Université de Lille

Design and Development of Intelligent Air Quality Indicators for Environmental and Health Preservation Support

Idriss JAIRI

Submitted in fulfillment of the requirements for the degree of

PhD in Computer Science

Doctoral School of MADIS – MAthematics and DIgital Sciences
University of Lille

CRIStAL UMR CNRS 9189 – Centre de Recherche en Informatique,
Signal et Automatique de Lille

Defense Date: 26 September 2025

Examining Committee:

President:	Prof. Slim Hammadi, École Centrale de Lille
Reviewers:	Prof. Germain Forestier, ENSISA / University of Haute Alsace
	Prof. Maguelonne Teisseire, Research Director at INRAE Montpellier
Examiners:	Prof. Stéphane Le Calvé, Research Director at CNRS Strasbourg
	Dr. Agnieszka Rorat, Modeling Studies Project Manager at Atmo Hauts-de-France
Supervisors:	Prof. Hayfa Zgaya, Thesis Director & Supervisor, University of Lille
	Dr. Ludivine Canivet, Co-Supervisor, University of Lille
	Dr. Sarah Ben-Othman, Co-Supervisor, Polytech Lille

Université de Lille

Conception et développement d'indicateurs intelligents de qualité de l'air pour soutenir la préservation de l'environnement et de la santé

Idriss JAIRI

Thèse présentée en vue de l'obtention du grade de

Doctorat en Informatique

École doctorale : MADIS – MAthematics and DIgital Sciences
Université de Lille

CRIStAL UMR CNRS 9189 – Centre de Recherche en Informatique,
Signal et Automatique de Lille

Date de soutenance : 26 September 2025

Président :	Prof. Slim Hammadi, École Centrale de Lille
Rapporteurs :	Prof. Germain Forestier, ENSISA / Université de Haute-Alsace
	Prof. Maguelonne Teisseire, Directrice de recherche à l'INRAE Montpellier
Examinateurs :	Prof. Stéphane Le Calvé, Directeur de recherche au CNRS Strasbourg
	Dr. Agnieszka Rorat, Cheffe de projet études modélisation à Atmo Hauts-de-France
Encadrement :	Prof. Hayfa Zgaya, Directrice de thèse, Université de Lille
	Dr. Ludivine Canivet, Co-encadrante, Université de Lille
	Dr. Sarah Ben-Othman, Co-encadrante, Polytech Lille

Author's Declaration

I declare that the work presented in this thesis is entirely my own original research, conducted under the supervision of Prof. Hayfa Zgaya and co-supervision of Dr. Ludivine Canivet and Dr. Sarah Ben-Othman at the University of Lille. Where the work of others has been used or referenced, it has been clearly acknowledged and properly cited.

This thesis has been carried out in accordance with the University of Lille's regulations and the Code of Practice for Research Degree Programs.

I confirm that this thesis has not been submitted, either in whole or in part, for any degree, diploma, or other qualification at this or any other institution. Where parts of this work have been published in peer-reviewed journals or presented at conferences, appropriate acknowledgment has been made within the thesis.

The datasets compiled and methodologies developed as part of this research are made available for the benefit of the scientific community, subject to appropriate data sharing agreements and ethical considerations.

I take full responsibility for the accuracy of the information presented and the validity of the conclusions drawn from this research.

Idriss JAIRI

Abstract

Air pollution poses an escalating global threat to public health and environmental sustainability, with traditional monitoring systems struggling to provide the timely, accurate, and interpretable information essential for effective decision-making. This thesis addresses these critical limitations by developing intelligent air quality indicators that integrate advanced artificial intelligence (AI) techniques with environmental science to create more effective tools for environmental preservation and health protection.

This research makes three fundamental contributions to environmental intelligence systems. First, we introduce a novel granular computing (GrC) framework for explainable air quality classification that transforms opaque machine learning (ML) models into transparent decision support systems. We propose the AirQ-RuleGrCEx algorithm as our primary theoretical contribution—a novel and effective approach that systematically extracts human-readable rules by leveraging granular computing principles to explicitly link meteorological conditions to air pollution levels. This algorithm represents a significant methodological advancement in the field of explainable artificial intelligence (XAI) for environmental applications, achieving competitive accuracy while providing enhanced and meaningful interpretability for environmental stakeholders and policymakers. Through systematic development and refinement, we aim to establish this approach as one of the most promising algorithms for interpretable environmental classification, bridging the critical gap between predictive performance and decision transparency.

Second, we develop a comprehensive neural transfer learning paradigm through three innovative applications that significantly improve air quality forecasting

efficiency and scalability. Our first application demonstrates successful cross-pollutant knowledge transfer using artificial neural networks (ANNs), where models pre-trained on $PM_{2.5}$ data are effectively fine-tuned to predict other pollutants (PM_{10}, SO_2, NO_2, CO, O_3), enabling efficient modeling in data-scarce scenarios. The second application extends temporal transfer learning from hourly to multi-horizon forecasting (6-hour, 12-hour, and daily), allowing models to adapt across different prediction timescales with minimal computational overhead. The third application introduces our novel LagEnsembleForecasting framework, which combines four specialized deep learning models trained on different temporal lag sequences (1h, 6h, 12h, 24h) to provide enhanced prediction accuracy and natural uncertainty quantification through ensemble variance, with successful transferability to other pollutants demonstrating the framework's versatility.

Third, we advance particulate matter (PM) risk assessment through machine learning-driven toxicity threshold prediction, addressing the limitations of costly and time-intensive laboratory approaches. By leveraging a comprehensive dataset compiled from over 40 peer-reviewed studies, our framework employs five complementary algorithms with SHAP-based explainability (SHapley Additive exPlanations, a method for interpreting machine learning predictions) to distinguish toxic from non-toxic PM concentrations. This approach provides more nuanced toxicity assessments than conventional methods while identifying key compositional factors that drive toxicity outcomes.

The integration of these contributions establishes a comprehensive framework for intelligent environmental monitoring that enhances prediction capabilities, provides interpretable decision support, and enables more effective health risk assessment. Validation across multiple real-world datasets confirms the practical applicability and scalability of our approaches. Importantly, while this research focuses primarily on air quality, the methodological foundations and transfer learning principles established here create a pathway toward integrated water-air-soil monitoring strategies, where the same intelligent frameworks could be adapted and extended to create unified environmental management systems that holistically address the interconnected nature of environmental challenges across multiple domains.

This research demonstrates how advanced AI techniques can transform environ-

mental monitoring from reactive to proactive systems, providing decision-makers with the transparent, accurate, and actionable intelligence needed to address complex pollution challenges and protect public health in an era of increasing environmental pressures.

Résumé

La pollution atmosphérique représente une menace mondiale croissante pour la santé publique et la durabilité environnementale, les systèmes de surveillance traditionnels peinant à fournir les informations opportunes, précises et interprétables essentielles à une prise de décision efficace. Cette thèse aborde ces limitations critiques en développant des indicateurs intelligents de qualité de l'air qui intègrent des techniques avancées d'intelligence artificielle (IA) à la science environnementale pour créer des outils plus efficaces de préservation environnementale et de protection de la santé. Cette recherche apporte trois contributions fondamentales aux systèmes d'intelligence environnementale. Premièrement, nous introduisons un nouveau cadre de calcul granulaire (GrC) pour la classification explicable de la qualité de l'air qui transforme des modèles d'apprentissage automatique (ML) opaques en systèmes d'aide à la décision transparents. Nous proposons l'algorithme `AirQ-RuleGrCEx` comme notre principale contribution théorique—une approche novatrice et efficace qui extrait systématiquement des règles lisibles par l'humain en exploitant les principes du calcul granulaire pour relier explicitement les conditions météorologiques aux niveaux de pollution atmosphérique. Cet algorithme représente une avancée méthodologique significative dans le domaine de l'intelligence artificielle explicable (XAI) pour les applications environnementales, atteignant une précision compétitive tout en offrant une interprétabilité améliorée et significative pour les parties prenantes environnementales et les décideurs politiques. Grâce à un développement et un raffinement systématiques, nous visons à établir cette approche comme l'un des algorithmes les plus prometteurs pour la classification environnementale interprétable, comblant ainsi le fossé critique entre performance prédictive et transparence décisionnelle. Deuxièmement, nous développons un paradigme complet d'apprentissage par transfert neuronal à travers trois applications innovantes qui améliorent significativement l'efficacité et l'évolutivité de la prévision de la qualité de l'air. Notre première application démontre un transfert de connaissances inter-polluants réussi utilisant des réseaux de neurones artificiels (ANN), où des modèles pré-entraînés sur des données de $PM_{2.5}$ sont efficacement affinés pour prédire

d'autres polluants (PM_{10}, SO_2, NO_2, CO, O_3), permettant une modélisation efficace dans des scénarios de données rares. La deuxième application étend l'apprentissage par transfert temporel des prévisions horaires aux prévisions multi-horizons (6 heures, 12 heures et quotidiennes), permettant aux modèles de s'adapter à différentes échelles temporelles de prédiction avec une surcharge computationnelle minimale. La troisième application introduit notre nouveau cadre `LagEnsembleForecasting`, qui combine quatre modèles d'apprentissage profond spécialisés entraînés sur différentes séquences de décalage temporel (1h, 6h, 12h, 24h) pour fournir une précision de prédiction améliorée et une quantification naturelle de l'incertitude à travers la variance d'ensemble, avec une transférabilité réussie vers d'autres polluants démontrant la polyvalence du cadre. Troisièmement, nous faisons progresser l'évaluation des risques liés aux particules fines (PM) grâce à la prédiction des seuils de toxicité pilotée par l'apprentissage automatique, abordant les limitations des approches de laboratoire coûteuses et chronophages. En exploitant un ensemble de données complet compilé à partir de plus de 40 études évaluées par les pairs, notre cadre emploie cinq algorithmes complémentaires avec explicabilité basée sur SHAP (*SHapley Additive exPlanations*, une méthode d'interprétation des prédictions d'apprentissage automatique) pour distinguer les concentrations de PM toxiques des non-toxiques. Cette approche fournit des évaluations de toxicité plus nuancées que les méthodes conventionnelles tout en identifiant les facteurs compositionnels clés qui déterminent les résultats de toxicité. L'intégration de ces contributions établit un cadre complet pour la surveillance environnementale intelligente qui améliore les capacités de prédiction, fournit un support décisionnel interprétable et permet une évaluation plus efficace des risques sanitaires. La validation sur plusieurs ensembles de données réelles confirme l'applicabilité pratique et l'évolutivité de nos approches. Il est important de noter que, bien que cette recherche se concentre principalement sur la qualité de l'air, les fondements méthodologiques et les principes d'apprentissage par transfert établis ici créent une voie vers des stratégies intégrées de surveillance eau-air-sol, où les mêmes cadres intelligents pourraient être adaptés et étendus pour créer des systèmes unifiés de gestion environnementale qui abordent de manière holistique la nature interconnectée des défis environnementaux à travers de multiples domaines. Cette recherche démontre comment les techniques avancées d'IA peuvent transformer la surveillance environnementale de systèmes réactifs en systèmes proactifs, fournissant aux décideurs l'intelligence transparente, précise et exploitable nécessaire pour relever les défis complexes de la pollution et protéger la santé publique à une époque de pressions environnementales croissantes.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof. Hayfa Zgaya, for her invaluable guidance, support, and encouragement throughout this research journey. I am equally grateful to my co-supervisors, Dr. Ludivine Canivet and Dr. Sarah Ben-Othman, for their constructive feedback, insightful discussions, and continuous support.

I would also like to thank the OSL (Optimisation des Systèmes Logistiques) research team and its head, Prof. Slim Hammadi, for providing a stimulating research environment and valuable resources. My appreciation extends to Dr. Yu Fang, my former supervisor at Southwest Petroleum University (SWPU), China, whose mentorship during my Master's studies laid the foundation for my current research pursuits.

I am deeply grateful to my beloved parents, whose love, sacrifices, and constant support have been the foundation of all my achievements. I also thank my brother and my sisters for their encouragement, understanding, and unwavering belief in me throughout this journey.

Finally, I acknowledge the CRIStAL laboratory for hosting this work and offering the infrastructure that made this research possible. I am deeply thankful to all those who contributed, directly or indirectly, to the success of this thesis.

Contents

List of Tables

List of Figures

1. Introduction

1.1. Background

Air quality degradation represents one of the most significant environmental challenges of our time, with profound implications for both ecosystem integrity and human well-being. According to the World Health Organization (WHO), air pollution is responsible for millions of premature deaths annually, positioning it among the most critical environmental health risks globally [WHO 2022]. The economic burden is equally substantial, with global costs estimated at over 5% of GDP (Gross Domestic Product) in many developing countries due to healthcare expenses and lost productivity [World Bank 2023]. Despite increasing recognition of its severity, traditional approaches to air quality monitoring and management have struggled to provide decision-makers and the public with sufficiently accurate, timely, and interpretable information. While many regulatory agencies across Europe have implemented sophisticated air quality monitoring infrastructures, these systems often rely on fixed-station networks and traditional modeling techniques that may not fully capture local heterogeneity, provide predictive insights, or offer interpretability at scale. These limitations hinder the timely identification of pollution hotspots, complicate real-time response strategies, and reduce the effectiveness of public health interventions.

The challenge of air quality preservation exists within a broader environmental context characterized by declining biodiversity, degraded ecosystems, and pressured natural resources. These environmental challenges are interconnected through what environmental scientists term the "water-air-soil ecosystem" (Figure 1.1), forming a complex system with non-linear effects and feedback loops [J. Liu et al. 2007]. Within this framework, pollutants frequently transfer between atmospheric, aquatic, and terrestrial compartments, creating intricate patterns of environmental degradation that extend beyond any single medium. While our research focuses primarily on air quality indicators, we acknowledge this holistic perspective as essential context for comprehensive environmental management.

As illustrated in Figure 1.1, pollutants originating from anthropogenic sources such as incineration, industrial activities, and waste spreading disperse through the atmosphere, soil, and water. Key transfer mechanisms—including evaporation, deposition, leaching, percolation, and biological uptake—enable cross-media movement, leading to various exposure pathways that impact human health, biodiversity, and ecosystems.

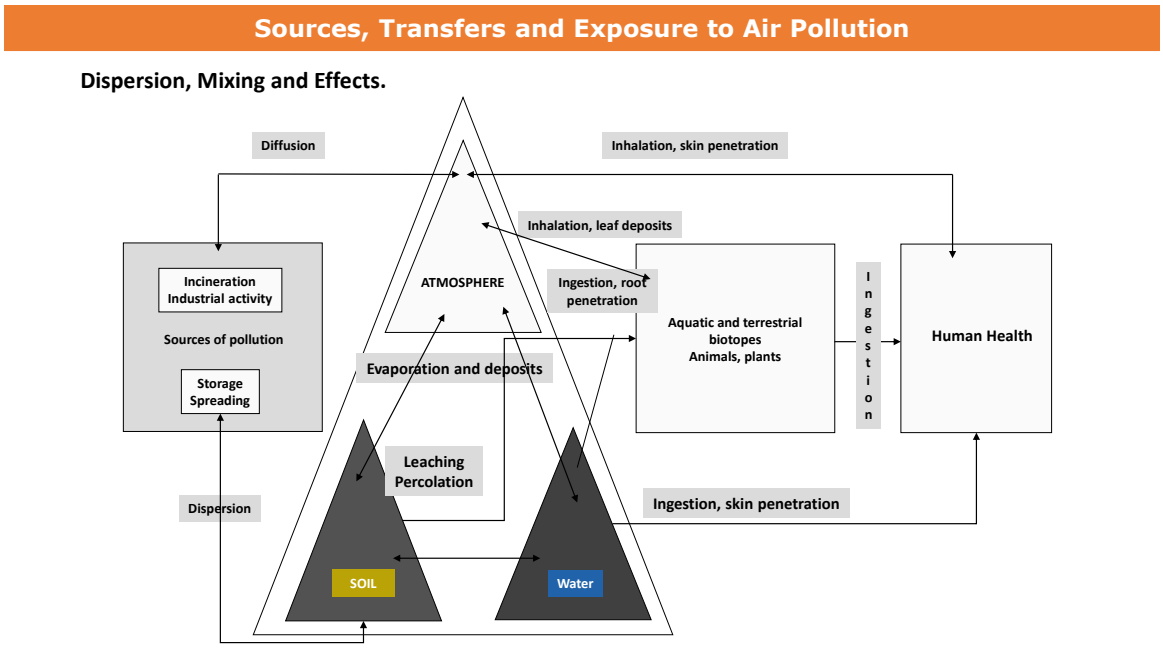


Figure 1.1.: Conceptual diagram of the water-air-soil pollutant transfer cycle, showing how pollutants from anthropogenic sources disperse across environmental media and impact human health and ecosystems through various transfer and exposure pathways.

Public authorities and decision-makers face mounting challenges in effectively monitoring and protecting air quality, due primarily to:

- The complex spatial and temporal dynamics of pollutant dispersion
- The intricate interactions between various pollutants and environmental factors
- The difficulty in translating technical measurements into meaningful health risk assessments

- The limitations of traditional monitoring infrastructure in providing comprehensive coverage (and real-time assessments)

Traditional approaches to air quality monitoring often lack the sophistication to capture these complexities or to predict future air quality scenarios with sufficient accuracy for proactive intervention. The limitations of these conventional methods have become increasingly apparent as our understanding of air pollution health impacts has grown more nuanced.

Among airborne pollutants, particulate matter (PM) is of particular concern due to its strong association with adverse health effects, including respiratory and cardiovascular diseases. However, the toxicity of PM is not solely determined by its concentration but also by its complex physico-chemical properties, which influence its biological impact. Despite extensive research, traditional toxicity assessment methods rely heavily on costly and time-consuming laboratory experiments, limiting their scalability and real-time applicability. Therefore, there is a pressing need for data-driven approaches that can effectively predict PM toxicity thresholds and improve risk assessment.

Recent advances in artificial intelligence (AI) offer promising new avenues for addressing these challenges. Machine learning (ML) techniques have demonstrated significant potential for improving pollution prediction accuracy across varied environmental conditions, while explainable AI approaches can make these predictions more transparent and actionable for decision-makers. Notable examples include the use of deep learning (DL) for accurate $PM_{2.5}$ forecasting in urban environments [Pak et al. 2020] and the application of ensemble methods for more reliable ozone prediction [J. Du et al. 2022; Zang et al. 2021].

In particular, machine learning-based PM toxicity prediction enables automated classification of toxic versus non-toxic PM concentrations by leveraging large datasets containing physico-chemical properties and exposure information. By integrating domain knowledge with data-driven approaches, these models can complement traditional laboratory-based assessments, enabling more rapid and scalable toxicity classification. This approach enhances air quality monitoring by supporting targeted interventions to mitigate health risks. The application of these technologies to environmental monitoring represents a paradigm shift in our approach to air quality management.

However, several important challenges remain in the application of AI to air quality monitoring and assessment, including:

- **Technical and Methodological Challenges:**

  - The need for more interpretable models that can explain their reasoning to non-technical stakeholders
  - The challenge of effectively transferring knowledge between models for different pollutants, different stations, and different temporal resolutions
  - The incorporation and propagation of uncertainty throughout modeling processes, including input data uncertainty, model structural uncertainty, and prediction uncertainty

- **Domain-Specific Challenges:**
  - The difficulty in developing accurate health risk assessments that account for varying population vulnerabilities
  - The challenge of predicting PM toxicity thresholds with high reliability while ensuring model interpretability (to foster regulatory acceptance and real-world deployment)

- **Implementation and Infrastructure Challenges:**
  - The integration of diverse data sources with different temporal and spatial resolutions
  - The computational resources required for real-time analysis of high-dimensional environmental data
  - The need for cost-effective deployment strategies in resource-constrained regions

1.1.1. Interdisciplinary Positioning of This Research

This thesis is positioned at the intersection of environmental science, computer science, and public health, drawing from and contributing to all three domains. From environmental science, we adopt foundational concepts of pollutant behavior, atmospheric dynamics, and ecosystem interactions. From computer science, particularly artificial intelligence, we leverage advances in machine learning, knowledge representation, and computational modeling. From public health, we incorporate frameworks for risk assessment, exposure analysis, and health impact evaluation. This interdisciplinary approach is essential for addressing the complex, multifaceted challenges of air quality management that cannot be adequately addressed within the confines of any single discipline.

The integration of these distinct yet complementary perspectives enables novel approaches to environmental intelligence that transcend traditional disciplinary boundaries. Rather than simply applying computational methods to environmental data, this research seeks to develop a new methodological framework that fundamentally reimagines how we conceptualize, monitor, and respond to air quality challenges—drawing from multiple disciplines while contributing to the emerging transdisciplinary domain of environmental intelligence [Hitziger et al. 2018].

1.1.2. Policy Context and Relevance

This research is situated within a broader policy context characterized by increasing recognition of air quality as a critical public health and environmental governance challenge. International frameworks such as the World Health Organization's Air Quality Guidelines [WHO 2022], the United Nations Sustainable Development Goals (particularly Goal 3: Good Health and Well-being, Goal 11: Sustainable Cities and Communities, and Goal 13: Climate Action), and regional initiatives such as the European Union's Clean Air Policy Package establish standards and targets that demand more sophisticated monitoring and assessment approaches.

These policy frameworks increasingly recognize the limitations of traditional threshold-based approaches to air quality management and call for more nuanced, context-sensitive methodologies that can account for complex exposure patterns, varying population vulnerabilities, and interactions between multiple pollutants. By developing intelligent air quality indicators that address these complexities, this research directly responds to these policy imperatives, potentially contributing to more effective implementation of air quality management strategies at local, national, and international levels.

The explainability focus of our research is particularly relevant in regulatory contexts where transparency in decision-making is not merely a technical consideration but a fundamental governance requirement. By developing classification approaches that can explain their reasoning in terms accessible to diverse stakeholders, this research addresses a critical gap between technical capabilities and governance needs in environmental management.

Addressing these challenges requires innovative approaches that combine advanced artificial intelligence techniques with domain expertise in atmospheric science, public health, and environmental management. This thesis presents such approaches, focusing on the

development of intelligent air quality indicators that enhance our ability to monitor, predict, and respond to air pollution challenges while acknowledging the broader water-air-soil context in which these challenges exist.

1.2. Research Problems

This research addresses fundamental challenges in contemporary air quality monitoring and health impact assessment systems, challenges that persist despite significant technological advancements in sensor technologies and computational capabilities. The core research problem encompasses multiple interrelated dimensions that collectively impede the development of truly intelligent environmental decision support systems.

Central to this problem is the epistemological disconnect—the gap between raw data collection and meaningful knowledge extraction—in air quality management. This disconnect refers to the difficulty in transforming environmental measurements into actionable insights that can inform policy and intervention. Despite the proliferation of environmental monitoring data, current frameworks exhibit critical limitations in their capacity to transform these data into actionable intelligence for environmental and public health preservation. These limitations manifest in several distinct yet interconnected domains:

1. **Limited explainability in air quality classification systems:** Contemporary machine learning approaches to air quality classification, while often achieving statistical accuracy, operate as epistemologically opaque "black boxes" that fail to provide interpretable rationales and rules for their classifications. This opacity creates significant barriers to trust and adoption among decision-makers and limits the practical utility of these systems in policy formulation and public communication. The challenge lies in developing classification frameworks that maintain high accuracy while providing transparent reasoning accessible to diverse stakeholders.

2. **Fragmented pollutant-specific and temporal-specific modeling approaches:** The current paradigm treats different atmospheric pollutants (e.g., $PM_{2.5}$, PM_{10}, and O_3) and temporal resolutions (e.g., hourly, 6-hourly, and daily) as isolated modeling tasks, developing separate models for each. This dual fragmentation overlooks the underlying interdependencies among pollutants as well as temporal patterns in pollutant behavior. As a result, it prevents the effective use of transfer learning techniques

that could exploit shared atmospheric processes across pollutants and time scales. Leveraging cross-pollutant and cross-temporal knowledge transfer could significantly enhance predictive accuracy, model robustness, and generalizability.

3. **Inadequate toxicity assessment frameworks:** Existing approaches to particulate matter toxicity assessment rely predominantly on fixed thresholds derived from resource-intensive laboratory experiments with limited ecological validity. These frameworks inadequately capture the complex, non-linear relationships between particulate matter characteristics and health outcomes across diverse population segments and environmental contexts. The critical challenge lies in developing more sophisticated, data-driven approaches to toxicity assessment that can adapt to varying pollutant compositions and population vulnerabilities.

4. **Infrastructure limitations in monitoring networks:** Current monitoring infrastructures typically rely on spatially sparse, fixed-location stations that cannot capture the fine-grained spatial heterogeneity of air pollution, particularly in complex urban environments. This infrastructure gap creates significant blind spots in our understanding of pollution exposure patterns and limits the effectiveness of targeted intervention strategies. The prohibitive cost of high-precision monitoring equipment further constrains network density, especially in regions with limited resources where air quality concerns are often most acute. These limitations are exacerbated by maintenance challenges and calibration requirements that can interrupt data continuity.

5. **Computational constraints in real-time analysis:** Despite advances in computational capabilities, existing systems struggle to process and analyze high-dimensional air quality data in real-time, creating critical latency between data collection and decision-making that undermines the effectiveness of rapid response to deteriorating air quality conditions. The computational intensity of sophisticated predictive models often forces a trade-off between model complexity and operational responsiveness. Additionally, the integration of heterogeneous data streams—from fixed stations, mobile sensors, satellite observations, and meteorological inputs—creates substantial computational overhead that further challenges real-time processing capabilities.

These challenges are exacerbated by the inherently distributed nature of air quality phenomena, which span multiple spatial and temporal scales and involve complex interactions

between natural and anthropogenic systems. The effective management of such distributed phenomena requires decentralized decision-making frameworks that can operate with local autonomy while maintaining system-level coherence—a capability largely absent from current centralized management approaches.

The research problem thus centers on the need for a paradigmatic shift from conventional air quality monitoring toward truly intelligent environmental indicator systems. These systems must go beyond data collection to extract meaningful patterns, provide explainable decisions, transfer knowledge across pollutant domains and temporal resolutions, accurately assess health risks, and support distributed decision-making across multiple scales of environmental governance.

1.3. Research Objectives and Questions

This research pursues a systematic inquiry into the development of intelligent air quality indicators for environmental and health preservation, guided by a structured methodological framework that addresses the limitations identified in current approaches. The investigation is organized around three interconnected research streams, each corresponding to a critical dimension of air quality intelligence.

1.3.1. Primary Research Objectives

1. **Develop explainable classification frameworks for air quality assessment:** This objective directly addresses the challenge of limited explainability in current classification systems by creating transparent, interpretable models for air quality classification through the integration of granular computing (GrC) and rule extraction techniques. The aim is to transform opaque/black-box machine learning models into explainable decision support tools that provide transparent reasoning accessible to diverse stakeholders, thereby enhancing trust and facilitating more effective implementation in policy contexts.

2. **Enhance cross-pollutant and cross-temporal prediction accuracy through transfer learning:** This objective confronts the fragmentation of pollutant-specific and temporal-specific modeling by developing neural transfer learning approaches that leverage knowledge gained from one pollutant/temporal resolution domain/task to

enhance prediction accuracy in others. The research aims to identify and exploit the shared underlying patterns and relationships between different air pollutants and temporal resolutions to create more robust, efficient, and accurate prediction models, particularly valuable in scenarios with limited monitoring data.

3. **Advance particulate matter toxicity assessment through machine learning:** This objective tackles the inadequacies of current toxicity assessment frameworks by developing novel machine learning approaches for predicting toxicity thresholds for particulate matter that better capture the complex relationships between physicochemical attributes, exposure characteristics and health outcomes. The aim is to transcend the limitations of fixed thresholds derived from laboratory experiments by creating adaptive, data-driven models that can accommodate varying pollutant compositions and population vulnerabilities.

4. **Establish methodological foundations for integrated environmental intelligence:** This cross-cutting objective aims to develop coherent methodological frameworks that integrate the above approaches within a unified paradigm of environmental intelligence. The research seeks to identify synergies between explainable AI, transfer learning, and toxicity prediction methods that can enhance overall system performance and applicability to real-world environmental challenges.

1.3.2. Research Questions

The investigation is guided by the following specific research questions that operationalize the broader research objectives:

1. **On explainable air quality classification:**
    - How can granular computing and rule extraction techniques be effectively integrated to develop explainable models for air quality classification?
    - To what extent do explainable models based on granular computing maintain classification accuracy compared to fundamental "black box" machine learning approaches?
    - What granularity levels and rule structures provide optimal balance between explainability and model performance in the air quality domain?

2. **On neural transfer learning for cross-pollutant and cross-temporal resolutions prediction:**
    - What architectural and training approaches enable most effective knowledge transfer between neural models for different air pollutants/temporal resolutions?
    - To what degree does transfer learning across pollutant and temporal resolutions domains improve prediction accuracy compared to pollutant-specific and temporal-specific models, particularly in data-scarce scenarios?
    - Which pollutant relationships exhibit strongest potential for beneficial knowledge transfer, and what underlying factors drive these relationships?
3. **On machine learning for toxicity threshold prediction:**
    - Which machine learning algorithms and feature representations most effectively capture the complex relationships between particulate matter characteristics and toxicity outcomes?
    - How do machine learning-based toxicity predictions compare with traditional laboratory-derived thresholds in terms of accuracy, generalizability, and applicability to diverse exposure scenarios?
    - What novel insights into particulate matter toxicity mechanisms can be derived from interpretable machine learning models trained on comprehensive toxicity datasets?
4. **On future integration with multi-agent systems:** *(Note: This section represents potential future extensions beyond the core thesis work)*
    - How might the developed intelligent air quality indicators be effectively integrated within distributed multi-agent systems for comprehensive environmental monitoring?
    - What coordination mechanisms would enable effective integration of air quality intelligence with water and soil monitoring within a broader water-air-soil strategic framework?
    - What architectural approaches could support adaptive learning and knowledge sharing across environmental domains in a multi-agent implementation?

These research objectives and questions establish a comprehensive framework for investigating the development of intelligent air quality indicators while providing specific pathways for advancing the state of the art in environmental monitoring and assessment. The research explicitly addresses critical limitations in current approaches while establishing methodological foundations for future extensions to broader environmental intelligence systems.

1.4. Significance and Contributions

This research makes substantial contributions to both theoretical discourse and practical applications in the domains of environmental intelligence, air quality management, and public health protection. The significance of this work extends across multiple dimensions of scientific, methodological, and societal impact.

1.4.1. Theoretical and Practical Significance

- **Advancement of explainable artificial intelligence in environmental contexts:** This research extends the theoretical foundations of explainable AI beyond conventional application domains into environmental science, developing novel frameworks for generating interpretable models in the context of complex environmental data. By integrating granular computing with rule extraction techniques, the work contributes to fundamental understanding of how domain-specific knowledge can be effectively incorporated into explainable AI architectures.

- **Extension of transfer learning application to environmental monitoring:** The research advances practical understanding of knowledge transfer mechanisms across different pollutant and temporal resolutions domains, identifying underlying commonalities that facilitate effective transfer while characterizing domain-specific factors that constrain transferability. These insights contribute to broader practical and theoretical discourse on cross-domain knowledge transfer in environmental modeling.

- **Reconceptualization of PM toxicity assessment paradigms:** By developing machine learning approaches for toxicity threshold prediction, this research challenges conventional paradigms in toxicological assessment that rely primarily on direct experimental observation. It establishes theoretical and practical frameworks for integrating

computational and experimental approaches to toxicity assessment that transcend the limitations of either approach in isolation.

- **Conceptual foundations for integrated environmental intelligence:** The work contributes to emerging theoretical and practical discourse on environmental intelligence by establishing conceptual frameworks for integrating diverse AI techniques within coherent systems that address multiple dimensions of environmental assessment and prediction simultaneously.

1.4.2. Integration of Theoretical and Practical Framework Components

Figure 1.2 presents the comprehensive theoretical framework that supports this research. It organizes and integrates key components across three pillars: **Core AI Approaches**, **Enhancement Approaches**, and **Computational Intelligence**. Together, these pillars form a conceptual architecture that guides the development of intelligent air quality indicators tailored to the challenges of environmental monitoring and public health protection.

This framework enables the systematic integration of cutting-edge artificial intelligence techniques with domain-specific knowledge, ensuring that developed models are both technically sound and environmentally meaningful.

- **Core AI Approaches:** This pillar includes mainstream artificial intelligence methodologies such as machine learning, deep learning, and transfer learning. These approaches form the core predictive engine of the research. Machine learning algorithms are used for classification and regression tasks, deep learning architectures capture complex nonlinear temporal dynamics in air pollution data, and transfer learning enables cross-domain and cross-resolution knowledge adaptation, particularly valuable in data-scarce scenarios.

- **Enhancement Approaches:** This central pillar addresses critical limitations of conventional AI methods by introducing advanced techniques to enhance model reliability and interpretability. *Uncertainty management* methods (e.g., probabilistic modeling, fuzzy logic, and evidence theory) allow the models to represent, reason with, and propagate uncertainty in environmental data. *Explainable AI* approaches—especially those based on granular computing and rule extraction—support transparent decision-making by providing intelligible justifications for predictions, a crucial requirement in

regulatory and policy contexts.

- **Computational Intelligence:** This pillar incorporates soft computing paradigms that support the representation of vague, imprecise, and incomplete information—common characteristics of environmental and health-related data. Techniques such as granular computing, fuzzy systems, and rough sets theory are employed to structure knowledge, handle imprecision, and facilitate interpretable classifications. These methods serve as the foundation for explainable models that move beyond black-box predictions.

Finally, throughout this layered architecture, domain knowledge from environmental science, toxicology, and public health is systematically embedded. This ensures that the models are not only computationally efficient and interpretable but also scientifically grounded and policy-relevant.

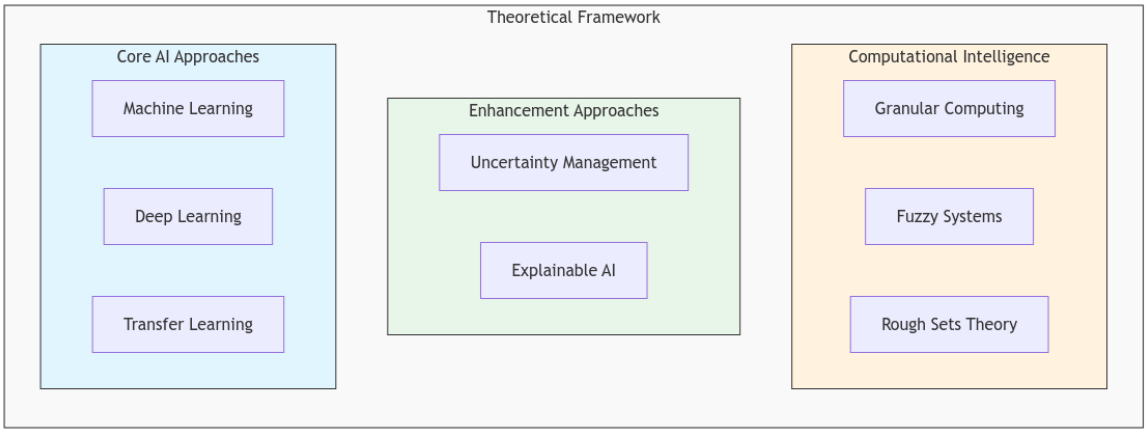


Figure 1.2.: A comprehensive theoretical framework integrating core AI methods, enhancement approaches (uncertainty and explainability), and soft computing paradigms to develop intelligent air quality indicators.

1.4.3. Methodological Significance

- **Novel integration of granular computing with environmental classification:** This research introduces methodological innovations in the application of granular computing to air quality classification, developing systematic approaches for determining optimal granularity levels and rule extraction mechanisms in environmental data contexts that balance interpretability with classification performance.

- **Innovative applications of neural transfer learning for air quality forecasting:** The work establishes methodological frameworks for designing and training neural networks that effectively transfer knowledge across pollutant domains and temporal scales, providing structured approaches to architecture selection, feature representation, and training protocols that maximize transfer effectiveness.
- **Advancement of computational toxicology methodologies:** By developing machine learning approaches for toxicity threshold prediction, this research establishes novel methodological frameworks for integrating diverse toxicological data within predictive models that capture complex non-linear relationships between particulate characteristics and health impacts.
- **Methodological foundations for multi-domain environmental intelligence:** The research develops methodological approaches that establish foundations for future integration of air quality intelligence within broader environmental monitoring systems, particularly through multi-agent architectures that can coordinate across environmental domains.

1.4.4. Practical and Societal Significance

- **Enhanced decision support for environmental governance:** By developing explainable air quality classification models, this research provides public authorities and environmental managers with transparent, interpretable decision support tools that can facilitate more effective communication, build stakeholder trust, and support evidence-based policy development.
- **Improved air pollution forecasting capabilities:** The neural transfer learning approaches developed in this research enhance the accuracy of air pollution predictions, particularly in data-sparse contexts, enabling more effective early warning systems for adverse air quality events that can mitigate public health impacts through timely interventions.
- **Advanced health risk assessment:** The machine learning approaches for toxicity threshold prediction provide more nuanced, context-sensitive assessments of health risks associated with particulate matter exposure, supporting more targeted public health

interventions and environmental management strategies that prioritize highest-risk scenarios.

- **Foundations for integrated environmental protection:** While focusing primarily on air quality, the methodological approaches developed in this research establish foundations for more integrated environmental protection strategies that can eventually incorporate water and soil monitoring within unified management frameworks.
- **Bridging technical capabilities with governance needs:** By emphasizing explainability and interpretability alongside predictive performance, this research addresses critical gaps between technical capabilities and governance requirements in environmental management, facilitating more effective translation of environmental intelligence into policy action.

1.4.5. Limitations and Boundary Conditions

While this research makes significant contributions across multiple dimensions, it is important to acknowledge several limitations and boundary conditions that contextualize these contributions:

- **Data availability constraints:** The models and methodologies developed are necessarily limited by the availability, quality, and representativeness of existing environmental monitoring and toxicological assessment data. Particularly for toxicity prediction, where comprehensive datasets linking particulate characteristics to health outcomes remain relatively scarce, the generalizability of our findings may be constrained by these data limitations.
- **Computational complexity trade-offs:** The development of sophisticated AI approaches for environmental monitoring inevitably encounters trade-offs between model complexity, computational efficiency, and practical deployability. While we strive to balance these considerations, certain approaches developed in this research may require further optimization before widespread practical implementation.
- **Contextual specificity:** The effectiveness of the developed methodologies may vary across different environmental, social, and governance contexts. Models trained

and validated in specific geographical or institutional settings may require significant adaptation before application in substantially different contexts.

- **Scope limitations:** While this research acknowledges the broader water-air-soil ecosystem context, its primary focus remains on air quality indicators. The extension of these approaches to integrate water and soil monitoring represents an important direction for future research but falls beyond the scope of this thesis.

These limitations do not diminish the significance of the research contributions but rather define the boundary conditions within which these contributions should be interpreted and applied. They also highlight important directions for future research that can address these limitations and extend the approaches developed in this thesis.

1.4.6. Future Impact Trajectories

The significance of this research extends beyond its immediate contributions to its potential for catalyzing future developments in environmental intelligence. Particularly noteworthy is its potential for integration with multi-agent system architectures that could extend the developed approaches to broader water-air-soil monitoring frameworks. Such extensions would represent a paradigmatic shift from compartmentalized environmental monitoring toward truly integrated environmental intelligence systems capable of tracking pollutant transfers between environmental domains and identifying complex cross-domain interactions that influence environmental and health outcomes.

In summary, this research makes substantial contributions across theoretical, methodological, and practical dimensions of environmental intelligence, establishing new approaches to air quality monitoring and assessment while laying foundations for future extensions to more comprehensive environmental protection frameworks.

1.5. Thesis Structure

This thesis is organized into six interconnected chapters that systematically address the development of intelligent air quality indicators through complementary methodological approaches:

Chapter 1: Introduction establishes the foundational context by articulating the critical challenges in contemporary air quality monitoring, positioning the research within the broader

water-air-soil ecosystem framework, and defining specific research objectives. This chapter demonstrates how traditional monitoring approaches fall short of providing the timely, accurate, and interpretable information needed for effective environmental decision-making.

Chapter 2: Methodological Background and Materials provides a comprehensive theoretical foundation covering granular computing, time series analysis, machine learning algorithms, deep learning architectures, and transfer learning principles. This chapter also introduces the Beijing air quality datasets that serve as the empirical foundation for model development and validation across all subsequent investigations.

Chapter 3: Advancing Particulate Matter Risk Assessment addresses the critical gap in scalable toxicity assessment by developing machine learning models capable of predicting PM toxicity thresholds based on physico-chemical and exposure characteristics. This chapter demonstrates how data-driven approaches can complement traditional laboratory-based methods while providing interpretable insights through SHAP analysis.

Chapter 4: An Explainable Granular Computing Approach for Air Quality Classification tackles the interpretability challenge in environmental AI by developing the AirQ-RuleGrCEx algorithm based on granular computing, which extracts human-readable classification rules linking meteorological conditions to $PM_{2.5}$ concentration levels. This approach bridges the gap between predictive accuracy and explanatory transparency essential for regulatory applications.

Chapter 5: Advancing Air Quality Forecasting Through Transfer Learning introduces transfer learning as a transformative paradigm for environmental forecasting, systematically exploring knowledge transfer across different pollutants and temporal resolutions. This chapter presents three complementary studies: cross-pollutant prediction, the LagEnsembleForecasting framework, and multi-temporal resolution adaptation.

Chapter 6: Conclusions and Perspectives synthesizes the key contributions across all research streams, discusses their theoretical and practical implications, acknowledges current limitations, and outlines promising directions for future work, particularly the extension to integrated water-air-soil monitoring systems.

The three core research chapters (3–5) each correspond to published or submitted peer-reviewed papers, with significantly expanded methodological details, comprehensive experimental results, and in-depth discussions that extend well beyond the constraints of journal article formats. Together, these chapters demonstrate a progression from explainable toxicity assessment through interpretable classification to efficient forecasting—collectively

establishing a comprehensive framework for intelligent environmental monitoring.

1.6. Publications

The work in this dissertation primarily relates to the following peer-reviewed scientific papers (in order of publication):

1. **Idriss Jairi**, Sarah Ben-Othman, Ludivine Canivet, Hayfa Zgaya-Biau, "Explainable based approach for the air quality classification on the granular computing rule extraction technique" *Engineering Applications of Artificial Intelligence*, 2024. (Q1, IF: 7.5, AI/Software)
   https://doi.org/10.1016/j.engappai.2024.108096

2. **Idriss Jairi**, Sarah Ben-Othman, Ludivine Canivet, Hayfa Zgaya-Biau, "Enhancing Air Pollution Prediction: A Neural Transfer Learning Approach across Different Air Pollutants" *Environmental Technology & Innovation*, 2024. (Q1, IF: 6.7, Environmental Engineering)
   https://doi.org/10.1016/j.eti.2024.103793

3. **Idriss Jairi**, Amelle Rekbi, Sarah Ben-Othman, Slim Hammadi, Ludivine Canivet, Hayfa Zgaya-Biau, "Enhancing particulate matter risk assessment with novel machine learning-driven toxicity threshold prediction" *Engineering Applications of Artificial Intelligence*, 2025. (Q1, IF: 7.5, AI/Software)
   https://doi.org/10.1016/j.engappai.2024.109531

Conferences & Workshops

- Doctoral Consortium Participant, ECAI "European Conference on Artificial Intelligence", Santiago de Compostela, Spain – October 2024. (Top-tier European AI conference)
  https://anaellewilczynski.pages.centralesupelec.fr/ecai-2024-dc/accepted.html

- Natalia Grabar, **Idriss Jairi**, Hayfa Zgaya-Biau, "Enhancing Efficient Semantic Similarity Computing with Optimized BERT Models" *Medical Informatics Europe (MIE)*,

Glasgow, Scotland – May 2025. (Leading European conference in health informatics) https://pubmed.ncbi.nlm.nih.gov/40380588/
Note: This contribution is not directly related to this thesis.

