as a vital tool for advancing sustainable energy infrastructures.

7.1 Recommendations for Energy Producers

This study highlights several actionable recommendations for energy producers, particularly those involved in solar energy generation, to enhance operational efficiency, forecasting accuracy, and long-term sustainability.

First and foremost, the research demonstrates that advanced AI-driven forecasting techniques—particularly deep learning-based ensemble models—can significantly improve the precision of solar energy production forecasts. Data Analyzers within this business are encouraged to adopt hybrid and ensemble learning strategies that integrate both endogenous and exogenous variables. Models such as LSTM, GRU, and optimized stacked meta-models not only capture temporal dependencies but also outperform traditional forecasting models in accuracy and robustness. Investing in such high-precision forecasting systems can contribute directly to more effective grid integration, optimized energy trading, and improved maintenance scheduling.

Through ongoing discussions with Tegnatia Energy—a prominent EPC provider with over 400 MWp of installed solar capacity across 44 sites in Turkey—several business-driven insights have emerged. Tegnatia's extensive experience in real-time monitoring, performance analysis, and maintenance underlines the importance of tailoring forecasting strategies to the different stages of a solar project's lifecycle. During the initial investment and construction phase, higher-level timeframes such as daily forecasts are more valuable for strategic planning and investor decision-making. Conversely, in the post-installation and operational phase, when SCADA systems are actively monitoring the plant, lower-frequency data (10-minute to hourly forecasts) become more critical. These allow for timely detection of anomalies, quick response

to faults, and prevention of potential production losses—directly supporting maintenance and repair operations.

Moreover, industry practitioners emphasize that even a 1% increase in forecasting accuracy holds significant commercial value. This seemingly small improvement can directly impact pricing strategies, energy market positioning, and investor confidence. High-precision forecasts enable producers to develop more competitive bids, align supply with contractual obligations, and establish themselves as reliable energy partners in volatile markets. Given this, energy producers should not underestimate the compound financial and operational benefits of incremental gains in prediction accuracy.

Ultimately, the findings of this research reinforce the need for energy producers to prioritize data quality, invest in adaptive and scalable AI tools, and align forecasting strategies with the technical and economic demands of each project phase. These recommendations support the broader objective of building intelligent, resilient, and sustainable solar energy systems that can meet both current and future energy challenges.

7.2 Future Research Recommendations and Directions

Despite the achievements in this thesis, several avenues remain open for further inquiry into solar energy forecasting. First, an ongoing challenge in this field arises from the widespread use of black-box models, which provide limited understanding of the mathematical relationships between input variables and forecast targets. This lack of transparency can be an obstacle to informed policymaking, as stakeholders must rely on forecasts without fully understanding the underlying mechanics. As a result, developing more interpretable models is a critical research direction. A scarcity of sufficiently transparent machine learning (ML) approaches persists in both energy production and consumption contexts, and only a handful of recent studies have explicitly addressed interpretable ML techniques for large-scale energy consumption forecasting (see Eskandari et al. (2024) for an example). Future work should therefore focus on creating robust explanatory frameworks that elucidate model behavior while retaining forecasting accuracy.

A second area for future exploration involves the forecasting horizon. Although very short-term and short-term forecasts receive considerable attention in the literature, medium- and long-term forecasting has been relatively underexplored. Concentrating research efforts on longer time horizons is crucial for optimizing maintenance schedules, capacity planning, and power system operations. By generating more accurate forecasts over extended intervals, energy providers and policymakers can make better-informed decisions that directly impact infrastructure investments and resource allocation strategies.

Lastly, additional topics of rising importance warrant closer investigation, most notably hierarchical forecasting. This advanced method—often referred to as forecast reconciliation (Wickramasuriya et al, 2019) or high-dimensional forecasting (Pinson, 2013)—addresses the need to reconcile forecasts at multiple levels of aggregation, whether temporal, geographical, or both. In the context of solar energy, hierarchical forecasting becomes particularly relevant for validating that the combined output of individual photovoltaic strings aligns with the total generation of the entire plant. Progress in the area of hierarchical forecasting has been propelled by Rob Hyndman's research group (see Athanasopoulos et al. (2009) as a preliminary framework), which has demonstrated how base forecasts can be refined through post-processing in a regression framework.

Implementing such approaches more broadly in solar forecasting could significantly enhance overall prediction accuracy and operational coherence. Embracing these emerging methodologies, along with further innovations in model interpretability and forecasting horizons, holds considerable promise for shaping the next generation of solar energy forecasting research.

REFERENCES

Aggarwal, C. C. (2017). *Outlier Analysis*. Springer.

Aggarwal, S., & Saini, L. (2014). Solar energy prediction using linear and non-linear regularization models: A study on AMS (American Meteorological Society) 2013–14 Solar Energy Prediction Contest. *Energy, 78*, 247-256. https://doi.org/10.1016/j.energy.2014.10.012.