Submitted Papers

- **Idriss Jairi**, Sarah Ben-Othman, Ludivine Canivet, Hayfa Zgaya-Biau, "Enhancing Air Quality Forecasting Accuracy at Multiple Temporal Resolutions Through Deep Transfer Learning" *Engineering Applications of Artificial Intelligence*, 2025. (Under review – Q1, IF: 7.5)

- **Idriss Jairi**, Sarah Ben-Othman, Ludivine Canivet, Hayfa Zgaya-Biau, "LagEnsemble-Forecasting: Multi-Lag Ensemble Learning and Cross-Pollutant Transfer Learning for Comprehensive Air Quality Forecasting" *Environmental Modelling & Software*, 2025. (Under review – Q1, IF: 4.8, Environmental Science/Modelling)

2. Methodological Background and Materials

Addressing the complex challenges of air quality monitoring, classification, forecasting, and health risk assessment requires a robust methodological foundation that integrates diverse computational approaches tailored to specific aspects of environmental data analysis. This chapter establishes the theoretical and practical foundations underlying the research presented in subsequent chapters by systematically examining the key methodologies employed throughout this thesis. We begin with granular computing theory, which provides the framework for extracting interpretable classification rules from air quality data, enabling the development of explainable decision support systems essential for environmental policy-making. Time series analysis techniques are explored because air quality data inherently exhibits temporal dependencies and seasonal patterns that must be properly understood and modeled for effective forecasting. Classical machine learning algorithms are examined as they form the core of the PM toxicity prediction framework, providing reliable and interpretable approaches for distinguishing between toxic and non-toxic particulate matter concentrations. Deep learning architectures are investigated for their application in transfer learning scenarios and for building advanced models capable of capturing complex nonlinear relationships in air quality time series data. The chapter also covers essential data preparation techniques, evaluation metrics, and cross-validation strategies that ensure rigorous experimental design and reliable model assessment. Finally, we present the Beijing air quality datasets that serve as the empirical foundation for validating our proposed approaches. This comprehensive methodological framework provides the necessary tools for developing intelligent air quality analysis systems that can effectively support public health protection and environmental decision-making across the diverse applications explored in this thesis.

2.1. Granular Computing

2.1.1. Introduction

Granular Computing (GrC) has emerged as a paradigm of fundamental importance in the landscape of information processing and computational intelligence. It represents a systematic approach to complex problem-solving through the use of *granules*—information entities that encapsulate meaning at varying levels of abstraction and resolution [Pedrycz 2018]. The central principle of GrC is that complex systems and problems can be more effectively understood, represented, and solved when approached through multiple levels of granularity, allowing for shifts between detailed and abstract perspectives as necessitated by the problem context [Yiyu Yao 2008].

The significance of GrC lies in its capacity to formalize human cognitive processes that naturally decompose complex problems into simpler components, establish hierarchical structures, and focus attention at appropriate levels of detail (Figure 2.1). This mimicking of human information processing provides a powerful framework for addressing the challenges of uncertainty, imprecision, and complexity that characterize real-world problems [Bargiela and Pedrycz 2003]. As [Lotfi A Zadeh 1997] aptly noted, "the human ability to process precisely that which is imprecise" finds its computational parallel in granular computing.

In essence, GrC offers a unified theoretical framework that integrates multiple formalisms dealing with information granulation, including rough sets [Pawlak 1982], fuzzy sets [Lotfi Asker Zadeh 1965], interval analysis [Moore 1966], and quotient spaces [L. Zhang and B. Zhang 2004], among others. This integration fosters methodological synergy and knowledge transfer across these previously disparate domains, enhancing the collective capacity to address complex computational challenges [Pedrycz 2018].

2.1.2. Definition and Core Concepts

Granular Computing can be formally defined as "a general computation theory for effectively using granules such as classes, clusters, subsets, groups, and intervals to build an efficient computational model for complex real-world problems" [YY Yao 2000]. This definition encapsulates the essence of GrC as a systematic approach to structuring and processing complex information through the use of granules at various levels of granularity.

The foundational entity in GrC is the *information granule*, which represents an elementary

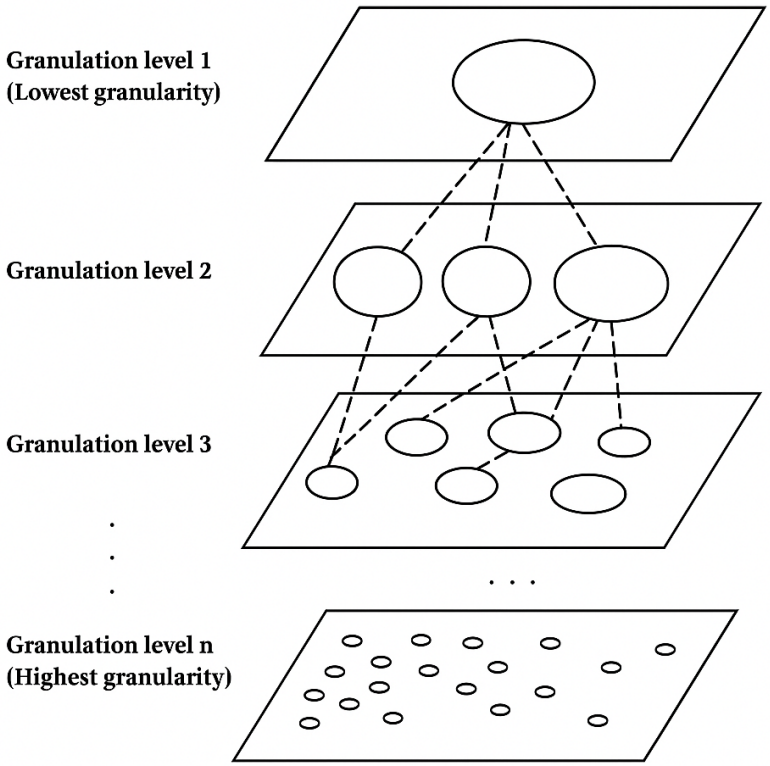


Figure 2.1.: Multi-level granulation hierarchy illustrating the progressive refinement of information granules from coarsest granularity (level 1) to finest granularity (level n) in granular computing paradigm.

unit of meaning in a particular problem domain. Formally, an information granule G can be defined as an aggregation of elements drawn together by indistinguishability, similarity, proximity, or functionality [Pedrycz 2011]. Mathematically, granules may be expressed through various formalisms:

- In crisp set theory: $G = \{x \in X | P(x)\}$ where P is a predicate
- In fuzzy set theory: $G = \{(x, \mu_G(x)) | x \in X\}$ where $\mu_G(x)$ is the membership function
- In rough set theory: $G = (G_{lower}, G_{upper})$ where G_{lower} and G_{upper} are the lower and upper approximations
- In interval mathematics: $G = [a, b]$ where a and b define the interval boundaries

The process of *granulation* involves the construction, interpretation, and utilization of information granules. [Bargiela and Pedrycz 2006] identify three fundamental aspects of granulation:

1. *Construction*: The process of forming granules from elemental data or from other granules 2. *Interpretation*: Assigning meaning to granules within the problem context 3. *Utilization*: Employing granules for computation, reasoning, and problem-solving

A *granular structure* $\mathcal{G} = \{G_1, G_2, \ldots, G_n\}$ represents a collection of granules organized according to certain relationships. These structures may take various forms, including hierarchies, trees, and networks, each imposing specific constraints on the relationships between granules [YY Yao 2006]. A particularly important concept is that of *multi-level granular representation*, which enables abstraction and zooming capabilities essential for managing complexity.

The principle of *justifiable granularity* [Pedrycz 2018] stipulates that the level of granularity should be determined by:

$$\text{Specificity}(G) \times \text{Coverage}(G) \to \max \tag{2.1}$$

where specificity relates to the precision of information carried by the granule, and coverage relates to its experimental evidence.

- Specificity(G): Measures how precise, informative, or narrow the granule is. A highly specific granule conveys detailed and fine-grained information (e.g., a small interval or tight cluster).

- Coverage(G): Measures how well the granule is supported by the data—that is, how many data points fall within the granule or how representative it is of the observed evidence.

GrC operates through three primary methodological perspectives as articulated by [Yiyu Yao 2008]:

- *Philosophical perspective*: GrC as structured thinking
- *Methodological perspective*: GrC as structured problem-solving
- *Computational perspective*: GrC as structured information processing

These perspectives collectively provide a comprehensive framework for understanding and applying granular computing principles across diverse domains.

A. Historical Development

The evolution of granular computing represents a fascinating convergence of multiple intellectual traditions in computer science, mathematics, and cognitive science. While the term itself was formally introduced only in the mid-1990s, its conceptual foundations have deeper historical roots that deserve study and examination.

The foundational roots of granular computing (GrC) can be attributed to Zadeh's seminal introduction of fuzzy set theory in 1965, which introduced the revolutionary concept of graded membership, challenging the binary logic of classical set theory [Lotfi Asker Zadeh 1965]. This pioneering work laid the groundwork for representing and reasoning with imprecise information—a core capability of granular computing. The 1970s witnessed further development of this foundation through Zadeh's introduction of linguistic variables and the theory of approximate reasoning [Lotfi Asker Zadeh 1975], which formalized the concept of computing with words—an early manifestation of information granulation.

A parallel and equally significant development occurred with Pawlak's introduction of rough set theory in 1982 [Pawlak 1982]. This theory provided a formal methodology for dealing with vagueness and uncertainty through the use of approximation spaces, offering another powerful approach to information granulation. Concurrently, developments in cluster analysis, interval mathematics, and quotient space theory contributed additional perspectives on information granulation [Moore 1966; B. Zhang and L. Zhang 1992]. The formal inception

of granular computing as a unified field occurred in the mid-1990s, with the term first appearing in a paper by Lin in 1996 [Lin 1997] and gaining prominence through Zadeh's 1997 paper on "Toward a Theory of Fuzzy Information Granulation and Its Centrality in Human Reasoning and Fuzzy Logic" [Lotfi A Zadeh 1997]. This period marked the recognition of information granulation as a common thread connecting various theories and methodologies.

Between 2000 and 2010, GrC entered a phase of theoretical consolidation. Yiyu Yao introduced the triarchic theory of GrC, framing it as structured thinking, structured problem solving, and structured information processing [Yiyu Yao 2004]. In parallel, Pedrycz and Bargiela formalized the principle of justifiable granularity in their foundational textbook [Bargiela and Pedrycz 2003], providing design principles for constructing meaningful granules. This era also saw the institutionalization of the field through the first IEEE International Conference on Granular Computing in 2005 [*IEEE International Conference on Granular Computing* 2005] and the publication of the *Handbook of Granular Computing* in 2008 [Pedrycz, Skowron, et al. 2008].

The post-2015 era marks a shift toward integration with modern AI and machine learning. The launch of the Granular Computing journal by Springer in 2016 ["Granular Computing: An International Journal" 2016] symbolized disciplinary maturity. Cutting-edge research now explores granular deep learning for enhanced interpretability and uncertainty handling, explainable AI through granular knowledge representation, federated learning frameworks incorporating granular computing principles, spatial granular systems for geographic and environmental applications, and data-driven granular cognitive computing that bridges human cognition patterns with machine learning architectures.

In summary, granular computing has evolved from loosely connected theoretical traditions into a unified paradigm for uncertainty modeling and cognitive reasoning. Its distinctive contribution lies in its emphasis on granulation, multi-level abstraction, and human-aligned computation—principles that remain vital as AI systems face increasingly complex, uncertain, and heterogeneous environments.

B. Motivations and Applications

The development and application of granular computing are driven by fundamental motivations that address inherent challenges in complex information processing. These motivations not only justify the theoretical significance of GrC but also explain its practical utility across

diverse application domains.

Core Motivations:

1. *Management of Complexity*: Perhaps the most compelling motivation for GrC is its capacity to manage complexity through multi-level representations [YY Yao 2006]. By enabling transitions between different levels of granularity, GrC provides mechanisms to balance precision and simplicity, facilitating comprehension and manipulation of complex systems. As [Bargiela and Pedrycz 2003] note, "complexity disappears when viewed from the appropriate level of abstraction."

2. *Handling Uncertainty and Imprecision*: Real-world data is inherently characterized by uncertainty, imprecision, and incompleteness [Lotfi A Zadeh 1997]. The various formalisms incorporated within GrC—fuzzy sets, rough sets, and interval analysis—provide complementary approaches to representing and reasoning with such imperfect information.

3. *Cognitive-Inspired Computing*: GrC draws inspiration from human cognitive processes that naturally employ multiple levels of granularity in perception and problem-solving [Hobbs 1990]. This cognitive grounding enhances the development of computational systems that better align with human reasoning patterns, facilitating human-computer interaction and interpretability.

4. *Computational Efficiency*: Appropriate granulation can significantly reduce computational complexity by focusing on relevant aspects of a problem at suitable levels of detail [Pedrycz 2018]. This efficiency gain is particularly crucial when dealing with big data and real-time applications.

5. *Knowledge Integration*: GrC provides a unifying framework for integrating knowledge represented at different levels of granularity and from different perspectives [Yiyu Yao 2008]. This integration capability is essential for knowledge synthesis in multidisciplinary contexts.

A particularly illustrative case study of successful GrC application is found in [Kok and Chan 2016], who developed a Granular Computing-Based Crowd Segmentation (GrCS) framework that addresses complex visual scene analysis by conceptualizing problems at different granularity levels. Their approach successfully aggregates structurally similar pixels

into meaningful atomic structure granules, effectively outlining natural boundaries between crowd and background regions across diverse scene layouts.

Similarly, [M. Song, Y. Li, et al. 2023] applied granular neural networks to time series prediction, demonstrating how connecting model uncertainty with information granularity concepts enhances both predictive performance and explainability. Their approach provided optimized interval outputs that proved more robust than single-value predictions while maintaining high accuracy across multiple datasets.

Furthermore, [Ayoub et al. 2025] developed a granular computing framework for credit card fraud detection that implements a three-pillar strategy combining case-based reasoning, fuzzy rough sets for feature selection, and optimized Support Vector Data Description. This multi-stage granular approach successfully addressed class imbalance and noise reduction while surpassing current algorithms in both accuracy and efficiency.

These applications collectively illustrate how the core motivations of granular computing—such as managing complexity, handling uncertainty, and aligning with human cognition—translate into tangible advantages across a wide array of real-world domains. From healthcare and finance to cybersecurity and environmental monitoring, GrC has been successfully applied to enhance interpretability, reduce computational overhead, and improve decision-making performance. As the field continues to mature, the number and diversity of applications are rapidly increasing, making it infeasible to review them all comprehensively within this section. Moreover, recent methodological innovations such as the Three-Way Decision (3WD) framework [Yiyu Yao 2009, 2010, 2011] and its extensions like Sequential Three-Way Decision (S3WD) [Yiyu Yao 2013, 2016] further expand the capabilities of GrC by incorporating decision deferment, active learning, and context-aware granule formation. These developments underscore the growing importance of GrC as a foundational paradigm in the design of intelligent, interpretable, and uncertainty-aware computational systems.

C. Relation to Other Computational Paradigms

Granular Computing does not exist in isolation but rather occupies a distinctive position within the broader ecosystem of computational paradigms. Understanding these relationships is essential for positioning GrC within the contemporary computational landscape and for identifying opportunities for cross-paradigm integration and synergy.

Soft Computing:

Perhaps the closest relationship exists between GrC and soft computing, which encompasses fuzzy logic, neural networks, and evolutionary computation. As [Lotfi A Zadeh 1996] noted, both paradigms share a philosophical orientation toward tolerating imprecision and uncertainty to achieve tractability and robustness. GrC can be viewed as complementary to soft computing, with the former providing structural frameworks for information representation and the latter offering mechanisms for learning and adaptation [Pedrycz 2018]. The integration of these paradigms has led to hybrid approaches such as granular fuzzy systems [Pedrycz 2018] and granular neural networks [Pedrycz and Vukovich 2001; M. Song and Xinyu Zhao 2024].

Hierarchical Information Processing:

Hierarchical approaches to information processing, including hierarchical clustering, decision trees, and pyramid algorithms, share GrC's emphasis on multilevel structures [Gordon 1987]. However, GrC offers a more comprehensive theoretical framework that formalizes the relationships between levels and provides mechanisms for navigating across granularity levels. [YY Yao 2006] established formal mappings between hierarchical information processing structures and granular structures, demonstrating that the former can be viewed as specific instances of the latter.

Deep Learning:

Contemporary developments in deep learning and GrC exhibit fascinating parallels and potential for integration. Both paradigms involve multiple layers of representation, with complex features emerging at higher levels [LeCun et al. 2015]. However, they differ fundamentally in their approach to knowledge representation: deep learning typically employs distributed, subsymbolic representations learned from data, while GrC emphasizes symbolic, interpretable representations constructed through explicit granulation processes. Recent work on granular neural networks [M. Song, Y. Li, et al. 2023] and deep granular computing [Behzadidoost et al. 2024] represents promising attempts to integrate these paradigms, combining the learning capabilities of deep architectures with the interpretability of granular structures.

Explainable AI:

In the contemporary context of increasing emphasis on AI explainability, GrC offers valuable contributions through its inherent focus on interpretable representations [Liang and D. Liu 2015]. The multi-level granular structures characteristic of GrC naturally support explanation generation at varying levels of detail, facilitating communication of AI reasoning

to human users. [Pedrycz and S.-M. Chen 2011] demonstrated how granular models could provide explanations for complex deep learning systems, establishing GrC as a bridge between subsymbolic learning and symbolic explanation.

This analysis reveals that GrC occupies a distinctive position within the computational landscape, offering unique perspectives on information structuring and processing while exhibiting meaningful relationships with other paradigms. The most promising direction for future development appears to be not the isolated advancement of GrC, but rather its integration with complementary paradigms to address the multifaceted challenges of complex information processing in contemporary applications.

2.1.3. Theoretical Foundations of Granular Computing

A. Information Granulation Principles

Information granulation represents the fundamental cognitive process of structuring knowledge by abstracting and grouping entities based on similarity, functionality, or coherence [Lotfi A Zadeh 1996]. This process aligns naturally with human cognition, as humans instinctively perceive and organize the world through varying levels of detail depending on context and purpose [Hobbs 1990].

The principle of *justifiable granularity* introduced by [Bargiela and Pedrycz 2003] emphasizes that the granulation process should balance specificity and generality, ensuring that the resulting granules preserve essential information while eliminating unnecessary details. This principle guides the formation of granules that are both computationally efficient and semantically meaningful.

Yao's tri-level framework [Yiyu Yao 2007] offers a comprehensive perspective on information granulation through three interconnected levels:

- **Philosophical level**: Addresses the conceptual aspects of granular computing, including structural thinking paradigms and granular perspectives on complex systems.
- **Representational level**: Focuses on the models used to represent information granules, such as intervals, fuzzy sets, rough sets, and clusters.
- **Computational level**: Encompasses algorithms and methodologies for processing granules, including granulation, organization, and computing with granules.

The principle of *uncertainty tolerance* is central to information granulation, acknowledging that granular representations inherently accommodate uncertainty and vagueness in data [Pedrycz 2018]. This tolerance enables robust processing of imprecise or incomplete information, particularly valuable in real-world applications where data quality may be compromised.

The principle of *multiple representations* suggests that multiple granular perspectives can coexist, each offering complementary views of the same underlying phenomenon [YY Yao 2006]. This multi-view approach enables comprehensive analysis by capturing different aspects of complex systems through varying granular perspectives.

B. Granular Structures

Granular structures refer to the organizational frameworks that define relationships among information granules, enabling systematic navigation across different levels of abstraction [Yiyu Yao 2008]. These structures serve as the foundation for granular computing methodologies, facilitating operations such as granule comparison, composition, and transformation.

Hierarchical Structures: Hierarchical organization represents one of the most common forms of granular structures, where granules are arranged in levels of increasing or decreasing granularity [Pedrycz 2018]. In hierarchical structures, granules are organized in ordered levels where finer granules are contained within coarser ones.

Nested Structures: Nested granular structures exhibit the property that each granule at a coarser level completely contains one or more granules from finer levels [Pedrycz 2005]. This property ensures consistency across granularity levels and facilitates smooth transitions between different resolutions.

Covering Structures: In covering granular structures, granules at the same or different levels may partially overlap, relaxing the strict nesting requirement [Qian et al. 2010]. This flexibility accommodates scenarios where clear-cut boundaries between granules are impractical, allowing for more natural representations of real-world phenomena.

Granularity Measures: Quantifying the resolution or specificity of granules is essential for comparing and manipulating granular structures. Various measures have been proposed to assess granule size, complexity, and information content, enabling systematic evaluation and optimization of granular representations.

C. Granulation Methods

Granulation methods encompass the algorithmic approaches and techniques for forming, refining, and manipulating information granules [Pedrycz 2018]. These methods bridge theoretical principles and practical applications, enabling the implementation of granular computing systems across diverse domains.

Top-down vs. Bottom-up Approaches: Granulation methods can be broadly categorized into two fundamental approaches:

- **Top-down granulation** begins with coarse, high-level granules and progressively refines them to achieve finer granularity. This approach aligns with deductive reasoning, starting from general concepts and moving toward specific details.
- **Bottom-up granulation** starts with atomic or fine-grained entities and aggregates them to form larger granules. This inductive approach consolidates detailed information into increasingly abstract representations, building complexity from simple components.

Both approaches offer distinct advantages depending on the application domain and the nature of available data, with many practical systems employing hybrid strategies that combine elements of both methodologies.

D. Mathematical Frameworks

Granular computing leverages diverse mathematical frameworks to formalize the representation and manipulation of information granules [Pedrycz, Skowron, et al. 2008]. These frameworks provide rigorous foundations for granular methods and enable systematic analysis of granular systems across different domains. Each framework contributes unique capabilities for handling uncertainty, imprecision, and varying levels of abstraction inherent in granular computing.

Fuzzy Set Theory Fuzzy set theory, introduced by [Lotfi Asker Zadeh 1965], offers a natural framework for representing information granules with imprecise or gradual boundaries. This approach aligns perfectly with granular computing's need to handle uncertain and context-dependent information groupings.

In fuzzy granular computing, an information granule A in universe X is characterized by a membership function:

$$\mu_A : X \to [0, 1] \tag{2.2}$$

where $\mu_A(x)$ represents the degree to which element x belongs to granule A. This gradual membership allows granules to have soft boundaries, reflecting the natural imprecision in real-world categorizations.

The *extension principle* [Lotfi Asker Zadeh 1975] enables operations on fuzzy granules, allowing granular computing systems to process and transform uncertain information while preserving the gradual nature of granular boundaries.

This framework is particularly valuable in granular computing applications requiring flexible categorization, such as natural language processing, pattern recognition, and decision support systems where crisp boundaries are neither realistic nor desirable.

Rough Set Theory Rough set theory, developed by [Pawlak 1982], provides a complementary approach to granular computing by formalizing approximation and indiscernibility in information systems. This framework excels in scenarios where granules must be formed from incomplete or imprecise data.

Given an information system (U, A) where U represents objects and A represents attributes, rough set theory creates elementary granules through indiscernibility relations. Objects that cannot be distinguished based on available attributes form natural granules, represented by equivalence classes $[x]_B$ for attribute subset $B \subseteq A$.

For any concept $X \subseteq U$, rough set theory approximates it using these elementary granules through lower and upper approximations:

$$\underline{B}X = \{x \in U | [x]_B \subseteq X\} \tag{2.3}$$

$$\overline{B}X = \{x \in U | [x]_B \cap X \neq \emptyset\} \tag{2.4}$$

The boundary region $BN_B(X) = \overline{B}X - \underline{B}X$ represents the granular uncertainty, containing objects that cannot be definitively classified. This three-region structure directly supports granular computing by providing different levels of certainty in classification and decision-making.

The three-way decision framework [Yiyu Yao 2010] extends this approach by explicitly incorporating delayed decision strategies, allowing granular computing systems to defer decisions when information is insufficient, thus improving overall system reliability.

Interval Analysis Interval analysis [Moore 1966] supports granular computing by representing granules with precise boundaries but uncertain internal content. An interval granule $[a, b]$ encapsulates a range of possible values, enabling granular arithmetic operations:

$$[a, b] + [c, d] = [a + c, b + d] \tag{2.5}$$

This framework proves essential for numerical granular computing applications where uncertainty bounds must be preserved throughout computational processes. Interval-valued fuzzy sets [Gorzałczany 1987] combine interval and fuzzy approaches, allowing granules to have both uncertain boundaries and uncertain membership degrees, creating a rich granular representation suitable for complex real-world phenomena.

Cluster Analysis Clustering provides a data-driven approach to granule formation through similarity-based grouping [A. K. Jain et al. 1999]. Traditional hard clustering creates crisp granules, while fuzzy clustering allows overlapping granules that better reflect the gradual nature of many real-world categorizations.

The fuzzy c-means algorithm [Bezdek et al. 1981] creates overlapping granules by minimizing:

$$J_m = \sum_{i=1}^{n} \sum_{j=1}^{c} u_{ij}^m \|x_i - c_j\|^2 \tag{2.6}$$

where u_{ij} represents the membership degree of object x_i in granule j, and m controls the degree of fuzziness. This approach enables granular computing systems to automatically discover meaningful granular structures from data without prior domain knowledge.

Framework Integration These mathematical frameworks complement each other in granular computing applications. Fuzzy sets handle gradual boundaries, rough sets manage incomplete information, intervals preserve uncertainty bounds, and clustering discovers data-driven granules. Modern granular computing systems often combine multiple frameworks to

leverage their complementary strengths, creating hybrid approaches that can handle the full spectrum of uncertainty and imprecision encountered in complex real-world applications.

2.1.4. Granular Rule Induction and Extraction

Granular rule induction refers to the process of deriving interpretable if-then rules from granulated datasets. This approach is particularly suitable for domains requiring transparent decision-making, such as healthcare, finance, and environmental monitoring. The rules are typically extracted through algorithms that minimize entropy and maximize granularity coherence, thereby maintaining a balance between generality and specificity.

The fundamental advantage of granular rule induction lies in its ability to produce human-interpretable rules while maintaining computational efficiency. Unlike black-box approaches such as deep neural networks, granular rules provide explicit logical statements that domain experts can validate and understand.

For example, in air quality monitoring, granular rule induction has been applied in this research to extract interpretable classification rules that relate environmental conditions—such as $PM_{2.5}$ levels, temperature, dew point, atmospheric pressure, and wind speed—to air pollution severity. These rules are derived by forming information granules based on threshold-based discretization of continuous features and optimizing for minimal entropy and maximal generality. The resulting rules enable transparent decision-making, such as classifying air quality into "Good," "Moderate," or "Unhealthy" categories. Furthermore, integration with fuzzy logic enhances the robustness of rule induction under uncertainty, allowing the system to handle imprecise boundaries between air quality levels. Recent developments also include hybrid approaches that combine granular computing with genetic algorithms and neural-symbolic systems to refine rule sets and improve classification performance in dynamic environmental conditions.

Building upon this foundation, we now present the theoretical framework for constructing granular computing models, beginning with the formal representation of data structures and proceeding through the rule extraction methodology.

A. Information Table

An information table serves as a foundational structure for the systematic representation of data. It is organized in a tabular form, encapsulating entities (or objects), their associated

attributes, and the corresponding attribute values. Formally, an information table is defined as:

$$S = (U, A_t, L, \{V_a \mid a \in A_t\}, \{f_a \mid a \in A_t\}) \tag{2.7}$$

In this representation, U denotes a non-empty set of objects (the universe), A_t is a non-empty set of attributes, V_a represents the domain of possible values associated with attribute a (where $a \in A_t$), and f_a is an information function that maps each object in U to a specific value in V_a, i.e., $f_a : U \to V_a$. The component L refers to the formal language defined over the attribute set A_t, enabling the construction of logical formulas that describe object properties. The atomic formulas in L take the form $a = v$ with $a \in A_t$ and $v \in V_a$, and more complex formulas can be constructed using logical connectives such as conjunction ($\wedge$), disjunction ($\vee$), and negation ($\neg$).

The process of granulating the universe U involves partitioning the objects into clusters, subsets, or groups based on some measure of similarity. Each such subset, known as a *granule*, may be either crisp or fuzzy in nature [Yiyu Yao 2004]. These granules can be systematically derived from the language L. Specifically, for an atomic expression $a = v$, the corresponding basic granule is denoted as $m(a = v)$ and represents the set of all objects in U that satisfy the condition $a = v$, formally defined as $m(a = v) = \{x \in U : f_a(x) = v\}$ [J. Yao and YY Yao 2002].

Example: Consider an air quality dataset where U contains daily measurements, $A_t = \{PM_{2.5}, Temperature, Humidity\}$, and we have discrete values such as $V_{PM_{2.5}} = \{Low, Medium, High\}$. The granule $m(PM_{2.5} = High)$ would contain all measurement days where particulate matter levels were classified as high.

B. Granule Measures and Evaluation Metrics

In the process of extracting classification rules from data, a variety of evaluation metrics are employed to assess the quality of individual granules and the relationships between granule pairs. These metrics collectively determine the usefulness and reliability of extracted rules, guiding the selection process during rule induction. One of the fundamental metrics is *Generality* (Equation 2.8), which evaluates how broadly a granule, defined by a formula ϕ, covers the universe U. It is computed as the ratio of the number of objects contained in the granule $m(\phi)$ to the total number of elements in U:

$$Generality(\phi) = \frac{|m(\phi)|}{|U|} \tag{2.8}$$

A higher generality value indicates that the granule covers a larger portion of the dataset, suggesting broader applicability but potentially lower specificity.

A second important metric is *Confidence* (Equation 2.9). This measure quantifies the reliability of the implication between two formulas, ϕ and ψ, by computing the proportion of objects that satisfy both formulas relative to those satisfying ϕ:

$$Confidence(\phi \rightarrow \psi) = \frac{|m(\phi \wedge \psi)|}{|m(\phi)|} = \frac{|m(\phi \cap \psi)|}{|m(\phi)|} \tag{2.9}$$

where $m(\phi \wedge \psi)$ represents the granule containing objects that satisfy both conditions ϕ and ψ simultaneously.

Another related metric, *Coverage* (Equation 2.10), assesses how much of the target formula ψ is accounted for by the antecedent formula ϕ. It is defined as the proportion of the overlapping instances $m(\phi \wedge \psi)$ relative to all instances satisfying ψ:

$$Coverage(\phi \rightarrow \psi) = \frac{|m(\phi \wedge \psi)|}{|m(\psi)|} \tag{2.10}$$

The most critical metric is *Conditional Entropy* (Equation 2.11), which quantifies the internal consistency or homogeneity of the granule induced by ϕ with respect to a family of outcome formulas $\psi = \{\psi_1, \ldots, \psi_n\}$. This metric reflects the uncertainty in predicting ψ given ϕ, and is calculated as follows:

$$ConditionalEntropy(\psi|\phi) = -\sum_{i=1}^{n} P(\psi_i|\phi) \log_2(P(\psi_i|\phi)) \tag{2.11}$$

Here, the conditional probability distribution $P(\psi_i|\phi) = \frac{|m(\phi \wedge \psi_i)|}{|m(\phi)|}$ captures the likelihood of each ψ_i occurring given the granule defined by ϕ. Lower conditional entropy values indicate more homogeneous granules, suggesting better classification potential. The conditional entropy thus serves as a powerful tool to evaluate how well the rule $\phi \rightarrow \psi$ partitions the dataset in a meaningful and informative way.

C. Induction of Classification Rules and Granule Tree Construction

The process begins with the generation of a set of basic concepts derived from atomic formulas of the form $a = v$. These atomic concepts form the foundation for granule

construction. Subsequently, each basic granule is assessed using the evaluation metrics discussed in Section B. Rule induction is guided by optimizing these metrics—specifically, by minimizing conditional entropy while maximizing generality, confidence, and coverage.

The procedure for constructing the granule network is outlined in Algorithm 1, which is adapted from the approach described in [J. Yao and YY Yao 2002].

Algorithm 1 Granule Network Construction

Initialize the basic concept family using atomic formulas: $BC(U) = \{(a = v, m(a = v)) \mid a \in A_t, v \in V_a\}$.

Assign unused basic concepts: $UBC(U) = BC(U)$.

Initialize the granule network: $GN = (U, \emptyset)$, consisting of a single node without edges.

while the smallest granules in GN do not fully solve the classification task **do**

Evaluate each unused basic concept using the fitness criteria.

Select the concept $C = (a = v, m(a = v))$ with the highest fitness score (low entropy, high generality, confidence, and coverage).

Update $UBC(U) = UBC(U) \setminus \{C\}$.

Expand the granule network GN by generating new nodes as intersections of $m(a = v)$ with existing nodes, and connect them using arcs labeled with $a = v$.

The granular computing (GrC) framework for rule induction has been explored in several domains, though its adoption remains relatively limited. For instance, [Rozehkhani and Mohammadzad 2022] applied GrC to classify patients and diagnose COVID-19 based on symptoms. Similarly, it has been used to assess seismic vulnerability in various studies [Khamespanah et al. 2013; Samadi Alinia and Delavar 2011; Hossein Sheikhian et al. 2017]. In another case, [Rozehkhani and Mahan 2022] utilized GrC to derive rules for estimating the required number of virtual machines based on relevant system parameters.

Despite these efforts, the application of GrC for rule extraction remains underdeveloped. The primary limitations include: (1) sensitivity to parameter selection, particularly the weight values in the fitness function (the function that combines the granule measurement metrics); (2) computational complexity that scales poorly with attribute dimensionality; and (3) limited theoretical guarantees on rule optimality. In this study, a novel GrC-based algorithm specifically designed for meteorological data analysis was constructed from the ground up to derive interpretable and actionable rules associated with $PM_{2.5}$ concentration levels. Our

approach introduces several key innovations: (1) an advanced GrC rule extraction model that prioritizes granules with zero entropy, ensuring maximum class purity and eliminating ambiguous classification boundaries; (2) an intelligent granule selection mechanism that systematically addresses remaining unclassified objects while minimizing redundancy with previously classified instances through Jaccard Index optimization; and (3) a robust fallback strategy for handling test instances that do not match any extracted rules, incorporating hierarchical rule relaxation and similarity-based classification to ensure comprehensive coverage. To the best of our knowledge, existing algorithms are often poorly generalized, lack rigorous explanation, and have typically been validated only on small-scale datasets with limited real-world applicability.

This comprehensive examination of granular computing establishes its fundamental importance as a paradigm for interpretable, uncertainty-aware information processing. The key contributions of GrC to intelligent systems include:

- **Cognitive Alignment**: By mimicking human reasoning processes, GrC creates more intuitive and explainable AI systems
- **Uncertainty Management**: Multiple mathematical frameworks provide robust handling of imprecise and incomplete information
- **Multi-level Abstraction**: Hierarchical granular structures enable flexible navigation between detail and abstraction
- **Rule Interpretability**: Extracted rules provide transparent decision logic accessible to domain experts

The evolution from theoretical foundations to practical applications demonstrates GrC's maturity as a computational paradigm. However, several challenges remain:

Current Limitations:

1. Computational complexity scaling with high-dimensional data
2. Parameter sensitivity in granule formation algorithms
3. Limited theoretical guarantees for optimal granule selection
4. Integration challenges with modern deep learning architectures

Future Research Directions:

1. **Hybrid Architectures**: Integrating GrC with deep learning for explainable neural networks
2. **Adaptive Granulation**: Developing self-organizing granular structures that evolve with data
3. **Distributed Computing**: Scaling GrC algorithms for big data environments
4. **Domain-Specific Applications**: Extending GrC to emerging fields like quantum computing and biotechnology

As demonstrated in this thesis through air quality applications, GrC provides a robust foundation for developing interpretable, reliable, and human-centered intelligent systems. The integration of granular computing with modern AI techniques represents a promising direction for addressing the growing demand for explainable artificial intelligence.

2.2. Time Series Analysis

Time series analysis is a fundamental methodological framework for extracting meaningful insights and making predictions from data collected over time. It is especially critical in domains where observations are recorded at consistent intervals, such as finance, meteorology, and environmental monitoring [Hyndman and Athanasopoulos 2018]. In the context of air quality research, time series analysis is essential for forecasting pollutant concentrations, identifying seasonal patterns, and informing policy-making aimed at mitigating public health risks. This section presents a comprehensive overview of time series analysis, covering basic definitions, key components, classical statistical models, modern machine learning techniques, and associated challenges.

2.2.1. Definition and Importance

A time series is defined as a sequence of data points $\{x_t\}$ collected at regular, equally spaced time intervals $t = 1, 2, \ldots, T$ [Box et al. 2015]. The main objective of time series analysis is to uncover underlying patterns such as trend, seasonality, and cycles, and to develop models that can forecast future values based on past observations.

In air quality studies, forecasting pollutants such as $PM_{2.5}$ and PM_{10} can help provide early warnings, support regulatory compliance, and reduce adverse health effects. Time series analysis enables researchers and practitioners to better understand pollution dynamics and react proactively.

2.2.2. Components of Time Series Data

Time series data typically consist of the following components (Figure 2.2):

- **Trend (T_t):** The long-term progression in the data, which may be increasing, decreasing, or stable.
- **Seasonality (S_t):** Repetitive and predictable patterns occurring at fixed intervals (e.g., daily, weekly, annually).
- **Cyclic Patterns (C_t):** Fluctuations over longer, non-fixed durations often driven by economic or environmental cycles.
- **Residual/Noise (e_t):** Irregular or random variations that remain after trend, seasonality, and cycles are removed.

These components can be combined additively or multiplicatively:

$$x_t = T_t + S_t + C_t + e_t \quad \text{(Additive Model)} \tag{2.12}$$

$$x_t = T_t \cdot S_t \cdot C_t \cdot e_t \quad \text{(Multiplicative Model)} \tag{2.13}$$

2.2.3. Stationarity and Differencing

A stationary time series has statistical properties—such as mean, variance, and autocorrelation—that do not change over time [Box et al. 2015]. Stationarity is a prerequisite for many classical forecasting techniques, including autoregressive (AR) and moving average (MA) models.

To assess stationarity, the Augmented Dickey-Fuller (ADF) test is commonly used [Dickey and Fuller 1979]. If a series is non-stationary, it can be transformed through differencing:

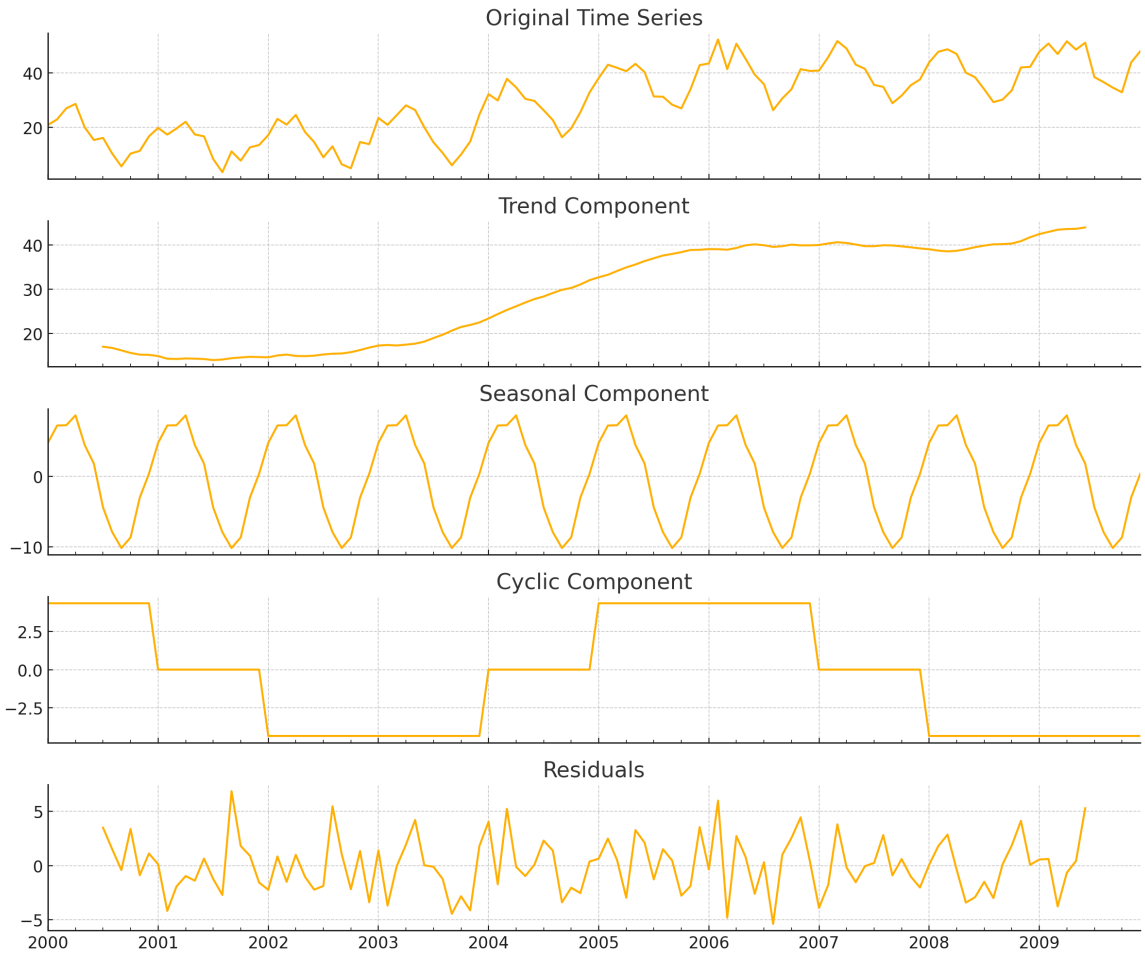


Figure 2.2.: Decomposition of a time series into its constituent components: (a) Original time series data, (b) Trend component, (c) Seasonal component, (d) Residual component.

$$y_t = x_t - x_{t-1} \tag{2.14}$$

where y_t denotes the first-order differenced series. Higher-order differencing may be applied if necessary.

2.2.4. Autocorrelation and Partial Autocorrelation

The Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) are essential tools in time series analysis. They help identify the dependency structure in sequential data and guide the selection of appropriate lag terms for autoregressive (AR) and moving average (MA) models.

Autocorrelation Function (ACF) The ACF measures the linear correlation between a time series and its own past values at various lags. For a stationary time series $\{x_t\}$, the autocorrelation at lag k is given by:

$$\rho_k = \frac{\sum_{t=k+1}^{T}(x_t - \bar{x})(x_{t-k} - \bar{x})}{\sum_{t=1}^{T}(x_t - \bar{x})^2} \tag{2.15}$$

where $\bar{x}$ is the sample mean of the series. The ACF provides insight into the presence of MA components.

Partial Autocorrelation Function (PACF) The PACF measures the correlation between x_t and x_{t-k} after removing the influence of intermediate lags $x_{t-1}, \ldots, x_{t-k+1}$. It captures the direct effect of past observations on the current value. Mathematically, for lag k, PACF corresponds to the coefficient ϕ_{kk} in the autoregressive model, and ϵ_t is the white noise error term at time t:

$$x_t = \phi_{k1}x_{t-1} + \phi_{k2}x_{t-2} + \cdots + \phi_{kk}x_{t-k} + \epsilon_t \tag{2.16}$$

Model Identification ACF and PACF plots (correlograms) are used for identifying appropriate orders of ARIMA models:

- ACF with a sharp cutoff and gradually decaying PACF suggests an MA process.
- PACF with a sharp cutoff and gradually decaying ACF suggests an AR process.

- Both ACF and PACF decaying slowly indicate an ARMA process.