Ahmed, R., Sreeram, V., Mishra, Y., & Arif, M. (2020). A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization. *Renewable and Sustainable Energy Reviews, 124*, 109792. https://doi.org/10.1016/j.rser.2020.109792.

AlHakeem, D., Mandal, P., Haque, A. U., Yona, A., Senjyu, T., & Tseng, T.-L. (2015). A new strategy to quantify uncertainties of wavelet-GRNN-PSO based solar PV power forecasts using bootstrap confidence intervals. In *2015 IEEE power & energy society general meeting* (pp. 1-5). IEEE. https://doi.org/10.1109/PESGM.2015.7286233.

Amiri, B., Gómez-Orellana, A. M., Gutiérrez, P. A., Dizène, R., Hervás-Martínez, C., & Dahmani, K. (2021). A novel approach for global solar irradiation forecasting on tilted plane using Hybrid Evolutionary Neural Networks. *Journal of cleaner production, 287*, 125577. https://doi.org/10.1016/j.jclepro.2020.125577.

Antonanzas, J., Osorio, N., Escobar, R., Urraca, R., Martinez-de-Pison, F. J., & Antonanzas-Torres, F. (2016). Review of photovoltaic power forecasting. *Solar Energy, 136*, 78-111. https://doi.org/10.1016/j.solener.2016.06.069.

Assouline, D., Mohajeri, N., & Scartezzini, J.-L. (2017). Quantifying rooftop photovoltaic solar energy potential: A machine learning approach. *Solar Energy, 141*, 278-296. https://doi.org/10.1016/j.solener.2016.11.045.

Athanasopoulos, G., Ahmed, R. A., & Hyndman, R. J. (2009). Hierarchical forecasts for Australian domestic tourism. *International Journal of Forecasting, 25*(1), 146-166. https://doi.org/10.1016/j.ijforecast.2008.07.004.

Aybar-Ruiz, A., Jiménez-Fernández, S., Cornejo-Bueno, L., Casanova-Mateo, C., Sanz-Justo, J., Salvador-González, P., & Salcedo-Sanz, S. (2016). A novel grouping genetic algorithm–extreme learning machine approach for global solar radiation prediction from numerical weather models inputs. *Solar Energy, 132*, 129-142. https://doi.org/10.1016/j.solener.2016.03.015.

Barbieri, F., Rajakaruna, S., & Ghosh, A. (2017). Very short-term photovoltaic power forecasting with cloud modeling: A review. *Renewable and Sustainable Energy Reviews, 75*, 242-263. https://doi.org/10.1016/j.rser.2016.10.068.

Basurto, N., Arroyo, Á., Vega, R., Quintián, H., Calvo-Rolle, J. L., & Herrero, Á. (2019). A hybrid intelligent system to forecast solar energy production. *Computers & electrical engineering, 78*, 373-387. https://doi.org/10.1016/j.compeleceng.2019.07.023.

Bebis, G., & Georgiopoulos, M. (1994). Feed-forward neural networks. *Ieee Potentials*, 13(4), 27-31; Doi: 10.1109/45.329294.

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for Hyper-Parameter Optimization. *Advances in Neural Information Processing Systems (NeurIPS),* 24.

Bergstra, J., Yamins, D., & Cox, D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning* (pp. 115-123). PMLR.

Bergstra, J., Yamins, D., & Cox, D. D. (2013). Hyperopt: A Python Library for Optimizing the Hyperparameters of Machine Learning Algorithms. *SciPy*, 13, 20.

Bianchi, F. M., Livi, L., & Alippi, C. (2017). Investigating echo state networks dynamics by means of recurrence analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 29(2), 627-638.

Bizzarri, F., Bongiorno, M., Brambilla, A., Gruosso, G., & Gajani, G. S. (2012). Model of photovoltaic power plants for performance analysis and production forecast. *IEEE transactions on sustainable energy, 4(2)*, 278-285. https://doi.org/10.1109/TSTE.2012.2219563.

Blaga, R., Sabadus, A., Stefu, N., Dughir, C., Paulescu, M., & Badescu, V. (2019). A current perspective on the accuracy of incoming solar energy forecasting. *Progress in energy and combustion science, 70*, 119-144. https://doi.org/10.1016/j.pecs.2018.10.003.

Box, G. E. P. & Jenkins, J. M. (1976). *Time Series Analysis: Forecasting and Control*. San Francisco, CA.: Holden-Day.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5-32.

Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 93–104.

Brown, R. G. (1959). *Statistical Forecasting for Inventory Control*. McGraw-Hill.

Brownlee, J. (2018). *Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery.

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. *Proceedings of the 23rd International Conference on Machine Learning*, 161-168.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3), 1-58.