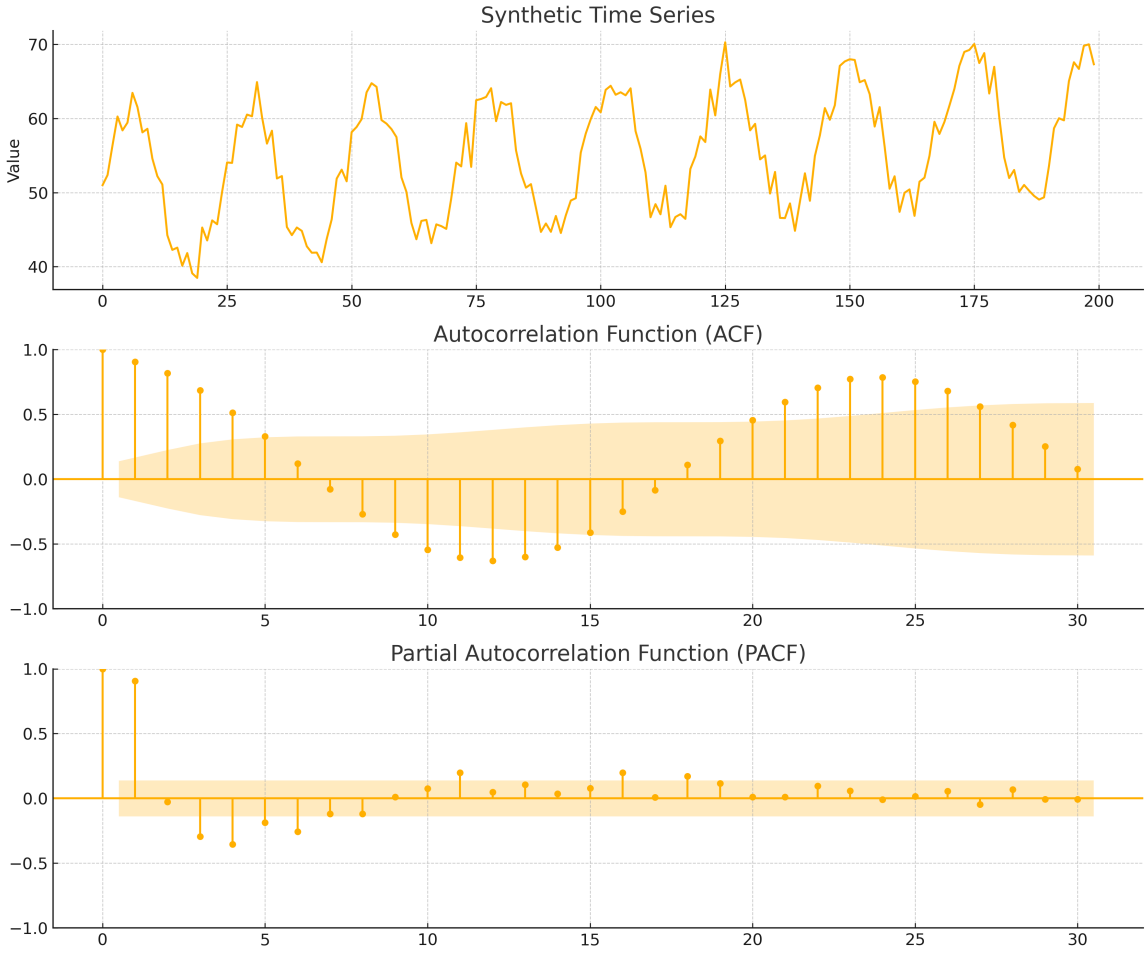


Figure 2.3.: Synthetic time series (top), Autocorrelation Function (middle), and Partial Autocorrelation Function (bottom). The ACF and PACF help determine lag dependencies in the data.

2.2.5. Lag Features and Temporal Feature Engineering

In machine learning-based time series forecasting, temporal dependencies must be explicitly encoded using engineered features. Common transformations include:

- **Lag features:** Previous time steps, e.g., $x_{t-1}, x_{t-2}, \ldots$

- **Rolling statistics:** Moving average or standard deviation over a window.
- **Date/time components:** Hour of day, day of week, season, etc., to capture periodicity.

These features allow non-sequential models (e.g., random forests) to capture time-based dynamics and enhance performance on structured temporal tasks.

2.2.6. Classical Time Series Models

Classical models form the foundation of time series analysis and remain widely used for their interpretability and robustness.

A. Autoregressive Integrated Moving Average (ARIMA)

The ARIMA model, denoted ARIMA(p, d, q), integrates autoregressive (AR), differencing (I), and moving average (MA) components to handle non-stationary time series [Box et al. 2015]:

$$\phi(B)(1 - B)^d x_t = \theta(B)\epsilon_t \tag{2.17}$$

Here, B is the backward shift operator, $\phi(B)$ and $\theta(B)$ are polynomials of order p and q, d is the order of differencing, and ϵ_t is white noise.

B. Seasonal ARIMA (SARIMA)

SARIMA extends ARIMA by incorporating seasonal effects:

$$\phi(B)\Phi(B^s)(1 - B)^d(1 - B^s)^D x_t = \theta(B)\Theta(B^s)\epsilon_t \tag{2.18}$$

where s is the seasonal period, and $\Phi(B^s)$ and $\Theta(B^s)$ represent seasonal AR and MA components, with D indicating seasonal differencing.

2.2.7. Machine Learning Models for Time Series

Unlike ARIMA-based models that assume linearity and stationarity, machine learning models can learn complex nonlinear mappings and are generally data-driven rather than assumption-driven. These models are particularly advantageous when the time series exhibits nonlinear

dynamics, contains high-dimensional features, or includes exogenous variables that interact in complex ways.

Machine learning (ML) techniques have increasingly been applied to time series forecasting due to their ability to capture nonlinear relationships and high-dimensional dependencies. Unlike classical models, ML models often do not require assumptions of stationarity or linearity.

- **Random Forests and Gradient Boosting:** These tree-based models can handle multivariate inputs and interactions but require feature engineering (e.g., lags) [Dudek 2022; Mallala et al. 2025; Taieb and Hyndman 2014].
- **Support Vector Regression (SVR):** Effective for capturing nonlinear patterns in small datasets [Dhiman et al. 2019].
- **Artificial Neural Networks (ANNs):** Can approximate complex functions but are prone to overfitting without sufficient data [G. P. Zhang 2003].
- **Recurrent Neural Networks (RNNs), LSTM, GRU:** Specifically designed for sequential data; LSTMs and GRUs mitigate vanishing gradients and are well-suited for long-term dependencies [Bandara et al. 2020].
- **Temporal Convolutional Networks (TCNs):** Use 1D dilated convolutions to model long-range dependencies and allow for efficient parallelization [Hewage et al. 2020].
- **DeepAR and Transformer Models:** Enable probabilistic and multivariate forecasting using autoregressive or attention-based mechanisms [Flunkert et al. 2017].

These models have been successfully employed in air pollution forecasting, especially for pollutants like $PM_{2.5}$, where complex meteorological interactions play a significant role [W. Li and Jiang 2023; Yadav et al. 2024].

2.2.8. Challenges in Time Series Forecasting

Several challenges must be addressed to develop effective time series models [Bontempi et al. 2012; Gama et al. 2014; Hyndman and Athanasopoulos 2018]:

- **Non-stationarity:** Many real-world series exhibit changing mean or variance over time.
- **Data Quality Issues:** Missing values, noise, and outliers are common in environmental data.
- **High Dimensionality:** Multivariate time series increase the risk of overfitting and model complexity.
- **Concept Drift:** Evolving data distributions over time necessitate model retraining or adaptation.
- **Long-Horizon Forecasting:** Error accumulation reduces accuracy over extended prediction windows.

Overcoming these challenges involves preprocessing techniques (e.g., imputation, normalization), robust model design, and often, ensemble or hybrid methods.

2.3. Fundamental Machine Learning Algorithms

Machine learning algorithms form the foundation of predictive modeling by automatically identifying patterns in data. This section covers fundamental classification algorithms that remain essential in modern data science applications despite the rise of deep learning. Each algorithm represents a different approach to decision boundary construction, from linear separators to tree-based ensembles, with distinct mathematical foundations, strengths, and limitations. These algorithms also serve broader purposes beyond classification - for instance, K-Nearest Neighbors was utilized in this study for filling missing values in PM2.5 environmental data, demonstrating the versatility of these classical methods. Understanding these classical algorithms provides crucial insights into more complex modern techniques and helps inform algorithm selection for specific applications.

2.3.1. Logistic Regression

Logistic Regression is a fundamental classification algorithm derived from linear regression, but adapted to predict binary outcomes through the use of the logistic function [Hosmer Jr et al. 2013].

A. Mathematical Formulation

Unlike linear regression which models continuous outputs directly, logistic regression models the probability that an input belongs to a particular class. For binary classification, the model estimates:

$$P(Y = 1|X) = \sigma(w^T X + b) \tag{2.19}$$

where σ is the sigmoid function defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2.20}$$

Variable Definitions:

- $P(Y = 1|X)$ - Probability that the output Y equals 1 given input features X
- $X \in \mathbb{R}^n$ - Input feature vector of dimension n
- $w \in \mathbb{R}^n$ - Weight vector (model parameters)
- $b \in \mathbb{R}$ - Bias term (intercept)
- $\sigma(\cdot)$ - Sigmoid function
- $z = w^T X + b$ - Linear combination of inputs (logits)

The decision boundary is defined by $w^T X + b = 0$, which forms a hyperplane in the feature space.

B. Algorithm Components

Linear Decision Boundary: Logistic regression creates a linear decision boundary in the feature space, making it interpretable and computationally efficient.

Probabilistic Output: Unlike other classifiers, logistic regression provides well-calibrated probability estimates, making it suitable for applications requiring confidence measures.

Regularization Options: Supports L1 (Lasso), L2 (Ridge), and Elastic Net regularization to prevent overfitting:

- L1 regularization: $R(w) = \lambda_1 \sum_{j=1}^{n} |w_j|$, where w_j is the j^{th} model weight, n is the number of features, and λ_1 is the L1 regularization strength.

- L2 regularization: $R(w) = \lambda_2 \sum_{j=1}^{n} w_j^2$, where w_j is the j^{th} model weight, n is the number of features, and λ_2 is the L2 regularization strength.
- Elastic Net: $R(w) = \alpha_1 \sum_{j=1}^{n} |w_j| + \alpha_2 \sum_{j=1}^{n} w_j^2$, where α_1 and α_2 control the contributions of the L1 and L2 penalties respectively.

C. Training and Optimization

Logistic regression is trained by minimizing the binary cross-entropy loss function:

$$J(w, b) = -\frac{1}{m} \sum_{i=1}^{m} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \tag{2.21}$$

where $\hat{y}_i = \sigma(w^T X_i + b)$ is the predicted probability for the i-th sample.

Training Variables:

- $J(w, b)$ - Binary cross-entropy loss function
- m - Number of training samples
- $y_i \in \{0, 1\}$ - True binary label for the i-th sample
- $\hat{y}_i \in [0, 1]$ - Predicted probability for the i-th sample

Parameters are updated using gradient descent:

$$w := w - \alpha \frac{\partial J}{\partial w} \tag{2.22}$$

$$b := b - \alpha \frac{\partial J}{\partial b} \tag{2.23}$$

where $\alpha > 0$ is the learning rate, and $\frac{\partial J}{\partial w}$ and $\frac{\partial J}{\partial b}$ are the partial derivatives of the loss function J with respect to the weight vector w and bias term b, respectively.

2.3.2. Support Vector Machines

Support Vector Machines (SVMs) are powerful supervised learning models that find the optimal hyperplane to separate classes by maximizing the margin between the decision boundary and the closest data points [Cortes and Vapnik 1995].

A. Mathematical Formulation

SVM finds a hyperplane that optimally separates classes while maximizing the margin. For non-separable data, it uses a soft margin approach:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \\ \text{subject to} \quad & y_i(w^T X_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \end{aligned} \tag{2.24}$$

Variable Definitions:

- $w \in \mathbb{R}^n$ - Weight vector (normal to the hyperplane)
- $X_i \in \mathbb{R}^n$ - Feature vector for the i-th sample
- $b \in \mathbb{R}$ - Bias term
- $y_i \in \{-1, +1\}$ - Binary class label for the i-th training sample
- $\xi_i \geq 0$ - Slack variable allowing margin violations
- $C > 0$ - Regularization parameter balancing margin maximization and error minimization
- m - Number of training samples

The decision function is:

$$f(X) = \text{sign}(w^T X + b) \tag{2.25}$$

B. Algorithm Components

Margin Maximization: SVM maximizes the margin between classes, leading to better generalization and robustness.

Support Vectors: Only uses a subset of training data (support vectors) for decision making, making it memory efficient.

Kernel Trick: Handles non-linear decision boundaries by mapping features to higher-dimensional spaces using kernel functions:

$$\text{Linear:} \quad K(X_i, X_j) = X_i^T X_j \tag{2.26}$$

$$\text{Polynomial:} \quad K(X_i, X_j) = (X_i^T X_j + r)^d \tag{2.27}$$

$$\text{RBF (Gaussian):} \quad K(X_i, X_j) = \exp(-\gamma \|X_i - X_j\|^2) \tag{2.28}$$

Kernel Parameters:

- $r \geq 0$ - Polynomial offset parameter
- d - Polynomial degree
- $\gamma > 0$ - RBF bandwidth parameter

C. Training and Optimization

SVM training involves solving a quadratic optimization problem, typically using Sequential Minimal Optimization (SMO) or similar algorithms. The kernelized decision function becomes:

$$f(X) = \text{sign}\left(\sum_{i \in SV} \alpha_i y_i K(X_i, X) + b\right) \tag{2.29}$$

Training Variables:

- α_i - Lagrange multipliers (learned during training)
- SV - Set of support vectors (subset of training data)
- $K(X_i, X)$ - Kernel function computing similarity between vectors

2.3.3. Decision Trees

Decision Trees are non-parametric supervised learning algorithms that create a tree-like model of decisions, where internal nodes represent features, branches represent decision rules, and leaf nodes represent class labels [Quinlan 1986].

A. Mathematical Formulation

Decision trees recursively partition the feature space using impurity measures to guide splitting decisions:

Gini Impurity:

$$\text{Gini}(t) = 1 - \sum_{i=1}^{K} p(i|t)^2 \tag{2.30}$$

Entropy:

$$\text{Entropy}(t) = -\sum_{i=1}^{K} p(i|t) \log_2 p(i|t) \tag{2.31}$$

Information Gain:

$$\text{IG}(t, a) = \text{Entropy}(t) - \sum_{v \in \text{Values}(a)} \frac{|t_v|}{|t|} \text{Entropy}(t_v) \tag{2.32}$$

Variable Definitions:

- t - A node in the decision tree
- K - Total number of classes
- $p(i|t)$ - Proportion of samples belonging to class i at node t
- a - An attribute (feature) used for splitting
- t_v - Subset of samples at node t for which attribute a has value v
- $|t|$ - Total number of samples at node t
- $\text{IG}(t, a)$ - Information gain from splitting node t on attribute a

B. Algorithm Components

Recursive Partitioning: The algorithm recursively splits the data based on feature values that maximize information gain or minimize impurity.

Stopping Criteria: Tree growth is controlled by criteria such as maximum depth, minimum samples per leaf, or minimum information gain.

Pruning Techniques: Prevent overfitting through:

- Pre-pruning: Early stopping based on criteria
- Post-pruning: Remove branches that don't improve validation performance

C. Training and Optimization

Decision trees use a greedy algorithm to find optimal splits at each node:

$$\begin{aligned} D_{left} &= \{(x, y) \in D : x_f \leq \theta\} \\ D_{right} &= \{(x, y) \in D : x_f > \theta\} \end{aligned} \tag{2.33}$$

Training Variables:

- D - Dataset (set of training samples)
- f - Selected feature index for splitting
- θ - Splitting threshold for continuous features
- D_{left}, D_{right} - Left and right subsets after splitting

The algorithm selects the feature and threshold that maximizes information gain at each node.

2.3.4. Random Forest

Random Forest (RF) is a powerful ensemble learning algorithm that constructs multiple decision trees during training and combines their outputs through majority voting (for classification) or averaging (for regression) to improve predictive accuracy and reduce overfitting [Breiman 2001].

A. Mathematical Formulation

Given a training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$, Random Forest builds B decision trees using bootstrap sampling and feature randomness.

For classification tasks, the final ensemble prediction is obtained by majority vote. On the other hand, for regression tasks, the final ensemble prediction is obtained by averaging:

$$\hat{y}(x') = \text{mode}\{f_1(x'), f_2(x'), ..., f_B(x')\} \tag{2.34}$$

Variable Definitions:

- $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$ – Training dataset with n input-output pairs

- x' – New input instance for prediction
- $\hat{y}(x')$ – Final predicted class label (regression value) for x'
- $f_b(\cdot)$ – Prediction function of the b-th decision tree
- B – Number of trees in the forest
- mode$\{\cdot\}$ – Majority class or average value among all tree predictions

B. Algorithm Components

Bootstrap Aggregating (Bagging):

Each tree is trained on a different bootstrap sample—i.e., a random sample of the training data drawn with replacement—leading to diverse decision boundaries.

Feature Randomness:

At each split, a random subset of features is considered instead of all features. The number of features to consider at each split is given by:

$$|F_{\text{split}}| = m_{\text{try}} \tag{2.35}$$

where:

$$m_{\text{try}} = \begin{cases} \sqrt{p}, & \text{(commonly used in classification)} \\ p/3, & \text{(commonly used in regression)} \end{cases} \tag{2.36}$$

Variable Definitions:

- p – Total number of input features
- m_{try} – Number of features randomly selected at each split
- F_{split} – Subset of features used at a split node

Out-of-Bag (OOB) Error: Each bootstrap sample excludes around one-third of the training data. These unused samples can be used to estimate model performance without needing a separate validation set.

Feature Importance: The importance of feature j is calculated by aggregating the impurity decrease caused by splits on that feature across all trees:

$$\text{Importance}(j) = \frac{1}{B} \sum_{b=1}^{B} \sum_{\text{splits on } j} \Delta I_{j,b} \tag{2.37}$$

Variable Definitions:

- $\Delta I_{j,b}$ – Reduction in impurity due to splits on feature j in tree b
- $\text{Importance}(j)$ – Average importance of feature j across all trees

C. Training and Optimization

Random Forest training involves the following steps:

1. **Bootstrap Sampling:** Generate B training subsets by randomly sampling with replacement from the original dataset.
2. **Tree Construction:** Train each tree on its bootstrap sample, using only a random subset of features at each split.
3. **Aggregation:** For classification, take the majority vote across trees. For regression, average the predictions.

The training process is inherently parallelizable since each decision tree is built independently. This makes Random Forest both powerful and scalable for large-scale datasets.

2.3.5. eXtreme Gradient Boosting (XGBoost)

XGBoost is a scalable, efficient, and regularized ensemble learning algorithm based on gradient boosting that sequentially builds an ensemble of weak learners (typically decision trees), with each new model trained to minimize the errors made by its predecessors [T. Chen and Guestrin 2016].

A. Mathematical Formulation

Let $\hat{y}_i^{(0)}$ denote the initial prediction for sample i (commonly the mean value in regression tasks). At iteration t, the model updates its prediction as:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{2.38}$$

Variable Definitions:

- $\hat{y}_i^{(t)}$ – Prediction for instance i after t boosting rounds
- $\hat{y}_i^{(t-1)}$ – Prediction after the previous round
- $f_t(x_i)$ – Output of the t-th decision tree (weak learner) for input x_i
- x_i – Feature vector for sample i

B. Algorithm Components

Gradient-Based Learning:

XGBoost uses both first-order (gradient) and second-order (Hessian) derivatives of the loss function to improve optimization:

$$g_i = \frac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}} \tag{2.39}$$

$$h_i = \frac{\partial^2 l(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2} \tag{2.40}$$

Variable Definitions:

- g_i – First-order gradient of the loss with respect to prediction $\hat{y}_i^{(t-1)}$
- h_i – Second-order derivative (Hessian) of the loss function
- $l(y_i, \hat{y}_i^{(t-1)})$ – Loss between true label y_i and predicted value $\hat{y}_i^{(t-1)}$

Regularization:

To control model complexity and reduce overfitting, XGBoost includes a regularization term:

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{2.41}$$

Variable Definitions:

- $\Omega(f_t)$ – Regularization penalty for tree f_t
- γ – Regularization coefficient for the number of leaves
- λ – L2 regularization term for leaf weights
- T – Total number of leaves in the tree
- w_j – Score/weight assigned to leaf j

Optimal Leaf Weights:
Leaf weights are computed to minimize the regularized loss:

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \tag{2.42}$$

Variable Definitions:

- w_j^* – Optimal weight for leaf j
- I_j – Set of sample indices assigned to leaf j
- g_i, h_i – First and second-order gradients for instance i
- λ – Regularization parameter (as above)

C. Training and Optimization

The final objective combines the loss function over all samples and the regularization term:

$$\mathcal{L} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{t=1}^{T} \Omega(f_t) \tag{2.43}$$

Variable Definitions:

- $\mathcal{L}$ – Total training loss with regularization
- n – Total number of training samples
- $l(y_i, \hat{y}_i)$ – Prediction loss for instance i
- $\Omega(f_t)$ – Regularization for tree f_t

- T – Total number of trees in the ensemble

Training Features

- **Parallel Processing:** XGBoost parallelizes tree construction for scalability.
- **Missing Value Handling:** Learns the best default direction for missing values at each split.
- **Early Stopping:** Stops training when validation performance does not improve for a set number of rounds.
- **Feature Importance:** Provides multiple measures of feature importance (e.g., gain, frequency).

2.3.6. K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is a non-parametric, instance-based learning algorithm widely used for both classification and regression tasks. Unlike parametric models that assume a fixed functional form, KNN makes predictions based on the similarity of new data points to existing samples in the training set [Cover and Hart 1967]. In this study, KNN was particularly applied for **regression-based imputation** to fill missing values in $PM_{2.5}$ air quality data by leveraging proximity in feature space.

A. Mathematical Formulation

Given a training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$, and a query sample x_q, KNN identifies the k nearest neighbors $\mathcal{N}_k(x_q)$ in feature space and predicts the output based on their labels.

Classification prediction:

$$\hat{y}_q = \arg\max_{c \in \{1,2,\ldots,C\}} \sum_{x_i \in \mathcal{N}_k(x_q)} \mathbb{I}[y_i = c] \tag{2.44}$$

Regression prediction:

$$\hat{y}_q = \frac{1}{k} \sum_{x_i \in \mathcal{N}_k(x_q)} y_i \tag{2.45}$$

Variable Definitions:

- $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$: Labeled training data

- $x_q \in \mathbb{R}^d$: Query point
- $\hat{y}_q$: Predicted output for x_q
- k: Number of nearest neighbors
- $\mathcal{N}_k(x_q)$: Set of k nearest neighbors to x_q
- C: Number of classes (classification only)
- $\mathbb{I}[\cdot]$: Indicator function (1 if condition is true, 0 otherwise)

B. Algorithm Components

Distance Metrics: KNN uses a distance function to compute similarity:

$$\text{Euclidean:} \quad d(x_i, x_q) = \sqrt{\sum_{j=1}^{d} (x_{i,j} - x_{q,j})^2} \tag{2.46}$$

$$\text{Manhattan:} \quad d(x_i, x_q) = \sum_{j=1}^{d} |x_{i,j} - x_{q,j}| \tag{2.47}$$

$$\text{Minkowski (generalized):} \quad d(x_i, x_q) = \left(\sum_{j=1}^{d} |x_{i,j} - x_{q,j}|^p \right)^{1/p} \tag{2.48}$$

Weighting Schemes: Optionally, neighbors can be weighted by their distance (e.g., inverse distance) to give closer neighbors more influence.

C. Training and Optimization

KNN does not require explicit training (lazy learner), but optimization occurs during inference:

- **Efficient Search:** Data structures such as KD-Trees or Ball Trees are used to speed up nearest neighbor searches.
- **Hyperparameter Selection:** The value of k, choice of distance metric, and weighting scheme significantly affect performance.

- **Scalability Considerations:** Large datasets can increase prediction time and memory usage; dimensionality reduction may be applied.
- **Missing Value Imputation:** In this study, KNN regression was applied to estimate missing $PM_{2.5}$ values based on similarity to nearby fully observed instances.

2.4. Deep Learning

Deep learning is a subset of machine learning that employs artificial neural networks with multiple layers to progressively extract higher-level features from raw input. This section explores fundamental deep learning architectures, their mathematical foundations, and their applications to time series analysis [Goodfellow et al. 2016; LeCun et al. 2015].

2.4.1. Artificial Neural Networks

Artificial Neural Networks (ANNs) are computing systems inspired by biological neural networks, consisting of interconnected nodes (neurons) organized in layers that transform input data through weighted connections and non-linear activation functions [Goodfellow et al. 2016].

A. Mathematical Formulation

A feedforward neural network with L layers processes information sequentially from input to output. For the l-th layer, the forward propagation is defined as:

$$Z^{[l]} = W^{[l]} A^{[l-1]} + b^{[l]} \tag{2.49}$$

$$A^{[l]} = g^{[l]}(Z^{[l]}) \tag{2.50}$$

Variable Definitions:

- L - Total number of layers in the network
- l - Index of the current layer, where $l \in \{1, 2, \ldots, L\}$
- $X \in \mathbb{R}^{n \times d}$ - Input data matrix with n samples and d features
- $W^{[l]} \in \mathbb{R}^{n_l \times n_{l-1}}$ - Weight matrix connecting layer $(l-1)$ to layer l

- $b^{[l]} \in \mathbb{R}^{n_l}$ - Bias vector for layer l
- $Z^{[l]} \in \mathbb{R}^{n_l \times n}$ - Linear transformation (pre-activation) at layer l
- $A^{[l]} \in \mathbb{R}^{n_l \times n}$ - Activation output of layer l, with $A^{[0]} = X^T$
- $g^{[l]}(\cdot)$ - Activation function for layer l
- n_l - Number of neurons in layer l

B. Architecture and Key Components

Activation Functions: Introduce non-linearity, enabling networks to learn complex patterns:

$$\text{Sigmoid:} \quad \sigma(z) = \frac{1}{1 + e^{-z}} \tag{2.51}$$

$$\text{ReLU:} \quad \text{ReLU}(z) = \max(0, z) \tag{2.52}$$

$$\text{Tanh:} \quad \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{2.53}$$

$$\text{Softmax:} \quad \text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{2.54}$$

Activation Function Variables:

- z - Input to the activation function (pre-activation value)
- z_i - The i-th component of the input vector for softmax
- K - Number of classes (for softmax output layer)
- $\sigma(\cdot)$ - Sigmoid function notation

Loss Functions: The network learns by minimizing a loss function appropriate to the task:

Binary Cross-Entropy (Binary Classification):

$$\mathcal{L}_{BCE}(y, \hat{y}) = -\frac{1}{n} \sum_{i=1}^{n} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \tag{2.55}$$

Categorical Cross-Entropy (Multi-class Classification):

$$\mathcal{L}_{CCE}(y, \hat{y}) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{K} y_{ij} \log(\hat{y}_{ij}) \tag{2.56}$$

Mean Squared Error (Regression):

$$\mathcal{L}_{MSE}(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.57}$$

Loss Function Variables:

- $\mathcal{L}(\cdot)$ - Loss function measuring prediction error
- y - Vector of true labels for all samples
- $\hat{y}$ - Vector of predicted values for all samples
- y_i - True label for the i-th sample
- $\hat{y}_i$ - Predicted value for the i-th sample
- y_{ij} - One-hot encoded true label (1 if sample i belongs to class j, 0 otherwise)

C. Training and Optimization

Neural networks are trained using backpropagation [Rumelhart et al. 1986], which computes gradients of the loss function with respect to network parameters using the chain rule.

The backpropagation algorithm works by first computing how much each layer's pre-activation values $Z^{[l]}$ should change to reduce the loss, then using this information to update the weights and biases.

Step 1: Gradient with respect to pre-activations

$$\frac{\partial \mathcal{L}}{\partial Z^{[l]}} = \frac{\partial \mathcal{L}}{\partial A^{[l]}} \odot g'^{[l]}(Z^{[l]}) \tag{2.58}$$

Step 2: Gradient with respect to weights

$$\frac{\partial \mathcal{L}}{\partial W^{[l]}} = \frac{1}{n}\frac{\partial \mathcal{L}}{\partial Z^{[l]}}(A^{[l-1]})^T \tag{2.59}$$

Step 3: Gradient with respect to biases

$$\frac{\partial \mathcal{L}}{\partial b^{[l]}} = \frac{1}{n}\text{sum}(\frac{\partial \mathcal{L}}{\partial Z^{[l]}}, \text{axis} = 1) \tag{2.60}$$

Parameters are updated using gradient descent:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}(\theta_t) \tag{2.61}$$

Backpropagation Variables:

- $\frac{\partial \mathcal{L}}{\partial Z^{[l]}} \in \mathbb{R}^{n_l \times n}$ - Gradient of loss with respect to pre-activations at layer l
- $\frac{\partial \mathcal{L}}{\partial A^{[l]}} \in \mathbb{R}^{n_l \times n}$ - Gradient of loss with respect to activations at layer l
- $\frac{\partial \mathcal{L}}{\partial W^{[l]}} \in \mathbb{R}^{n_l \times n_{l-1}}$ - Gradient of loss with respect to weights at layer l
- $\frac{\partial \mathcal{L}}{\partial b^{[l]}} \in \mathbb{R}^{n_l}$ - Gradient of loss with respect to biases at layer l
- $g'^{[l]}(\cdot)$ - Derivative of activation function at layer l
- $\odot$ - Element-wise multiplication (Hadamard product)
- $(A^{[l-1]})^T \in \mathbb{R}^{n_{l-1} \times n}$ - Transpose of activations from previous layer
- $\text{sum}(\cdot, \text{axis} = 1)$ - Sum across samples (columns) for each neuron
- θ - Vector of all network parameters (weights and biases)
- $\eta > 0$ - Learning rate (step size for parameter updates, typically 10^{-4} to 10^{-1})
- t - Training iteration index, where $t \in \{0, 1, 2, \ldots\}$
- $\nabla_\theta \mathcal{L}(\theta_t)$ - Gradient of loss function with respect to all parameters

Regularization Techniques: To prevent overfitting and improve generalization, several regularization methods are commonly employed:

L1 Regularization: Adds a penalty term based on the absolute values of weights:

$$\mathcal{L}_{reg} = \mathcal{L} + \lambda_1 \sum_{l=1}^{L} \|W^{[l]}\|_1 \tag{2.62}$$

L2 Regularization: Adds a penalty term based on the squared weights:

$$\mathcal{L}_{reg} = \mathcal{L} + \lambda_2 \sum_{l=1}^{L} \|W^{[l]}\|_F^2 \tag{2.63}$$

Dropout: Randomly sets neurons to zero during training with probability p:

$$A^{[l]}_{dropout} = A^{[l]} \odot M^{[l]} \tag{2.64}$$

where $M^{[l]} \in \{0,1\}^{n_l}$ is a binary mask with each element drawn from Bernoulli$(1-p)$. Note that dropout is only applied during training, not during inference.

Regularization Variables:

- $\mathcal{L}_{reg}$ - Regularized loss function
- $\lambda_1 \geq 0$ - L1 regularization strength parameter (typically 10^{-6} to 10^{-2})
- $\lambda_2 \geq 0$ - L2 regularization strength parameter (typically 10^{-6} to 10^{-2})
- $\|W^{[l]}\|_1$ - L1 norm (sum of absolute values) of weight matrix $W^{[l]}$
- $\|W^{[l]}\|_F^2$ - Frobenius norm (sum of squared elements) of weight matrix $W^{[l]}$
- $p \in [0,1]$ - Dropout probability (typically 0.2-0.5 for hidden layers)
- $M^{[l]} \in \{0,1\}^{n_l}$ - Binary dropout mask for layer l
- $A^{[l]}_{dropout}$ - Activations after applying dropout
- Bernoulli$(1-p)$ - Bernoulli distribution with success probability $(1-p)$

Intuitive Explanation:

- The weight gradient $\frac{\partial \mathcal{L}}{\partial W^{[l]}}$ tells us how to adjust each weight to reduce the loss
- It's computed by multiplying how much each neuron's pre-activation should change $\left(\frac{\partial \mathcal{L}}{\partial Z^{[l]}}\right)$ with the input it received from the previous layer $(A^{[l-1]})^T$
- The bias gradient is simpler - it's just the average of how much each neuron's pre-activation should change across all training samples
- In practice, these gradients guide the learning process: large gradients indicate parameters that strongly influence the loss and should be adjusted more significantly

2.4.2. Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are specialized neural networks designed to process sequential data by maintaining an internal memory state that captures information from previous time steps [Elman 1990; Goodfellow et al. 2016].

A. Mathematical Formulation

Unlike feedforward networks, RNNs have connections that form directed cycles, allowing information to persist across time steps. For an input sequence $X = (x_1, x_2, \ldots, x_T)$, the RNN computes at each time step t:

$$h_t = g(W_{hh}h_{t-1} + W_{hx}x_t + b_h) \tag{2.65}$$

$$y_t = f(W_{yh}h_t + b_y) \tag{2.66}$$

with initial hidden state h_0 typically initialized to zero.

Variable Definitions:

- $X = (x_1, x_2, \ldots, x_T)$ - Input sequence of length T
- t - Current time step index, where $t \in \{1, 2, \ldots, T\}$
- $x_t \in \mathbb{R}^d$ - Input feature vector at time step t
- $h_t \in \mathbb{R}^h$ - Hidden state at time step t with h hidden units
- $y_t \in \mathbb{R}^o$ - Output at time step t with o output units
- $W_{hh} \in \mathbb{R}^{h \times h}$ - Hidden-to-hidden weight matrix (recurrent weights)
- $W_{hx} \in \mathbb{R}^{h \times d}$ - Input-to-hidden weight matrix
- $W_{yh} \in \mathbb{R}^{o \times h}$ - Hidden-to-output weight matrix
- b_h, b_y - Bias vectors for hidden and output layers
- $g(\cdot), f(\cdot)$ - Activation functions for hidden and output layers
- T - Sequence length (number of time steps)
- d, h, o - Dimensions of input, hidden, and output spaces respectively

B. Architecture and Key Components

Memory Mechanism: The hidden state h_t serves as the network's memory, carrying information from previous time steps through the recurrent connection $W_{hh}h_{t-1}$. This enables RNNs to model temporal dependencies and sequential patterns.

RNN Architectures for Air Quality Forecasting: In air quality time series modeling, Recurrent Neural Networks (RNNs) are typically employed using specific architectural configurations that align with the forecasting task. While RNNs can support diverse structures, the following architectures are most relevant for air pollution prediction tasks:

- **Many-to-One:** A sequence of historical observations is used to predict a single future value.
  - *Example:* Using the past 24 hours of $PM_{2.5}$, temperature, and humidity readings to forecast $PM_{2.5}$ at the next hour.
- **Many-to-Many (Synchronized):** Input and output sequences are aligned in time; a prediction is made for each step in the input.
  - *Example:* Predicting hourly $PM_{2.5}$ values for the next 24 hours, using aligned hourly sequences of meteorological and pollutant data.
- **Many-to-Many (Encoder–Decoder):** A past input sequence is encoded into a context vector, which is then decoded to produce a separate sequence of future predictions.
  - *Example:* Encoding the past 72 hours of multivariate pollutant data to forecast the next 24 hours of $PM_{2.5}$ concentration.

Other RNN configurations such as *One-to-One* and *One-to-Many* are less common in air quality forecasting but may be used in simplified or sensor-level classification tasks.

Note: Additional RNN architectures, including bidirectional RNNs, attention-based sequence models, and hierarchical RNNs, are more frequently applied in other domains such as natural language processing (e.g., machine translation) and computer vision (e.g., video classification), but are not the focus of this work.

Gradient Flow Problem: Standard RNNs suffer from vanishing/exploding gradients due to the multiplicative nature of gradient propagation through time:

$$\frac{\partial h_t}{\partial h_k} = \prod_{j=k+1}^{t} \frac{\partial h_j}{\partial h_{j-1}} \tag{2.67}$$

Gradient Flow Variables:

- $\frac{\partial h_t}{\partial h_k}$ - Gradient of hidden state at time t with respect to earlier time k
- k - Earlier time step index, where $k < t$
- j - Product index for gradient chain

C. Training and Optimization

RNNs are trained using Backpropagation Through Time (BPTT) [Werbos 1990], which unfolds the network through time and applies standard backpropagation:

$$\frac{\partial \mathcal{L}}{\partial W} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}_t}{\partial W} \tag{2.68}$$

Parameters are updated using gradient descent:

$$W := W - \eta \frac{\partial \mathcal{L}}{\partial W} \tag{2.69}$$

Training Variables:

- $\mathcal{L}$ - Total loss across all time steps
- $\mathcal{L}_t$ - Loss at time step t
- W - Generic weight parameter (applies to all weight matrices)
- $\eta > 0$ - Learning rate

Training Challenges [Bengio et al. 1994; Hochreiter and Schmidhuber 1997; Sutskever et al. 2013]:

- **Vanishing Gradients**: Gradients exponentially decrease through time, preventing learning of long-term dependencies
- **Exploding Gradients**: Gradients exponentially increase, causing training instability
- **Sequential Processing**: Limited parallelization due to temporal dependencies

- **Limited Memory**: Difficulty retaining information across long sequences

Common Solutions [Cho et al. 2014; Le et al. 2015; Pascanu et al. 2013; Williams and Peng 1990]:

- **Gradient Clipping**: Limit gradient magnitudes to prevent explosion
- **Proper Initialization**: Initialize recurrent weights carefully (e.g., orthogonal initialization)
- **Advanced Architectures**: Use LSTM or GRU units to address gradient flow problems
- **Sequence Truncation**: Limit sequence length for computational efficiency

2.4.3. Gated Recurrent Units

Gated Recurrent Units (GRUs) were introduced by [Cho et al. 2014] to address the vanishing gradient problem in standard RNNs by incorporating gating mechanisms that regulate information flow through the network while maintaining computational efficiency.

A. Mathematical Formulation

A GRU uses two gating mechanisms to control information flow and memory retention. At each time step t, the GRU computes:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{2.70}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{2.71}$$

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{2.72}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{2.73}$$

Variable Definitions:

- $x_t \in \mathbb{R}^d$ - Input vector at time step t
- $h_t \in \mathbb{R}^h$ - Hidden state at time step t
- $z_t \in \mathbb{R}^h$ - Update gate vector (controls information retention)

- $r_t \in \mathbb{R}^h$ - Reset gate vector (controls information forgetting)
- $\tilde{h}_t \in \mathbb{R}^h$ - Candidate hidden state (new information)
- $W_z, W_r, W_h \in \mathbb{R}^{h \times d}$ - Input-to-hidden weight matrices
- $U_z, U_r, U_h \in \mathbb{R}^{h \times h}$ - Hidden-to-hidden weight matrices
- b_z, b_r, b_h - Bias vectors
- $\sigma(\cdot)$ - Sigmoid activation function
- $\tanh(\cdot)$ - Hyperbolic tangent activation function
- $\odot$ - Element-wise multiplication (Hadamard product)
- d, h - Input and hidden dimensions respectively

B. Architecture and Key Components

Gating Mechanisms:

- **Reset Gate (r_t)**: Controls how much past information to forget when computing the candidate state. When $r_t \approx 0$, effectively ignores previous hidden state, useful when past patterns become irrelevant.
- **Update Gate (z_t)**: Determines balance between retaining previous information and incorporating new information. Acts as a weighted combination controller.

Information Flow Control: The final hidden state combines retained memory and new information:

- When $z_t \approx 1$: Favor new candidate state $\tilde{h}_t$ (emphasize recent information)
- When $z_t \approx 0$: Favor previous state h_{t-1} (maintain long-term memory)

Advantages over Standard RNNs:

- **Gradient Flow**: Gating mechanisms create pathways for gradients to flow through long sequences without vanishing
- **Selective Memory**: Can adaptively choose what information to remember or forget

- **Computational Efficiency**: Fewer parameters than LSTMs while maintaining similar performance

C. Training and Optimization

GRUs address gradient flow problems through their gating structure. The update gate creates direct connections that help preserve gradient flow:

$$\frac{\partial h_t}{\partial h_{t-1}} = (1 - z_t) + z_t \frac{\partial \tilde{h}_t}{\partial h_{t-1}} \tag{2.74}$$

Gradient Flow Variables:

- $\frac{\partial h_t}{\partial h_{t-1}}$ - Gradient flow between consecutive time steps
- $(1 - z_t)$ - Direct connection component (bypass pathway)
- $z_t \frac{\partial \tilde{h}_t}{\partial h_{t-1}}$ - Candidate state contribution

When $z_t \approx 0$, gradients flow directly through the $(1 - z_t)$ term, providing a shortcut connection that preserves gradient magnitudes across long sequences.

Training Benefits:

- **Reduced Vanishing Gradients**: Gating mechanisms provide gradient flow pathways
- **Faster Training**: Fewer parameters compared to LSTMs
- **Better Long-term Dependencies**: Can capture patterns across longer sequences than standard RNNs
- **Adaptive Learning**: Gates learn to control information flow based on data patterns

2.4.4. Long Short-Term Memory

Long Short-Term Memory (LSTM) networks are a sophisticated variant of RNNs designed to overcome the vanishing gradient problem through an elaborate gating mechanism that regulates information flow through a dedicated cell state [Hochreiter and Schmidhuber 1997].

A. Mathematical Formulation

An LSTM cell uses three gates and a cell state to control information flow. At each time step t, the LSTM computes:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{2.75}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2.76}$$

$$\tilde{C}_t = \tanh(W_C x_t + U_C h_{t-1} + b_C) \tag{2.77}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{2.78}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{2.79}$$

$$h_t = o_t \odot \tanh(C_t) \tag{2.80}$$

Variable Definitions:

- x_t - Input vector at time step t
- h_t - Hidden state at time step t
- C_t - Cell state at time step t (long-term memory)
- f_t - Forget gate vector (controls information removal)
- i_t - Input gate vector (controls information addition)
- o_t - Output gate vector (controls information exposure)
- $\tilde{C}_t$ - Candidate cell state (new information to potentially store)
- W_f, W_i, W_C, W_o - Input-to-hidden weight matrices
- U_f, U_i, U_C, U_o - Hidden-to-hidden weight matrices
- b_f, b_i, b_C, b_o - Bias vectors
- $\sigma(\cdot)$ - Sigmoid activation function
- $\tanh(\cdot)$ - Hyperbolic tangent activation function
- $\odot$ - Element-wise multiplication (Hadamard product)

B. Architecture and Key Components

Three-Gate System:

- **Forget Gate (f_t)**: Determines what information from previous cell state C_{t-1} should be discarded. Crucial for forgetting outdated patterns when new patterns emerge.
- **Input Gate (i_t)**: Controls what new information from candidate state $\tilde{C}_t$ should be stored. Allows selective incorporation of relevant information while ignoring noise.
- **Output Gate (o_t)**: Regulates what parts of cell state should be exposed as hidden state output. Provides additional control layer for prediction.

Cell State as Long-Term Memory: The cell state C_t serves as the network's long-term memory, flowing through the network with minimal transformations. It combines:

- **Retained Memory**: $f_t \odot C_{t-1}$ (filtered previous cell state)
- **New Information**: $i_t \odot \tilde{C}_t$ (filtered candidate state)

Advantages over Standard RNNs:

- **Long-term Dependencies**: Can maintain information across very long sequences
- **Selective Memory**: Three gates provide fine-grained control over information flow
- **Gradient Stability**: Cell state provides direct gradient pathways
- **Robust Learning**: Less susceptible to vanishing gradient problems

C. Training and Optimization

LSTMs address gradient flow problems through the cell state, which provides a direct pathway for gradient propagation:

$$\frac{\partial C_t}{\partial C_{t-1}} = f_t \tag{2.81}$$

Gradient Flow Benefits:

- When $f_t \approx 1$, gradients flow directly through the cell state without diminishing
- Enables learning of dependencies spanning long sequences

- Provides stable training compared to standard RNNs
- Allows adaptive control over what information to retain or discard

Training Characteristics:

- **Stability**: Less prone to vanishing gradients than standard RNNs
- **Complexity**: More parameters require careful initialization and regularization
- **Flexibility**: Three gates adapt to different sequence patterns
- **Performance**: Excellent for tasks requiring long-term memory

2.5. Transfer Learning

Transfer learning represents a paradigm shift in machine learning, enabling models to leverage knowledge gained from solving one problem to improve performance on a different but related problem. This section explores the theoretical foundations, historical development, and practical implementations of transfer learning techniques [Pan and Q. Yang 2010; Ruder 2019; Tan et al. 2018; Weiss et al. 2016; Zhuang et al. 2020].

2.5.1. Introduction to Transfer Learning

A. Definition and Core Concepts

Transfer learning is formally defined as improving the performance of target learners on target domains by transferring the knowledge contained in different but related source domains and tasks [Pan and Q. Yang 2010; Ruder 2019]. In the context of deep learning, this typically involves reusing all or parts of a model trained on a source task as the starting point for a model on a target task (Figure 2.4).