Chen, S.-M., Chang, Y.-C., Chen, Z.-J., & Chen, C.-L. (2013). Multiple fuzzy rules interpolation with weighted antecedent variables in sparse fuzzy rule-based systems. *International Journal of Pattern Recognition and Artificial Intelligence, 27(05)*, 1359002. https://doi.org/10.1142/S0218001413590027.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794.

Chen, T., He, T., & Benesty, M. (2015). XGBoost: Extreme Gradient Boosting. R package version 0.4-2, 1(4).

Cheung, Y. W., & Lai, K. S. (1995). Lag order and critical values of the augmented Dickey–Fuller test. *Journal of Business & Economic Statistics*, 13(3), 277-280; Doi: https://doi.org/10.1080/07350015.1995.10524601.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv*:1406.1078.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv*:1412.3555.

Colak, T., & Qahwaji, R. (2007). Automatic sunspot classification for real-time forecasting of solar activities. In *2007 3rd international conference on recent advances in space technologies* (pp. 733-738). IEEE. https://doi.org/10.1109/RAST.2007.4284089**.**

Cutler, D. R., Edwards Jr., T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J., & Lawler, J. J. (2007). Random forests for classification in ecology. *Ecology*, 88(11), 2783-2792.

Dambreville, R., Blanc, P., Chanussot, J., & Boldo, D. (2014). Very short term forecasting of the global horizontal irradiance using a spatio-temporal autoregressive model. *Renewable energy, 72*, 291-300. https://doi.org/10.1016/j.renene.2014.07.012.

Das, U. K., Tey, K. S., Seyedmahmoudian, M., Mekhilef, S., Idris, M. Y. I., Van Deventer, W., Horan, B., Stojcevski, A. (2018). Forecasting of photovoltaic power generation and model optimization: A review. *Renewable and Sustainable Energy Reviews, 81*, 912-928. https://doi.org/10.1016/j.rser.2017.08.017.

David, M., Ramahatana, F., Trombe, P.-J., & Lauret, P. (2016). Probabilistic forecasting of the solar irradiance with recursive ARMA and GARCH models. *Solar Energy, 133*, 55-72. https://doi.org/10.1016/j.solener.2016.03.064.

de Freitas Viscondi, G., & Alves-Souza, S. N. (2019). A Systematic Literature Review on big data for solar photovoltaic electricity generation forecasting. *Sustainable Energy Technologies and Assessments, 31*, 54-63. https://doi.org/10.1016/j.seta.2018.11.008.

Diagne, M., David, M., Lauret, P., Boland, J., & Schmutz, N. (2013). Review of solar irradiance forecasting methods and a proposition for small-scale insular grids. *Renewable and Sustainable Energy Reviews, 27*, 65-76. https://doi.org/10.1016/j.rser.2013.06.042.

Dietterich, T. G. (2000). Ensemble methods in machine learning. *International Workshop on Multiple Classifier Systems.* Springer.

Ding, M., Wang, L., & Bi, R. (2011). An ANN-based approach for forecasting the power output of photovoltaic system. *Procedia Environmental Sciences, 11*, 1308-1315. https://doi.org/10.1016/j.proenv.2011.12.196.

Dolara, A., Grimaccia, F., Leva, S., Mussetta, M., & Ogliari, E. (2015). A physical hybrid artificial neural network for short term forecasting of PV plant power output. *Energies, 8(2)*, 1138-1153. https://doi.org/10.3390/en8021138.

Doorga, J. R. S., Dhurmea, K. R., Rughooputh, S., & Boojhawon, R. (2019). Forecasting mesoscale distribution of surface solar irradiation using a proposed hybrid approach combining satellite remote sensing and time series models. *Renewable and Sustainable Energy Reviews, 104*, 69-85. https://doi.org/10.1016/j.rser.2018.12.055.

Eskandari, H., Saadatmand, H., Ramzan, M., & Mousapour, M. (2024). Innovative framework for accurate and transparent forecasting of energy consumption: A fusion of feature selection and interpretable machine learning. *Applied Energy, 366*, 123314. https://doi.org/10.1016/j.apenergy.2024.123314.

Francisco, M. M., Alves-Souza, S. N., Campos, E. G., & De Souza, L. S. (2017). Total data quality management and total information quality management applied to costumer relationship management. In *Proceedings of the 9th international conference on information management and engineering* (pp. 40-45). https://doi.org/10.1145/3149572.3149575.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189-1232.

Fu, T., & Wang, C. (2018). A hybrid wind speed forecasting method and wind energy resource analysis based on a swarm intelligence optimization algorithm and an artificial intelligence model. *Sustainability, 10(11)*, 3913. https://doi.org/10.3390/su10113913.

Gala, Y., Fernández, Á., Díaz, J., & Dorronsoro, J. R. (2016). Hybrid machine learning forecasting of solar radiation values. *Neurocomputing, 176*, 48-59. https://doi.org/10.1016/j.neucom.2015.02.078.

Gandelli, A., Grimaccia, F., Leva, S., Mussetta, M., & Ogliari, E. (2014). Hybrid model analysis and validation for PV energy production forecasting. In *2014 international joint conference on neural networks (IJCNN)* (pp. 1957-1962). IEEE. https://doi.org/10.1109/IJCNN.2014.6889786.

Garnett, R. (2023). *Bayesian Optimization*. Cambridge University Press.

Glahn, H. R., & Lowry, D. A. (1972). The use of model output statistics (MOS) in objective weather forecasting. *Journal of Applied Meteorology (1962-1982)*, 1203-1211. Doi: https://www.jstor.org/stable/26176961.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Graves, A. (2013). *Supervised sequence labelling with recurrent neural networks* (Doctoral dissertation). Retrieved from http://scholar.google.com/

Guermoui, M., Melgani, F., Gairaa, K., & Mekhalfi, M. L. (2020). A comprehensive review of hybrid models for solar radiation forecasting. *Journal of cleaner production, 258*, 120357. https://doi.org/10.1016/j.jclepro.2020.120357.

Haessig, P., Multon, B., Ahmed, H. B., Lascaud, S., & Bondon, P. (2015). Energy storage sizing for wind power: impact of the autocorrelation of day-ahead forecast errors. *Wind Energy, 18(1)*, 43-57. https://doi.org/10.1002/we.1680.

Hanna, R., Kleissl, J., Nottrott, A., & Ferry, M. (2014). Energy dispatch schedule optimization for demand charge reduction using a photovoltaic-battery storage system with solar forecasting. *Solar Energy, 103*, 269-287. https://doi.org/10.1016/j.solener.2014.02.020.

Hegger, R., Kantz, H., & Schreiber, T. (1999). Practical implementation of nonlinear time series methods: The TISEAN package. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 9(2), 413-435.

Hernández-Torres, D., Bridier, L., David, M., Lauret, P., & Ardiale, T. (2015). Technico-economical analysis of a hybrid wave power-air compression storage system. *Renewable energy, 74*, 708-717. https://doi.org/10.1016/j.renene.2014.08.070.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 359–366. Doi: https://doi.org/10.1016/0893-6080(89)90020-8.

Hu, J., Gao, P., Yao, Y., & Xie, X. (2014). Traffic flow forecasting with particle swarm optimization and support vector regression. In *17th international ieee conference on intelligent transportation systems (itsc)* (pp. 2267-2268). IEEE.https://doi.org/10.1109/ITSC.2014.6958049.

Huang, G. B., Wang, D. H., & Lan, Y. (2012). Extreme learning machines: A survey. *International Journal of Machine Learning and Cybernetics*, 2(2), 107-122.

Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1-3), 489-501.

Huang, Y., Lu, J., Liu, C., Xu, X., Wang, W., & Zhou, X. (2010). Comparative study of power forecasting methods for PV stations. In *2010 International Conference on Power System Technology* (pp. 1-6). IEEE. https://doi.org/10.1109/POWERCON.2010.5666688.

Hyndman, R. & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts.

Jaeger, H. (2001). The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *German National Research Center for Information Technology GMD Technical Report*. Bonn, Germany, 148(34), 13.

Jiang, H., Dong, Y., & Xiao, L. (2017). A multi-stage intelligent approach based on an ensemble of two-way interaction model for forecasting the global horizontal radiation of India. *Energy conversion and management, 137*, 142-154. https://doi.org/10.1016/j.enconman.2017.01.040.

Jiménez-Pérez, P. F., & Mora-López, L. (2014). Modeling daily profiles of solar global radiation using statistical and data mining techniques. In *Advances in Intelligent Data Analysis XIII: 13th International Symposium, IDA 2014, Leuven, Belgium, October 30–November 1, 2014. Proceedings 13* (pp. 155-166). Springer International Publishing. https://doi.org/10.1007/978-3-319-12571-8_14.

Jolliffe, I. (2011). Principal component analysis in International encyclopedia of statistical science. *Berlin, Heidelberg: Springer Berlin Heidelberg*, 1094-1096.

Karim, S., Lucey, B. M., Naeem, M. A., & Uddin, G. S. (2022). Examining the interrelatedness of NFTs, DeFi tokens and cryptocurrencies. *Finance Research Letters*, 47, 102696. Doi: https://doi.org/10.1016/j.frl.2022.102696.

Karim, S., Lucey, B. M., Naeem, M. A., & Vigne, S. A. (2023). The dark side of Bitcoin: do Emerging Asian Islamic markets help subdue the ethical risk? *Emerging Markets Review*, 54, 100921. Doi: https://doi.org/10.1016/j.ememar.2022.100921.