Let us formalize this definition:

Definition 1 (Domain). *A domain $\mathcal{D}$ consists of a feature space $\mathcal{X}$ and a marginal probability distribution $P(X)$ where $X = \{x_1, x_2, \ldots, x_n\} \in \mathcal{X}$. That is, $\mathcal{D} = \{\mathcal{X}, P(X)\}$.*

Definition 2 (Task). *Given a domain $\mathcal{D}$, a task $\mathcal{T}$ consists of a label space $\mathcal{Y}$ and a conditional probability distribution $P(Y|X)$ that is typically learned from training data consisting of pairs $\{x_i, y_i\}$, where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$. Thus, $\mathcal{T} = \{\mathcal{Y}, P(Y|X)\}$.*

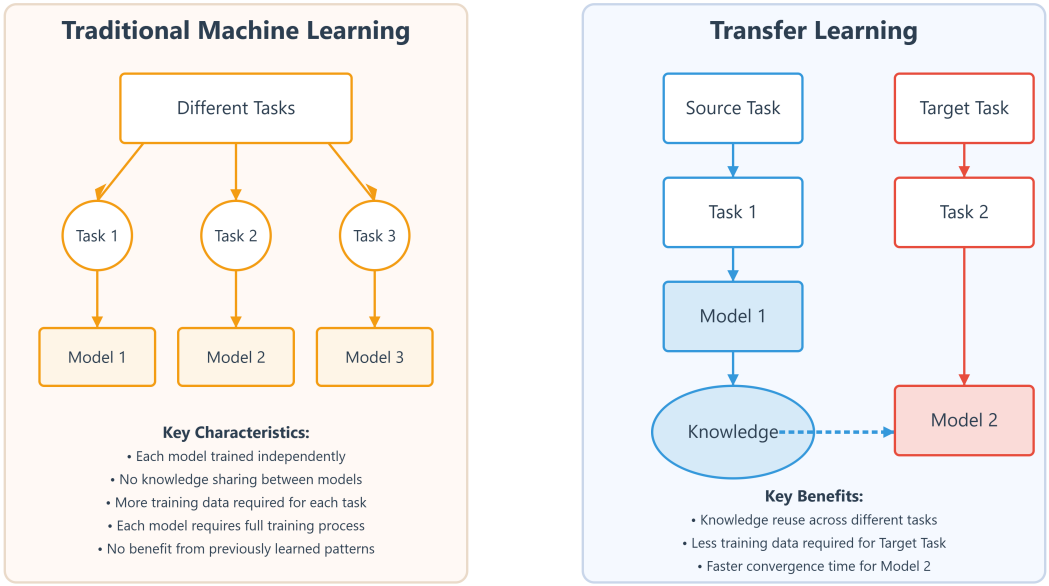


Figure 2.4.: Traditional ML versus transfer learning: Independent models versus knowledge transfer.

Definition 3 (Transfer Learning). *Given a source domain $\mathcal{D}_S$ with a corresponding task $\mathcal{T}_S$ and a target domain $\mathcal{D}_T$ with a corresponding task $\mathcal{T}_T$, transfer learning aims to improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$.*

Transfer learning can be categorized based on the relationship between source and target domains and tasks:

1. **Inductive Transfer Learning:** $\mathcal{T}_S \neq \mathcal{T}_T$, regardless of whether $\mathcal{D}_S = \mathcal{D}_T$. Knowledge is transferred to improve performance on a different task.

2. **Transductive Transfer Learning:** $\mathcal{D}_S \neq \mathcal{D}_T$ but $\mathcal{T}_S = \mathcal{T}_T$. The tasks are the same, but the domains differ (e.g., different feature distributions).

3. **Unsupervised Transfer Learning:** Both domains and tasks differ, and no labeled data is available in the target domain.

In deep learning implementations, transfer learning commonly manifests as:

- **Feature Extraction:** Using the pre-trained model as a fixed feature extractor, where higher layers of the neural network are removed and the rest is treated as a feature extractor for the new task.
- **Fine-tuning:** Initializing a model with pre-trained weights and then updating all or some of these weights during training on the new task.
- **Domain Adaptation:** Adjusting model components to reduce the gap between source and target domains, often using additional adaptation layers or specialized loss functions.

B. Motivation: Why Transfer Learning?

Transfer learning has emerged as a critical approach in machine learning for several compelling reasons:

Data Efficiency One of the principal motivations for transfer learning is addressing the data-hungry nature of deep learning models:

- **Limited Labeled Data:** In many real-world applications, obtaining large amounts of labeled data is expensive, time-consuming, or sometimes impossible. Transfer learning allows models to perform well even with limited target domain data.
- **Statistical Evidence:** Empirical studies consistently show that transfer learning can reduce the required training data by orders of magnitude. For instance, a model fine-tuned with transfer learning might achieve with 100 examples what would otherwise require 10,000 examples [Zhuang et al. 2020].
- **Mathematical Advantage:** From a theoretical perspective, transfer learning provides a favorable initialization in the weight space, placing the model closer to an optimal solution for the target task and requiring fewer examples to converge.

Computational Efficiency Beyond data considerations, transfer learning offers significant computational benefits:

- **Reduced Training Time:** Starting from pre-trained weights typically results in faster convergence compared to random initialization, often reducing training time by 50-90%.
- **Lower Resource Requirements:** Many organizations lack the computational infrastructure required to train large models from scratch. Transfer learning democratizes access to state-of-the-art models.
- **Environmental Impact:** Training large neural networks from scratch requires substantial energy and produces significant carbon emissions. Transfer learning reduces this environmental footprint.

Performance Improvements Perhaps most importantly, transfer learning often leads to superior model performance:

- **Higher Accuracy:** Models initialized with transferred knowledge typically achieve higher accuracy than those trained from scratch, especially in data-constrained scenarios.
- **Better Generalization:** Pre-trained models have often learned robust, generalizable representations from diverse datasets, improving performance on new tasks.
- **Regularization Effect:** The transferred knowledge can serve as an implicit regularizer, reducing overfitting on the target task.

2.5.2. Transfer Learning Approaches in Deep Learning

Deep learning has significantly amplified the effectiveness of transfer learning through the development of specialized architectures and techniques. This section provides an overview of the principal approaches to transfer learning in deep neural networks, highlighting the core mechanisms, strengths, and applications of each method.

A. Fine-tuning Pre-trained Models

Fine-tuning represents one of the most straightforward yet powerful transfer learning approaches, where a model pre-trained on a source task has some or all of its parameters updated to adapt to a target task.

The process typically involves:

1. Initializing a model with weights learned from a source domain (often using models pre-trained on large datasets like ImageNet for vision or large text corpora for NLP)
2. Replacing the task-specific output layers to match the target task requirements
3. Updating model parameters through training on the target dataset, often with a smaller learning rate than would be used for training from scratch

The effectiveness of fine-tuning depends significantly on the similarity between source and target domains and the layer-wise transferability of features. As demonstrated by Yosinski et al. [Yosinski et al. 2014], lower layers in deep networks tend to capture general features (e.g., edges, textures in vision) that transfer well across domains, while higher layers encode more task-specific information.

Different fine-tuning strategies can be employed based on dataset size and task similarity:

- **Full fine-tuning**: Update all model parameters
- **Layer-wise fine-tuning**: Selectively update certain layers, often freezing early layers and fine-tuning later ones
- **Progressive fine-tuning**: Gradually unfreeze layers during training, starting from the output layers

The learning rate schedule plays a crucial role in fine-tuning, with lower learning rates typically used to preserve pre-trained knowledge while allowing for adaptation. The discriminative fine-tuning approach (using different learning rates for different layers) has proven effective for many applications.

B. Feature Extraction

Feature extraction treats pre-trained models as fixed feature extractors, using their intermediate representations as input features for a new model on the target task. This approach is particularly useful when the target dataset is small or computational resources are limited.

The process involves:

1. Running the target dataset through a pre-trained model

2. Extracting activations from intermediate layers (typically after non-linearities)
3. Using these representations as features for a new model (often a simpler classifier like logistic regression or SVM)

The choice of which layer to extract features from significantly impacts performance. Early layers capture low-level features (e.g., edges, corners), while deeper layers represent more abstract concepts (e.g., object parts, semantic categories). For closely related tasks, later layers often provide more relevant features, while for more distant tasks, intermediate layers may be optimal.

Feature extraction offers several advantages:

- Computational efficiency, as the pre-trained model needs to be run only once
- Minimal risk of catastrophic forgetting, as the pre-trained model remains unchanged
- Simplicity of implementation and reduced risk of overfitting on small datasets

In practice, feature extraction is often used as a stepping stone toward fine-tuning, providing a useful baseline and insights into which transferred representations might be most valuable.

C. Multi-task Learning

Multi-task learning (MTL) extends transfer learning by simultaneously optimizing performance across multiple related tasks. While traditional transfer learning is typically sequential (train on source, transfer to target), MTL trains on all tasks concurrently, allowing for bidirectional knowledge transfer.

The key components of MTL include:

- **Shared representations**: Lower layers learn task-agnostic features beneficial to multiple tasks
- **Task-specific components**: Specialized layers or heads for each individual task
- **Joint optimization**: A combined objective function that balances performance across tasks

Mathematically, the MTL objective function can be represented as:

$$\mathcal{L}_{MTL} = \sum_{i=1}^{T} \lambda_i \mathcal{L}_i(\theta_s, \theta_i) \tag{2.82}$$

where $\mathcal{L}_i$ is the loss for task i, λ_i is the task weight, θ_s represents shared parameters, and θ_i represents task-specific parameters.

The effectiveness of MTL depends on selecting complementary tasks that induce a beneficial inductive bias in the shared representations. Hard parameter sharing (using identical lower layers across tasks) is the most common approach, though soft parameter sharing (allowing task-specific layers with regularization to encourage similarity) offers greater flexibility.

MTL has proven particularly effective in:

- Natural language processing (e.g., joint parsing, named entity recognition, and sentiment analysis)
- Computer vision (e.g., simultaneous object detection, segmentation, and depth estimation)
- Autonomous driving (e.g., joint perception, prediction, and planning)

D. Domain-Adversarial Neural Networks

Domain-adversarial neural networks (DANNs) address domain adaptation challenges by learning domain-invariant representations that perform well on both source and target domains. This approach is particularly valuable when labeled data is abundant in the source domain but scarce or absent in the target domain.

The DANN architecture, introduced by Ganin et al. [Ganin et al. 2016], consists of:

- A feature extractor network G_f that maps inputs to a feature space
- A label predictor G_y that classifies based on these features
- A domain classifier G_d that attempts to identify whether features came from the source or target domain

The key innovation in DANNs is the gradient reversal layer placed between the feature extractor and domain classifier, which encourages the learning of features that are discriminative for the main classification task but indistinguishable between domains.

Domain-adversarial training has been successfully applied to:

- Image classification across visual domains (e.g., photos to sketches)
- Sentiment analysis across different text domains (e.g., book reviews to electronics reviews)
- Speech recognition across different accents or recording conditions

Recent variants have extended this approach with multiple domain discriminators, conditional adversarial networks, and cycle-consistent transformations.

2.5.3. Transfer Learning for Time Series and Sequential Data

Time series and sequential data present unique challenges for transfer learning due to their temporal dependencies, varying lengths, and domain-specific characteristics. This section explores specialized approaches that adapt transfer learning techniques to these data types, enabling knowledge transfer across different sequential domains.

A. Recurrent Neural Network Approaches

Recurrent Neural Networks (RNNs) and their variants have traditionally been the architecture of choice for sequential data. Transfer learning with RNNs involves several specialized techniques to address the temporal nature of the data.

Weight Transfer in RNN Architectures The foundational approach to RNN transfer learning involves pre-training an RNN (typically LSTM or GRU) on a large source domain dataset and transferring some or all of the weights to a target domain model. The effectiveness of this transfer depends on:

- **Layer-wise Transferability**: Similar to CNNs, lower layers in deep RNNs tend to capture more general temporal patterns, while higher layers encode task-specific information. Empirical studies by Raghu et al. [Raghu et al. 2019] demonstrated that transferring the first few layers often provides the most benefit.
- **State Initialization**: Beyond weight transfer, RNNs can benefit from hidden state initialization, where the final hidden state from the source task becomes the initial state for the target task, providing a contextual "warm start".

- **Gating Adaptation**: In gated RNN variants (LSTM/GRU), selectively adapting certain gates (e.g., input gate in LSTM) while freezing others can preserve general temporal dynamics while allowing adaptation to target-specific patterns.

Sequential Multi-task Learning Extensions of multi-task learning for sequential data have shown promise for transfer learning:

- **Hierarchical RNNs**: These architectures use different recurrent layers to capture task-specific and shared temporal dynamics, with lower layers encoding domain-invariant features [H. Tang et al. 2020].
- **Progressive Neural Networks**: Originally proposed by Rusu et al. [Rusu et al. 2016], these have been adapted for sequential data, allowing lateral connections between task-specific columns of recurrent units while preventing catastrophic forgetting.
- **Temporal Attention Transfer**: This approach transfers attention weights from a source RNN to guide the learning of the target RNN, effectively transferring knowledge about which time steps are most important for prediction.

LSTM-based Domain Adaptation Domain adaptation techniques specifically designed for recurrent architectures include:

- **Adversarial Domain Adaptation**: Similar to DANNs but with temporal considerations, these approaches train RNNs to learn domain-invariant sequential representations through adversarial training with a temporal domain discriminator [Purushotham et al. 2017].
- **Discrepancy-based Adaptation**: Methods like maximum mean discrepancy (MMD) have been extended to account for temporal dependencies in sequential data [Wilson and Cook 2020].

RNN transfer approaches have shown particular success in applications such as speech recognition, sensor data analysis, and financial time series, especially when the source and target domains share similar temporal dynamics despite different marginal distributions.

2.5.4. Conclusion and Future Directions

Transfer learning has emerged as a powerful paradigm that enables models to reuse knowledge across tasks and domains, significantly improving performance, particularly in data-scarce scenarios. By leveraging strategies such as fine-tuning, feature extraction, domain adaptation, multi-task learning, and knowledge distillation, transfer learning has led to substantial advancements in computer vision, natural language processing, and sequential data modeling.

This thesis contributes to the growing body of work by extending and operationalizing transfer learning specifically within the context of time series forecasting and environmental modeling. It demonstrates how knowledge acquired from high-resolution pollutant data and pre-trained models can be effectively transferred across different pollutants, locations, and temporal resolutions. In doing so, the research not only improves predictive performance but also enhances model efficiency and generalizability in complex environmental monitoring tasks.

Looking ahead, several challenges and opportunities remain. Future research should focus on mitigating negative transfer, improving the interpretability of transferred representations, and expanding transferability across heterogeneous domains and modalities. Furthermore, integrating transfer learning with emerging paradigms such as continual learning, causal inference, and meta-learning holds great promise for building more robust, adaptive, and context-aware AI systems capable of supporting long-term environmental intelligence and sustainable decision-making.

2.6. Data Preparation and Model Development

2.6.1. Data Preprocessing

Data preprocessing is a fundamental step in any machine learning and deep learning pipeline. Raw data is often noisy, inconsistent, and varies in scale across features, which can degrade the performance of most models. Proper preprocessing improves learning efficiency, ensures fair feature contribution, and enhances overall model accuracy and generalization. In this study, the following preprocessing techniques were employed:

A. Correlation Analysis

Correlation analysis is a fundamental statistical technique used to quantify the strength and direction of linear relationships between pairs of variables. This is particularly important in predictive modeling, where detecting and addressing multicollinearity—strong correlations between independent variables—can significantly enhance model stability, interpretability, and generalizability. Multicollinearity can inflate the variance of coefficient estimates, reduce model performance, and obscure the effect of individual predictors.

One widely used measure is the *Pearson correlation coefficient* r, which captures the degree of linear association between two continuous variables X and Y. It is defined as:

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{2.83}$$

where:

- X_i and Y_i are the individual values of variables X and Y for observation i,
- $\bar{X}$ and $\bar{Y}$ are the sample means of X and Y, respectively,
- n is the total number of paired observations.

The coefficient r takes values in the range $[-1, +1]$:

- $r = +1$: perfect positive linear correlation,
- $r = -1$: perfect negative linear correlation,
- $r = 0$: no linear correlation.

In practical applications, pairs of features with a high absolute correlation (e.g., $|r| > 0.9$) are considered highly collinear. Such redundancy can be addressed by removing one of the correlated variables or applying dimensionality reduction techniques such as Principal Component Analysis (PCA). Conducting correlation analysis during the preprocessing stage ensures more robust, interpretable, and efficient modeling.

B. Feature Scaling

Feature scaling is a critical preprocessing step that transforms input variables to a common scale, ensuring that no single feature disproportionately influences the learning process due

to its magnitude. This step is especially important for machine learning algorithms that are sensitive to the scale of input features, including gradient descent-based methods, support vector machines (SVMs), k-nearest neighbors (KNN), and neural networks.

Without proper scaling, features with larger numerical ranges can dominate the objective function, leading to biased models and suboptimal convergence behavior. Feature scaling improves numerical stability, speeds up the training process, and ensures that all features contribute fairly to model learning.

Two widely used scaling methods are described below:

- **Normalization (Min-Max Scaling):** This method rescales the feature values to a specified range, typically $[0, 1]$. It is defined as:

$$X^{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{2.84}$$

  where:

  - X is the original feature value,
  - X_{min} and X_{max} are the minimum and maximum values of the feature in the dataset,
  - X^{norm} is the normalized feature value.

- **Standardization (Z-score Scaling):** This approach transforms the feature to have zero mean and unit variance, preserving the shape of the distribution while shifting and scaling it. It is defined as:

$$X^{\text{std}} = \frac{X - \mu}{\sigma} \tag{2.85}$$

  where:

  - X is the original feature value,
  - μ is the mean of the feature,
  - σ is the standard deviation of the feature,
  - X^{std} is the standardized feature value.

The choice between normalization and standardization depends on the specific algorithm and the distribution of the data. For instance, normalization is often preferred when the data are bounded or uniformly distributed, while standardization is more suitable for features

that follow a Gaussian distribution. Regardless of the method, applying feature scaling is essential for ensuring effective and consistent model training.

2.6.2. Hyper-parameters Optimization

Hyper-parameters are external configuration settings that govern the behavior of machine learning models, such as the number of layers in a neural network, the depth of a decision tree, or the learning rate in gradient-based algorithms. These parameters are not learned during training and must be carefully selected to ensure optimal model performance [J. Bergstra and Bengio 2012; Feurer and Hutter 2019].

Optimizing hyper-parameters is essential for improving generalization, reducing overfitting, and enhancing predictive accuracy. Common optimization strategies include:

- **Grid Search:** Exhaustive evaluation across a manually specified set of values.
- **Random Search:** Randomly samples combinations from the hyper-parameter space, often more efficient than grid search [J. Bergstra and Bengio 2012].
- **Bayesian Optimization:** Builds a surrogate model (e.g., Gaussian Processes) to model the objective function and select promising hyper-parameter sets [Snoek et al. 2012].
- **Tree-structured Parzen Estimators (TPE):** A probabilistic model used in frameworks like Hyperopt [J. Bergstra, Yamins, et al. 2013] and Optuna [Akiba et al. 2019] for efficient search [J. S. Bergstra et al. 2011].
- **Evolutionary Algorithms:** Population-based optimization inspired by natural selection.

In this thesis, hyper-parameter tuning was conducted using GridSearch and Optuna with TPE, enabling efficient exploration of the search space and robust model configuration.

2.6.3. Evaluation Metrics

Evaluation metrics are essential tools to assess model performance and guide the selection of the best-performing model. Depending on the task—classification or regression—different sets of metrics are used to provide a comprehensive performance evaluation.

A. Classification Metrics

For binary or multi-class classification problems, the following metrics are widely used:

- **Accuracy:** The proportion of correctly predicted samples (both positive and negative) among all samples.
$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$
where TP, TN, FP, and FN refer to true positives, true negatives, false positives, and false negatives, respectively.

- **Precision:** The proportion of predicted positive instances that are actually positive.
$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall (Sensitivity):** The proportion of actual positive instances that are correctly identified.
$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-score:** The harmonic mean of precision and recall, offering a balanced measure when there is class imbalance.
$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **AUC-ROC (Area Under the Receiver Operating Characteristic Curve):** This metric evaluates the model's ability to distinguish between classes across all classification thresholds. The ROC curve plots the *True Positive Rate (TPR)* against the *False Positive Rate (FPR)* at various threshold settings:
$$\text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN}$$
The AUC (Area Under the Curve) summarizes the entire ROC curve into a single value ranging from 0 to 1:
  - AUC = 1 indicates a perfect classifier.
  - AUC = 0.5 indicates a model with no discriminative ability (equivalent to random guessing).

- A higher AUC indicates better model performance in distinguishing between the positive and negative classes.

AUC-ROC is particularly useful when dealing with imbalanced datasets, as it considers the trade-off between sensitivity and specificity at different thresholds rather than relying on a single cutoff point.

B. Regression Metrics

For regression tasks such as pollutant concentration forecasting, the following metrics are commonly employed to evaluate model performance:

- **Mean Absolute Error (MAE):** Measures the average absolute difference between predicted and actual values.

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

  where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the number of samples.

- **Mean Squared Error (MSE):** Measures the average of the squared differences between predicted and actual values, penalizing larger errors more heavily.

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

- **Root Mean Squared Error (RMSE):** The square root of MSE, often preferred for its interpretability in the same units as the target variable.

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

- **Mean Absolute Percentage Error (MAPE):** Represents prediction error as a percentage of the actual value.

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

- **R-squared (R^2):** Indicates the proportion of the variance in the target variable that is explained by the model.

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

Relative Improvement (RI):

To comprehensively assess the performance benefits of the proposed *transfer learning* framework and the use of pre-trained models with fine-tuning, the **Relative Improvement (RI)** metric is computed for each regression metric. It quantifies the percentage change in performance between a baseline model (from scratch training) and the proposed approach:

$$RI = \frac{M_{\text{baseline}} - M_{\text{proposed}}}{M_{\text{baseline}}} \times 100\% \tag{2.86}$$

where:

- M_{baseline} is the metric value obtained using a baseline model (e.g., training from scratch or a different model),
- M_{proposed} is the value obtained using the proposed method (e.g., deep ensemble (*LagEnsembleForecasting*) or transfer learning).

Important distinction: For error-based metrics (MAE, MSE, RMSE, MAPE), a **positive RI** indicates an improvement (i.e., a reduction in error). However, for R^2, which increases with better performance, RI is defined differently to account for its upper bound of 1:

$$RI_{R^2} = \frac{R^2_{\text{proposed}} - R^2_{\text{baseline}}}{R^2_{\text{baseline}}} \times 100\% \tag{2.87}$$

These metrics collectively support a rigorous and interpretable evaluation of model effectiveness across both classification and regression tasks in this study.

C. Cross-Validation Techniques

To ensure model robustness and generalizability, we employed the following cross-validation strategies:

- **K-Fold Cross-Validation:** The dataset is randomly divided into k equal-sized folds. The model is trained on $k-1$ folds and validated on the remaining fold. This process is repeated k times, with each fold used once as the validation set. The performance metrics are averaged across the k iterations to obtain a more stable estimate of model performance. This method is particularly useful in small datasets to avoid overfitting and to assess variance across different data splits.
- **Time Series Split:** Unlike K-Fold CV, which randomly shuffles the data, time series cross-validation maintains the temporal order of observations. It incrementally expands the training set and validates on the next chronological chunk. This is crucial for time-dependent data (e.g., pollutant concentrations) to prevent information leakage and ensure realistic forecasting scenarios.

By incorporating these cross-validation techniques, we aimed to mitigate overfitting and obtain a reliable estimation of the model's performance in both static and sequential data contexts.

2.7. Air Quality Datasets

This thesis employs two comprehensive and common air quality datasets from Beijing, China: the Beijing Multi-Site Air Quality dataset [S. Chen 2017] and the Beijing $PM_{2.5}$ dataset [S. Chen 2015]. Both datasets provide valuable information for understanding air pollution patterns and developing predictive models for air quality analysis.

2.7.1. Beijing Multi-Site Air Quality Dataset

The Beijing Multi-Site Air Quality dataset [S. Chen 2017] is a comprehensive and high-resolution record of air pollutant concentrations and meteorological parameters collected across multiple monitoring stations in Beijing. This dataset spans from January 1st, 2013 to February 28th, 2017 and contains 420,768 hourly measurements, making it an essential resource for studying the spatial-temporal dynamics of urban air pollution across different regions of the city.

Unlike single-station datasets, this multi-site dataset includes measurements from over 35 official monitoring stations distributed throughout Beijing's urban and suburban areas.

It captures a diverse range of air pollutants, including $PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, and O_3, alongside essential meteorological variables such as temperature, pressure, dew point, wind direction and speed, and rainfall. The dataset's station-wise structure enables detailed intra-city comparisons and region-specific forecasting, essential for pollution control and public health planning (Table 2.1).

Characteristic	Description
Time period	January 1, 2013 – February 28, 2017
Time resolution	Hourly
Monitoring coverage	35+ monitoring stations across Beijing
Total instances	420,768 (hourly readings)
Missing values	Present in pollutant fields (e.g., 2.1% in $PM_{2.5}$)
Data source	Beijing Municipal Environmental Monitoring Center

Table 2.1.: Overview of the Beijing Multi-Site Air Quality Dataset.

Table 2.2 summarizes the dataset's attributes, covering both pollution and meteorological indicators. Each entry is tagged with a station identifier, allowing for spatial modeling and comparative studies across locations.

Table 2.3 reports the statistical summary of the key air pollutants and meteorological variables. $PM_{2.5}$ and PM_{10} show wide ranges with values reaching up to 999 and 994 $\mu g/m^3$ respectively, indicative of severe pollution episodes. Variability in temperature, wind speed, and rainfall highlights the climatic influence on pollution dispersion and deposition processes.

The inclusion of station-level and meteorological information allows for advanced spatial modeling and region-specific pollution analysis. Moreover, the hourly temporal resolution enables detection of diurnal cycles and rapid pollution shifts, making the dataset highly suitable for deep learning-based prediction tasks, spatiotemporal interpolation, and air quality policy evaluation in megacities.

2.7.2. Beijing PM2.5 Dataset

The Beijing $PM_{2.5}$ dataset represents a comprehensive record of particulate matter concentrations monitored at the US Embassy in Beijing over a five-year period from January 1st, 2010 to December 31st, 2014. Comprising 43,824 hourly observations, this dataset offers

Variable	Description	Non-Null Count	Data Type
No	Row identifier	420,768 non-null	int64
Year	Year of observation	420,768 non-null	int64
Month	Month of observation	420,768 non-null	int64
Day	Day of observation	420,768 non-null	int64
Hour	Hour of observation (0–23)	420,768 non-null	int64
$PM_{2.5}$	Fine particulate matter concentration ($\mu g/m^3$)	412,029 non-null	float64
PM_{10}	Coarse particulate matter concentration ($\mu g/m^3$)	414,319 non-null	float64
SO_2	Sulfur dioxide concentration ($\mu g/m^3$)	411,747 non-null	float64
NO_2	Nitrogen dioxide concentration ($\mu g/m^3$)	408,652 non-null	float64
CO	Carbon monoxide concentration ($\mu g/m^3$)	400,067 non-null	float64
O_3	Ozone concentration ($\mu g/m^3$)	407,491 non-null	float64
TEMP	Ambient temperature (°C)	420,370 non-null	float64
PRES	Atmospheric pressure (hPa)	420,375 non-null	float64
DEWP	Dew point temperature (°C)	420,365 non-null	float64
RAIN	Hourly rainfall (mm)	420,378 non-null	float64
WD	Wind direction (categorical)	418,946 non-null	string
WSPM	Wind speed (m/s)	420,450 non-null	float64
Station	Monitoring station identifier	420,768 non-null	string

Table 2.2.: Attribute summary of the Beijing Multi-Site Air Quality Dataset.

Variable	Min	1st Qu.	Median	Mean	3rd Qu.	Max
$PM_{2.5}$	0.0	34.0	72.0	87.6	122.0	999.0
PM_{10}	1.0	53.0	101.0	118.2	162.0	994.0
SO_2	0.0	7.0	17.0	22.6	32.0	360.0
NO_2	0.0	24.0	45.0	49.6	70.0	315.0
CO	0.0	300.0	700.0	755.3	1100.0	5960.0
O_3	0.0	7.0	23.0	34.9	54.0	307.0
TEMP	-25.0	-1.1	11.0	10.2	22.0	42.0
PRES	980.0	1008.2	1016.0	1015.2	1023.0	1046.0
WSPM	0.0	1.4	2.6	2.9	4.0	21.6
RAIN	0.0	0.0	0.0	0.4	0.0	74.0

Table 2.3.: Statistical summary of the Beijing Multi-Site Air Quality Dataset.

exceptional temporal granularity for analyzing both short-term fluctuations and long-term trends in urban air quality. The high-resolution hourly measurements enable detailed investigation of diurnal patterns, meteorological influences, and seasonal variations in $PM_{2.5}$ concentrations within one of the world's most air pollution-challenged megacities (Table 2.4).

The dataset's extended temporal coverage (5 years) is particularly valuable for capturing the full spectrum of air quality conditions across multiple seasons and years, allowing for robust analysis of cyclic patterns and exceptional pollution events. The geographical significance of Beijing—a densely populated urban center with complex pollution sources including industrial emissions, vehicular traffic, and regional transport—makes this dataset an ideal case study for investigating urban air pollution dynamics in rapidly developing economic regions.

Table 2.5 presents the complete attribute listing of the dataset, including variable descriptions, data completeness, and data types. The dataset encompasses not only the target $PM_{2.5}$ concentrations but also critical meteorological parameters that potentially influence particulate matter levels, such as temperature, dew point, pressure, wind characteristics, and precipitation indicators. This comprehensive collection of meteorological covariates enables in-depth analysis of the complex relationships between atmospheric conditions and air pollution concentrations.

Table 2.6 provides the statistical summary of key variables in the dataset. Notable is the

Characteristic	Description
Time period	January 1, 2010 – December 31, 2014
Time resolution	Hourly
Monitoring location	US Embassy in Beijing (39.954N, 116.468E)
Total instances	43,824 (hourly readings across 5 years)
Missing values	Approximately 4.7% (primarily in $PM_{2.5}$ measurements)
Data source	U.S. Department of State Air Quality Monitoring Program

Table 2.4.: Overview of the Beijing $PM_{2.5}$ Dataset.

Variable	Description	Non-Null Count	Data Type
No	Row identifier	43,824 non-null	int64
Year	Year of observation	43,824 non-null	int64
Month	Month of observation	43,824 non-null	int64
Day	Day of observation	43,824 non-null	int64
Hour	Hour of observation (0-23)	43,824 non-null	int64
$PM_{2.5}$	Fine particulate matter concentration ($\mu g/m^3$)	41,757 non-null	float64
DEWP	Dew point temperature (°C)	43,824 non-null	int64
TEMP	Ambient air temperature (°C)	43,824 non-null	float64
PRES	Atmospheric pressure (hPa)	43,824 non-null	float64
CBWD	Combined wind direction (NE, NW, SE, SW)	43,824 non-null	categorical
Iws	Cumulated wind speed (m/s)	43,824 non-null	float64
Is	Cumulated hours of snow	43,824 non-null	int64
Ir	Cumulated hours of rain	43,824 non-null	int64

Table 2.5.: Attribute summary of the Beijing $PM_{2.5}$ Dataset.

extensive range of $PM_{2.5}$ concentrations, spanning from 0.0 to 994.0 $\mu g/m^3$, which captures both pristine air quality conditions and severe pollution episodes. The substantial difference between the median (72.0 $\mu g/m^3$) and mean (98.6 $\mu g/m^3$) $PM_{2.5}$ values indicates a right-skewed distribution, suggesting frequent occurrence of high pollution events. Temperature values range from -19.0°C to 42.0°C, reflecting Beijing's continental climate with cold winters and hot summers—a characteristic that significantly influences seasonal pollution patterns.

Variable	Min	1st Qu.	Median	Mean	3rd Qu.	Max
$PM_{2.5}$	0.0	23.0	72.0	98.6	140.0	994.0
DEWP	-40.0	-10.0	2.0	1.8	14.8	28.0
TEMP	-19.0	0.0	14.0	13.1	25.0	42.0
PRES	991.0	1008.0	1016.0	1015.3	1023.0	1038.0
IWS	0.0	0.9	1.8	2.0	2.9	10.3
Ir	0.0	0.0	0.0	0.1	0.0	25.4
Is	0.0	0.0	0.0	0.0	0.0	18.3

Table 2.6.: Statistical summary of Beijing $PM_{2.5}$ Dataset.

The combined wind direction (CBWD) variable categorizes wind patterns into four cardinal quadrants (NE, NW, SE, cv), allowing for analysis of how pollution transport correlates with regional airflow patterns. The precipitation indicators (cumulated hours of rain and snow) enable examination of wet deposition effects on particulate matter concentrations. This comprehensive dataset thus provides a solid foundation for investigating the complex interplay between meteorological conditions and urban air quality dynamics.

2.8. Conclusions

This chapter has established a comprehensive methodological foundation by reviewing key computational paradigms—granular computing, time series analysis, fundamental machine learning algorithms, deep learning, and transfer learning—that collectively address the complex challenges of air quality prediction and interpretable rule extraction. The integration of granular computing with modern machine learning techniques provides a unique framework that balances predictive accuracy with explainability, essential for environmental decision-making. The granular rule induction methodology enables extraction of interpretable decision

rules from complex datasets, while transfer learning offers efficient knowledge transfer across different pollutants, locations, and temporal resolutions—particularly valuable given the data scarcity challenges in environmental monitoring. The fundamental machine learning algorithms—Logistic Regression, Support Vector Machines, Decision Trees, Random Forest, and XGBoost—serve as the foundation for PM toxicity classification tasks, providing essential baseline comparisons and intuitive understanding of model performance characteristics across different algorithmic approaches. These fundamental methods enable systematic evaluation of which algorithmic paradigms are most suitable for distinguishing between toxic and non-toxic particulate matter concentrations under various environmental conditions. In contrast, deep learning architectures (ANNs, RNNs, GRUs, and LSTMs), are specifically employed for building advanced air quality time series forecasting models that can capture complex temporal dependencies and non-linear relationships in sequential pollutant data. These deep learning approaches also form the backbone for implementing neural transfer learning strategies, enabling sophisticated knowledge transfer mechanisms across different temporal resolutions, geographic locations, and pollutant types. The systematic treatment of data preprocessing, evaluation metrics, and cross-validation techniques establishes a rigorous experimental framework for reliable model assessment. The two Beijing air quality datasets provide ideal testbeds with rich temporal resolution and comprehensive meteorological covariates for evaluating the proposed approaches. This methodological foundation supports the development of more intelligent, adaptive, and transparent environmental monitoring systems that can effectively bridge the gap between computational sophistication and practical applicability in real-world air quality management.

3. Advancing Particulate Matter Risk Assessment: Machine Learning-Driven Toxicity Threshold Prediction

While accurate air quality forecasting and classification systems provide essential tools for environmental monitoring, the assessment of particulate matter (PM) health risks requires a deeper understanding of the specific toxicity thresholds that determine when PM concentrations transition from relatively harmless to potentially dangerous for human health. Traditional approaches to PM toxicity assessment rely heavily on labor-intensive laboratory experiments and epidemiological studies that, despite their scientific rigor, are time-consuming, costly, and limited in scalability—often requiring extensive experimental protocols and lengthy data collection periods to establish toxicity thresholds for different PM compositions and exposure scenarios. This creates a critical gap in our ability to rapidly assess the health risks associated with emerging PM sources, novel atmospheric conditions, or newly monitored environments where extensive historical toxicological data may not be available. This chapter addresses this fundamental challenge by introducing a comprehensive machine learning framework for predicting PM toxicity thresholds based on physico-chemical characteristics and exposure conditions. By leveraging a meticulously curated dataset compiled from over 40 peer-reviewed studies and employing five complementary ML algorithms—including ensemble methods and gradient boosting techniques—we demonstrate how data-driven approaches can effectively distinguish between toxic and non-toxic PM concentrations while providing unprecedented insights into the specific compositional and exposure factors that drive toxicity outcomes. Through the integration of explainable artificial intelligence (XAI) techniques, particularly SHAP (SHapley Additive exPlanations) analysis, this work not only achieves robust predictive performance but also delivers interpretable, scientifically credible results that can directly inform regulatory decision-making and public

health policy development.

3.1. Background and Motivation

Air pollution remains a major environmental and public health challenge, contributing to millions of premature deaths worldwide. Among airborne pollutants, particulate matter (PM) poses one of the most severe health risks due to its ability to penetrate deep into the respiratory system and enter the bloodstream. Fine particles, particularly $PM_{2.5}$ (particles with a diameter $\leq 2.5\ \mu m$), have been strongly linked to respiratory diseases, cardiovascular disorders, and other adverse health effects [K.-H. Kim et al. 2015; Rorat et al. 2020]. The toxicity of PM, however, is not solely determined by particle size but also by its physico-chemical composition and exposure conditions, making its risk assessment a complex task [Kelly and Fussell 2012].

Particulate matter (PM) refers to a mixture of tiny solid and liquid particles suspended in the air, varying in size, composition, and origin. It can be composed of various substances, including carbon, organic compounds, and metals. PM can be broadly categorized based on particle size. PM_{10} refers to particles with a diameter of 10 micrometers or less, while $PM_{2.5}$ refers to particles with a diameter of 2.5 micrometers or less. Additionally, ultra-fine particles, with diameters less than 0.1 micrometers ($PM_{0.1}$), represent an important component of PM due to their ability to penetrate biological barriers and induce systemic effects (Figure 3.1).

Exposure to PM can have harmful effects on human health, particularly on the respiratory and cardiovascular systems, but, $PM_{2.5}$ are much more dangerous and deadly than PM_{10}, due to its small size, which can easily get deep into the lungs and pass into the bloodstream, therefore, causing serious health problems (Figure 3.2).

The assessment of PM toxicity is a critical concern in environmental and public health research. Traditional approaches rely on labor-intensive laboratory experiments and epidemiological studies, which, despite their accuracy, are often time-consuming, costly, and limited in scalability. These conventional methods involve extensive biological testing and long-term health studies, making it difficult to assess large datasets efficiently. Consequently, there is a pressing need for data-driven methodologies that can enhance the understanding of PM toxicity and provide efficient, scalable, and interpretable risk assessment tools.

In response to these challenges, machine learning (ML) techniques have emerged as promising tools for toxicity prediction. By leveraging large datasets containing diverse

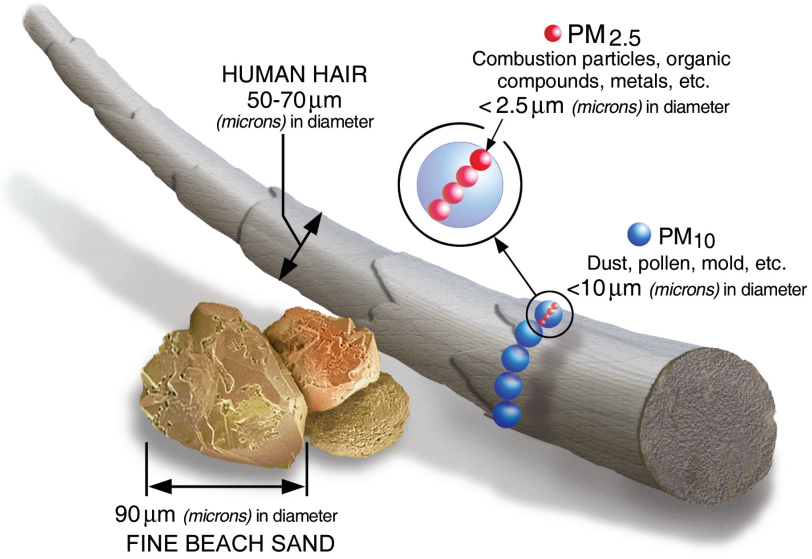


Figure 3.1.: Size comparisons for PM particles. Source: [United States Environmental Protection Agency 2023]

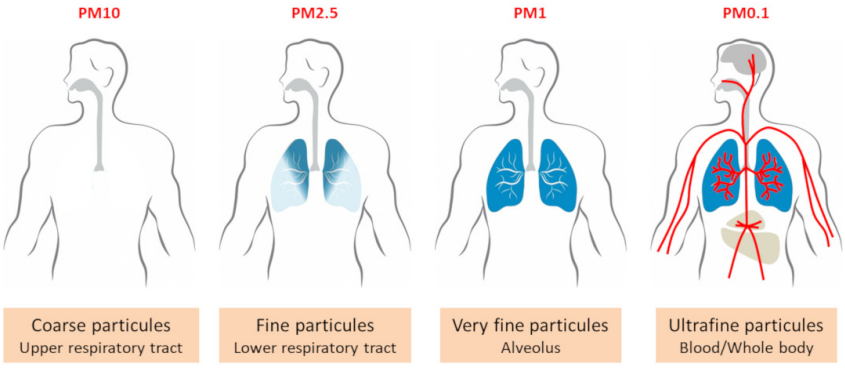


Figure 3.2.: Lung penetration of PM particles. Source : [Vincent 2019]

PM characteristics and exposure information, ML models can identify complex, non-linear relationships that traditional statistical methods struggle to capture. Furthermore, the integration of explainable artificial intelligence (XAI) techniques, such as SHAP (SHapley Additive exPlanations) values, allow for greater transparency and interpretability, providing valuable insights into the physico-chemical and exposure attributes that contribute most to PM toxicity.

This chapter presents a robust ML-driven framework for classifying PM toxicity thresholds based on key physico-chemical and exposure characteristics. The study employs five classification models—Logistic Regression, Support Vector Classifier, Decision Tree, Random Forest, and XGBoost—to distinguish between toxic and non-toxic PM concentrations. By applying SHAP values, this research also offers a deeper understanding of how different PM attributes influence toxicity, ensuring the models' predictions are interpretable and actionable. Figure 3.3 provides a visual representation of the binary classification task used in this study, where varying concentrations of particulate matter (PM) are labeled as either toxic or non-toxic based on their biological effects. The x-axis shows the tested concentrations of PMs, while the y-axis represents the observed biological effect (e.g., oxidative stress or inflammation), modeled as a binary outcome: 0 (no effect) or 1 (effect). A classification threshold is shown, illustrating how the model separates toxic from non-toxic concentrations. This figure helps conceptualize how the toxicity threshold is used to train and evaluate machine learning classifiers. It is important to note that these effects are not always linear; for example, among tested concentrations of 10, 20, 30, 40, and 50 $\mu g/mL$, a toxic effect may only be observed at 20 $\mu g/mL$, highlighting the potential for non-linear or dose-specific biological responses.

The integration of machine learning into PM toxicity assessment not only enhances predictive accuracy but also introduces a new paradigm in environmental health risk evaluation. By leveraging data-driven insights, this approach has the potential to streamline toxicity threshold determination, reduce reliance on costly laboratory testing, and improve regulatory decision-making. To this end, this study makes several key contributions:

1. A novel ML-based approach for predicting PM toxicity thresholds, an area that has not been extensively explored in prior research.

2. A comprehensive dataset, aggregating diverse physico-chemical characteristics of PM and corresponding toxicity assessments.

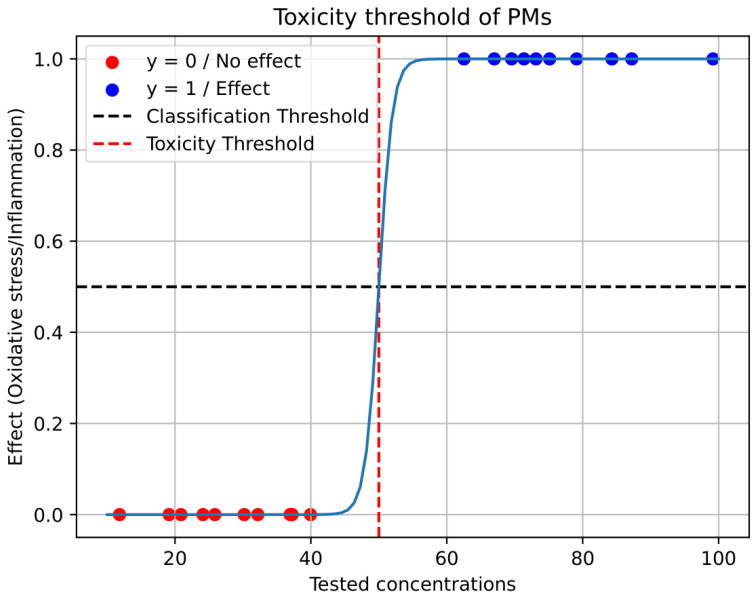


Figure 3.3.: Visual representation of the toxicity threshold-based classification task.

3. A comparative evaluation of five ML algorithms—Logistic Regression, Support Vector Classifier, Decision Tree, Random Forest, and XGBoost—for toxicity prediction.

4. An interpretable predictive model, employing SHAP values to quantify feature contributions and enhance trust in model predictions.

5. Actionable insights for regulatory policies, guiding targeted interventions to mitigate PM-associated health risks.

The findings of this research not only advance our understanding of PM toxicity but also provide a practical tool for regulators and policymakers. By offering a scalable and interpretable ML-based toxicity assessment framework, this work contributes to establishing more informed exposure limits and developing predictive strategies to reduce environmental degradation and health impacts. Identifying critical contributors to PM toxicity lays the foundation for targeted interventions and risk mitigation strategies in environmental health, ultimately helping to improve air quality and public well-being.