Kennel, M. B., Brown, R., & Abarbanel, H. D. (1992). Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Physical review A*, 45(6), 3403.

Korstanje, J. (2021). *Time Series Forecasting with LSTM Networks*. Springer.

Kostylev, V., & Pavlovski, A. (2011). Solar power forecasting performance–towards industry standards. *1st international workshop on the integration of solar power into power systems*, Aarhus, Denmark (pp. 1-8).

Kühnert, J., Lorenz, E., & Heinemann, D. (2013). Satellite-based irradiance and power forecasting for the German energy market. *Solar energy forecasting and resource assessment*, 267-297.

Lauret, P., Voyant, C., Soubdhan, T., David, M., & Poggi, P. (2015). A benchmarking of machine learning techniques for solar radiation forecasting in an insular context. *Solar Energy, 112*, 446-457. https://doi.org/10.1016/j.solener.2014.12.014.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. Doi: https://doi.org/10.1038/nature14539.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.

Li, J., Ward, J. K., Tong, J., Collins, L., & Platt, G. (2016). Machine learning for solar irradiance forecasting of photovoltaic system. *Renewable energy, 90*, 542-553. https://doi.org/10.1016/j.renene.2015.12.069.

Li, L.-L., Zhao, X., Tseng, M.-L., & Tan, R. R. (2020). Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm. *Journal of cleaner production, 242*, 118447. https://doi.org/10.1016/j.jclepro.2019.118447.

Li, Y., Sun, Q., Lehman, B., Lu, S., Hamann, H. F., Simmons, J., & Black, J. (2016). A machine-learning approach for regional photovoltaic power forecasting. In *2016 IEEE Power and Energy Society General Meeting (PESGM)* (pp. 1-5). IEEE. https://doi.org/10.1109/PESGM.2016.7741991.

Li, Z., Rahman, S. M., Vega, R., & Dong, B. (2016). A hierarchical approach using machine learning methods in solar photovoltaic energy production forecasting. *Energies, 9(1)*, 55. https://doi.org/10.3390/en9010055.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R News*, 2(3), 18-22.

Lorenz, E., Hurka, J., Heinemann, D., & Beyer, H. G. (2009). Irradiance forecasting for the power prediction of grid-connected photovoltaic systems. *IEEE Journal of selected topics in applied earth observations and remote sensing, 2(1)*, 2-10. Doi: 10.1109/JSTARS.2009.2020300.

Lou, S., Li, D. H., Lam, J. C., & Chan, W. W. (2016). Prediction of diffuse solar irradiance using machine learning and multivariable regression. *Applied energy, 181*, 367-374. https://doi.org/10.1016/j.apenergy.2016.08.093.

Lukoševičius, M., & Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3), 127-149.

Muntean, M., Tulbure, A., & Rotar, C. (2016). Data mining methods for parameters forecasting of a small solar plant. In *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies VIII* (Vol. 10010, pp. 343-348). SPIE. https://doi.org/10.1117/12.2245896.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. *In Proceedings of the 27th International Conference on Machine Learning* (pp. 807–814).

Pedro, H. T., & Coimbra, C. F. (2012). Assessment of forecasting techniques for solar power production with no exogenous inputs. *Solar Energy, 86(7)*, 2017-2028. https://doi.org/10.1016/j.solener.2012.04.004.

Pinson, P. (2013). Wind energy: Forecasting challenges for its operational management. *Statistical Science, 28*(4), 564-585. https://doi.org/10.1214/13-STS445.

Rao, A., Lucey, B., Kumar, S., & Lim, W. M. (2023). Do green energy markets catch cold when conventional energy markets sneeze? *Energy Economics*, 127, 107035. Doi: https://doi.org/10.1016/j.eneco.2023.107035.

Reikard, G. (2009). Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy, 83(3)*, 342-349. https://doi.org/10.1016/j.solener.2008.08.007.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088), 533-536. Doi: https://doi.org/10.1038/323533a0.

Russo, M., Leotta, G., Pugliatti, P., & Gigliucci, G. (2014). Genetic programming for photovoltaic plant output forecasting. *Solar Energy, 105*, 264-273. https://doi.org/10.1016/j.solener.2014.02.021.

Saadati, T., & Barutcu, B. (2024a). Forecasting solar energy: Leveraging artificial intelligence and machine learning for sustainable energy solutions. *Journal of Economic Surveys*. Doi: 10.1111/joes.12678.

Saadati, T., & Barutcu, B. (2024b). Optimized solar energy forecasting for sustainable development using machine learning, deep learning, and chaotic models. *International Journal of Energy Economics and Policy*, 15(1), (pp.110-120). Doi: https://doi.org/10.32479/ijeep.17766.

Saadati, T., & Barutcu, B. (2024c). Enhancing solar energy forecasting through Chaos theory and Echo State Networks: A case study of the Konya Eregli solar power plant. *The 4th International Conference on Energy, Environment and Storage of Energy (ICEESEN)*. Cappadocia, Turkiye: Cappadocia University.

Saadati, T., & Barutcu, B. (2024d). Solar Energy Production Time series data, *Mendeley Data,* V2, Doi: 10.17632/2tpv28kr83.2.

Saadati, T., & Taskin, G. (2024). A Sensitivity-Based Explainable Method For Remote Sensing Scene Classification. In *IGARSS 2024-2024 IEEE International Geoscience and Remote Sensing Symposium* (pp. 3026-3029). IEEE. Doi: 10.1109/IGARSS53475.2024.10641899.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85-117.

Seabold, S., & Perktold, J. (2010). Statsmodels: econometric and statistical modeling with python. *SciPy*, 7(1).

Serana. (2020). Time Series Modelling — ARIMA and ETS. *Medium*. Retrieved December 23, 2024, from https://medium.com/@serana.ai/time-series-modelling-arima-and-ets-cafc904b9183.

Shahi, S., Fenton, F. H., & Cherry, E. M. (2022). Prediction of chaotic time series using recurrent neural networks and reservoir computing techniques: A comparative study. *Machine learning with applications*, *8*, 100300; Doi: https://doi.org/10.1016/j.mlwa.2022.100300.

Shahi, T. B., Ma, W., Zhang, Q., & Zhang, Z. (2022). A review of reservoir computing and its applications. *Neural Networks*, 152, 151-170.

Sobri, S., Koohi-Kamali, S., & Rahim, N. A. (2018). Solar photovoltaic generation forecasting methods: A review. *Energy conversion and management, 156*, 459-497. https://doi.org/10.1016/j.enconman.2017.11.019.

Soman, S. S., Zareipour, H., Malik, O., & Mandal, P. (2010). A review of wind power and wind speed forecasting methods with different time horizons. In *North American power symposium 2010* (pp. 1-8). IEEE. Doi: https://doi.org/10.1109/NAPS.2010.5619586.

Sözen, A., Arcaklioğlu, E., Özalp, M., & Kanit, E. G. (2004). Use of artificial neural networks for mapping of solar potential in Turkey. *Applied energy, 77(3)*, 273-286. https://doi.org/10.1016/S0306-2619(03)00137-5.

Tiba, S., & Belaid, F. (2021). Modeling the nexus between sustainable development and renewable energy: the African perspectives. *Journal of Economic Surveys*, 35(1), 307-329. Doi: https://doi.org/10.1111/joes.12401.

Voyant, C., Notton, G., Kalogirou, S., Nivet, M.-L., Paoli, C., Motte, F., & Fouilloy, A. (2017). Machine learning methods for solar radiation forecasting: A review. *Renewable energy, 105*, 569-582. https://doi.org/10.1016/j.renene.2016.12.095.

Vrettos, E., & Gehbauer, C. (2019). A hybrid approach for short-term PV power forecasting in predictive control applications. In *2019 IEEE Milan PowerTech* (pp. 1-6). IEEE. Doi: 10.1109/PTC.2019.8810672.

Watanabe, S. (2023). Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance. *arXiv preprint* arXiv:2304.11127. Doi: https://doi.org/10.48550/arXiv.2304.11127.

Watanabe, S., & Hutter, F. (2022). c-TPE: Tree-structured Parzen estimator with inequality constraints for expensive hyperparameter optimization. *arXiv preprint arXiv*:2211.14411.

Watanabe, S., Awad, N., Onishi, M., & Hutter, F. (2022). Speeding up multi-objective hyperparameter optimization by task similarity-based meta-learning for the tree-structured parzen estimator. *arXiv preprint arXiv*:2212.06751.

Wickramasuriya, S. L., Athanasopoulos, G., & Hyndman, R. J. (2019). Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Association, 114*(526), 804-819. https://doi.org/10.1080/01621459.2018.1448825.

Wolpert, D. H. (1992). Stacked generalization. *Neural networks, 5*(2), 241-259.

Yang, D., & van der Meer, D. (2021). Post-processing in solar forecasting: Ten overarching thinking tools. *Renewable and Sustainable Energy Reviews, 140*, 110735. https://doi.org/10.1016/j.rser.2021.110735.

Yona, A., Senjyu, T., Funabashi, T., & Kim, C.-H. (2013). Determination method of insolation prediction with fuzzy and applying neural network for long-term ahead PV power output correction. *IEEE transactions on sustainable energy, 4(2)*, 527-533. https://doi.org/10.1109/TSTE.2013.2246591.

Yona, A., Senjyu, T., Saber, A. Y., Funabashi, T., Sekine, H., & Kim, C.-H. (2008). Application of neural network to 24-hour-ahead generating power forecasting for PV system. In *2008 IEEE Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century* (pp. 1-6). IEEE. https://doi.org/10.1109/PES.2008.4596295.