3.2. Related Works on PM Toxicity

Understanding and predicting the toxicity threshold of particulate matter is a key focus in environmental health research, as it has significant implications for public health and regulatory policies. Identifying the point at which particulate matter shifts from being less harmful to potentially harmful helps communities, industries, and policymakers take early action to protect human health. Therefore, it is important to understand how different physico-chemical properties of particles, along with exposure conditions, influence variations in the toxicity threshold. This knowledge supports the establishment of scientifically based exposure limits, guiding the creation of effective regulations for controlling emissions and pollution. Additionally, it allows for a more targeted approach to risk assessment and resource allocation, ensuring efforts are directed towards reducing the most critical air pollution-related health risks. By integrating scientific research with policy development, predicting toxicity thresholds contributes to cleaner environments and improved public health. To address this, environmental scientists have conducted various statistical studies to determine the toxicity threshold of particulate matter. These studies aim to identify the concentration level at which exposure to PM becomes harmful to human health. However, determining this threshold often requires extensive time-consuming and costly scientific investigations, including epidemiological and toxicological research on the health effects of different PM exposure levels. Table 3.1 highlights studies reviewed from over 40 research papers that examined the toxicity threshold of particulate matter.

The studies presented in Table 3.1 focus on determining the toxicity threshold of particulate matter (PM). These studies explore various aspects, including proinflammatory effects, oxidative stress, cytotoxicity, and mitochondrial function impairment. The research covers different locations, such as Cotonou (Benin), the Beijing-Tianjin-Hebei region, and areas affected by dust storms. It also examines different types of particulate matter, including spray paint particles and atmospheric PM. The main objective is to gain a comprehensive understanding of the health effects of PM through in vitro and in vivo experiments, toxicity assessments, and compositional analyses.

However, despite these efforts, traditional methods for determining PM toxicity thresholds remain slow, complex, and expensive. These methods often fail to accurately identify the specific physico-chemical and exposure characteristics that contribute most to PM toxicity. Additionally, they do not account for the complex interactions between different

Reference	Title	About the paper
[Cachon et al. 2014]	Proinflammatory effects and oxidative stress within human bronchial epithelial cells exposed to atmospheric particulate matter ($PM_{2.5}$ and $PM_{>2.5}$) collected from Cotonou, Benin	Experiments on $PM_{2.5}$ and $PM_{>2.5}$ collected in Cotonou (Benin), by evaluating their adverse health effects using in vitro culture of human lung cells.
[X. Niu et al. 2017]	Atmospheric levels and cytotoxicity of polycyclic aromatic hydrocarbons and oxygenated-PAHs in $PM_{2.5}$ in the Beijing-Tianjin-Hebei region	Priority PAHs and OPAHs were quantified in the $PM_{2.5}$ samples collected at four cities in the Beijing-Tianjin-Hebei region, and the toxicity of extracts prepared from the fine particles was investigated in vitro.
[Shao et al. 2018]	The pro-inflammatory effects of particulate matter on epithelial cells are associated with elemental composition	Roof space PM samples as a proxy of residential cumulative exposure to outdoor air pollution. The goal is to investigate their pro-inflammatory effects on human lung cells and the contribution of the endotoxin and chemical content.
[Leclercq et al. 2018]	Air pollution-derived $PM_{2.5}$ impairs mitochondrial function in healthy and chronic obstructive pulmonary diseased human bronchial epithelial cells	In order to find whether the mitochondrial dysfunction is closely related to the cell homeostasis maintenance after particulate matter ($PM_{2.5}$) exposure, oxidative, inflammatory, apoptotic and mitochondrial endpoints were carefully studied in human bronchial epithelial BEAS-2B, normal human bronchial epithelial (NHBE) and chronic obstructive pulmonary disease (COPD)-diseased human bronchial epithelial (DHBE) cells acutely or repeatedly exposed to air pollution-derived $PM_{2.5}$
[G. Wang et al. 2019]	High-content analysis of particulate matters-induced oxidative stress and organelle dysfunction in vitro	The PMs-induced oxidative stress and organelle dysfunction study, by comparing the toxicity threshold/effect of three different PMs ($PM_{2.5}$, DEP, and CB.)
[Y.-C. Chen et al. 2019]	Environmental concentration of spray paint particulate matters causes pulmonary dysfunction in human normal bronchial epithelial BEAS-2B cell	Normal human lung epithelial cells as a model to provide more realistic information related to the adverse health impact of SPPMs (spray paint particulate matters). This research shows that exposure to different sizes of SPPMs at realistic occupational levels can stimulate early COPD development through the induction of epithelial barrier dysfunction.
[D. Zhang et al. 2022]	Toxicity assessment and heavy metal components of inhalable particulate matters ($PM_{2.5}$ and PM_{10}) during a dust storm invading the city	The spatial-temporal differences and relationships of PMs from urban and suburban areas during a case event of long-range dust were investigated through compositional analyses and toxicity tests in order to understand the health risks posed by dust $PM_{2.5}$ and PM_{10}

Table 3.1.: Summary of reviewed studies on PM toxicity.

PM characteristics, which limits the accuracy of their findings.

To overcome these limitations, this study introduces a machine learning-based approach to predicting PM toxicity thresholds using a well-structured dataset compiled from existing literature. This method not only improves predictive accuracy but also incorporates explainable AI techniques, specifically SHAP values, to identify and quantify the contribution of each PM characteristic to its toxicity. Traditional methods cannot provide this level of insight.

Our study takes a new approach by integrating machine learning and explainable AI into PM toxicity research. This provides a more efficient and insightful tool for health risk assessment compared to traditional methods. The proposed methodology has the potential to significantly improve the way toxicity thresholds are determined, leading to important implications for public health policies and environmental regulations.

The traditional laboratory approach for determining PM toxicity thresholds is difficult, expensive, and time-consuming. However, a large amount of data from previous studies can be collected and structured to develop machine learning models that simplify and accelerate the prediction process. To our knowledge, this approach has not yet been explored using ML techniques, making it a promising area for innovation. By applying advanced ML algorithms, we aim to gain new insights and improve the accuracy and efficiency of toxicity threshold predictions.

The prediction of toxicity is not limited to airborne particulate matter but extends to other scientific fields. For example, in pharmaceuticals, researchers study how different substances affect the human body [Vo et al. 2019]. In environmental science, scientists analyze the impact of chemicals on ecosystems and wildlife [Idakwo et al. 2018]. Machine learning has also been applied in toxicology, such as using deep learning to predict the toxicity of chemical compounds [Unterthiner et al. 2015]. Some studies have employed ML models like regression decision trees and random forests to assess toxic risks and the factors contributing to the toxicity of nanomaterials [Gernand 2013; Gernand and Casman 2014]. Additionally, a recent review summarizes the use of AI-based approaches for toxicological research across various disciplines [Perez Santin et al. 2021].

Ultimately, the ability to predict and evaluate toxicity is essential in multiple scientific fields, playing a crucial role in protecting human health and the environment.

3.3. Data Collection and Management

Building a comprehensive dataset for machine learning-based toxicity prediction of particulate matter was a challenging task. One of the main difficulties was the absence of an open-access dataset and the complexity of compiling a unified dataset from diverse sources. This section describes the process of collecting and managing data from multiple experimental studies, each providing valuable insights into the physico-chemical properties of PM and associated exposure conditions.

To construct a diverse and representative dataset, an extensive review of existing scientific literature was conducted. This included studies covering laboratory experiments, field measurements, and controlled exposure trials, with a particular focus on toxicity research related to oxidative stress and inflammation. The final dataset comprises over 1000 instances (representing different PM samples and exposure tests) with more than 30 features, extracted from over 40 peer-reviewed studies. Table 3.1, presented earlier in the literature review section (Section 3.2), provides an overview of the key studies used in the data collection process. Priority was given to studies that reported detailed physico-chemical properties and exposure conditions of PM samples, including particle size distribution, chemical composition, and aerodynamic diameter, among others. Table 3.2 summarizes the key physico-chemical and exposure characteristics of the collected PM data.

During the data collection process, several challenges were encountered:

1. Many physico-chemical characteristics were not consistently reported across studies, with more than 50% missing values for parameters such as size distribution, aerodynamic diameter (average), and agglomeration state.

2. Variability in the classification of PM composition, with different research groups using diverse categorizations and measurement units (e.g., ng/g, ng/m^3, %).

3. Inconsistencies in exposure concentration units across studies, with some reporting values in $\mu g/mL$ and others in $\mu g/cm^2$.

After data collection, a rigorous preprocessing phase was undertaken, including unit standardization, handling missing values, and correcting inconsistencies to ensure a structured and unified dataset.

Physico-chemical characteristics		Exposure characteristics	
Characteristic	Description	Characteristic	Description
Aerodynamic diameter class	Categorization of PM based on size ($PM_{0.1}$, PM_1, $PM_{2.5}$, PM_{10}), considering its aerodynamic behavior.	In vitro/ In vivo	Experimental conditions: In vitro (outside a living organism) or In vivo (within a living organism).
Size distribution	Distribution of PM sizes within a sample.	Species	Type of test subjects (e.g., human cells, rats).
Aerodynamic diameter (average)	Average aerodynamic diameter of PM in a sample.	Mode of exposure	How exposure occurs (e.g., inhalation, cellular exposure).
Agglomeration state	Clustering of PM particles due to weak adhesion forces.	Exposure duration	Total time of exposure in experimental studies.
Aggregation state	Stronger bonding of PM particles into clusters.	Unique or chronic exposure	Single vs. repeated exposure scenarios.
Composition	Chemical composition of PM (e.g., carbon, trace metals, polycyclic aromatic hydrocarbons).	Cells	Type of cells used in experiments.
Surface chemistry	Chemical interactions on PM surfaces.	Cell viability test	Laboratory assays assessing the percentage of viable cells post-exposure.
Zeta potential	Electrical potential at the PM surface in a surrounding medium.	Markers of oxidative stress or inflammation	Indicators of PM-induced oxidative stress or inflammation.
Oxidative potential	Ability of PM to generate reactive oxygen species, leading to cellular damage.	Concentrations	Tested exposure concentrations of PM.
Oxygen content	Relative oxygen composition of PM.	Toxicity threshold	Minimum PM concentration causing toxic effects.

Table 3.2.: Overview of physico-chemical and exposure characteristics of PM.

This dataset serves as the foundation for our research, enabling the development of predictive models for PM toxicity threshold estimation. The careful curation and integration of physico-chemical and exposure characteristics ensure that our models capture the complex relationships between PM properties and their potential health impacts. Additionally, based on this dataset, we are developing a thesaurus for toxicity threshold research, which will facilitate broader scientific use and potential integration into international research efforts.

3.4. Theoretical Background

Machine learning (ML) has become an essential tool in environmental health sciences, offering the capability to analyze complex, high-dimensional datasets and detect intricate patterns that traditional statistical models often overlook. In the context of toxicity classification, ML models facilitate automated, data-driven decision-making by learning from historical experimental findings. Supervised learning, particularly classification techniques, is well-suited for toxicity threshold determination, where the objective is to differentiate between toxic and non-toxic PM concentrations based on key physico-chemical and exposure-related features.

One of the key strengths of ML models lies in their ability to process heterogeneous data sources, capture interactions among multiple features, and generalize predictions to new PM samples. However, ensuring model interpretability remains a major challenge, as black-box predictions may lack scientific credibility unless supplemented by explainability techniques such as SHAP values, which decompose model outputs into human-understandable feature contributions.

3.4.1. Mathematical Formulation

The problem of PM toxicity classification can be formally described as follows:

1. **Feature space:**

$$\mathbf{x}_i \in \mathbb{R}^d, \quad i = 1, 2, \ldots, n$$

where $\mathbf{x}_i$ is the feature vector representing the i-th PM sample, encompassing physico-chemical properties such as size, composition, surface area, and exposure characteristics.

2. **Binary target variable:**

$$y_i = \begin{cases} 1 & \text{if the PM concentration is toxic (class 1)} \\ 0 & \text{if the PM concentration is non-toxic (class 0)} \end{cases}$$

3. **Classification model:** The objective is to learn a function $f : \mathbb{R}^d \to \{0, 1\}$ that maps the input features to a binary target variable:

$$\hat{y}_i = f(\mathbf{x}_i)$$

where $\hat{y}_i$ is the predicted class label for the i-th PM sample.

3.4.2. Machine Learning Algorithms

To model the relationship between the physico-chemical properties of PM, exposure characteristics, and toxicity outcomes, a diverse set of classical machine learning algorithms was employed (Section: 2.3). Each of these algorithms offers unique strengths, enabling a comprehensive analysis of the dataset:

- **Logistic Regression (LR)**: A widely used method for binary classification that models the probability of a sample belonging to the toxic or non-toxic class using a logistic function.
- **Support Vector Classifier (SVC)**: A powerful classification technique that works well for both linear and non-linear separable data, aiming to find the optimal decision boundary by maximizing the margin between classes.
- **Decision Tree (DT)**: A rule-based model that efficiently captures complex interactions within the data by recursively splitting the feature space based on entropy or Gini impurity.
- **Random Forest (RF)**: An ensemble learning method that combines multiple decision trees to improve predictive accuracy and reduce overfitting.
- **Extreme Gradient Boosting (XGBoost)**: A high-performance boosting algorithm that optimizes classification through an iterative learning process, improving upon errors made by previous trees.

By leveraging this ensemble of algorithms, the study aims to uncover a wide range of potential patterns and relationships within the data, ensuring robust and comprehensive analysis. Each model was fine-tuned through hyperparameter optimization to maximize performance and generalization across the dataset.

The integration of these ML techniques not only enhances predictive power but also provides a scalable approach to toxicity classification. However, given the complexity of ML models, the study emphasizes interpretability by employing explainability methods such as SHAP values, ensuring that predictions are both scientifically valid and actionable for regulatory decision-making.

3.5. Machine Learning Models for PM Toxicity Classification

This section introduces the proposed ML models for predicting PM toxicity thresholds, where class 0 represents non-toxic concentrations and class 1 represents toxic concentrations. Developed based on extensive research and tailored methodologies, these models offer a significant improvement in assessing airborne particulate matter toxicity. By utilizing ML techniques and a well-curated dataset that captures key physico-chemical and exposure characteristics, the models enhance our understanding of PM toxicity. Designed for precision, scalability, and adaptability, they not only provide accurate predictions but also establish a framework that can be further refined and expanded. This approach aims to offer a robust and scalable solution for toxicity assessment, contributing to improved public health and environmental protection.

3.5.1. Data Preprocessing and Exploratory Data Analysis

The dataset used in this study comprises over 1,000 particulate matter (PM) samples and 28 features, including the target variable. To ensure the dataset's quality, consistency, and suitability for subsequent modeling, a thorough data preprocessing and exploratory data analysis (EDA) phase was conducted. Table 3.3 presents the final set of collected features, along with the percentage of missing values for each attribute.

Column name	Null count	Null ratio (%)	Description	Data type
Sample Name	0	0	Name of the PM sample	Object
Source	0	0	Sample source (e.g., industrial, urban, etc.)	Object/Categorical
Aerodynamic diameter class	12	1.12	Categorization of PMs based on aerodynamic diameter	Object/Categorical
Aerodynamic diameter (average)	1009	94.04	Mean aerodynamic diameter of PMs	Object/Categorical
Size distribution	1022	95.25	Distribution of particle sizes in the sample	Object/Categorical
Agglomerated	898	83.69	State of particle agglomeration	Integer/Categorical
Aggregated	941	87.7	Aggregation state of the PMs	Integer/Categorical
Composition ETM	0	0	Concentration of metallic trace elements	Float/Numerical
Composition CARBON	0	0	Concentration of carbon	Float/Numerical
Composition HAP	0	0	Concentration of polycyclic aromatic hydrocarbons	Float/Numerical
Composition O-HAP	0	0	Concentration of oxygenated polycyclic aromatic hydrocarbons	Float/Numerical
Composition COV	0	0	Concentration of volatile organic compounds	Float/Numerical
Composition PCDD	0	0	Concentration of polychlorinated dibenzo-dioxins	Float/Numerical
Composition PCDF	0	0	Concentration of polychlorinated dibenzo-furans	Float/Numerical
Composition DL-PCB	0	0	Concentration of polychlorinated biphenyls-Dioxin Like	Float/Numerical
Composition WSE	0	0	Concentration of water-soluble elements	Float/Numerical
Composition IONS	0	0	Concentration of ions	Float/Numerical
Composition OTHERS	0	0	Concentration of other pollutants	Float/Numerical
In vitro/In vivo	0	0	Experimental setting (in vitro or in vivo)	Object/Categorical
Exposure duration	0	0	Length of PM exposure	Float/Numerical
Exposure Unique Chronic	0	0	Single or repeated exposure classification	Object/Categorical
Cell	0	0	Type of cells used in experiments	Object/Categorical
Cell viability test	340	31.69	Method to determine viable (living) cells in a sample	Integer/Categorical
MSO/Infla Name	0	0	Name of oxidative stress or inflammation markers	Object/Categorical
Infla 0/MSO 1	0	0	Indicator of inflammation (0) or oxidative stress (1)	Integer/Categorical
Concentration	0	0	Tested PM concentration	Float/Numerical
Unit of measure	0	0	Measurement unit for concentration values	Object/Categorical
Effect	0	0	Toxicity outcome (toxic or non-toxic)	Integer/Categorical

Table 3.3.: Overview of the collected PM features, missing values, and data types.

A. Data Preprocessing and Cleaning

Data preprocessing is a crucial step in machine learning workflows, ensuring that the dataset is structured, consistent, and ready for modeling. This phase involved handling missing values, detecting outliers, and standardizing data representations.

As shown in Table 3.3, six features had missing values: size distribution (95.25%), aerodynamic diameter (94.04%), agglomerated state (83.69%), aggregated state (87.7%), cell viability test (31.69%), and aerodynamic diameter class (1.12%). A commonly used guideline suggests removing features with over 70–80% missing values as they may introduce excessive uncertainty. Based on this, the following variables were removed from further analysis:

- Size distribution
- Aerodynamic diameter (average)
- Agglomerated state
- Aggregated state

For the cell viability test feature, missing values were replaced with -1 to indicate unavailable data. Meanwhile, missing values in aerodynamic diameter class were imputed using the most frequently occurring class ($PM_{2.5}$).

After preprocessing, the final dataset included 11 categorical features and 13 numerical features. Table 3.4 presents an overview of categorical variables and their possible values, while Table 3.5 provides descriptive statistics of numerical variables.

B. Exploratory Data Analysis (EDA)

EDA plays a crucial role in identifying patterns, relationships, and potential biases within the dataset before applying machine learning models. This step involves:

- **Understanding feature distributions** to detect skewness, outliers, and missing patterns.
- **Examining relationships between PM characteristics and toxicity outcomes** to assess their potential contribution to classification tasks.

Categorical column	Values
Aerodynamic diameter class	PM_{10}, $PM_{2.5}$, PM_1, $PM_{0.1}$
In vitro/In vivo	In vitro, In vivo
Exposure Unique Chronic	Unique, Chronic
Cell	BEAS-2B, A549, NHBE, Co-culture A549/THP-1, 16HBE, No cells, HELF, HUVEC, HAT2E, WL-38, Rat Epithelial Lung Cells
Cell viability test	0, 1, -1
MSO/Infla Name	ROS, IL-8, IL-6, TNF-Alpha, IL-1B, GSSG/GSH, MDA, SOD, GSH, 8OHdG
Infla0 or MSO1	Marker of Oxidative Stress (1), Inflammation (0)
Unit of measure	ug/cm^2, ug/mL
Effect	Toxic (1), Non-toxic (0)

Table 3.4.: List of categorical columns in the final dataset with corresponding values.

Column name	Count	Mean	Std	Min	25%	50%	75%	Max
Composition ETM	1073.0	35.18	41.20	-1.0	3.45	12.37	87.11	100.00
Composition CARBON	1073.0	7.07	25.41	-1.0	-1.00	-1.00	-1.00	100.00
Composition HAP	1073.0	14.94	25.72	-1.0	0.05	0.23	23.90	100.00
Composition O-HAP	1073.0	0.59	7.14	-1.0	-1.00	-1.00	-1.00	49.52
Composition COV	1073.0	-0.91	0.28	-1.0	-1.00	-1.00	-1.00	0.050
Composition PCDD	1073.0	-0.97	0.16	-1.0	-1.00	-1.00	-1.00	0.00474
Composition PCDF	1073.0	-0.971	0.165	-1.0	-1.00	-1.00	-1.00	0.00606
Composition DL-PCB	1073.0	-0.97	0.16	-1.0	-1.00	-1.00	-1.00	0.00980
Composition WSE	1073.0	-0.19	6.13	-1.0	-1.00	-1.00	-1.00	68.59
Composition IONS	1073.0	32.79	37.13	-1.0	-1.00	-1.00	69.44	94.59
Composition OTHERS	1073.0	5.07	19.25	-1.0	-1.00	-1.00	-1.00	85.26
Exposure duration	1073.0	29.43	17.91	1.0	24.00	24.00	24.00	72.00
Concentration	1073.0	73.14	178.23	1.0	10.00	25.00	80.00	2000.00

Table 3.5.: Descriptive statistics of numerical columns in the final dataset.

- **Visualizing class distributions** to evaluate dataset balance and identify any imbalances between toxic and non-toxic classifications.

To further understand the data, visual analysis was performed on key categorical and numerical features. Figure 3.4 presents the distribution of the toxicity outcome (Effect) across some selected categorical variables such as aerodynamic diameter class, exposure duration, and biological model type (in vitro/in vivo). This allows for visual inspection of potential relationships between experimental conditions and observed effects. Figure 3.5 displays a correlation heatmap that quantifies the pairwise associations between physico-chemical and exposure properties and the toxicity label, offering insights into feature relevance and redundancy.

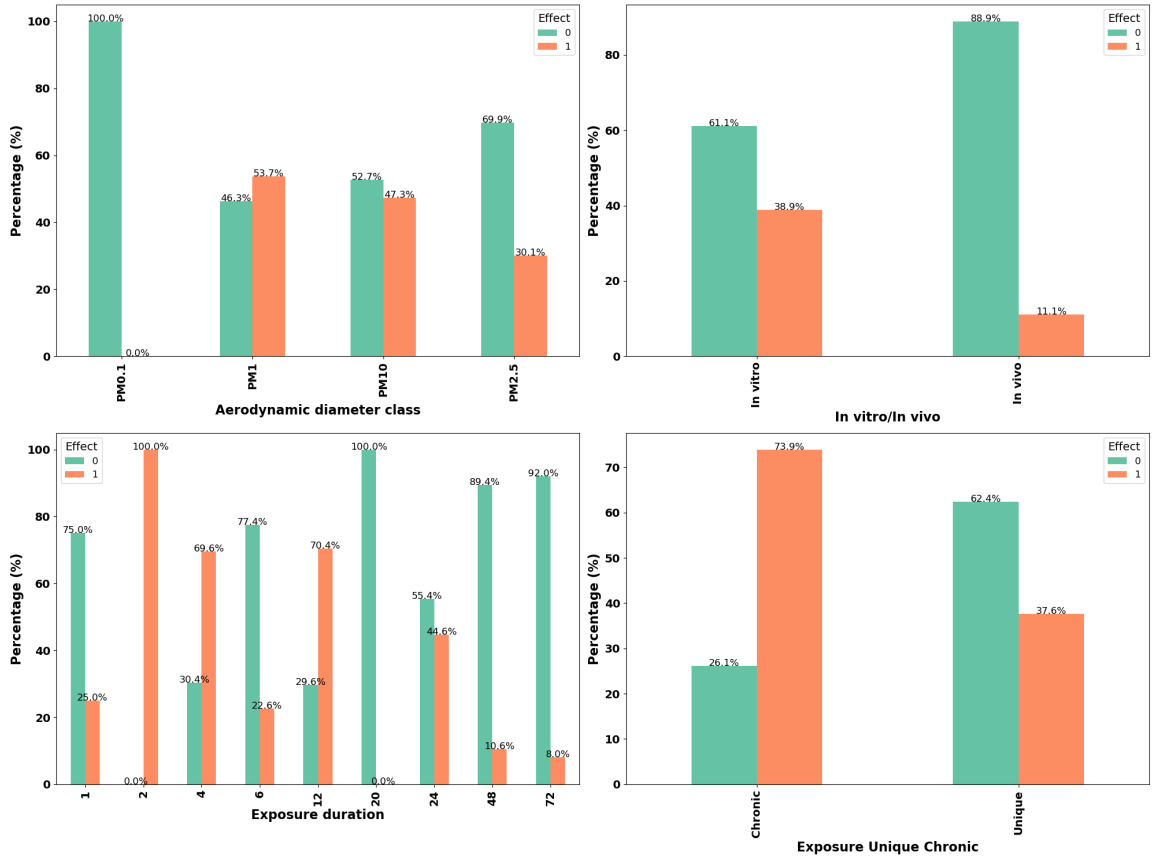


Figure 3.4.: Distribution of toxicity outcomes (Effect) across categorical features.

By conducting thorough data preprocessing and EDA, a well-structured dataset was ob-

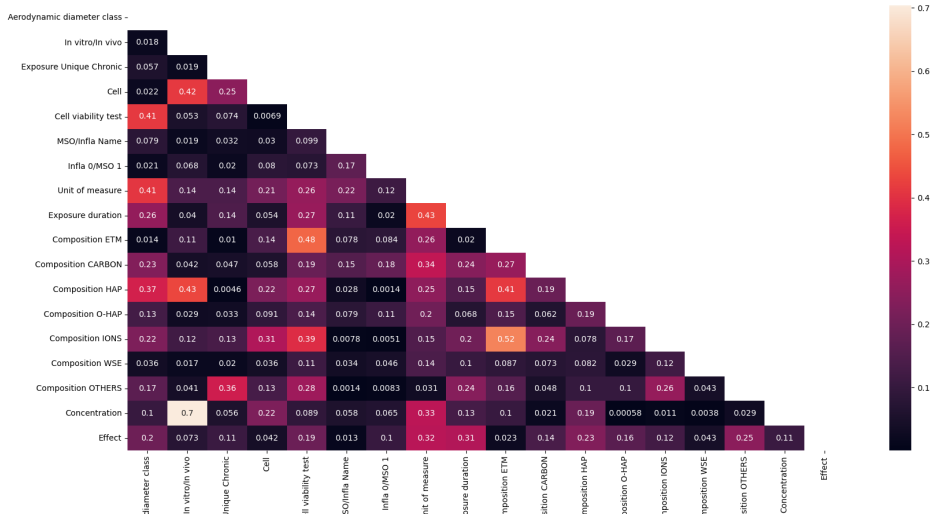


Figure 3.5.: Correlation heatmap between physico-chemical and exposure features and the toxicity outcome.

tained, providing a solid foundation for developing machine learning models. The next section delves into feature engineering techniques used to refine and optimize model performance.

3.5.2. Feature Engineering

Feature engineering is a crucial step in the development of machine learning models, significantly influencing their robustness and predictive accuracy. This process involves transforming and refining input variables to enhance their relevance and utility, ensuring that the model can effectively capture meaningful patterns. Feature engineering encompasses various tasks, including the careful selection, transformation, and creation of features from the preprocessed dataset, particularly in handling categorical and numerical variables.

Handling Categorical Features Categorical variables must be converted into a format suitable for ML algorithms, commonly using encoding techniques such as one-hot encoding or label encoding. In this study, different approaches were applied based on the nature of the categorical features:

- **Ordinal Encoding**: Applied to *Aerodynamic diameter class*, *Cell*, and *MSO/Infla*

Name, where each category was assigned a unique numerical identifier.

- **Dummy Encoding**: Used for *In vitro/In vivo*, *Exposure Unique Chronic*, and *Unit of measure*. Unlike one-hot encoding, dummy encoding represents N categories using $N-1$ features, preventing redundancy while retaining categorical information.

Processing Numerical Features The numerical attributes include several physico-chemical and exposure characteristics. As shown in Table 3.5, the composition-related features represent concentration ratios or percentages of different chemical components in each PM sample. These include: *Composition ETM, Composition CARBON, Composition HAP, Composition O-HAP, Composition COV, Composition PCDD, Composition PCDF, Composition DL-PCB, Composition WSE, Composition IONS, and Composition OTHERS*.

A value of -1 in these composition features indicates missing or unreported data in the original research studies, introducing uncertainty about whether the actual value is zero or simply not recorded. Given that *Composition COV, Composition PCDD, Composition PCDF, and Composition DL-PCB* contain mostly negligible values, they were excluded from further analysis. The retained composition features for modeling include: ETM, CARBON, HAP, O-HAP, WSE, IONS, and OTHERS.

The remaining numerical variables, *Exposure duration* and *Concentration*, were incorporated into the final dataset without additional transformations.

Final Feature Set The final list of selected features, including the target variable, is as follows:

1. **Physico-chemical Characteristics**
   - Aerodynamic diameter class
   - Composition ETM
   - Composition CARBON
   - Composition HAP
   - Composition O-HAP
   - Composition IONS
   - Composition WSE

- Composition OTHERS

2. **Exposure Characteristics**
   - In vitro/In vivo
   - Exposure Unique Chronic
   - Cell
   - Cell viability test
   - MSO/Infla Name
   - Infla 0/MSO 1
   - Exposure duration
   - Concentration
   - Unit of measure
   - Effect (target variable)

3.5.3. Hyperparameter Optimization

Hyperparameter optimization (Section: 2.6.2) is a critical step in fine-tuning machine learning models to enhance their performance before the training phase. Hyperparameters are configuration settings that govern how an algorithm learns from data, directly influencing its accuracy and generalization ability on unseen data. The goal of hyperparameter optimization is to identify the most effective combination of values that maximize model performance.

In this study, the Grid Search Cross-Validation (GridSearchCV) method was employed for hyperparameter tuning. GridSearchCV performs an exhaustive search over a predefined set of hyperparameter values to determine the optimal combination for each ML algorithm. The process involves:

- Defining a grid of hyperparameters and their potential values.
- Splitting the dataset into training and validation subsets.
- Iteratively training and evaluating the model for each hyperparameter combination using cross-validation.

- Selecting the best-performing hyperparameter set based on a chosen evaluation metric (e.g., accuracy or F1-score).

Cross-validation ensures that the selected hyperparameters contribute to a model that generalizes well to unseen data, reducing the risk of overfitting. This automated approach replaces the need for manual tuning, streamlining the process of selecting optimal hyperparameters for improved model performance.

Table 3.6 presents the tested hyperparameter values and the best-selected configurations for the employed ML algorithms.

Machine Learning Algorithm	Hyperparameters and Tested Values	Best Values
Logistic Regression	Penalty (Regularization): ['l1', 'l2'], Regularization parameter (C): [0.001, 0.01, 0.1, 1, 10, 100], Solver: ['saga', 'liblinear', 'lbfgs', 'newton-cg']	Penalty: 'l2', Regularization parameter (C): 100, Solver: 'newton-cg'
Support Vector Classifier	Regularization parameter (C): [0.1, 1, 10], Kernel: ['linear', 'poly', 'rbf']	Regularization parameter (C): 10, Kernel: 'rbf' (radial basis function)
Decision Tree Classifier	Criterion: ['gini', 'entropy'], Max depth: [None, 5, 10, 15], Min samples split: [2, 5, 10], Min samples leaf: [1, 2, 4]	Criterion: 'entropy', Max depth: None, Min samples split: 2, Min samples leaf: 1
Random Forest Classifier	Number of estimators: [50, 100, 200], Max depth: [None, 5, 10, 15], Min samples split: [2, 5, 10], Min samples leaf: [1, 2, 4]	Number of estimators: 50, Max depth: 15, Min samples split: 2, Min samples leaf: 1
XGBoost Classifier	Number of estimators: [50, 100, 200], Learning rate: [0.001, 0.01, 0.1], Max depth: [3, 4, 5], Subsample: [0.8, 0.9, 1.0], Gamma: [0, 1, 5], Min child weight: [1, 3, 5]	Number of estimators: 200, Learning rate: 0.1, Max depth: 4, Subsample: 1.0, Gamma: 0, Min child weight: 5

Table 3.6.: Tested hyperparameters and optimal values for the selected ML algorithms.

3.6. Experimental Results

3.6.1. Results with Initial Imbalanced Data

This section presents the results obtained from the developed ML models when trained on the initial imbalanced dataset. As outlined in the previous section (Section: 2.6.3), the evaluation is based on key performance metrics, including accuracy, precision, recall, F1-score, k-fold

cross-validation, and ROC-AUC, providing a baseline assessment of the models performance. By analyzing results without applying any imbalance-handling techniques, we gain insight into the impact of skewed class distributions and establish a reference for comparison with results obtained after addressing class imbalance.

The dataset was divided into training and test sets using a 70:30 split, ensuring that 70% of the data was allocated for training, while the remaining 30% was used for testing and validation. The split was stratified to maintain the proportion of toxic (class 1) and non-toxic (class 0) instances across both sets. Additionally, a fixed random state was used to ensure the reproducibility of results. Table 3.7 summarizes the performance metrics for the developed models.

It is particularly important to emphasize the recall metric (Section: 2.6.3) since misclassifying a toxic PM concentration (class 1) as non-toxic (class 0) (False Negatives, FN) is more critical than the reverse scenario (False Positives, FP). Hence, the goal is to minimize the number of FN cases, thereby maximizing recall.

Machine Learning Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Actual-Predicted			
					0-0	0-1	1-1	1-0
Logistic Regression	75	68	66	67	159	39	82	42
Support Vector Classifier	73	65	68	66	152	46	84	40
Decision Tree Classifier	84	78	79	79	171	27	98	26
Random Forest Classifier	87	85	81	83	180	18	101	23
XGBoost	84	80	76	78	175	23	94	30

Table 3.7.: Performance metrics of the developed ML models on initial imbalanced data.

Figure 3.6 presents the confusion matrices for all models, offering a visual representation of their classification outcomes. The matrices provide an intuitive comparison of True Positives, True Negatives, False Positives, and False Negatives across different classifiers.

Additionally, Figure 3.7 presents the ROC curves of the developed models along with their corresponding AUC (Area Under the Curve) values. The ROC curve provides insight into the trade-off between the true positive rate (sensitivity) and the false positive rate at different threshold values, while AUC quantifies the model's ability to differentiate between toxic and non-toxic samples.

To ensure that the models are not overfitting or memorizing the training data, k-fold cross-validation (k=5) was performed to evaluate their generalization capability. Table 3.8

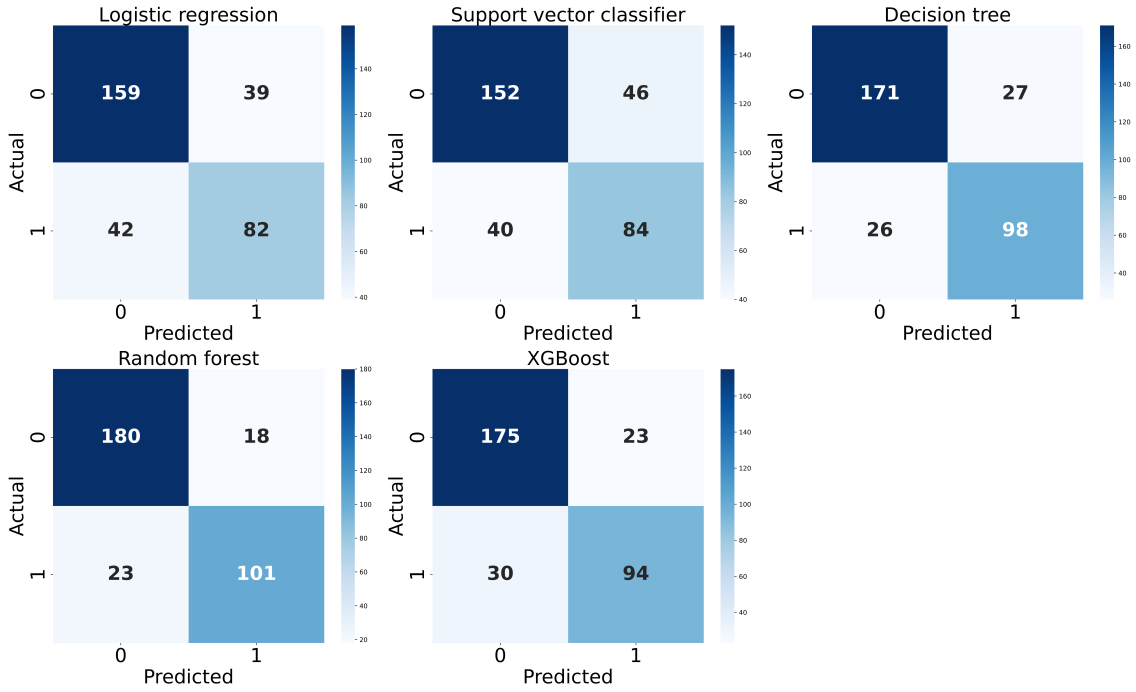


Figure 3.6.: Confusion matrices of the developed ML models on initial imbalanced data.

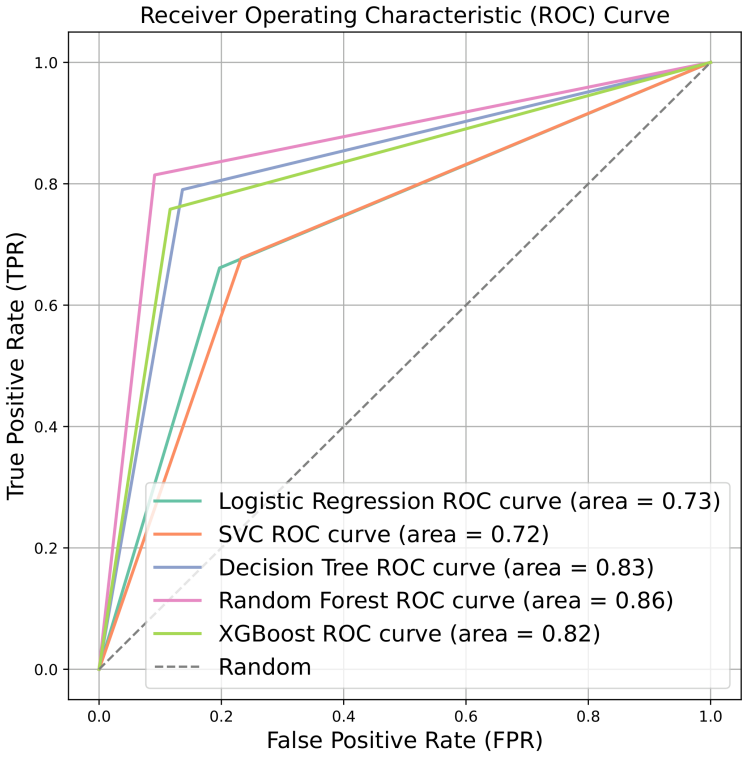


Figure 3.7.: Comparison of ROC curves and AUC values for the developed ML models on initial imbalanced data.

presents the mean performance metrics obtained from cross-validation.

Machine Learning Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Logistic Regression	73	66	61	64
Support Vector Classifier	74	65	73	68
Decision Tree Classifier	83	77	80	79
Random Forest Classifier	88	85	82	83
XGBoost	87	84	80	82

Table 3.8.: 5-Fold cross-validation performance metrics of the developed ML models on initial imbalanced data.

3.6.2. Results with Class Weight Adjustment for Imbalanced Data

To mitigate the effects of class imbalance, the class weighting technique was applied. As illustrated in Figure 3.8, the dataset exhibits an imbalance where over 60% of PM samples belong to the non-toxic class (0), whereas less than 40% are toxic (class 1).

By assigning higher weights to the minority class during training, the model is encouraged to reduce False Negatives (FN) and improve its ability to detect toxic PM concentrations. The adjusted loss function is given by Equation 3.1, [Kotsiantis et al. 2006]:

$$Weighted\ Loss = \frac{-1}{n}\sum_{i=1}^{n} w_{pos} * (y_i \log(\hat{y}_i)) + w_{neg} * ((1 - y_i) \log(1 - \hat{y}_i)) \tag{3.1}$$

where w_{pos} and w_{neg} represent the class weights computed as follows (Equation 3.2), [He and Garcia 2009]:

$$w_j = \frac{n}{k * n_j} \tag{3.2}$$

- n is the total number of training samples,
- k is the number of classes (in this case, $k = 2$),
- n_j is the number of samples belonging to class j (i.e., class 0 or class 1).

For the dataset in this study, class weights were computed as 1.3021 for toxic samples (class 1) and 0.8116 for non-toxic samples (class 0).

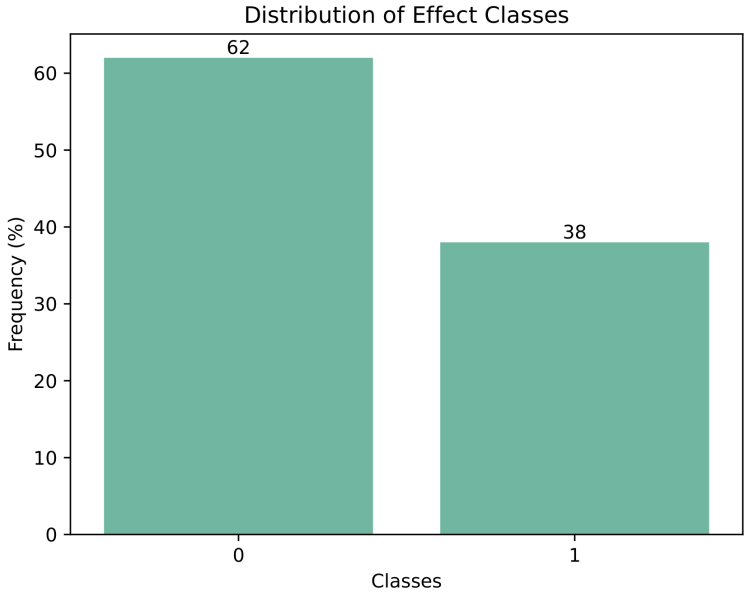


Figure 3.8.: Distribution of toxic (class 1) and non-toxic (class 0) samples in the dataset.

Tables 3.9 and 3.10 summarize the performance of the developed ML models after class weight adjustment.

Machine learning algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Actual-Predicted			
					0-0	0-1	1-1	1-0
Logistic regression	74	63	77	69	141	57	96	28
Support vector classifier	76	65	83	73	143	55	103	21
Decision tree classifier	83	79	77	78	172	26	96	28
Random forest classifier	86	84	80	82	179	19	99	25
XGBoost	84	78	80	79	170	28	99	25

Table 3.9.: Performance metrics of the developed ML models with class weight adjustment.

Figure 3.9 presents the confusion matrices of the developed models after applying class weight adjustment. This adjustment was implemented to address class imbalance by assigning higher penalties for misclassifying the positive class (toxic cases), which is typically underrepresented in imbalanced datasets. As a result, the number of false negatives (FN) was reduced, particularly in the Support Vector Classifier (FN = 21) and Random Forest models (FN = 25), indicating enhanced sensitivity in detecting toxic cases. The class weight

Machine learning algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Logistic regression	73	61	77	68
Support vector classifier	75	64	83	72
Decision tree classifier	83	78	80	79
Random forest classifier	88	85	82	83
XGBoost	86	82	83	82

Table 3.10.: 5-Fold cross-validation performance metrics of the developed ML models with class weight adjustment.

adjustment led to a moderate increase in recall across most models, reflecting improved sensitivity without compromising overall accuracy—an essential factor for accurately assessing the toxicity of particulate matter.

Figure 3.10 shows the ROC curves of the developed models, along with their corresponding AUC values, providing a comparative measure of model performance in distinguishing between toxic and non-toxic cases.

3.7. Model Explainability Using SHAP Values

This section explores the insights gained from SHAP (SHapley Additive exPlanations) values and the identification of key features that drive the predictions of the developed machine learning models. SHAP values provide a detailed breakdown of the model's output by attributing contributions to individual features, offering transparency into the decision-making process. Analyzing the significance of each feature helps uncover the underlying patterns influencing toxicity threshold predictions. This analysis emphasizes the critical role of specific physico-chemical and exposure-related characteristics in shaping the models' predictions. Through this approach, we aim to provide a clear and interpretable understanding of the factors driving toxicity outcomes, reinforcing the validity and reliability of our machine learning models.

3.7.1. Explainable AI

As machine learning and deep learning models grow in complexity, understanding their decision-making process becomes increasingly challenging. Explainable Artificial Intelligence

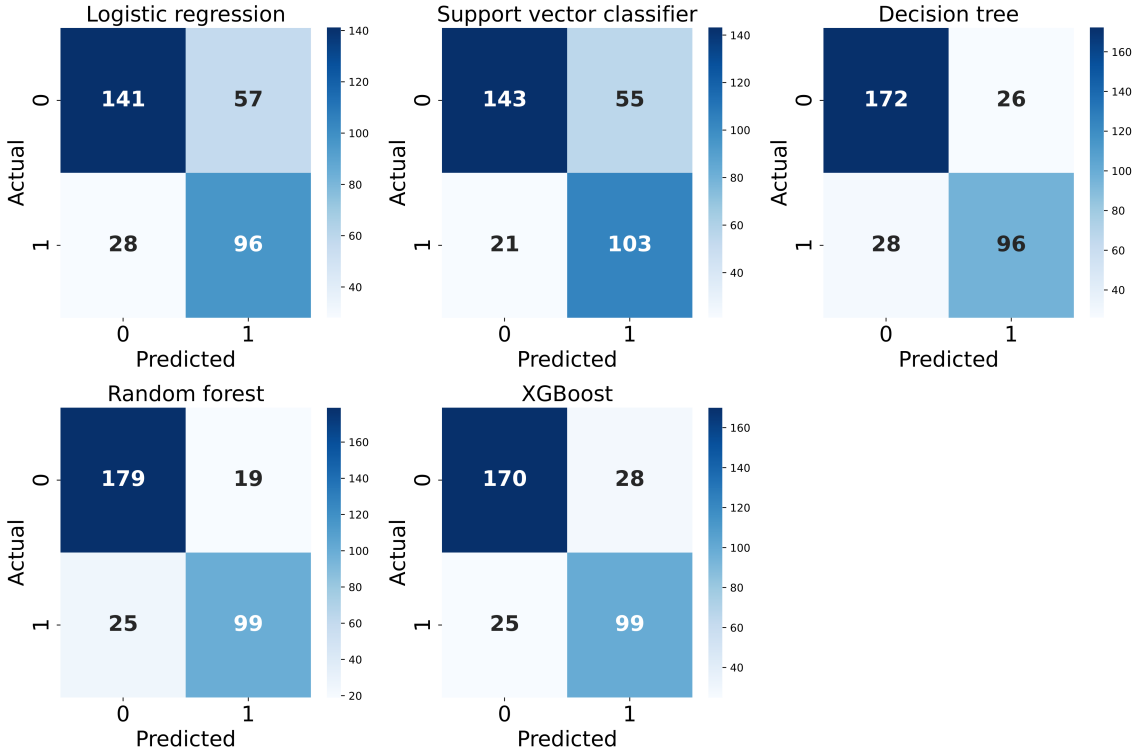


Figure 3.9.: Confusion matrices of the developed ML models with class weight adjustment.

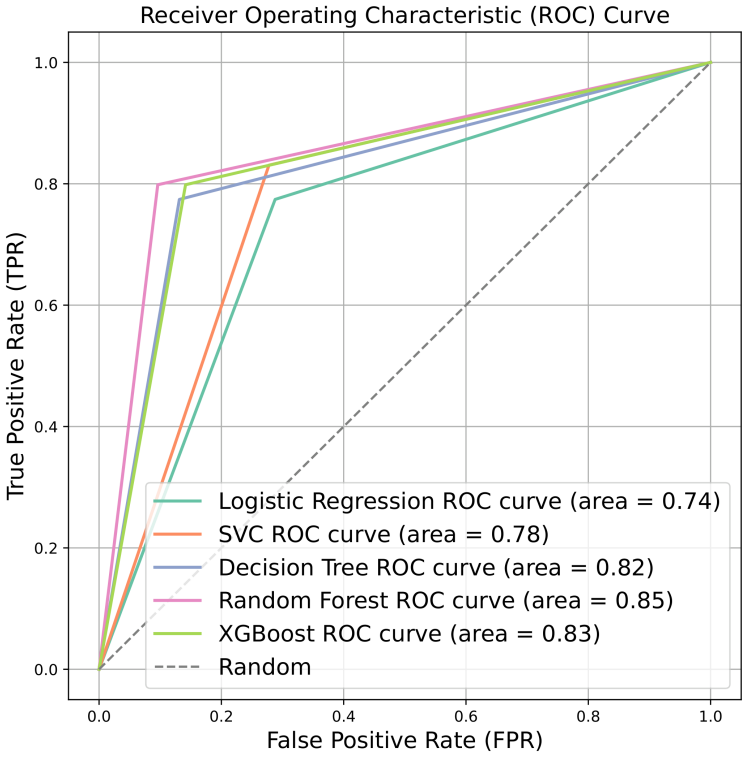


Figure 3.10.: Comparison of ROC curves and AUC values for the developed ML models with class weight adjustment.

(XAI) refers to a set of methods and techniques that make the predictions and internal logic of artificial intelligence (AI) models transparent and interpretable to humans [Doshi-Velez and B. Kim 2017; Gunning 2017]. This is particularly important in sensitive domains such as environmental science, healthcare, and autonomous systems, where accountability and trust are critical.

A. Need for Explainability

While complex models such as ensemble methods and neural networks often yield superior predictive performance, they are frequently criticized as "black boxes" due to their lack of transparency. Explainability is essential for:

- **Trust and Transparency:** Building user confidence by providing understandable justifications for model predictions.
- **Model Validation:** Identifying spurious correlations or biases in model behavior.
- **Regulatory Compliance:** Satisfying legal or ethical standards that require explanation of automated decisions.
- **Scientific Insight:** Gaining domain-specific knowledge from learned relationships in the data (e.g., pollution levels and weather conditions).

B. Categories of Explainability Methods

XAI methods can be broadly categorized as:

- **Global Explanations:** Provide insights into the overall model behavior. Examples include feature importance rankings and partial dependence plots.
- **Local Explanations:** Explain individual predictions. Examples include LIME (Local Interpretable Model-Agnostic Explanations) [Ribeiro et al. 2016] and SHAP (SHapley Additive exPlanations) [Lundberg and Lee 2017].

3.7.2. SHAP Values

SHAP is a unified framework based on cooperative game theory, which assigns each feature an importance value for a particular prediction. It ensures desirable properties such as

consistency and local accuracy, making it a powerful tool for interpreting complex models. SHAP values are particularly useful for:

- Identifying key contributing features for each prediction.
- Visualizing the impact and direction of each feature (e.g., summary and force plots).
- Comparing feature effects across different data instances or classes.

Figure 3.11 illustrates the SHAP beeswarm plots for the positive class (toxic) using the initial imbalanced data (before class weight adjustment). The beeswarm plot is a widely used tool for visualizing SHAP values across multiple instances in a dataset. In this plot, each point represents an individual prediction, with its vertical position indicating the SHAP value for a specific feature. The spread of points along the feature axis reflects the range and variability of feature contributions, highlighting the relative impact of each feature on model predictions. The y-axis lists the input features in descending order of importance, with the most influential features positioned at the top. This visualization helps identify consistent patterns, outliers, and the overall contribution of features to the model's decision-making process.

As shown in Figure 3.11, the most influential features vary slightly across models. However, some key features consistently appear as highly significant, including Composition ETM, Composition HAP, Concentration, and Composition IONS.

Figure 3.12 shows the SHAP beeswarm plots for the positive class after applying the class weight adjustment. The most important features remain largely consistent across models, though there are minor variations in the feature ranking order.

This comparison highlights the robustness of the feature importance rankings across different model adjustments, underscoring the consistent influence of certain features on toxicity threshold predictions regardless of data imbalance correction.

3.8. Discussion

This study explored the effectiveness of machine learning (ML) techniques in predicting the toxicity thresholds of airborne particulate matter (PM). The results presented in Section 3.6 provide valuable insights into the classification of PM concentrations and the impact of addressing class imbalance on model performance.

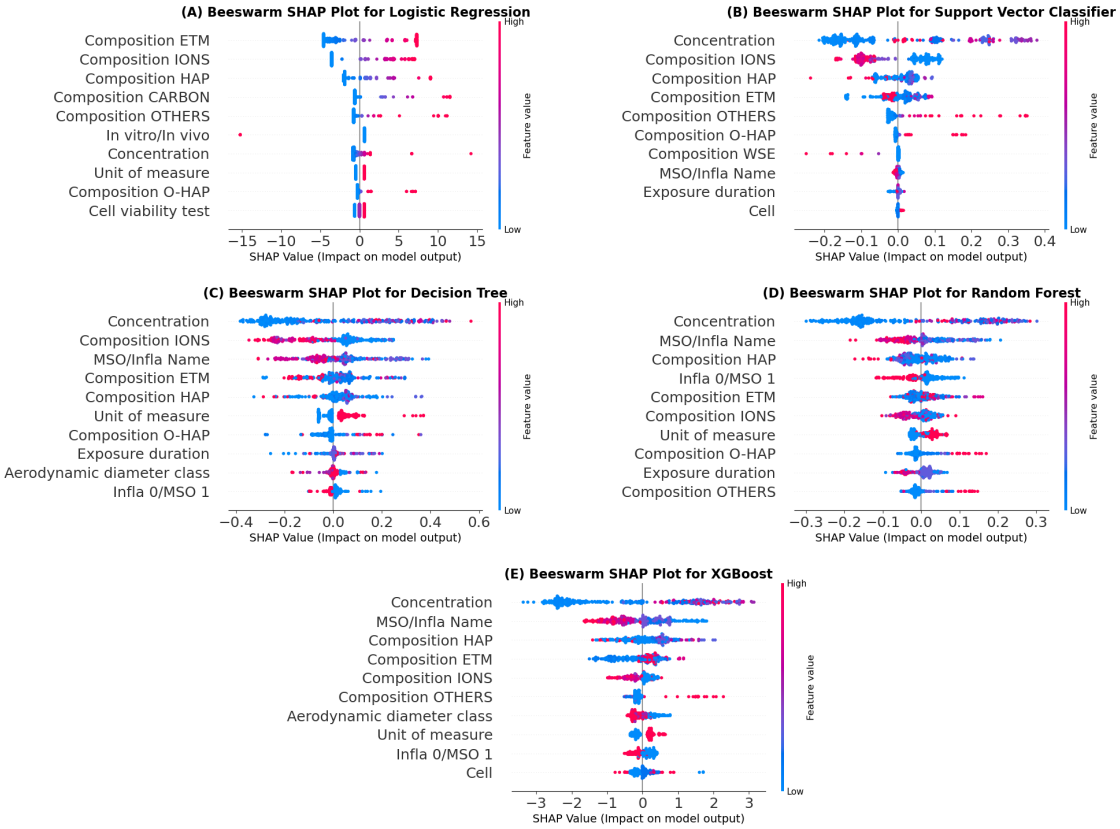


Figure 3.11.: SHAP beeswarm plots showing feature contributions for the developed ML models with initial imbalanced data.

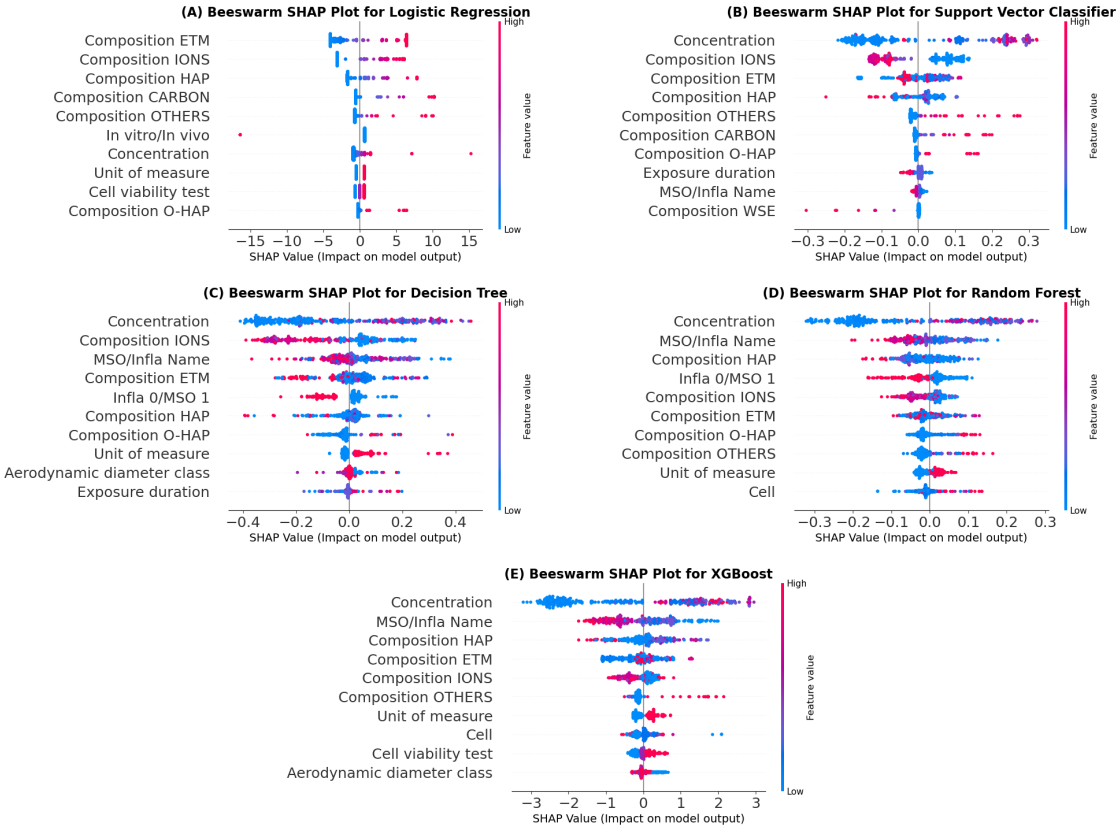


Figure 3.12.: SHAP beeswarm plots showing feature contributions for the developed ML models with class weight adjustment.

A. Model Performance and Imbalance Data Handling

Initially, we evaluated the models on an imbalanced dataset, where the Random Forest classifier achieved the highest performance with an accuracy of 87%, precision of 85%, recall of 81%, and an F1-score of 83% (Table 3.7). However, given the critical importance of recall in toxicity prediction—where false negatives could have severe consequences—we implemented a class weighting strategy to address the imbalance.

After applying class weights (Table 3.9), we observed notable improvements in recall across most models. In particular, the recall of the Support Vector Classifier (SVC) increased from 68% to 83%, with a minimal drop in overall accuracy (73% to 76%). This trade-off between accuracy and recall highlights a key challenge in environmental monitoring: ensuring that potentially toxic samples are correctly identified while maintaining overall model reliability. The confusion matrices (Figures 3.6 and 3.9) further confirm this improvement, showing a significant reduction in false negatives, particularly for the SVC and Random Forest models. Enhancing model sensitivity in this way is crucial for minimizing the risk of underestimating PM toxicity.

B. Cross-Validation and Model Robustness

To assess the robustness of our models, we conducted a 5-fold cross-validation (Tables 3.8 and 3.10). The consistency between initial performance metrics and cross-validation results suggests that our models generalize well to unseen data. Notably, the Random Forest and XGBoost models maintained strong performance across all metrics, even after addressing class imbalance. This stability underscores the reliability of these models in real-world scenarios where variations in PM data distributions are expected.

C. ROC-AUC Analysis

Further analysis using Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) values (Figures 3.7 and 3.10) confirmed the discriminative power of our models. High AUC values, particularly for the Random Forest and XGBoost models, indicate excellent differentiation between toxic and non-toxic PM samples. Importantly, the minimal change in AUC values after class weight adjustment suggests that while sensitivity to the minority class improved, the models' overall predictive power remained stable.

D. Feature Importance and SHAP Analysis

To enhance the interpretability of our models, we employed SHAP (SHapley Additive exPlanations) analysis. The SHAP beeswarm plots (Figures 3.11 and 3.12) consistently highlighted key features influencing PM toxicity predictions. Across all models, Composition ETM, Composition HAP, Concentration, and Composition IONS emerged as the most significant predictors. The stability of these findings, even after addressing class imbalance, reinforces their critical role in toxicity assessment.

The identification of these key features has important implications for environmental monitoring and public health policy. By prioritizing these compositional elements and concentration levels, regulatory agencies can develop more targeted and effective air quality assessment strategies.

E. Comparison with Traditional Toxicity Assessment Methods

Traditional methods for determining PM toxicity thresholds rely primarily on laboratory experiments and biological assays. These approaches involve controlled experimental conditions, making them fundamentally different from the data-driven ML approach employed in this study. Unlike experimental testing, which directly measures biological responses to PM exposure, our approach leverages statistical patterns in physico-chemical and exposure characteristics to predict toxicity thresholds.

Since no prior ML-based framework exists for direct comparison, our study represents a pioneering effort in this field. While traditional laboratory tests remain essential for validating biological responses, our method offers a complementary tool that provides rapid insights into PM toxicity risks without requiring extensive experimental procedures. This data-driven approach has the potential to accelerate preliminary risk assessments, guiding more focused laboratory investigations.

F. Practical Implications and Future Directions

The application of ML in toxicity prediction presents several advantages over conventional methods. Laboratory-based approaches, such as toxicological and epidemiological studies, are time-consuming, expensive, and require significant resources. In contrast, our ML models offer a scalable, cost-effective alternative capable of processing vast amounts of existing data to provide timely and reliable toxicity predictions.

Several key benefits of this approach include:

- **Speed and Efficiency:** ML models can rapidly analyze PM toxicity, significantly reducing the time required for assessments compared to traditional experimental methods.
- **Scalability:** The ability to incorporate data from various sources allows for broader environmental assessments across different geographic regions and pollution types.
- **Integration with Air Quality Monitoring Systems:** The potential for real-time toxicity prediction enables proactive public health interventions and regulatory decision-making.
- **Model Adaptability and Continuous Improvement:** As more data becomes available, the models can be continuously refined to enhance predictive accuracy and reliability.

Despite these advantages, some limitations must be acknowledged. The dataset used in this study, while extensive, remains relatively small, and the available literature does not provide all the necessary characteristics to develop highly sophisticated models. Future research should focus on expanding datasets, incorporating additional physico-chemical attributes, and exploring advanced modeling techniques such as deep learning to further improve toxicity predictions.

Nevertheless, this study establishes a robust methodological foundation for PM toxicity prediction. By addressing the challenges of class imbalance, enhancing model interpretability, and demonstrating practical applications, this research paves the way for more comprehensive studies in the future. Additionally, as part of our ongoing efforts, we are developing a thesaurus for toxicity threshold research, which will facilitate global scientific collaboration and integration into international research initiatives.

G. Conclusion

In summary, this study demonstrates the potential of ML-based approaches for predicting PM toxicity thresholds, offering an efficient and scalable alternative to traditional laboratory methods. The high recall achieved, particularly after class imbalance adjustments, ensures that toxic PM samples are less likely to be misclassified, thereby enhancing public health

protection. Moreover, SHAP analysis provides crucial insights into the key factors driving toxicity predictions, making the models more interpretable and actionable for regulatory agencies.

By leveraging the vast body of existing scientific literature, our approach enables rapid toxicity assessments that can complement traditional methods, guiding future research and policy development. With continued advancements in data availability and model refinement, ML-driven toxicity prediction has the potential to revolutionize air quality monitoring and environmental risk assessment.

3.9. Summary and Prospects

This study tackles the challenge of predicting the toxicity threshold of airborne particulate matter (PM). Traditional methods for assessing PM toxicity rely on labor-intensive and costly laboratory tests. Understanding PM toxicity is essential for mitigating the health risks of air pollution, highlighting the need for more efficient predictive techniques.

3.9.1. Summary

The primary goal of this research was to develop a machine learning (ML)-based approach to predict PM toxicity thresholds using existing data from the literature. A comprehensive dataset was constructed by aggregating PM toxicity test results from various sources. Key physico-chemical and exposure characteristics were extracted, and five ML models — logistic regression, support vector classifier, decision tree classifier, random forest classifier, and XGBoost — were trained and evaluated for their predictive performance.

The experimental results demonstrate that the ML-based approach effectively predicts PM toxicity thresholds. The tree-based models, particularly the Random Forest classifier, achieved strong performance in both the initial imbalanced data and after class weight adjustment. SHAP analysis provided insights into feature importance, helping to identify which PM characteristics most influence toxicity predictions.

3.9.2. Key Findings

The most important conclusions from this study are:

1. **Effectiveness of ML models** – The ML-based approach is effective in predicting PM toxicity thresholds, offering a faster and more scalable alternative to traditional laboratory tests.
2. **Importance of data quality** – Building a comprehensive dataset from existing literature is essential for model success. More data and standardized characteristics are needed, suggesting the value of an open-source dataset for broader research use.
3. **Handling imbalanced data** – The developed models showed good performance even in imbalanced classification scenarios, but further improvements are possible through more advanced algorithms.
4. **Feature interpretability** – SHAP analysis enhanced understanding of the impact of different PM characteristics on toxicity, supporting better feature selection and model interpretation.
5. **Future potential** – The data-driven approach marks a significant improvement over classical methods, with potential for further refinement through expanded data collection and more sophisticated predictive models.

3.9.3. Limitations and Future Work

This study is one of the first to use ML techniques to predict and classify PM toxicity thresholds based on existing data. However, several limitations should be addressed:

- **Data limitations** – The study relies on existing data, which may be incomplete or inconsistent in terms of physico-chemical and exposure characteristics, affecting the generalizability of the models.
- **Imbalanced classification** – Although addressed to some extent, data imbalance remains a challenge, especially with limited data availability.
- **Model complexity** – While effective, the models may benefit from further refinement and the integration of more complex algorithms to better capture the complexity of PM toxicity and account for uncertainty.

- **Data and feature uncertainty** – The dataset exhibits significant uncertainty in the input features themselves, with missing values exceeding 50% for several key physico-chemical characteristics (size distribution, aerodynamic diameter, agglomeration state). Additionally, inconsistencies in measurement units, analytical methods, and reporting standards across different studies introduce systematic uncertainties that propagate through the model predictions. This feature-level uncertainty, combined with the use of -1 values to represent missing compositional data, creates ambiguity about whether unreported values represent true zeros or simply unmeasured parameters.
- **Uncertainty quantification** – The current models provide deterministic predictions without confidence intervals or uncertainty estimates. This limitation is particularly critical for regulatory applications where decision-makers need to understand the reliability and confidence bounds of toxicity predictions. The absence of uncertainty quantification makes it difficult to assess prediction reliability, especially for PM samples with characteristics outside the training distribution.

Future research should focus on expanding and standardizing the dataset, ideally through an open-source repository where researchers can contribute new data. Addressing data and feature uncertainty should be a priority, potentially through the development of standardized measurement protocols, harmonized reporting guidelines, and imputation methods that explicitly account for the uncertainty introduced by missing or inconsistent data. Techniques such as multiple imputation, Bayesian approaches for handling missing data, or uncertainty-aware feature engineering could help mitigate the impact of incomplete information on model reliability. Exploring more advanced ML techniques, such as deep learning and ensemble methods, could further improve model performance, particularly in handling imbalanced data and managing uncertainty. Implementing uncertainty quantification methods, such as Bayesian neural networks, Monte Carlo dropout, or ensemble-based approaches, would provide crucial confidence estimates for each prediction. These uncertainty measures could help identify when models are making predictions on out-of-distribution samples or when additional laboratory validation may be warranted. Furthermore, developing probabilistic models that output prediction intervals rather than point estimates would be invaluable for risk assessment frameworks, allowing regulators to make more informed decisions based on both predicted toxicity and associated confidence levels. Integrating real-time environmental data may also enhance model accuracy and adaptability to diverse contexts. Additionally,

future work should explore the development of uncertainty-aware decision support systems that can flag high-uncertainty predictions for manual review or additional experimental validation. This hybrid approach would combine the efficiency of ML predictions with the reliability of traditional laboratory methods, ensuring that critical toxicity assessments maintain appropriate safety margins while leveraging the scalability advantages of data-driven approaches. By addressing these challenges and refining predictive models, including robust uncertainty quantification and systematic approaches to handling data uncertainty, future work can significantly improve our understanding of PM toxicity and its impact on human health.

3.9.4. Ongoing Work

Building upon the findings and limitations identified in this study, we are currently exploring advanced decision-making frameworks to enhance the reliability and interpretability of PM toxicity predictions. Specifically, our ongoing research focuses on integrating fuzzy logic and three-way decision (3WD) theory to address the inherent uncertainties and ambiguities in toxicity classification. Three-way decision theory, introduced by Yao [Yiyu Yao 2010, 2012], is closely related to granular computing (Section: 2.1) and extends traditional binary classification by introducing a third decision option: deferment or non-commitment. Instead of forcing a binary choice between "toxic" and "non-toxic," 3WD allows for an intermediate category of "uncertain" or "requires further investigation" [Yiyu Yao 2018]. This connection to granular computing enables the framework to work with information at different levels of granularity, allowing for hierarchical decision-making where uncertainty can be managed across multiple scales of data abstraction [Yiyu Yao 2018]. This approach is particularly valuable in high-stakes domains where incorrect classifications can have severe consequences.

Key advantages of 3WD include: (1) **Risk reduction** – by deferring uncertain decisions, the approach minimizes the cost of misclassification; (2) **Explicit uncertainty handling** – the framework formally incorporates uncertainty into the decision process rather than forcing binary outcomes; (3) **Regulatory compatibility** – the "defer for further investigation" option aligns well with regulatory practices where additional testing may be warranted; and (4) **Cost-effectiveness** – by identifying cases that require additional laboratory validation, resources can be allocated more efficiently. In our ongoing work, we are developing a hybrid framework that combines fuzzy logic to handle the imprecision in PM compositional

data with 3WD theory to manage classification uncertainty. This approach will enable the system to identify PM samples that fall into borderline toxicity regions, flagging them for additional experimental validation while providing confident predictions for clear-cut cases. The integration of these methodologies aims to create a more robust and practically applicable decision support system for environmental health professionals and regulatory agencies. Preliminary results suggest that this approach can significantly reduce false negative rates while maintaining computational efficiency, making it particularly suitable for real-time environmental monitoring applications. The framework also provides transparent decision rationales, which is crucial for regulatory acceptance and scientific credibility.

4. An Explainable Granular Computing Approach for Air Quality Classification Through Rule Extraction

Building upon the methodological foundations established in Chapter 2 and the successful application of machine learning for PM toxicity prediction demonstrated in Chapter 3, this chapter addresses a complementary yet distinct challenge in environmental intelligence: the development of transparent and interpretable classification systems for real-time air quality assessment. While the previous chapter leveraged SHAP analysis to explain black-box model predictions for toxicity thresholds, the operational deployment of air quality monitoring systems requires a fundamentally different approach—one that generates inherently interpretable decision rules that can be directly understood and validated by environmental scientists, policymakers, and public health officials without requiring specialized machine learning expertise. Traditional air quality classification models, despite achieving high predictive accuracy, often function as opaque "black boxes" that provide limited insight into their decision-making processes, creating a significant barrier in domains where understanding the rationale behind predictions is as crucial as the predictions themselves. This chapter introduces a novel granular computing approach that addresses this interpretability challenge by systematically extracting human-readable classification rules that explicitly link meteorological conditions to $PM_{2.5}$ concentration levels. Through the development of the AirQ-RuleGrCEx algorithm, we demonstrate how granular computing's natural ability to organize information into meaningful hierarchical structures can bridge the gap between predictive performance and explanatory transparency. By mimicking human cognitive processes that naturally decompose complex environmental phenomena into understandable patterns, this approach enables the creation of decision support systems that not only classify air quality levels accurately but also provide clear, actionable insights into the specific

meteorological conditions that drive air pollution episodes, ultimately supporting more informed environmental management and public health interventions.

4.1. Background and Motivation

The complex nature of air pollution dynamics, characterized by intricate interactions between various pollutants and meteorological conditions, presents significant challenges for traditional modeling approaches. While advancements in artificial intelligence (AI) and machine learning (ML) have enabled increasingly accurate predictions and classifications of air quality levels, many state-of-the-art models function as "black boxes," offering limited insight into their decision-making processes. This opacity poses a substantial barrier in domains where interpretability is paramount, such as environmental science and public health, where understanding the rationale behind predictions is as crucial as the predictions themselves.

Several machine learning approaches have been applied to air quality classification, including decision trees, random forests, support vector machines, and neural networks. While these methods often yield high accuracy, they frequently sacrifice transparency for performance. Decision trees offer some degree of interpretability but may lack robustness when handling complex data patterns. Ensemble methods like random forests improve accuracy but further diminish transparency. Neural networks, despite their impressive predictive capabilities, present perhaps the greatest challenge in terms of explainability. This trade-off between performance and interpretability represents a significant obstacle in the development of air quality classification systems that can effectively support decision-making processes.

Granular Computing (GrC) emerges as a promising approach to address these challenges. As a paradigm that mimics human cognitive processes by organizing information into meaningful units or "granules," GrC offers a natural framework for extracting interpretable rules. The fundamental principles of GrC align well with the needs of environmental science, where the ability to articulate clear relationships between variables is essential. By representing information at different levels of abstraction, GrC enables the development of models that not only predict air quality levels accurately but also provide transparent explanations for these predictions through rule extraction.

The approach leverages the inherent capacity of GrC to handle complex and imprecise information through the formation of granules or clusters, facilitating the capture of patterns and relationships within data. This granular representation simplifies complex information,

making it more manageable and interpretable—a critical advantage in the domain of air quality classification where datasets often exhibit intricate and complex patterns. The flexibility of GrC to represent data at multiple levels of abstraction allows for a more nuanced understanding of the relationships between meteorological features and air quality levels.

This chapter aims to explore the application of granular computing for extracting interpretable rules in air quality classification, with a specific focus on $PM_{2.5}$ levels. The objectives include:

1. Developing a novel approach leveraging GrC to extract meaningful classification rules that explain the relationships between meteorological conditions and air quality levels ($PM_{2.5}$ levels).

2. Demonstrating the effectiveness of the proposed approach through comprehensive experiments on real-world air quality data.

3. Comparing the performance of the GrC-based rule extraction model with machine learning algorithms in terms of accuracy, transparency, and reliability.

4. Analyzing the extracted rules to enhance understanding of the factors influencing air quality, thereby contributing to both the technical advancement of classification methodologies and practical applications in environmental management.

By addressing these objectives, this chapter contributes to advancing both the technical aspects of air quality classification and the broader goal of developing explainable AI systems for environmental monitoring and management.

4.2. Related Works

Air quality monitoring and classification have garnered significant attention in recent years due to increasing concerns about environmental health and its impact on human wellbeing. This section provides a structured overview of the existing approaches and methodologies in air quality prediction and classification, highlighting their strengths and limitations while positioning our granular computing rule extraction approach within the broader research landscape.

4.2.1. Traditional Modeling Approaches

Deterministic models have been widely employed to understand and predict air quality parameters. Among these, Gaussian dispersion models such as the Atmospheric Dispersion Modeling System (ADMS) [McHugh et al. 1997] have been particularly prevalent in Europe. These models operate under the assumption that pollutant dispersion follows a Gaussian distribution, with concentration decreasing from the source in a bell-shaped curve [Abdel-Rahman 2008].

More sophisticated approaches include 3D Eulerian chemistry-transport models, which combine Eulerian methods with detailed representations of atmospheric processes. Notable examples include CHIMERE (CHIMie-transport model for Emission and REgional scales) [Bessagnet et al. 2004; Menut et al. 2013] and CMAQ (Community Multiscale Air Quality) [Binkowski and Roselle 2003]. While these deterministic models provide valuable insights into atmospheric dynamics, they often struggle to capture the full complexity of air quality variations due to inherent uncertainties and limitations in representing all atmospheric processes.

4.2.2. Air Quality Forecasting Methods

The limitations of deterministic models have led to increased adoption of statistical and data-driven approaches, particularly machine learning techniques that can effectively handle complex relationships in environmental data.

A. Statistical and Traditional Machine Learning Methods

Statistical regression models establish relationships between air pollutant concentrations and relevant predictors, including meteorological parameters and historical pollution data [A. Kumar and Pramila Goyal 2011; Slini et al. 2002]. Time series analysis techniques, particularly autoregressive integrated moving average (ARIMA) models [Abhilash et al. 2018; U. Kumar and V. Jain 2010], have been employed to capture temporal patterns in air quality data.

Support Vector Machines (SVM) have shown promise in air quality forecasting, as demonstrated by [Moazami et al. 2016], who developed a support vector regression model for predicting carbon monoxide levels. Decision trees, random forests, and eXtreme Gradient Boosting (XGBoost) have also been widely applied to air quality forecasting tasks [Bozdağ

et al. 2020; Castelli et al. 2020; Lei et al. 2023; Jinghui Ma et al. 2020; Osowski and Garanty 2007], often achieving high predictive accuracy.

B. Deep Learning Approaches

Recent advances in deep learning have led to the emergence of more sophisticated models for air quality prediction. Artificial Neural Networks (ANNs) have been implemented by several researchers [Chelani et al. 2002; A. Kumar and Piyush Goyal 2013; Niska et al. 2004], with [Noori, Hoshyaripour, et al. 2010] proposing a deep learning model for predicting daily carbon monoxide concentrations using both ANN and adaptive neuro-fuzzy inference systems.

Recurrent Neural Networks (RNNs) [Biancofiore et al. 2017; Eren et al. 2023; Zaini et al. 2022] and more advanced architectures have further pushed the boundaries of air quality forecasting capabilities. [J. Wang and G. Song 2018] proposed a deep spatial-temporal ensemble model combined with LSTM networks, while [H. Liu et al. 2019] developed an intelligent hybrid model for forecasting multiple pollutant concentrations. [S. Du et al. 2019] introduced a deep learning approach based on 1D-CNN and Bi-directional LSTM specifically for $PM_{2.5}$ forecasting.

4.2.3. Air Quality Classification Methods

While numerous studies have focused on forecasting air pollutant concentrations, comparatively fewer have addressed the classification of air quality levels, which provides a more accessible framework for decision-making and public communication. Table 4.1 provides a comprehensive overview of existing air quality classification studies.

This literature review establishes a foundation for understanding the current landscape of air quality classification and rule extraction methodologies. As Table 4.1 demonstrates, there is a significant gap in research concerning interpretable air quality classification models and the extraction of meaningful rules that explain pollutant levels. A critical limitation of existing studies is the widespread omission of meteorological features in model development, despite their established significant impact on air quality levels [Jhun et al. 2015]. Furthermore, the field is dominated by complex, black-box approaches that lack transparency and interpretability—key requirements for environmental decision-making and policy formation. In contrast, granular computing offers a promising alternative by generating

Reference	Methodology	Research Objective	Data Source	Limitations
[Kujaroentavon et al. 2014]	Decision tree	Establish classification rules for air quality index (AQI)	Air quality data from Thailand's pollution control department (2012-2013)	Meteorological factors excluded; unclear contribution and results explanation
[Sugiarto and Sustika 2016]	Decision Tree (C4.5 algorithm)	Develop air quality classification using entropy and information gain for decision tree construction	Data collected from sensor nodes	Limited training dataset; restricted input parameters
[Corani and Scanagatta 2016]	Multi-label Bayesian networks	Simultaneous prediction of multiple air pollution variables ($PM_{2.5}$ and Ozone)	Shanghai dataset (Feb 2013-Feb 2014, 10 stations); Berlin ozone data (1997-1999, single station); Burgas ozone dataset (208 daily recordings)	None identified
[Gore and Deshpande 2017]	Naive Bayes and Decision tree J48	Classify AQI categories based on four pollutants	U.S. Pollution Data	Meteorological factors excluded; unclear contribution; substandard model quality
[Aggarwal et al. 2017]	Fuzzy logic and fuzzy interface system	Develop fuzzy interface system for AQI calculation using $PM_{2.5}$ and PM_{10} with six linguistic variables each	Five-day data from open source	Limited to only two pollutants ($PM_{2.5}$ and PM_{10})
[Teologo et al. 2018]	Fuzzy logic and Mamdani fuzzy inference	Classify AQI using fuzzy logic system with CO and NO_2	Data from Philippines air quality monitoring portal	Limited to only two pollutants (CO and NO_2)
[Xiaosong Zhao et al. 2018]	SVM, Random forest, and RNN	Predict daily air quality classification for three U.S. cities using RNN	U.S. EPA data (Jan 2010-Dec 2015, 2,191 observations)	Meteorological factors excluded; non-interpretable results
[Hamami and Fithriyah 2020]	Artificial neural network	Develop neural network for classifying three air pollution levels	IoT dataset from Open Data Jakarta (10 attributes, 1,827 records)	Meteorological factors excluded; unclear contribution and results
[Mangayarkarasi et al. 2021]	Logistic Regression and Random Forest	Classify AQI categories	World Air Quality Index historical data	Meteorological factors excluded; unclear contribution and results
[Haq 2022]	SMOTEDNN, XGBoost, Random Forest, SVM, k-NN	Develop and compare novel SMOTEDNN model against four ML models for air pollution classification	NAMP program dataset (Jan 2015-Jul 2020)	Meteorological factors excluded; unclear contribution; suspiciously perfect results
[Hamami and Dahlan 2022]	Logistic Regression, KNN, Decision Tree, Random Forest	Classify air quality into three categories	Jakarta's open data (12 months, multiple attributes)	Meteorological factors excluded; unclear contribution and results
[Saminathan and Malathy 2023]	Logistic Regression, SVM, Random Forest, XGBoost, Multi-layer perceptron	Classify $PM_{2.5}$ values into different categories	UCI Machine Learning Repository 2017	Application of existing approaches without innovation; unclear methodology and result interpretation

Table 4.1.: Comprehensive literature review of air quality classification methods.

coherent and understandable information units that align well with the multifaceted nature of air quality assessment. This approach enables the development of models that are both accurate and interpretable, addressing a critical need in both machine learning applications and environmental science research.

4.2.4. GrC for Rule Extraction Applications

The application of granular computing (GrC) for classification rule induction spans diverse scientific domains. In medical diagnostics, Rozehkhani et al. [Rozehkhani and Mohammadzad 2022] implemented GrC to classify COVID-19 patients based on symptomatic patterns, while in computational resource management, they developed rule extraction models to optimize virtual machine allocation [Rozehkhani and Mahan 2022]. Seismic vulnerability assessment has benefited from GrC applications in several studies [Khamespanah et al. 2013; Samadi Alinia and Delavar 2011; Hossein Sheikhian et al. 2017]. In environmental engineering, Ghiasi et al. [Ghiasi et al. 2022] integrated GrC with artificial neural networks (GrC-ANN) to create robust predictive models for pollutant dispersion in aquatic environments, while Noori et al. employed GrC techniques both for predicting scour hole dimensions downstream of ski-jump buckets [Noori, Hossien Sheikhian, et al. 2017] and for pioneering more accurate estimation methods for dispersion coefficients in natural river systems [Noori, Ghiasi, et al. 2017]. These diverse applications highlight GrC's versatility and effectiveness in extracting interpretable classification rules across scientific disciplines, from healthcare and geophysics to environmental modeling and computational resource management.

4.2.5. Research Gap

Despite the demonstrated potential of GrC in various domains, its application to air quality classification remains largely unexplored. Most existing air quality classification studies employ complex "black-box" models that lack interpretability, while those that offer some degree of transparency often do not integrate important meteorological features that significantly influence pollution levels.

Furthermore, the literature reveals insufficient attention to extracting interpretable rules that explain the relationships between meteorological conditions and air quality levels. This gap is particularly significant given the importance of transparent decision-making in environmental monitoring and public health applications.

Our work addresses these limitations by developing a novel GrC-based approach for extracting interpretable classification rules from air quality data, incorporating meteorological variables as key features. By minimizing entropy in the extracted rules, our method prioritizes both accuracy and interpretability, offering a valuable contribution to both the technical advancement of classification methodologies and practical applications in environmental management.

4.3. Proposed Methodology

This section introduces our methodology for extracting classification rules for $PM_{2.5}$ air quality levels (categorized as Good, Poor, and Extremely Poor). Figure 4.1 illustrates the comprehensive workflow of our approach.

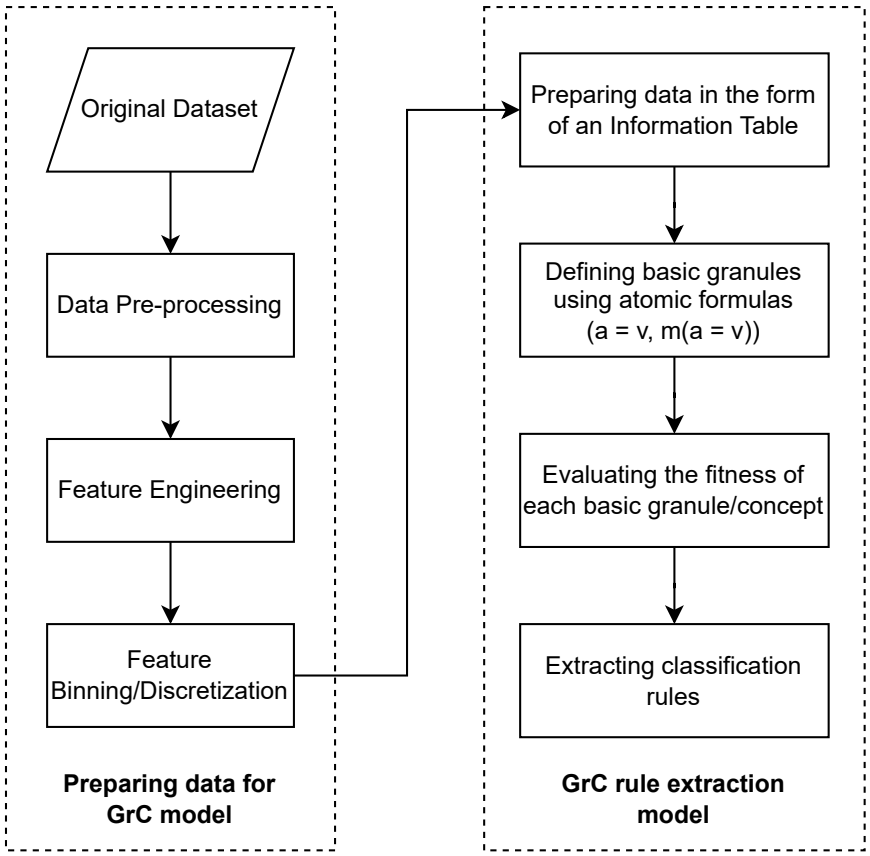


Figure 4.1.: Overview of the proposed Granular Computing (GrC) methodology.

As depicted in Figure 4.1, our framework consists of two primary components. The first component focuses on data preprocessing to ensure compatibility with the GrC model requirements. The second component encompasses three critical stages: (1) formulation of basic granules using 'attribute = value' structures, (2) quantitative assessment of granule/concept fitness through appropriate metrics, and (3) rule extraction using our novel heuristic algorithm designed to optimize both entropy minimization and coverage maximization. This final stage represents the most sophisticated aspect of our methodology, where we balance classification precision with rule generalizability.

4.3.1. Data Preparation and Information Table Construction

This study utilizes the Beijing $PM_{2.5}$ dataset (Section: 2.7.2), which comprises hourly measurements of $PM_{2.5}$ concentrations alongside seven meteorological variables: temperature, pressure, dew point temperature, combined wind direction, cumulated wind speed, cumulated hours of snow, and cumulated hours of rain.

A. Data Preprocessing

Our data preprocessing workflow began with a thorough examination of data quality. The $PM_{2.5}$ column was the only attribute containing missing values, with 2,067 instances (4.71% of the dataset) showing null values, which were subsequently removed from the analysis.

For feature selection, we focused on identifying the most relevant meteorological variables that influence $PM_{2.5}$ levels. Five key meteorological features were retained: dew point temperature (DEWP), temperature (TEMP), pressure (PRES), combined wind direction (CBWD), and cumulated wind speed (Iws). The cumulated hours of snow (Is) and rain (Ir) were excluded due to their extremely low variability, with 99% and 95% of values being zero, respectively.

B. Bivariate Analysis

Figure 4.2 presents a comprehensive bivariate analysis examining relationships between meteorological variables and $PM_{2.5}$ concentrations. The joint plots in Figures 4.2a, 4.2b, 4.2c, 4.2d, 4.2e, and 4.2f visualize associations between continuous meteorological variables and $PM_{2.5}$ levels through regression analysis. Additionally, Figures 4.2g and 4.2h utilize

strip and box plots to illustrate $PM_{2.5}$ distribution patterns across different wind direction categories.

C. Correlation Analysis

Figure 4.3 displays the pairwise Pearson correlation matrix for all variables in the dataset. Pearson's correlation coefficient, calculated using Equation 2.83 (Section: A), measures linear relationships between pairs of variables X and Y.

While strong correlations exist between certain meteorological variables (notably DEWP and PRES, TEMP and PRES), the analysis reveals relatively weak linear correlations between $PM_{2.5}$ and the meteorological features. However, this absence of strong linear relationships does not exclude the possibility of complex, non-linear associations between these variables, which motivates our application of granular computing for rule extraction.