Zhang, J., Ding, S., & Li, J. (2015a). Forecasting solar energy production using extreme learning machine. *Renewable Energy*, 81, 584-591.

Zhang, J., Florita, A., Hodge, B.-M., Lu, S., Hamann, H. F., Banunarayanan, V., & Brockway, A. M. (2015b). A suite of metrics for assessing the performance of solar power forecasting. *Solar Energy, 111*, 157-175. https://doi.org/10.1016/j.solener.2014.10.016.

Zhou, X., Liu, C., Luo, Y., Wu, B., Dong, N., Xiao, T., & Zhu, H. (2022). Wind power forecast based on variational mode decomposition and long short term memory attention network. *Energy Reports, 8*, 922-931. https://doi.org/10.1016/j.egyr.2022.08.159.

APPENDICES

APPENDIX A: Linear models forecasting results.
APPENDIX B: LSTM model implementation on embedded data.

APPENDIX A: Linear models forecasting results.

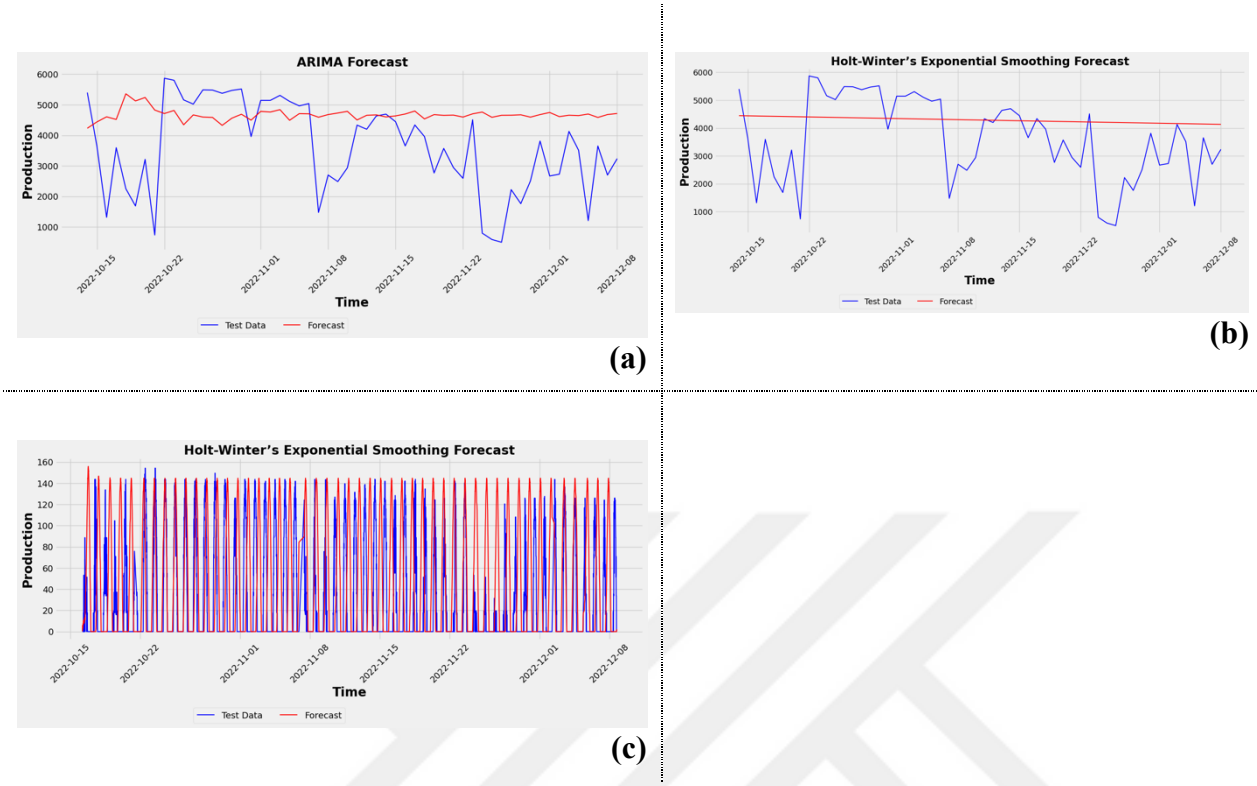

Figure A.1 : Forecasting results: (a) Auto regressive moving average. (b) Exponential smoothing on daily data (no seasonality). (c) Exponential smoothing on original data.

Table A.1 : Exponential smoothing model (with no seasonality) performance.

R-squared	MAPE	RMSE	MSE	MAE
-0.1691	0.3567	1612.23	2599309.8	1286.2

Table A.2 : Exponential smoothing model (with seasonality) performance.