D. Temporal Dependency Analysis

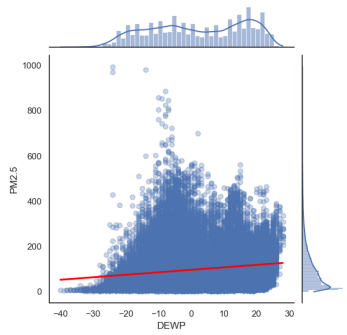
Recognizing that current $PM_{2.5}$ concentrations likely depend on previous values, we examined temporal autocorrelation patterns using both Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) analyses (Section: 2.2.4). ACF measures the correlation between observations at different time lags, including indirect effects, while PACF isolates the direct correlation between observations at specific time lags by controlling for intermediate observations. Figure 4.4 presents these analyses for 40 hourly lags of $PM_{2.5}$ concentrations.

The PACF analysis reveals a particularly strong correlation between current values and those from the previous hour (t-1), supporting our decision to incorporate a one-hour lag value of $PM_{2.5}$ as an additional feature in our model.

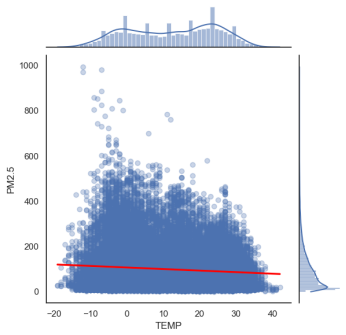
E. Feature Discretization

Our proposed GrC algorithm employs feature binning (discretization) to transform continuous numerical features into discrete categories. This approach simplifies the data representation, reduces the impact of outliers, and enhances interpretability by capturing potential non-linear relationships between features and the target variable. Table 4.2 details the discretization scheme applied to each variable.

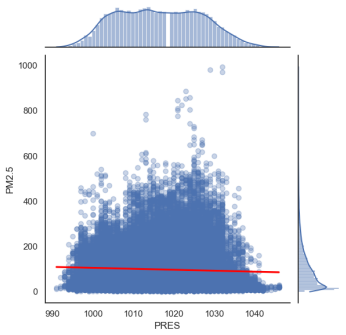
Table 4.3 presents a sample of the final information table constructed for the GrC model, incorporating six input features (One lag $PM_{2.5}$, TEMP, PRES, DEWP, Iws, and CBWD)



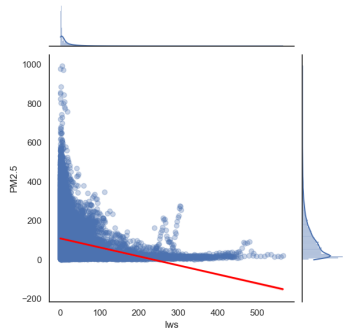
(a) Relationship between DEWP and $PM_{2.5}$.



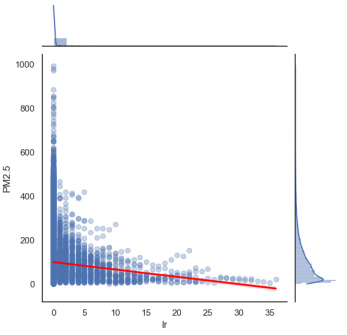
(b) Relationship between TEMP and $PM_{2.5}$.



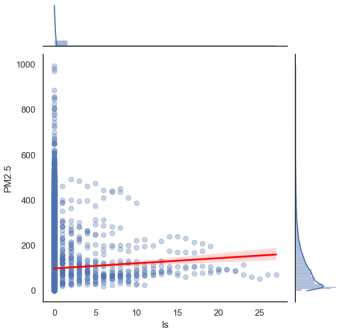
(c) Relationship between PRES and $PM_{2.5}$.



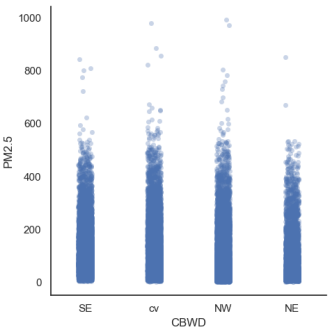
(d) Relationship between Iws and $PM_{2.5}$.



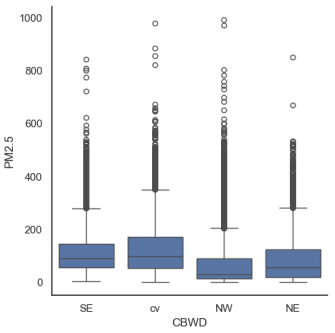
(e) Relationship between Ir and $PM_{2.5}$.



(f) Relationship between Is and $PM_{2.5}$.



(g) Relationship between CBWD and $PM_{2.5}$ (Strip plot).



(h) Relationship between CBWD and $PM_{2.5}$ (Box plot).

Figure 4.2.: Bivariate analysis between meteorological features and $PM_{2.5}$ concentrations.

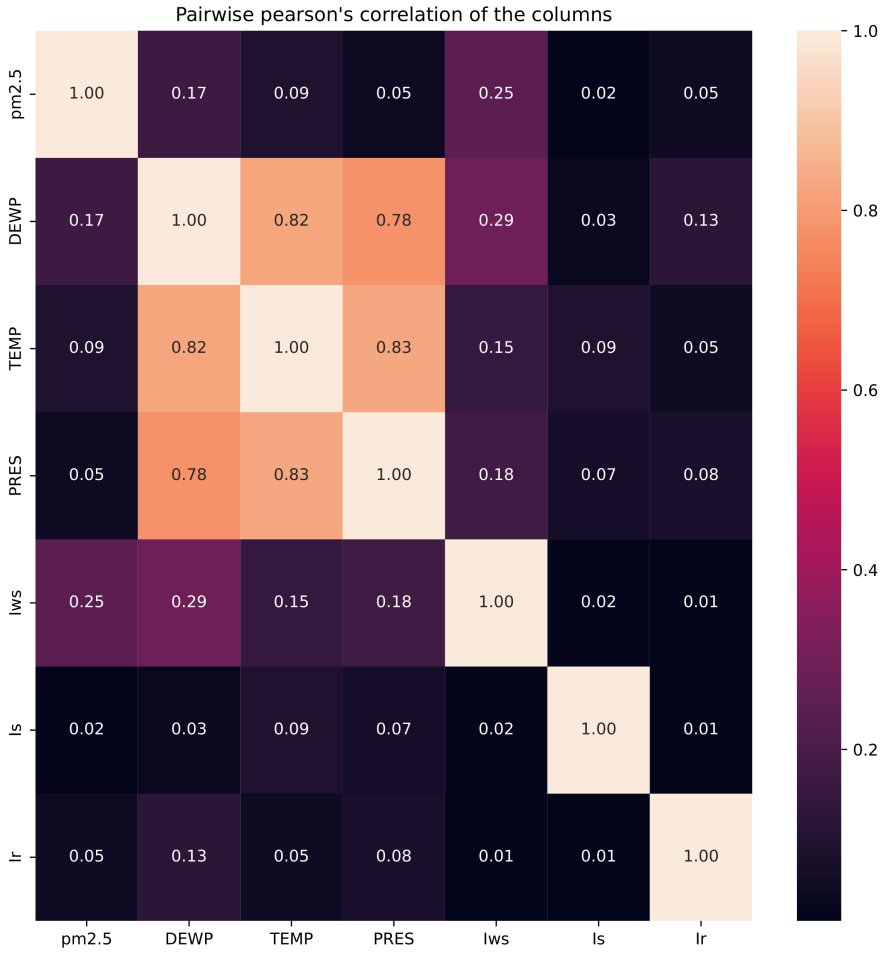


Figure 4.3.: Pairwise Pearson correlation matrix of dataset variables.

Temperature	Pressure	Dew-point	Wind Speed	Wind Direction	One Lag $PM_{2.5}$	$PM_{2.5}$
Very cold = $]-\infty, 0]$	Very low = $]-\infty, 980]$	Very dry = $]-\infty, 0]$	Calm = $[0, 0.5]$	NE	Good = $[0, 50]$	Good = $[0, 50]$
Cold = $]0, 10]$	Low = $]980, 1000]$	Dry = $]0,9]$	Light air = $]0.5, 1.5]$	SE	Poor = $]50, 300]$	Poor = $]50, 300]$
Cool = $]10, 20]$	Normal = $]1000, 1020]$	Comfortable = $]9, 15]$	Gentle breeze = $]1.5, 5]$	NW	Extremely poor = $]300, +\infty[$	Extremely poor = $]300, +\infty[$
Warm = $]20, 30]$	High = $]1020, 1030]$	Slightly uncomfortable = $]15, 20]$	Fresh breeze = $]5, 10.5]$	CV		
Hot = $]30, +\infty[$	Very high = $]1030, +\infty[$	Moderately uncomfortable = $]20, 23]$	Strong breeze = $]10.5, 13.5]$			
		Extremely uncomfortable = $]23, +\infty[$	Moderate gale = $]13.5, 20]$			
			Strong gale = $]20, 27]$			
			Violent storm = $]27, +\infty[$			

Table 4.2.: Discretization scheme for the attributes in the information table.

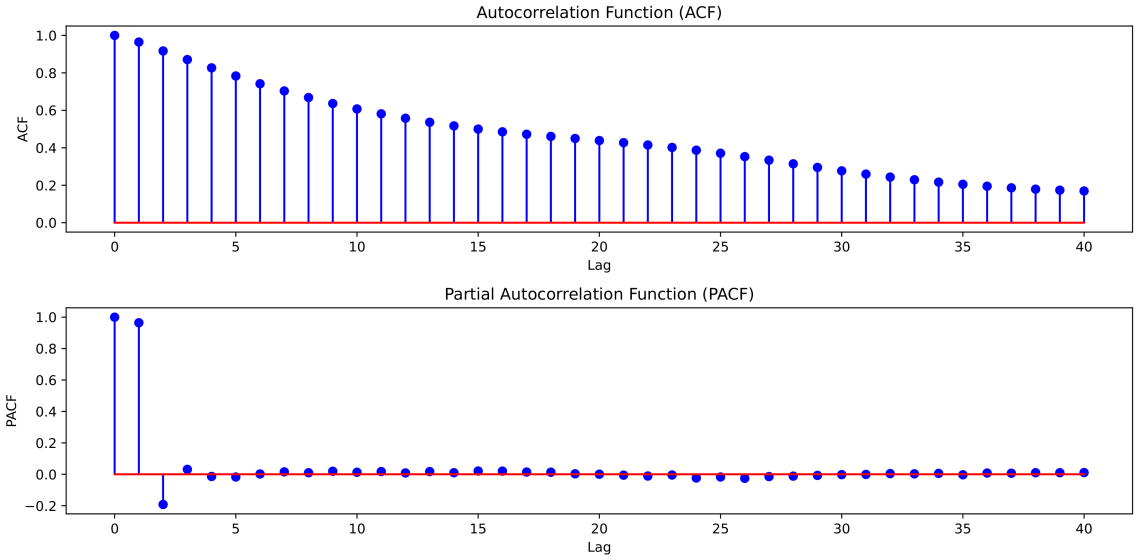


Figure 4.4.: ACF and PACF plots for $PM_{2.5}$ with 40 hourly lags.

and the target variable ($PM_{2.5}$ level).

Row ID	One lag $PM_{2.5}$	TEMP	PRES	DEWP	Iws	CBWD	$PM_{2.5}$ Level
821	Good	Cool	Low	Slightly Uncomfortable	Gentle Breeze	CV	Good
1123	Good	Cool	Normal	Slightly Uncomfortable	Moderate Gale	NW	Good
213	Poor	Warm	Normal	Moderately Uncomfortable	Gentle Breeze	CV	Poor
1182	Poor	Hot	Normal	Comfortable	Light Air	CV	Poor
136	Poor	Very cold	High	Very Dry	Gentle Breeze	CV	Poor
773	Poor	Cold	Normal	Very Dry	Strong Breeze	CV	Poor
1071	Good	Cool	Normal	Very Dry	Light Air	NE	Good
755	Poor	Hot	Low	Very Dry	Violent Storm	NW	Good
477	Good	Warm	Low	Slightly Uncomfortable	Light Air	CV	Poor
387	Poor	Cool	High	Very Dry	Strong Breeze	NE	Good

Table 4.3.: Sample rows from the final information table.

The data preparation phase establishes a robust foundation for our subsequent analyses. Through rigorous data cleaning, strategic feature selection, and thoughtful discretization, we have constructed an information table optimized for granular computing rule extraction algorithm. The inclusion of lagged $PM_{2.5}$ values captures important temporal dependencies, while the discretization scheme enhances interpretability. These preprocessing steps enable us to extract meaningful patterns and relationships between meteorological conditions and

$PM_{2.5}$ concentrations, ultimately facilitating the development of an interpretable rule-based model for air quality classification.

4.3.2. The AirQ-RuleGrCEx Algorithm: Air Quality Rule Extraction through Granular Computing

Following the data preparation phase, we now present our heuristic algorithm for extracting interpretable classification rules from the structured information table.

A. Basic Granules and Atomic Formulas

The first step in our approach involves constructing basic granules from the information table using atomic formulas expressed as pairs $(attribute = value, m(attribute = value))$, where $m(attribute = value)$ represents the collection of objects satisfying the condition $attribute = value$. It is important to note that while our implementation employs crisp sets for granulation as defined in Table 4.2, fuzzy sets could provide an alternative approach for future refinements.

The second step involves quantitatively evaluating these basic granules through four key metrics: generality, confidence, coverage, and entropy (defined in Equations 2.8, 2.9, 2.10, and 2.11). Table 4.4 displays the basic granules derived from the sample data in Table 4.3, while Table 4.5 presents the corresponding measurements for each formula/granule.

Our rule extraction algorithm operates by maintaining two critical sets. The first is the covering solution set (Equation 4.1), which is initially empty and gradually populated with classified objects. The second is the remaining objects set (Equation 4.2), which initially contains all objects from the information table and is progressively reduced as objects are classified.

$$covering\_solution = \{\emptyset\} \tag{4.1}$$

$$remaining\_objects = set(information\_table) - set(covering\_solution) \tag{4.2}$$

Two key metrics guide our rule selection process: conditional entropy (Equation 2.11) and generality (Equation 2.8). Entropy quantifies a granule's class purity—lower values indicate less impurity, with zero entropy signifying that all objects in the granule belong to the same class. Generality measures the proportion of total objects contained in a given granule.

Formula	Granule
One lag $PM_{2.5}$ = Good	[821, 1123, 1071, 477]
One lag $PM_{2.5}$ = Poor	[213, 1182, 136, 773, 755, 387]
TEMP = Cool	[821, 1123, 1071, 387]
TEMP = Warm	[213, 477]
TEMP = HOT	[1182, 755]
TEMP = Very cold	[136]
TEMP = Cold	[773]
PRES = Low	[821, 755, 477]
PRES = Normal	[1123, 213, 1182, 773, 1071]
PRES = High	[136, 387]
DEWP = Slightly uncomfortable	[821, 1123, 477]
DEWP = Moderately uncomfortable	[213]
DEWP = Comfortable	[1182]
DEWP = Very dry	[136, 773, 1071, 755, 387]
Iws = Gentle breeze	[821, 213, 136]
Iws = Moderate gale	[1123]
Iws = Light air	[1182, 1071, 477]
Iws = Strong breeze	[773, 387]
Iws = Violent storm	[755]
CBWD = CV	[821, 213, 1182, 136, 773, 477]
CBWD = NW	[1123, 755]
CBWD = NE	[1071, 387]

Table 4.4.: Basic granules derived from the sample data in Table 4.3.

Formula	Granule	Generality	Confidence			Coverage			Entropy
			Class 0	Class 1	Class 2	Class 0	Class 1	Class 2	
One lag $PM_{2.5}$ = Good	[821, 1123, 1071, 477]	0.4	0.75	0.25	0.0	0.6	0.2	0.0	0.244
One lag $PM_{2.5}$ = Poor	[213, 1182, 136, 773, 755, 387]	0.6	0.333	0.666	0.0	0.4	0.8	0.0	0.27
TEMP = Cool	[821, 1123, 1071, 387]	0.4	1.0	0.0	0.0	0.8	0.0	0.0	0.0
TEMP = Warm	[213, 477]	0.2	0.0	1.0	0.0	0.0	0.4	0.0	0.0
TEMP = HOT	[1182, 755]	0.2	0.5	0.5	0.0	0.2	0.2	0.0	0.30
TEMP = Very cold	[136]	0.1	0.0	1.0	0.0	0.0	0.2	0.0	0.0
TEMP = Cold	[773]	0.1	0.0	1.0	0.0	0.0	0.2	0.0	0.0
PRES = Low	[821, 755, 477]	0.3	0.66	0.33	0.0	0.4	0.2	0.0	0.27
PRES = Normal	[1123, 213, 1182, 773, 1071]	0.5	0.4	0.60	0.0	0.4	0.6	0.0	0.29
PRES = High	[136, 387]	0.2	0.50	0.5	0.0	0.2	0.2	0.0	0.30
DEWP = Slightly uncomfortable	[821, 1123, 477]	0.3	0.66	0.33	0.0	0.4	0.2	0.0	0.27
DEWP = Moderately uncomfortable	[213]	0.1	0.00	1.00	0.0	0.0	0.2	0.0	0.0
DEWP = Comfortable	[1182]	0.1	0.0	1.0	0.0	0.0	0.2	0.0	0.00
DEWP = Very dry	[136, 773, 1071, 755, 387]	0.4	0.50	0.50	0.0	0.4	0.4	0.0	0.30
Iws = Gentle breeze	[821, 213, 136]	0.3	0.33	0.66	0.0	0.2	0.4	0.0	0.27
Iws = Moderate gale	[1123]	0.1	1.0	0.0	0.0	0.2	0.0	0.0	0.0
Iws = Light air	[1182, 1071, 477]	0.3	0.33	0.66	0.0	0.2	0.4	0.0	0.27
Iws = Strong breeze	[773, 387]	0.2	0.50	0.50	0.0	0.2	0.2	0.0	0.30
Iws = Violent storm	[755]	0.1	1.0	0.0	0.0	0.2	0.0	0.0	0.00
CBWD = CV	[821, 213, 1182, 136, 773, 477]	0.6	0.166	0.833	0.0	0.2	1.0	0.0	0.19
CBWD = NW	[1123, 755]	0.2	1.0	0.00	0.0	0.4	0.0	0.0	0.0
CBWD = NE	[1071, 387]	0.2	1.00	0.00	0.0	0.4	0.0	0.0	0.0

Table 4.5.: Quantitative evaluation of basic granules.

B. The AirQ-RuleGrCEx Algorithm

Our algorithm begins by establishing a minimum entropy threshold, which we set to zero to extract the highest-quality rules. The process first identifies all formulas/rules meeting this threshold and updates the covering solution accordingly by adding the objects classified by these rules. The remaining objects set is then updated to reflect these classifications.

After extracting all formulas meeting the minimum entropy threshold, the algorithm addresses any remaining unclassified objects. This presents a challenge: from the available basic formulas/granules, how do we select the optimal rule considering two competing criteria? First, we must avoid reclassifying already processed objects. Second, we want to identify rules that encompass the maximum number of remaining unclassified objects.

To resolve this challenge, we employ the Jaccard Index (Equation 4.3), which measures the similarity between two sets A and B, with values ranging from 0% to 100%. Higher percentages indicate greater similarity between sets. In our implementation, set A comprises the granules of the remaining formulas, while set B contains the remaining unclassified objects.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4.3}$$

This recursive process continues until all objects are classified. Algorithm 2 outlines the complete procedure for constructing a granular decision tree, while Figure 4.5 provides a detailed flowchart of the rule extraction process.

Algorithm 2 The AirQ-RuleGrCEx algorithm

procedure GRANULENETBUILDING(inf_table)
Get the information table inf_table
Construct the family of basic concept with respect to atomic formulas (a = v, m(a = v))
Compute fitness (generality, confidence, coverage, and entropy) for the basic granules/formulas
Get the granules/formulas with min-entropy value (granules_min_entropy)
for *granule* in *granules_min_entropy* **do**
if granule['entropy_val'] $\neq$ 0 **then**
infTable $\leftarrow$ information_table[granule]
GranuleNetBuilding(infTable)
else
Update the covering_solution by adding the objects in the *granule*
Update the remaining_objs
remaining_objs $\leftarrow$ inf_table - covering_solution
while *remaining_objs* $\neq \{\emptyset\}$ **do**
Get the granule that has the highest Jaccard Index value with the remaining objects '*remaining_objs*' as granule
Update covering solution by adding the objects in the granule that has the highest Jaccard index with the remaining objects '*remaining_objs*'
Update the reaming objects '*remaining_objs*'
infTable $\leftarrow$ information_table[granule]
GranuleNetBuilding(infTable)

When applied to our sample data (Tables 4.3, 4.4, and 4.5), Algorithm 2 generates a granular tree as illustrated in Figure 4.6. This tree-like structure provides an interpretable

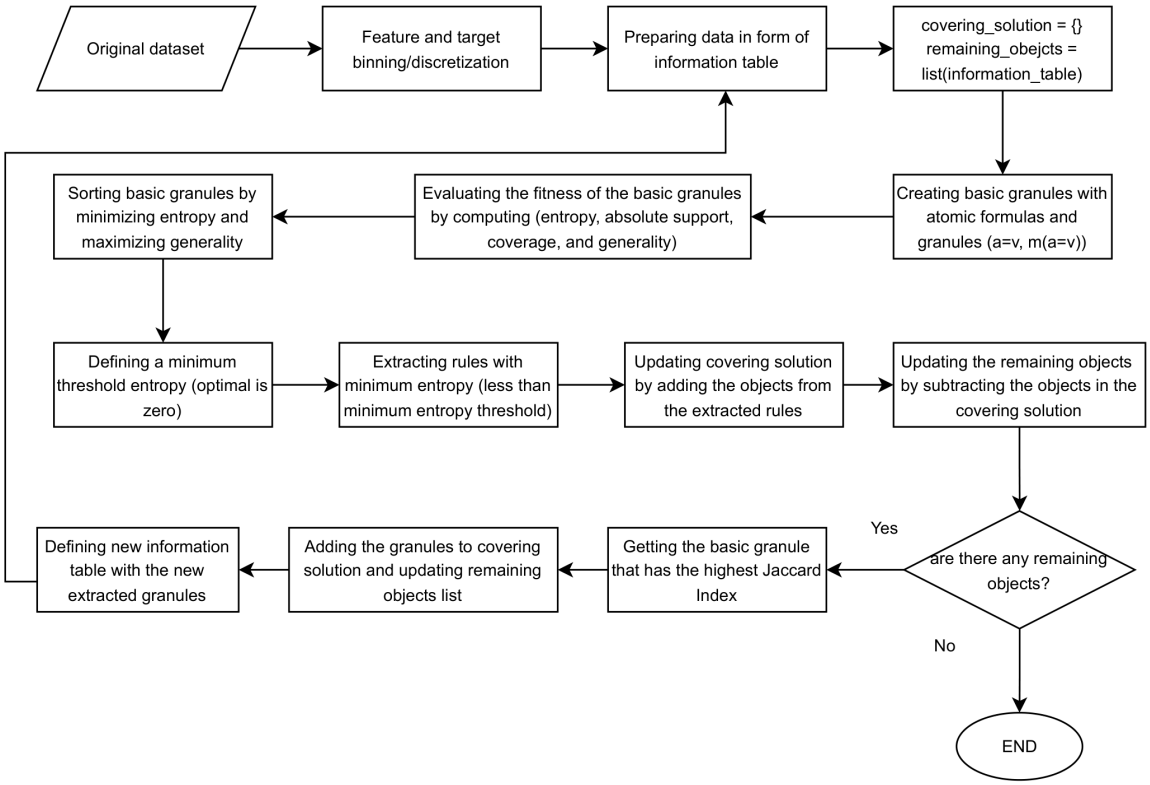


Figure 4.5.: Flowchart of the AirQ-RuleGrCEx algorithm.

flowchart with minimal entropy values, offering clear decision paths for classifying $PM_{2.5}$ levels based on meteorological conditions.

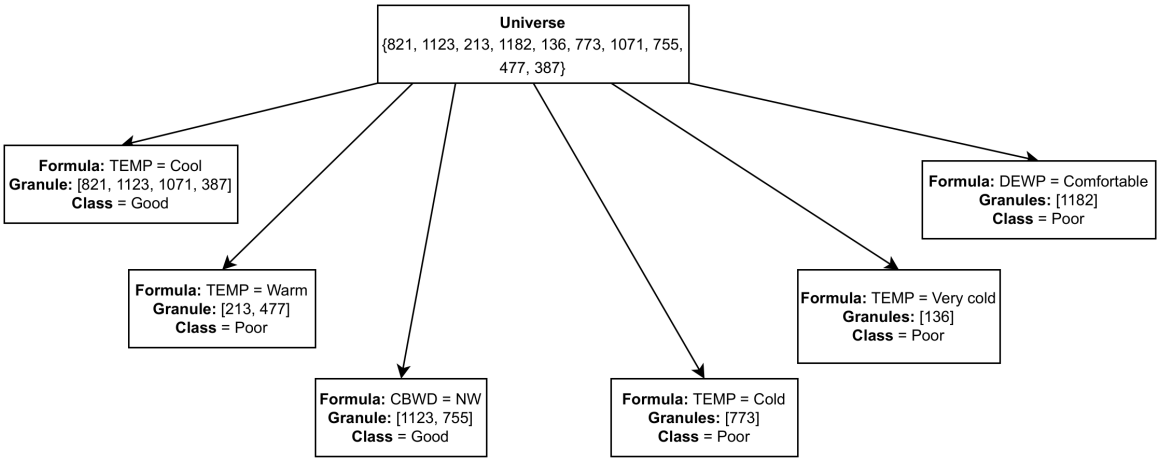


Figure 4.6.: Granular tree generated from the sample data.

The AirQ-RuleGrCEx methodology presents a comprehensive approach to extracting interpretable classification rules for air quality assessment. Beginning with careful data preparation and feature discretization, the process continues with the systematic creation of basic granules expressed as attribute-value pairs. These granules are evaluated using fitness metrics including generality, confidence, coverage, and entropy.

Our innovative heuristic algorithm introduces several key methodological advances that distinguish it from existing GrC approaches. First, the algorithm implements an advanced rule extraction model that exclusively prioritizes granules with zero entropy, ensuring maximum class purity and eliminating ambiguous classification boundaries that could compromise decision reliability. This strict entropy criterion guarantees that each extracted rule provides unambiguous classification guidance.

Second, the methodology incorporates an intelligent granule selection mechanism that systematically addresses remaining unclassified objects while minimizing redundancy with previously classified instances. This is achieved through Jaccard Index optimization, which measures the similarity between sets of remaining objects and candidate granules, ensuring efficient coverage without unnecessary overlap. The algorithm recursively applies this selection process until all training instances are classified, maintaining both completeness and computational efficiency.

Third, the approach includes a robust fallback strategy for handling test instances that do not match any extracted rules during the prediction phase. This challenge, commonly overlooked in existing GrC literature, is addressed through hierarchical rule relaxation and similarity-based classification mechanisms that ensure comprehensive coverage of the feature space. When a test instance falls outside the scope of extracted rules, the system employs a graduated relaxation process that progressively broadens rule conditions until a suitable match is found, maintaining classification consistency while preserving interpretability.

The result is a hierarchical, tree-structured model that provides transparent, interpretable rules for understanding the complex relationships between meteorological conditions and $PM_{2.5}$ concentrations. The combination of zero-entropy rule extraction, Jaccard Index-based object selection, and comprehensive fallback mechanisms creates a robust framework that addresses the primary limitations of existing GrC approaches: parameter sensitivity, incomplete coverage, and poor generalization to unseen data.

This approach offers significant advantages for environmental monitoring and policy-making contexts where model transparency and interpretability are essential. The methodological innovations ensure that the extracted rules are not only accurate but also comprehensive, providing decision-makers with reliable, interpretable guidance even for previously unseen environmental conditions. Future refinements could explore the integration of fuzzy sets for granulation or the adaptation of the algorithm to address temporal dependencies more explicitly.

4.4. Experiments and Results

4.4.1. Practical Applications

The AirQ-RuleGrCEx methodology represents an emerging tree-based machine learning approach for air quality classification that complements existing methods such as decision trees and random forests. By leveraging granular computing (GrC) for rule extraction, our approach provides a robust framework that significantly enhances air quality management systems.

The extracted interpretable rules offer clear insights into the complex relationships between meteorological conditions and air quality levels, empowering environmental monitoring initiatives with actionable intelligence. The adaptability of our GrC algorithm facilitates

seamless integration into existing air quality monitoring infrastructures, improving the accuracy of real-time assessments while providing policymakers and public health officials with transparent decision-support tools for implementing targeted interventions. This methodology constitutes a significant advancement with direct implications for effective air quality management and, by extension, community well-being.

4.4.2. Evaluation of the Proposed Model

We conducted comprehensive experiments to evaluate the effectiveness of our granular computing approach for air quality classification using the Beijing $PM_{2.5}$ dataset (Section: 2.7.2). The final dataset comprised 1,801 rows representing three $PM_{2.5}$ level categories (Good, Poor, and Extremely Poor), partitioned into training and testing sets using an 80-20 split.

From the 1,441 training instances, our algorithm extracted 588 classification rules. Table 4.6 presents a representative sample of these rules along with their corresponding outcomes.

Table 4.7 presents a comprehensive performance evaluation of our GrC model on both training and test sets. The model achieved 100% accuracy on the training set, successfully capturing all classification patterns. More importantly, it demonstrated strong generalization capability on the test set with 79% accuracy, indicating effective rule transfer to unseen data.

To rigorously assess model robustness and address potential overfitting concerns, we implemented k-fold cross-validation. We partitioned the dataset into five subsets (folds) and performed five training-evaluation iterations, using one fold for validation and the remaining four for training in each iteration. This approach provides a comprehensive assessment of the model's generalization capability across different data partitions. Table 4.8 presents the cross-validation results.

The 5-fold cross-validation results revealed consistent and promising performance even without addressing class imbalance issues. The model demonstrated remarkable stability across different data partitions, with accuracy ranging from 66% to 79%. This low variance in performance indicates that the model effectively learns underlying patterns rather than memorizing training data, confirming its robust generalization capability for unseen data.

Rule ID	Conditions	Outcome
264	PRES = Normal & $PM_{2.5}$_lagged_1 = Extremely poor & CBWD = NE & Iws = Gentle breeze & TEMP = Very cold	Poor
300	PRES = Low & $PM_{2.5}$_lagged_1 = Good & TEMP_level = Warm & CBWD = NW & DEWP = Dry & Iws = Moderate gale	Good
519	PRES = High & DEWP = Comfortable & TEMP = Warm & $PM_{2.5}$_lagged_1 = Good & CBWD = CV	Poor
212	PRES = Normal & $PM_{2.5}$_lagged_1 = Good & CBWD = CV & Iws = Gentle breeze & TEMP = Cold	Good
67	PRES_level = Normal & $PM_{2.5}$_lagged_1 = Poor & TEMP_level = Hot & CBWD = NW & DEWP = Comfortable & Iws = Moderate gale	Good
339	PRES = Low & $PM_{2.5}$_lagged_1 = Poor & TEMP = Warm & CBWD = SE & Iws = Moderate gale & Poor	
468	PRES = High & DEWP = Dry & $PM_{2.5}$_lagged_1 = Good & CBWD = NE & TEMP = Warm	Good
343	PRES = Low & $PM_{2.5}$_lagged_1 = Poor & TEMP = Warm & CBWD = SE & Iws = Violent storm	Poor
20	PRES = Normal & $PM_{2.5}$_lagged_1 = Poor & TEMP = Cool & DEWP =Dry & Iws = Fresh breeze	Poor
269	PRES_level=Normal & $PM_{2.5}$_lagged_1 = Extremely poor & CBWD = CV & TEMP = Warm & DEWP = Comfortable	Extremely poor

Table 4.6.: Representative extracted rules from the training set.

Model	Accuracy	Class	Metrics		
			Precision	Recall	F1-score
Training set (80%)	1.00	Good (Class 0)	1.00	1.00	1.00
		Poor (Class 1)	1.00	1.00	1.00
		Extremely poor (Class 2)	1.00	1.00	1.00
Test set (20%)	0.79	Good (Class 0)	0.82	0.81	0.81
		Poor (Class 1)	0.76	0.80	0.78
		Extremely poor (Class 2)	0.85	0.71	0.77

Table 4.7.: Performance metrics of the GrC model on training and test sets.

Folds	Accuracy	Class	Metrics		
			Precision	Recall	F1-score
1st fold	0.75	Good (Class 0)	0.82	0.77	0.79
		Poor (Class 1)	0.66	0.72	0.69
		Extremely poor (Class 2)	0.78	0.75	0.76
2nd fold	0.74	Good (Class 0)	0.78	0.79	0.79
		Poor (Class 1)	0.68	0.73	0.70
		Extremely poor (Class 2)	0.83	0.57	0.68
3rd fold	0.74	Good (Class 0)	0.74	0.81	0.77
		Poor (Class 1)	0.73	0.76	0.75
		Extremely poor (Class 2)	0.80	0.59	0.68
4th fold	0.79	Good (Class 0)	0.73	0.82	0.77
		Poor (Class 1)	0.85	0.77	0.81
		Extremely poor (Class 2)	0.00	0.00	0.00
5th fold	0.66	Good (Class 0)	0.69	0.57	0.63
		Poor (Class 1)	0.73	0.71	0.72
		Extremely poor (Class 2)	0.30	0.67	0.41

Table 4.8.: 5-fold cross-validation results for the AirQ-RuleGrCEx model.

4.4.3. Comparative Analysis with Machine Learning Models

We conducted a comprehensive comparison between our AirQ-RuleGrCEx approach and established machine learning models commonly used for air quality classification. This comparison evaluated both performance metrics and interpretability aspects against benchmark algorithms including decision tree classifier, random forest classifier, and CatBoost.

Decision tree classifiers recursively partition datasets based on features, creating interpretable tree-like structures where each leaf node represents a class. They excel in simplicity and interpretability but may be prone to overfitting. Random forest, an ensemble method, constructs multiple decision trees during training and outputs the mode of their classifications, reducing overfitting while increasing predictive accuracy through aggregation (Section: 2.3). CatBoost (Categorical Boosting) is a gradient boosting algorithm specifically designed for efficient handling of categorical features, offering robust performance with minimal hyperparameter tuning.

Table 4.9 presents a comparative analysis of these algorithms alongside our AirQ-RuleGrCEx approach, evaluating accuracy, precision, recall, F1-score, and interpretability.

Algorithm	Accuracy	Class	Metrics			Interpretability
			Precision	Recall	F1-score	
AirQ-RuleGrCEx	0.79	Good (Class 0)	0.82	0.81	0.81	Yes
		Poor (Class 1)	0.76	0.80	0.78	
		Extremely poor (Class 2)	0.85	0.71	0.77	
Decision Tree	0.79	Good (Class 0)	0.82	0.78	0.80	Yes
		Poor (Class 1)	0.75	0.80	0.78	
		Extremely poor (Class 2)	0.87	0.80	0.84	
Random Forest	0.84	Good (Class 0)	0.87	0.87	0.87	No
		Poor (Class 1)	0.82	0.84	0.83	
		Extremely poor (Class 2)	0.86	0.76	0.81	
CatBoost	0.86	Good (Class 0)	0.86	0.89	0.87	No
		Poor (Class 1)	0.88	0.80	0.84	
		Extremely poor (Class 2)	0.82	0.98	0.89	

Table 4.9.: Comparative analysis of air quality classification algorithms.

Our comparative analysis demonstrates that the AirQ-RuleGrCEx algorithm achieves competitive performance with widely used tree-based models. While ensemble methods like Random Forest and CatBoost exhibit marginally higher accuracy (84% and 86% respectively,

compared to 79% for our approach), they sacrifice interpretability for performance gains. Our approach matches the interpretability of decision trees while maintaining comparable classification performance.

Random Forest and CatBoost, while somewhat interpretable, offer significantly less transparency than our GrC model and individual decision trees. Random Forest's ensemble methodology aggregates predictions from multiple decision trees, while CatBoost's sequential tree-building process creates complex interdependencies that obscure decision pathways. In contrast, our AirQ-RuleGrCEx methodology provides explicit classification rules that directly link meteorological conditions to air quality outcomes, offering crucial transparency for environmental monitoring and policy applications where decision rationales are as important as predictive accuracy.

4.5. Discussion

In this comprehensive investigation of air quality classification methodologies, we examine the efficacy, interpretability, and practical implications of our proposed granular computing (GrC) approach. The experimental results reveal notable insights into both the strengths and limitations of GrC when compared to conventional machine learning algorithms in the context of environmental data analysis.

Our comparative analysis with established algorithms—including decision tree classifier, random forest, and CatBoost—demonstrates that the proposed GrC algorithm achieves competitive performance (79% accuracy) while offering distinct advantages in explainability. While advanced ensemble methods like Random Forest (84%) and CatBoost (86%) marginally outperform our approach in raw accuracy metrics, the GrC model strikes a more favorable balance between performance and interpretability—a critical consideration in environmental science applications where transparent decision-making processes are essential.

The most significant contribution of the GrC algorithm lies in its inherent interpretability. Unlike "black-box" models that obscure the reasoning behind predictions, our approach generates explicit, human-understandable rules that clearly explains the conditions leading to specific air quality classifications. These extracted rules provide valuable insights into the complex relationships between meteorological variables and $PM_{2.5}$ concentration levels, enabling domain experts to validate the model's reasoning against established environmental science knowledge. For instance, rules combining wind speed, temperature, and pressure

conditions yield meteorologically sound classifications that align with atmospheric dispersion principles.

The granular computing framework demonstrates particular effectiveness in handling the multifaceted nature of air quality dynamics. By conceptualizing data as coherent granules organized in hierarchical structures, the algorithm effectively captures the complex interdependencies between meteorological parameters and pollutant concentrations. This granular representation aligns well with the natural stratification observed in environmental systems, where different combinations of weather conditions create distinct air quality scenarios. The minimization of entropy in granule formation ensures that the resulting rules represent statistically significant patterns rather than random correlations.

A compelling aspect of our approach is its potential for knowledge transfer and rule generalization. The extracted rules represent fundamental relationships between meteorological variables and air quality levels that may transcend specific geographical and temporal contexts. This generalizability enhances the model's utility for environmental monitoring across diverse regions and conditions, potentially reducing the need for extensive location-specific retraining. Further investigation into the transferability of these rules across different urban environments would constitute a valuable extension of this research.

Despite these promising results, several challenges warrant consideration in future work. The computational complexity of granule formation and entropy minimization may impact scalability when applied to larger, high-dimensional datasets. Advanced optimization techniques or parallel processing implementations could address these efficiency concerns. Additionally, the current implementation's ability to simultaneously handle both categorical and continuous variables requires refinement, as does the development of mechanisms to mitigate potential overfitting in rule extraction. The introduction of pruning techniques or regularization mechanisms specifically designed for granular rule sets could enhance generalization performance.

The GrC approach also presents opportunities for integration with other advanced machine learning paradigms. Hybrid models combining the interpretability of granular computing with the predictive power of deep learning could leverage the complementary strengths of both methodologies. Similarly, the incorporation of fuzzy logic could enhance the model's ability to handle the inherent uncertainty and imprecision in environmental data measurements.

From an application perspective, the transparent nature of the GrC model makes it particularly suitable for deployment in environmental decision support systems. The clear

articulation of classification rules enables environmental managers and policymakers to understand the specific conditions that trigger poor air quality events, facilitating more targeted interventions and mitigation strategies. This transparency also enhances trust in the model's predictions, a crucial factor for operational adoption in public health and environmental management contexts.

In conclusion, our study establishes granular computing as a viable and valuable approach for air quality classification that effectively balances predictive performance with interpretability. The GrC algorithm, through its systematic extraction of meaningful classification rules, makes a significant contribution to the field of explainable AI in environmental monitoring. As air quality management continues to represent a critical environmental and public health challenge, our work provides a foundation for the development of transparent, accurate, and actionable prediction systems that can inform effective pollution control strategies and public health interventions.

4.6. Summary and Prospects

4.6.1. Summary of Key Findings

This research introduced a novel approach for air quality classification using granular computing (GrC) rule extraction, specifically targeting the classification of $PM_{2.5}$ concentration levels. Our comprehensive analysis revealed meaningful relationships between meteorological variables and air quality levels, demonstrating the following key findings:

- The proposed AirQ-RuleGrCEx algorithm successfully constructed a hierarchical granular tree that systematically organizes data into coherent and interpretable granules with minimal entropy.
- The extracted classification rules achieved 79% accuracy on unseen test data, comparable to traditional machine learning approaches like decision trees (79%), while providing greater interpretability than more complex models like Random Forest (84%) and CatBoost (86%).
- Five-fold cross-validation results confirmed the robustness of the model, with consistent performance across different data partitions, indicating good generalization capabilities.

- Meteorological variables including temperature, pressure, dew point, wind speed, and wind direction were identified as significant factors affecting $PM_{2.5}$ levels, with specific combinations of these variables leading to different air quality classifications.

4.6.2. Contributions to Explainable AI in Air Quality Classification

This research makes several significant contributions to the fields of explainable AI and environmental monitoring:

- **Enhanced Interpretability**: Unlike black-box models that prioritize accuracy over transparency, our GrC approach produces clear, human-understandable classification rules that explicitly define the conditions leading to specific air quality levels. This interpretability is crucial for environmental science applications where understanding the reasoning behind predictions is as important as the predictions themselves.
- **Novel Algorithm Development**: The AirQ-RuleGrCEx algorithm represents a methodological advancement in granular computing, particularly in its approach to minimizing entropy while extracting meaningful rules. The proposed method for handling multi-class classification problems demonstrates the versatility of granular computing techniques.
- **Balanced Performance**: The model achieves a favorable balance between accuracy and interpretability, making it particularly suitable for air quality management applications where transparency in decision-making is essential for policy implementation and public health interventions.
- **Framework for Environmental Data Analysis**: The research establishes a comprehensive framework for applying granular computing to environmental data, offering a systematic approach to data preparation, feature discretization, rule extraction, and model evaluation that can be adapted to similar environmental classification problems.

4.6.3. Potential Improvements and Future Research Directions

While the current research demonstrates the efficacy of granular computing for air quality classification, several promising directions for future work can enhance and extend this approach:

- **Extending GrC to Other Environmental Monitoring Tasks**:
  - Application to water quality classification, soil contamination assessment, and other environmental monitoring scenarios
  - Integration with remote sensing data for broader spatial coverage and environmental impact assessment
  - Development of domain-specific granulation strategies optimized for different environmental parameters
- **Hybrid Models Integrating GrC with Deep Learning**:
  - Creation of GrC-ANN (Artificial Neural Network) hybrid models that combine the interpretability of granular computing with the predictive power of deep learning
  - Exploration of knowledge distillation techniques to extract interpretable rules from more complex models
  - Development of attention mechanisms that focus on relevant granules for improved classification accuracy while maintaining interpretability
- **Deployment in Real-Time Air Quality Monitoring Systems**:
  - Optimization of the algorithm for real-time processing and classification of continuous air quality data streams
  - Integration with IoT sensor networks for distributed air quality monitoring and localized rule extraction
  - Development of adaptive rule systems that can evolve as environmental conditions change over time
  - Creation of explainable alert systems that not only predict poor air quality conditions but also explain the meteorological factors contributing to them
- **Methodological Improvements**:
  - Enhancement of the algorithm to handle imbalanced datasets more effectively
  - Development of techniques for dynamic discretization of continuous variables to optimize rule extraction

- Investigation of sequential three-way decision approaches within the granular computing framework to improve classification performance
- Exploration of fuzzy granular computing to better handle uncertainty in environmental data

The research presented in this chapter establishes a solid foundation for explainable air quality classification through granular computing. By pursuing these future directions, the approach can evolve into a comprehensive methodology for transparent, accurate, and actionable environmental monitoring systems that contribute to improved air quality management and public health protection.

4.6.4. Ongoing Work

Building upon the foundational AirQ-RuleGrCEx methodology, several promising research directions are currently under active development to further enhance the robustness, efficiency, and applicability of granular computing for environmental classification tasks.

Optimized Fitness Function Development: Current efforts focus on developing an enhanced fitness function that optimally balances the multiple evaluation criteria (generality, confidence, coverage, and conditional entropy) for rule selection. Rather than relying on predefined weight combinations, we are investigating adaptive weighting schemes that dynamically adjust based on dataset characteristics and classification complexity. This includes exploring multi-objective optimization approaches to identify non-dominated rule sets, as well as machine learning-based meta-optimization techniques that learn optimal weight configurations from training performance across diverse environmental datasets.