R-squared	MAPE	RMSE	MSE	MAE
-1.5544	1.8582	71.44	5104.2	50.5

Table A.3 : ARIMA model (with no seasonality) performance.

R-squared	MAPE	RMSE	MSE	MAE
-0.5891	0.4100	1879.63	3533039.2	1478.5

APPENDIX B: LSTM model implementation on embedded data.

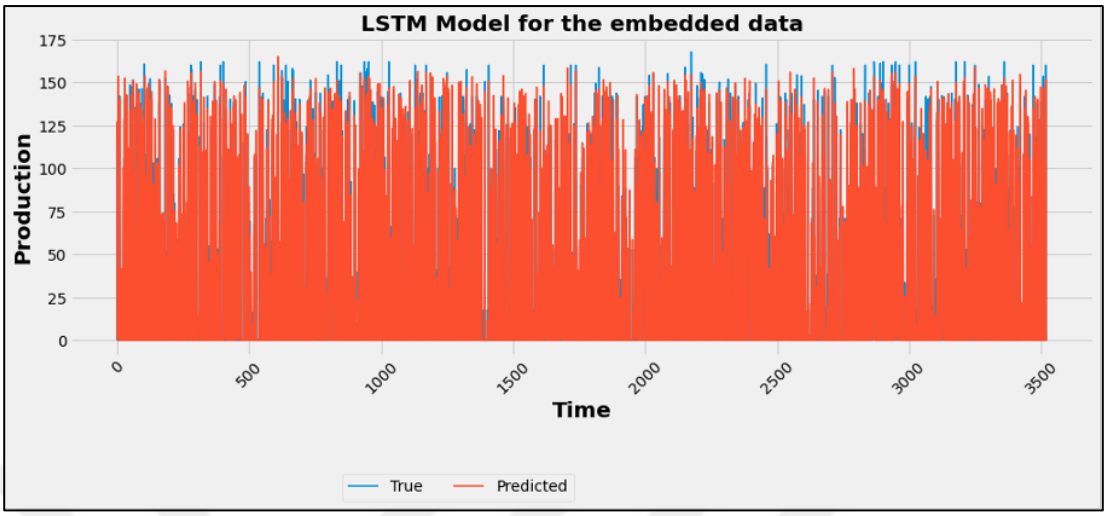

Figure B.1: LSTM forecasting result on the embedded data.

Table B.1 : LSTM model on the embedded data forecasting performance.

R-squared	MAPE	RMSE	MSE	MAE
0.9765	0.1190	8.08	65.3412	4.2295

The results indicate that the optimized LSTM model performs better on the original time series data ($R^2 = 0.9784$) compared to the embedded data.

CURRICULUM VITAE

Name Surname **: Taraneh SAADATI**

EDUCATION

- **B.Sc.** **:** 2010, University of Tehran, Mathematics Department

- **M.Sc.** **:** 2018, Kharazmi University, Industrial Engineerin

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2023 – 2024 Visiting PhD - Research Assistant at Cardiff University
- Jan 2023-Mar 2024 Scientific and Technological Research Council of Republic of Turkey (TUBITAK) - Research Scholar - R&D fellow in the 1001 granted project: "A Novel Explainable AI method based on Sensitivity Analysis and its Applications to Deep Learning Models for Remote Sensing Images".
- 2022 – 2025 Scientific Research Projects (BAP) Coordination Unit at Istanbul Technical University - Research Scholar [Project ID: 44133, Project code: MDK-2022-44133]

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Saadati, T., & Barutcu, B.** (2024). Forecasting solar energy: Leveraging artificial intelligence and machine learning for sustainable energy solutions. Journal of Economic Surveys. Doi: 10.1111/joes.12678.
- **Saadati, T., & Barutcu, B.** (2024). Optimized solar energy forecasting for sustainable development using machine learning, deep learning, and chaotic models. International Journal of Energy Economics and Policy, 15(1), (pp.110-120). Doi: https://doi.org/10.32479/ijeep.17766.
- **Saadati, T., & Barutcu, B.** (2024). Enhancing solar energy forecasting through Chaos theory and Echo State Networks: A case study of the Konya Eregli solar power plant. The 4th International Conference on Energy, Environment and Storage of Energy (ICEESEN). Cappadocia, Turkiye: Cappadocia University.
- **Saadati, T., & Barutcu, B.** (2024). Solar Energy Production Time series data, Mendeley Data, V2, Doi: 10.17632/2tpv28kr83.2.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- **Saadati, T., & Taskin, G.** (2024). A Sensitivity-Based Explainable Method For Remote Sensing Scene Classification. In IGARSS 2024-2024 IEEE International Geoscience and Remote Sensing Symposium (pp. 3026-3029). IEEE. Doi: 10.1109/IGARSS53475.2024.10641899.