Similarity Metric Evaluation and Selection: Beyond the current Jaccard Index implementation, comprehensive experimentation is underway to evaluate alternative similarity measures for granule selection and object classification. The investigation encompasses cosine similarity for high-dimensional feature spaces, Hamming distance for categorical attribute handling, and Euclidean distance variants for continuous meteorological variables. Additionally, we are exploring hybrid similarity measures that combine multiple distance metrics weighted according to attribute types and importance, as well as context-aware similarity functions that adapt to local data density and classification boundary complexity.

Comprehensive Test Set Coverage Strategies: A critical ongoing development addresses the challenge of test instances that do not correspond to any extracted training

rules. Current work involves implementing a multi-tier fallback mechanism that includes: (1) graduated rule relaxation through systematic attribute removal based on importance rankings, (2) k-nearest neighbor classification using rule centroids as reference points, (3) probabilistic rule matching with confidence-weighted predictions, and (4) ensemble voting among multiple relaxed rules. This comprehensive approach ensures robust classification coverage while maintaining interpretability and providing uncertainty quantification for predictions made outside the original rule space.

Fuzzy Inference System Integration: Perhaps most significantly, ongoing research explores the integration of extracted granular rules within fuzzy inference systems to enhance decision-making under uncertainty. This involves: (1) *fuzzification* of crisp granular rules by converting discrete attribute boundaries into fuzzy membership functions, enabling smoother transitions between classification regions; (2) construction of a comprehensive *fuzzy rule base* where granular rules serve as the foundation for fuzzy if-then statements with associated membership degrees; and (3) *defuzzification* processes that aggregate multiple rule activations to produce interpretable, confidence-weighted classification outputs. This integration leverages the interpretability strengths of granular computing while incorporating the uncertainty handling capabilities of fuzzy systems, creating a hybrid framework particularly well-suited for environmental decision-making where precise boundaries between air quality categories may be artificial or context-dependent.

These ongoing developments collectively aim to create a more robust, adaptive, and practically applicable granular computing framework that addresses current limitations while expanding the methodology's applicability to broader environmental monitoring and classification challenges.

5. Advancing Air Quality Forecasting: Transfer Learning Across Different Air Pollutants and Temporal Scales

While the previous chapters have established robust methodologies for PM toxicity threshold prediction and explainable air quality classification, the operational deployment of air quality forecasting systems continues to face significant challenges related to data scarcity, computational efficiency, and model scalability. Traditional approaches to air pollution forecasting typically require extensive historical datasets for each pollutant and temporal resolution, leading to fragmented, resource-intensive systems that struggle to adapt to new monitoring scenarios or emerging pollutants. However, the inherent relationships between atmospheric species—arising from common emission sources, shared chemical processes, and similar meteorological influences—present a compelling opportunity to leverage knowledge transfer across different forecasting tasks. This chapter introduces transfer learning as a transformative paradigm for air quality forecasting, systematically exploring how predictive knowledge acquired from one pollutant can enhance forecasting accuracy for others, and how models trained at fine temporal resolutions can be efficiently adapted to coarser time scales. Through three comprehensive studies, we demonstrate that transfer learning not only addresses the fundamental challenges of data efficiency and computational scalability but also enables the development of more robust, uncertainty-aware forecasting systems capable of supporting real-time environmental monitoring and public health decision-making across diverse operational contexts.

5.1. Background and Motivation

Accurate air pollution forecasting plays a critical role in environmental monitoring and public health management. However, conventional machine learning approaches often face limitations due to the need for extensive training data, high computational costs, and model redundancy—particularly when separate models are built for each pollutant or temporal resolution. These traditional methods typically construct isolated models for individual pollutants and specific time horizons, overlooking the intrinsic relationships between pollutants and across temporal scales.

Air pollutants such as $PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, and O_3 are often interrelated through common emission sources, similar atmospheric chemical processes, and shared meteorological influences. For example, $PM_{2.5}$ and PM_{10} originate from overlapping sources and exhibit related formation dynamics, while O_3 levels are heavily influenced by the presence of NO_2 and volatile organic compounds. These correlations present a valuable opportunity to improve forecasting performance through transfer learning, where knowledge acquired from one forecasting task can be repurposed to support another.

Similarly, forecasting air quality across different temporal resolutions (e.g., hourly, 6-hourly, 12-hourly, and daily) introduces its own set of challenges. Building separate models for each time scale demands large quantities of data and training time, often resulting in inefficient and fragmented forecasting systems. Traditional methods that average high-resolution forecasts to obtain lower-resolution predictions fail to capture the true temporal dynamics of air pollutant behavior. By applying transfer learning across temporal scales, it becomes possible to use knowledge from fine-grained (e.g., hourly) predictions to inform coarser time-scale models (6-hour, 12-hour, and daily), thereby reducing data and computational requirements while improving overall accuracy.

This research investigates the dual potential of transfer learning across both pollutant types and temporal resolutions. Specifically, it explores:

- The transfer of knowledge from forecasting $PM_{2.5}$ concentrations to predict other key pollutants within a single monitoring station.
- The extension of forecasting models trained on hourly $PM_{2.5}$ data to larger temporal horizons (6-hour, 12-hour, and daily).

By leveraging the shared physical and chemical characteristics of air pollutants, as well as

the temporal dependencies that govern their behavior, this approach aims to build more generalizable, data-efficient, and computationally effective forecasting systems. Ultimately, transfer learning offers a powerful framework for advancing air quality forecasting by unifying disparate prediction tasks through shared knowledge and representation learning.

5.2. Related Works

In recent years, transfer learning has emerged as a pivotal paradigm in air quality forecasting research, attracting substantial attention from both academic and practical domains. This machine learning approach offers significant advantages by enabling knowledge extraction from source domains or tasks to enhance prediction accuracy and model robustness in target applications. This section presents a systematic review of recent research that has investigated transfer learning methodologies for air quality prediction. Through synthesizing insights from diverse scholarly contributions, we identify prevailing techniques, persistent challenges, and promising directions for future investigation in this rapidly evolving field.

Several seminal studies have demonstrated the efficacy of transfer learning in enhancing air quality forecasting precision. [Jun Ma, Cheng, et al. 2019] developed a bidirectional long short-term memory (LSTM) model to capture long-term dependencies in $PM_{2.5}$ concentrations, applying transfer learning to transition knowledge from smaller to larger temporal resolutions, including daily, weekly, and monthly forecasts. Similarly, [Fong et al. 2020] employed transfer learning with LSTM-based recurrent neural networks to assist air quality monitoring stations with limited observational data, primarily transferring knowledge from PM_{10} to $PM_{2.5}$ prediction tasks. Extending this approach, [Jun Ma, Z. Li, et al. 2020] proposed a transfer learning-based stacked bidirectional LSTM network (TLS-BLSTM) to address air quality prediction challenges at new stations with data scarcity.

Further advancements in the field include the ensemble approach for multi-source transfer learning developed by [Dhole et al. 2021], which mitigates data shortage issues by generating cumulative predictions through knowledge transfer from multiple source stations to a target station. [Ni et al. 2022] introduced a sophisticated model incorporating maximum mean discrepancy (MMD) for optimal source station selection, an improved dual-stage two-phase model for spatial-temporal feature extraction, and a domain adversarial neural network (DANN) for domain-invariant feature identification. Additionally, [W. Ma et al. 2022] proposed coupling LSTM with transfer learning (TL-LSTM) to transfer knowledge from

hourly to daily ozone forecasting modules, while [J. Yang et al. 2023] developed a modified hybrid deep learning model within a transfer learning framework to address sparse data scenarios in air quality prediction. Recently, [B. Yao et al. 2024] introduced the Multi-source Variational Mode Transfer Learning (MSVMTL) framework, integrating data decomposition, deep learning, and multi-source transfer learning strategies.

Table 5.1 synthesizes the methodologies, objectives, datasets, and limitations of these transfer learning applications in air quality forecasting research. This comprehensive overview serves as a foundational reference for understanding the current knowledge landscape in this domain.

Reference	Methodology	Research Objective	Dataset Characteristics	Limitations
[Jun Ma, Cheng, et al. 2019]	Transferred bi-directional LSTM (TL-BLSTM) architecture for $PM_{2.5}$ prediction with deep learning integration.	Captures long-term $PM_{2.5}$ dependencies and transfers knowledge across temporal scales, enabling prediction at daily, weekly, and monthly resolutions.	Comprehensive three-year air quality monitoring from Guangdong, China, encompassing 26,304 hourly concentration measurements per station.	Exclusive focus on $PM_{2.5}$ without incorporating meteorological or geographical factors into the predictive framework.
[Fong et al. 2020]	LSTM-based recurrent neural networks with transfer learning for air pollutant concentration forecasting.	Employs pre-trained neural networks to enhance prediction accuracy at monitoring stations with limited observational data, focusing on knowledge transfer from PM_{10} to $PM_{2.5}$.	Extensive 12-year dataset (2001-2014) from Macao, containing daily measurements from four air quality monitoring stations and one automatic weather station.	Limited to cross-station transfer rather than cross-pollutant transfer, with narrow focus on PM_{10} to $PM_{2.5}$ knowledge transfer only.
[Jun Ma, Z. Li, et al. 2020]	Transfer learning-based stacked bidirectional LSTM network (TLS-BLSTM) for spatial knowledge transfer.	Addresses data scarcity at new monitoring stations by transferring spatial knowledge from established stations with robust historical data.	Hourly concentration data for $PM_{2.5}$, NO_2, and O_3 from multiple monitoring stations in Anhui Province, China.	Neglects integration of critical meteorological and geographical variables that influence pollutant dispersal and concentration.
[Dhole et al. 2021]	Multi-source transfer learning ensemble approach with CNN-LSTM and CNN-GRU architectures.	Mitigates data shortage issues through cumulative predictions derived from multiple source stations, enhancing target station forecasting capabilities.	Beijing Multi-Site Air Quality dataset with comprehensive pollutant measurements across multiple monitoring stations.	Restricted to $PM_{2.5}$ forecasting and cross-station transfer, without exploring cross-pollutant knowledge transfer potential.
[Ni et al. 2022]	Hybrid transfer learning model incorporating maximum mean discrepancy (MMD), dual-stage two-phase feature extraction, and domain adversarial neural networks.	Develops optimized source domain selection criteria and spatial-temporal feature extraction for $PM_{2.5}$ prediction at new sites with limited historical data.	Microsoft Research's Urban Air project dataset with multi-station air quality monitoring data.	Focuses exclusively on $PM_{2.5}$ prediction and station-to-station transfer, without addressing pollutant-to-pollutant knowledge transfer.
[W. Ma et al. 2022]	LSTM neural network coupled with transfer learning (TL-LSTM) and L2 regularization for enhanced generalization.	Transfers model configuration knowledge from hourly to daily forecasting modules, improving daily maximum 8-hour average O_3 prediction accuracy in Hong Kong.	Data from Tung Chung air quality monitoring and Hong Kong International Airport meteorological stations (2001-2018).	Limited application to O_3 forecasting only, without exploring transferability to other critical pollutants.
[J. Yang et al. 2023]	Modified hybrid deep learning framework combining LSTM and MLP architectures with transfer learning.	Addresses feature extraction limitations in transfer learning models for $PM_{2.5}$ prediction across multiple monitoring sites with sparse data.	Temporal data from Beijing (2013-2017) and Hengshui (2020-2022) monitoring networks.	Exclusive focus on $PM_{2.5}$ forecasting across stations without meteorological variable integration or cross-pollutant knowledge transfer exploration.
[B. Yao et al. 2024]	Multi-source Variational Mode Transfer Learning (MSVMTL) integrating data decomposition with deep learning and multi-source knowledge transfer.	Enhances $PM_{2.5}$ concentration forecasting accuracy at monitoring sites with limited historical data through advanced decomposition and transfer techniques.	Beijing multi-station air quality dataset with comprehensive spatio-temporal measurements.	Application restricted to $PM_{2.5}$ forecasting across different sites without exploring cross-pollutant knowledge transfer possibilities.
[Z. Ma et al. 2024]	Two-stage attention mechanism model based on transfer learning (TL-AdaBiGRU) incorporating temporal distribution characterization with temporal attention mechanism (Stage 1) and multi-head external attention mechanism (Stage 2)	Improvement of air pollution prediction capabilities at newly established atmospheric monitoring sites with limited historical data	Beijing Multi-Site Air-Quality Dataset	Methodological validation limited to new monitoring sites/stations
[Sangiorgio and Guariso 2024]	Long Short-Term Memory (LSTM) neural networks with transferability assessment procedure and clustering approach to reduce transferability uncertainty	Evaluation of neural network model transferability to unmonitored locations, forecasting of hourly ozone concentrations, and support for evidence-based decision-making in air pollution control	Data from 20 monitoring stations of diverse typologies (urban, rural, mountain) around the Alpine Arc measuring hourly ozone concentrations	Evaluation restricted to ozone forecasting applications

Table 5.1.: Comparative analysis of transfer learning applications in air quality forecasting research.

5.2.1. Research Gaps and Contributions

This systematic review establishes a robust foundation for understanding the evolution and current state of transfer learning applications in air quality forecasting. By examining methodological developments, practical implementations, and research challenges, we provide valuable context for subsequent investigations. The analysis reveals significant advancements in addressing data scarcity, domain shifts, and model generalization through transfer learning techniques, contributing to improved accuracy, robustness, and interpretability of air quality prediction models.

Despite these advances, our synthesis of the literature reveals two critical research gaps:

1. **Limited Application of Transfer Learning Across Different Pollutants:** As shown in Table 5.1, most existing studies focus on station-to-station or temporal transfer within a single pollutant (primarily $PM_{2.5}$), leaving cross-pollutant transfer learning largely unexplored. This gap is particularly evident within single-station contexts where multiple pollutants are monitored concurrently and are influenced by shared meteorological and chemical processes. To address this gap, our first study implements a Multi-Layer Perceptron (MLP) model, also known as an Artificial Neural Network (ANN), that transfers learned knowledge from $PM_{2.5}$ forecasting tasks to predict concentrations of PM_{10}, SO_2, NO_2, CO, and O_3. While MLPs are not the dominant architecture in time series forecasting, we leverage their strong capacity for modeling nonlinear relationships without requiring strict sequential dependencies, making them efficient and less prone to overfitting when transfer learning is applied across diverse pollutant types. In addition, the second study addresses the challenge of selecting an appropriate lag window—a crucial factor in time series modeling that can significantly affect performance—by introducing a deep ensemble strategy called **LagEnsembleForecasting**. This approach integrates multiple models trained on distinct lagged input sequences, allowing the system to capture short- and long-term pollutant dynamics more effectively. We systematically investigate the impact of different fine-tuning data ratios to optimize cross-pollutant performance within this flexible framework.

2. **Underutilization of Transfer Learning Across Temporal Resolutions:** While studies such as [Jun Ma, Cheng, et al. 2019; W. Ma et al. 2022] have applied transfer

learning from hourly to daily forecasts, they remain limited in scope and do not comprehensively explore a wider range of time scales or the integration of meteorological factors. Our third study addresses this gap by expanding the forecasting horizon from hourly to 6-hour, 12-hour, and daily resolutions. We incorporate meteorological variables to enrich model inputs and examine the effects of varying fine-tuning ratios (10%, 20%, 30%, 40%, and 50%) to assess model adaptability across temporal scales. This multi-resolution strategy offers an efficient and scalable framework for both short-term and long-term air quality prediction.

Together, these contributions aim to broaden the scope of transfer learning applications in environmental forecasting. By addressing underexplored areas such as pollutant-level knowledge transfer and temporal generalization with limited data, this research enhances the versatility, efficiency, and robustness of forecasting models. The findings contribute to a deeper understanding of how shared pollutant dynamics and temporal dependencies can be leveraged through transfer learning, offering promising directions for future studies in sustainable and intelligent environmental monitoring systems.

5.3. Enhancing air pollution prediction: A neural transfer learning approach across different air pollutants

5.3.1. Proposed model and result analysis

This section presents a neural transfer learning model architecture, based on an artificial neural network (ANN)/multi-layer perceptron (MLP), designed for multivariate air pollutant forecasting. The proposed approach leverages transfer learning to apply knowledge from $PM_{2.5}$ concentration forecasting to enhance predictive accuracy for other pollutants (PM_{10}, SO_2, NO_2, CO, and O_3) at the same monitoring station. We discuss data preprocessing techniques, feature engineering approaches, and provide comprehensive analysis of experimental results. Figure 5.1 illustrates the methodology framework proposed in this study.

A. Pre-trained model based on ANN/MLP for $PM_{2.5}$ forecasting

Dataset Overview This study utilizes the Beijing Multi-Site Air Quality dataset for the Aotizhongxin monitoring station (Section: 2.7.1). The dataset comprises hourly measure-

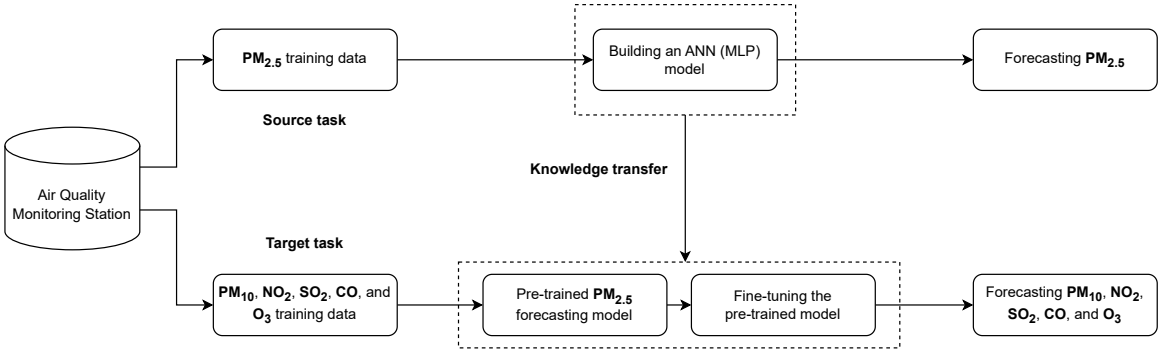


Figure 5.1.: The process of the proposed transfer learning approach in this study.

ments of six air pollutants ($PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, and O_3) alongside meteorological variables.

While the dataset contains sufficient data points for all pollutants, our research explores the potential of transfer learning within the same monitoring station—an approach not extensively addressed in previous studies. By demonstrating the effectiveness of this technique for these well-represented pollutants, we aim to establish a foundation for extending the methodology to pollutants with limited historical data. This research specifically examines how varying fine-tuning dataset sizes impact model performance, validating that transfer learning can be particularly beneficial in scenarios with constrained data availability.

Data Preprocessing and Feature Engineering

Missing Value Handling The initial data quality assessment identified columns with missing values, as illustrated in Table 2.2, Section 2.7.1. For numerical variables ($PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, O_3, TEMP, PRES, DEWP, RAIN, and WSPM), we evaluated the relationship between each variable and its lagged values through regression analysis. Figure 5.2 presents these regression plots along with their corresponding R^2 values.

The R^2 values quantify the proportion of variance in a variable's current value that can be explained by its previous value. Higher R^2 values indicate stronger linear relationships between a variable and its lagged counterpart, suggesting that the current value is closely related to its previous value. Based on these relationships, we employed forward or backward fill methods to address missing values in numerical variables, even for variables like RAIN that showed weaker linear correlations with their lagged values.

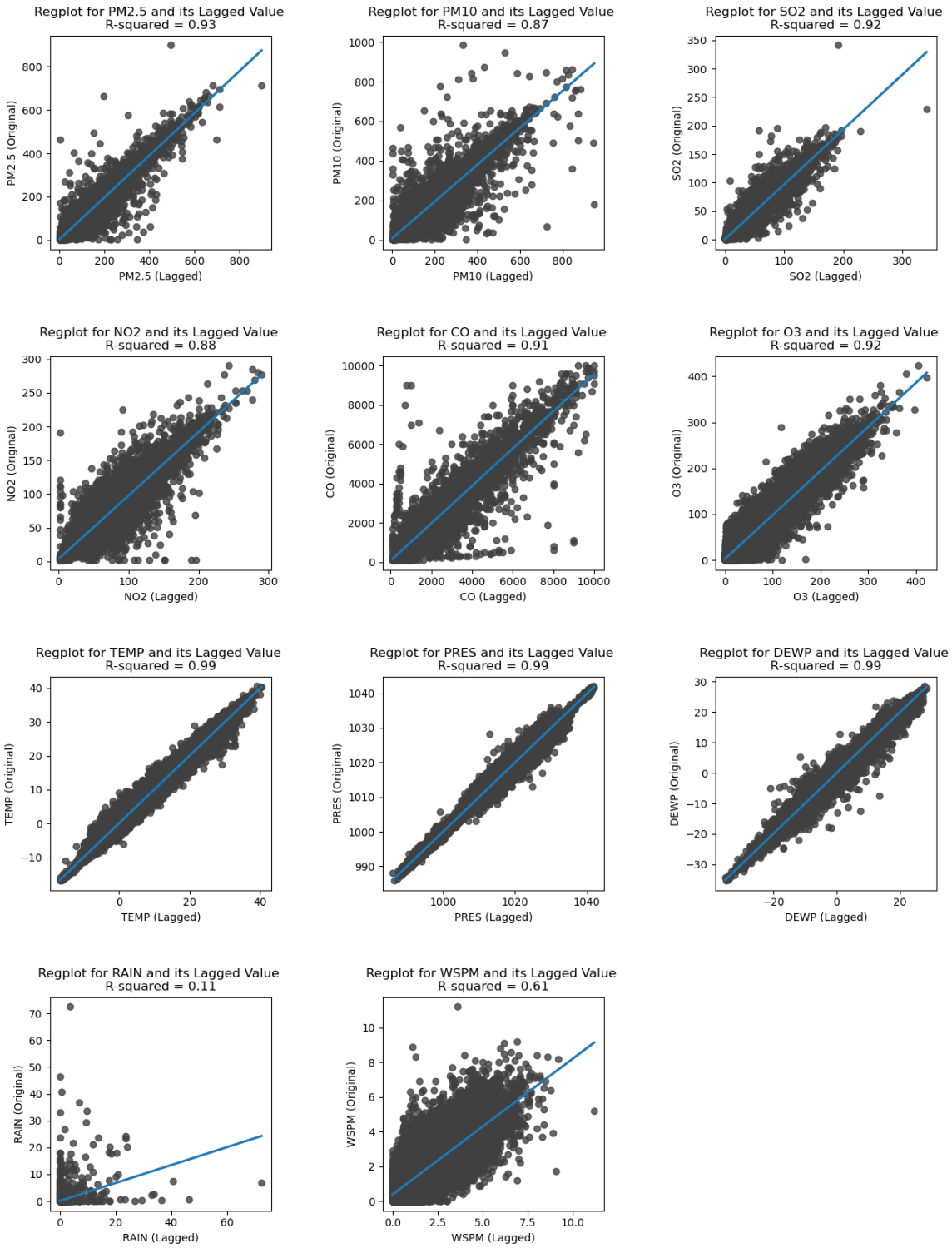


Figure 5.2.: Regression plots of the missing numerical columns with their one-lag values along with the R-squared values.

For the categorical wind direction variable (wd), which had 81 missing data points, we replaced missing values with the most frequent category, North East (NE).

Feature Encoding and Standardization To make it compatible with machine learning algorithms that need numbers as input, we applied ordinal encoding to the wind direction (wd) categorical variable, transforming the 16 distinct wind direction categories into sequential integer labels.

The significant disparities in variable scales—from wind speed (WSPM) ranging from 0 to 11.2 to pressure (PRES) ranging from 985.90 to 1042—necessitated standardization to ensure balanced feature contribution during model training. Without proper scaling, features with larger ranges could dominate the learning process, leading to biased performance and slower convergence. Therefore, we standardized all variables using the formula in Equation 2.85 in Section B.

Correlation Analysis Understanding variable relationships provides valuable insights into potential patterns and dependencies. We conducted correlation analysis to quantify the strength and direction of associations between variable pairs using Pearson correlation coefficients (Equation: 2.83)

The correlation matrix (Figure 5.3) revealed significant relationships between air pollutants. For example, the strong correlation (0.87) between PM_{10} and $PM_{2.5}$ indicates that changes in PM_{10} levels are closely associated with changes in $PM_{2.5}$ levels, suggesting shared sources or common atmospheric processes. This insight implies that monitoring one pollutant can help assess levels of related pollutants, aiding in comprehensive air quality management.

Time Series Analysis To examine temporal dependencies in air pollutant concentrations, we employed Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) analyses (Section: 2.2.4).

Figure 5.4 presents the ACF and PACF plots for all six air pollutants across 20 lag periods (hours). Since PACF captures direct correlations between data points, we used it to select appropriate lag values. The PACF plots revealed strong correlations between data points at time t and $t-1$ for all pollutants, leading us to incorporate one-lag values as additional features in our models.

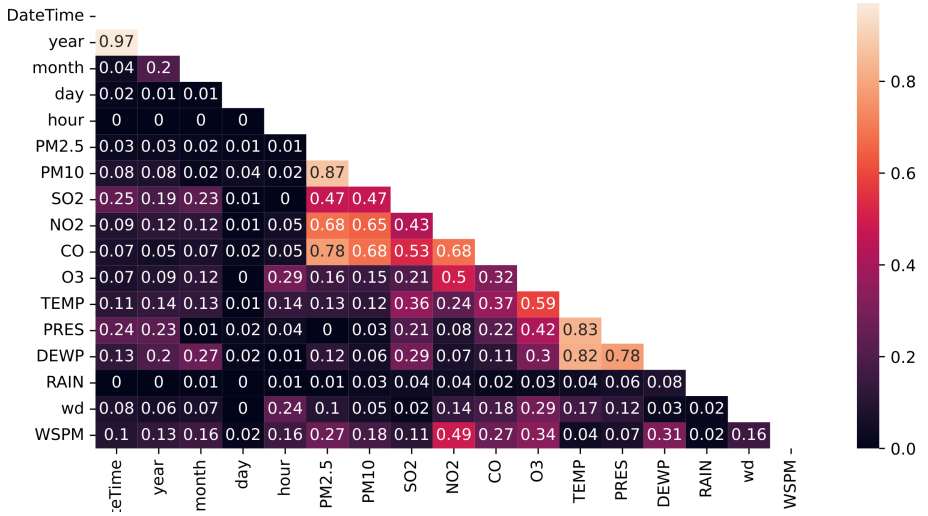


Figure 5.3.: Correlation matrix of the columns in the dataset.

B. ANN Architecture and Hyperparameters Optimization

Artificial neural networks (ANNs) (Section: 2.4.1) excel at capturing complex nonlinear relationships in air quality data. The architecture design—including layer configuration, neuron counts, and activation functions—significantly impacts predictive performance. We employed the Optuna framework for hyperparameter optimization, efficiently searching for optimal values using Tree-structured Parzen Estimator (TPE) techniques (Section: 2.6.2). Table 5.2 summarizes the range of tested hyperparameters and reports the best-performing configuration selected for the ANN/MLP model.

C. Pre-trained Model Training and Evaluation

We divided the dataset into training (60%, chronologically first portion), validation (20%), and test sets (20%). The model's input layer included meteorological variables (TEMP, PRES, DEWP, RAIN, encoded wind direction, WSPM) and the one-lag value of $PM_{2.5}$, while the output layer consisted of a single neuron for $PM_{2.5}$ concentration prediction.

We compiled the model using the Adam optimizer to minimize mean squared error loss. Figure 5.5 illustrates the training and validation loss curves across epochs, showing a consistent decrease in both curves that indicates effective learning and successful pattern

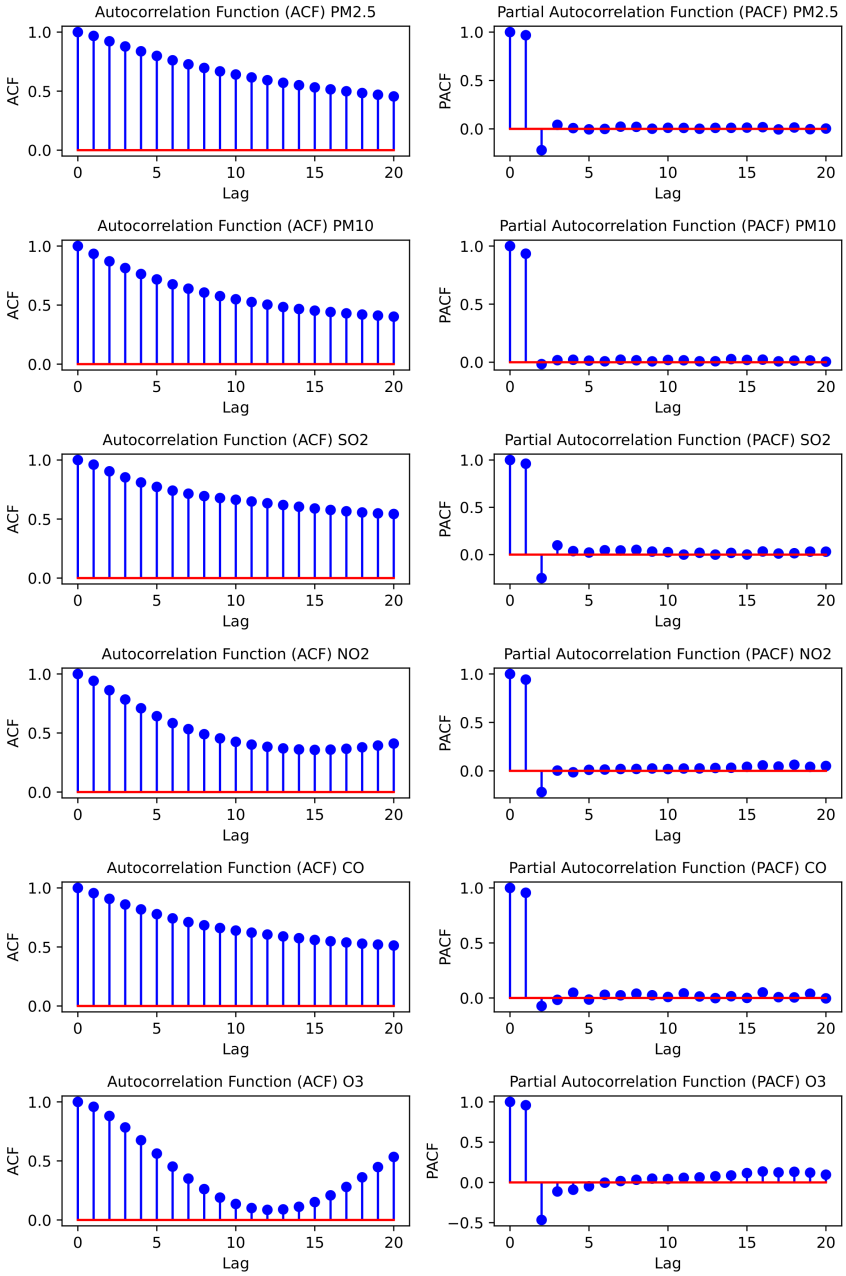


Figure 5.4.: ACF and PACF plots of the air pollutants variables.

Hyperparameter	Suggested values	Best value
Number of hidden layers	[1, 2, 3]	1
Number of units (hidden neurons)	[8, 16, 32, 64, 128]	8
Dropout rate	(0.1, 0.3)	0.1438
Learning rate	(1e-3, 1e-1)	0.0015
Regularization	['l1', 'l2', None]	None
Regularization strength	(1e-3, 1e-1)	0.063
Epochs	[50, 100, 150]	100

Table 5.2.: Summary of the chosen hyperparameters, tested, and best values.

recognition without overfitting.

Table 5.3 compares our ANN/MLP model performance with alternative time series forecasting architectures: Simple Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM). Our proposed ANN/MLP model demonstrates competitive or superior performance across all metrics and datasets. On the test set, our model achieves an MAE of 0.1467, MSE of 0.0649, RMSE of 0.2548, and R^2 of 0.9454, indicating high accuracy and robust generalization.

Model	Metric	MAE	MSE	RMSE	R^2
ANN/MLP (This study)	Training Set	0.1533	0.0737	0.2715	0.9262
	Validation Set	0.1536	**0.0909**	**0.3015**	**0.9423**
	Test Set	0.1467	**0.0649**	**0.2548**	**0.9454**
Simple RNN	Training Set	**0.1507**	0.0732	0.2705	0.9267
	Validation Set	**0.1505**	0.0918	0.3030	0.9417
	Test Set	**0.1442**	0.0653	0.2556	0.9451
GRU	Training Set	0.1578	0.0754	0.2747	0.9245
	Validation Set	0.1567	0.0946	0.3076	0.9399
	Test Set	0.1485	0.0681	0.2610	0.9428
LSTM	Training Set	0.1567	**0.0730**	**0.2702**	**0.9269**
	Validation Set	0.1590	0.0914	0.3023	0.9420
	Test Set	0.1507	0.0656	0.2562	0.9448

Table 5.3.: Pre-trained $PM_{2.5}$ forecasting performance comparative analysis of the proposed ANN/MLP model against other architectures.

The performance metrics across training, validation, and test sets remain consistent, with

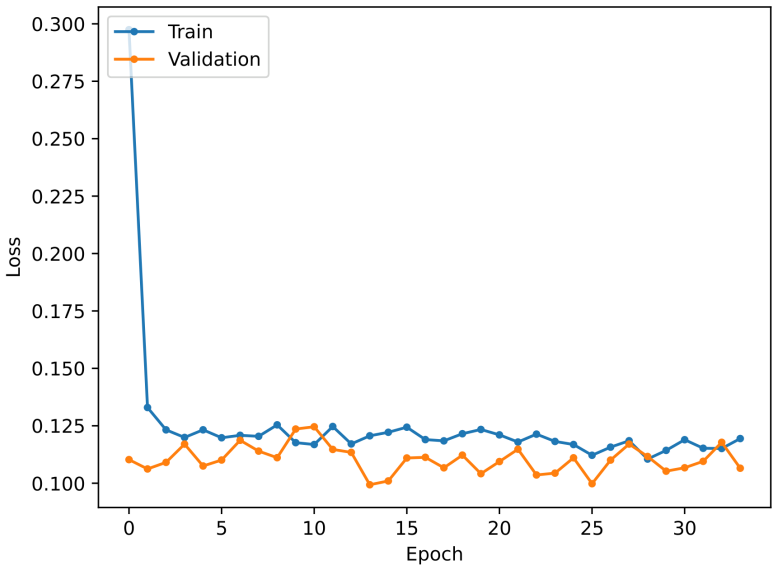


Figure 5.5.: Training and validation loss curves over epochs.

slightly better results on the test set, demonstrating the model's reliability and generalization capability. The high R^2 values across all datasets (0.9262, 0.9423, and 0.9454 respectively) indicate that the model explains a significant proportion of variance in $PM_{2.5}$ concentrations.

Figure 5.6 visually compares model predictions against ground truth values, further confirming the model's ability to capture temporal patterns and fluctuations in $PM_{2.5}$ concentrations.

To validate model robustness, we implemented 5-fold cross-validation using a TimeSeriesSplit approach (Table 5.4). This method divides the data into five folds, with each fold serving as a test set while the remaining data serves as the training set across five iterations.

The cross-validation results show slight performance variations across folds, with fold 3 achieving the lowest MSE (0.0524) and RMSE (0.2289), while folds 1 and 4 exhibit slightly higher errors. The average metrics—MSE (0.0758), MAE (0.1580), RMSE (0.2728), and R^2 (0.9338)—confirm the model's overall robustness and generalization capability.

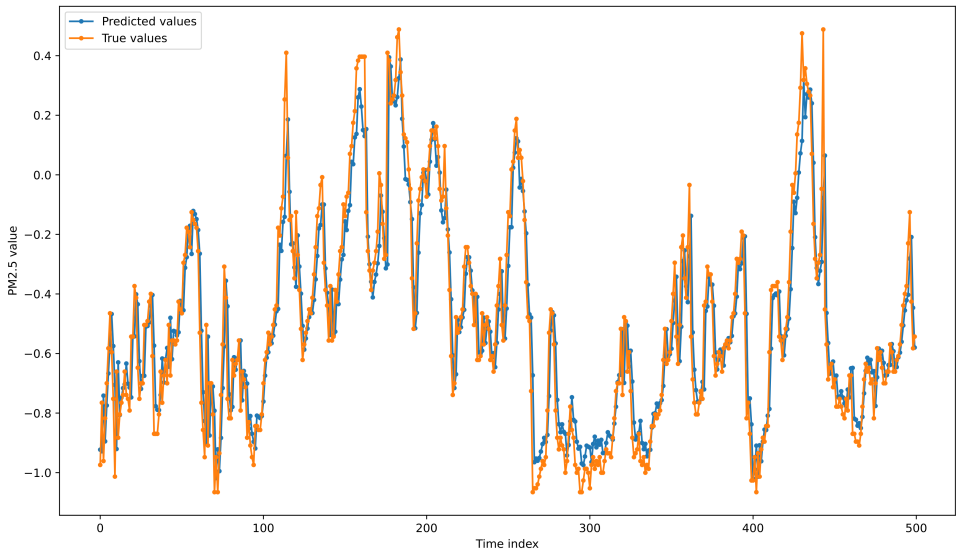


Figure 5.6.: Actual values against predicted values for $PM_{2.5}$ forecasting.

Fold	MSE	MAE	RMSE	R^2
1	0.1022	0.1865	0.3197	0.9257
2	0.0610	0.1453	0.2469	0.9429
3	0.0524	0.1577	0.2289	0.9137
4	0.0998	0.1559	0.3159	0.9388
5	0.0638	0.1444	0.2526	0.9477
Average	0.0758	0.1580	0.2728	0.9338

Table 5.4.: Performance metrics of the base (pre-trained) model with 5-fold cross-validation.

D. Forecasting of Other Air Pollutants Based on the Pre-trained Model

Building upon our successful $PM_{2.5}$ forecasting model, we extended our approach to predict five additional air pollutants: PM_{10}, SO_2, NO_2, CO, and O_3. By leveraging transfer learning principles, we utilized the knowledge captured in the pre-trained $PM_{2.5}$ model to enhance forecasting accuracy for these pollutants, even with limited training data.

We adapted the pre-trained model by modifying the input layer to incorporate the one-lag value of the target pollutant instead of $PM_{2.5}$ while retaining the same meteorological inputs. Two key hyperparameter adjustments were implemented during fine-tuning: (1) using a reduced learning rate to ensure cautious, incremental adjustments that preserve beneficial features from the pre-training phase; and (2) reducing the training epochs to 50 (half of the base model's epochs), significantly decreasing computational requirements.

As illustrated in Tables 5.5 and 5.6, we systematically evaluated how varying fine-tuning dataset sizes (10%, 20%, 30%, 40%, and 50% of available data) affected model performance for each pollutant, comparing transfer learning results against models trained from scratch with identical architecture.

The comparative evaluation of transfer learning against training from scratch reveals consistent advantages across all target pollutants. Key performance metrics, including MAE, MSE, RMSE, and R^2, indicate that the pre-trained ANN model, fine-tuned on limited data, significantly outperforms models trained independently from scratch—especially when fine-tuning with only 10% to 30% of the available data.

- **PM_{10}:** Transfer learning achieved superior results across all fine-tuning sizes, with a marked improvement at 10% (MAE: 0.2380 vs. 0.3298, R^2: 0.8750 vs. 0.8150). The performance gap persists, albeit slightly narrowing, as more data is used.
- **SO_2 and CO:** The pre-trained model consistently outperformed the scratch model at every fine-tuning level, confirming the efficiency of transfer learning for these pollutants.
- **NO_2:** Notable improvements were observed at lower fine-tuning ratios, with diminishing differences as data size increased—highlighting transfer learning's benefit in low-resource scenarios.
- **O_3:** While performance fluctuations were minimal, the pre-trained model maintained a consistent edge in both accuracy and R^2 across all sizes.

Pollutant	Fine-Tuning Size (%)	With Transfer Learning	MAE	MSE	RMSE	R^2
PM_{10}	10	**Yes**	**0.2380**	**0.1573**	**0.3966**	**0.8750**
		No	0.3298	0.2330	0.4827	0.8150
	20	**Yes**	**0.2351**	**0.1694**	**0.4116**	**0.8785**
		No	0.2680	0.1855	0.4307	0.8670
	30	**Yes**	**0.2064**	**0.1239**	**0.3520**	**0.8825**
		No	0.2369	0.1377	0.3711	0.8693
	40	**Yes**	**0.2092**	**0.1346**	**0.3669**	**0.8913**
		No	0.2433	0.1604	0.4005	0.8706
	50	**Yes**	**0.1939**	**0.1148**	**0.3388**	**0.8877**
		No	0.2181	0.1246	0.3530	0.8781
SO_2	10	**Yes**	**0.1322**	**0.0531**	**0.2305**	**0.9217**
		No	0.2287	0.0966	0.3109	0.8576
	20	**Yes**	**0.1315**	**0.0603**	**0.2457**	**0.9274**
		No	0.1934	0.0774	0.2782	0.9070
	30	**Yes**	**0.0974**	**0.0268**	**0.1637**	**0.9087**
		No	0.1196	0.0315	0.1775	0.8927
	40	**Yes**	**0.0961**	**0.0322**	**0.1796**	**0.9133**
		No	0.1080	0.0359	0.1896	0.9033
	50	**Yes**	**0.0897**	**0.0268**	**0.1639**	**0.9089**
		No	0.1087	0.0298	0.1727	0.8988
NO_2	10	**Yes**	**0.2593**	**0.1410**	**0.3756**	**0.8913**
		No	0.3366	0.2100	0.4582	0.8382
	20	**Yes**	**0.2515**	**0.1399**	**0.3741**	**0.8953**
		No	0.2791	0.1549	0.3936	0.8841
	30	**Yes**	**0.2411**	**0.1284**	**0.3584**	**0.8945**
		No	0.2419	0.1325	0.3640	0.8911
	40	**Yes**	**0.2425**	**0.1370**	**0.3702**	**0.9019**
		No	0.2653	0.1505	0.3880	0.8922
	50	**Yes**	**0.2216**	**0.1156**	**0.3401**	**0.8958**
		No	0.2278	0.1207	0.3474	0.8913

Table 5.5.: Performance metrics of PM_{10}, SO_2, and NO_2 forecasting with different fine-tuning sizes.

Pollutant	Fine-Tuning Size (%)	With Transfer Learning	MAE	MSE	RMSE	R^2
CO	10	**Yes**	**0.2497**	**0.2387**	**0.4886**	**0.9183**
		No	0.3156	0.2941	0.5423	0.8938
	20	**Yes**	**0.2183**	**0.1901**	**0.4360**	**0.9140**
		No	0.2685	0.2088	0.4569	0.9056
	30	**Yes**	**0.1798**	**0.1329**	**0.3645**	**0.9140**
		No	0.1845	0.1337	0.3657	0.9135
	40	**Yes**	**0.2046**	**0.1793**	**0.4235**	**0.9129**
		No	0.2148	0.1827	0.4275	0.9112
	50	**Yes**	**0.1897**	**0.1553**	**0.3941**	**0.9129**
		No	0.2077	0.1631	0.4038	0.9085
O_3	10	**Yes**	**0.1887**	**0.0832**	**0.2884**	**0.9251**
		No	0.2086	0.1066	0.3265	0.9040
	20	**Yes**	**0.1941**	**0.0883**	**0.2972**	**0.9248**
		No	0.2055	0.0974	0.3121	0.9170
	30	**Yes**	**0.2177**	**0.1116**	**0.3341**	**0.9226**
		No	0.2260	0.1108	0.3328	0.9232
	40	**Yes**	**0.1894**	**0.0821**	**0.2866**	**0.9245**
		No	0.1994	0.0924	0.3041	0.9150
	50	**Yes**	**0.2120**	**0.1012**	**0.3182**	**0.9219**
		No	0.2323	0.1077	0.3282	0.9169

Table 5.6.: Performance metrics of CO and O_3 forecasting with different fine-tuning sizes.

Key Observations:

1. **Data Efficiency:** Transfer learning enables high-accuracy forecasting even with minimal training data, making it suitable for data-scarce environments.
2. **Computational Gains:** Fine-tuned models converge faster and require fewer training epochs, reducing overall computational overhead.
3. **Model Robustness:** Across all pollutants and data sizes, the pre-trained model demonstrated stable and superior performance compared to baseline models.
4. **Scalability:** The consistent success of transfer learning supports its potential scalability to other pollutants or monitoring stations.

Model Benchmarking with Deep Learning Architectures Table 5.7 presents a detailed comparison of the proposed ANN model against RNN, GRU, and LSTM across all pollutants and fine-tuning sizes. The ANN demonstrated consistently strong performance, achieving competitive or superior results in MAE, RMSE, and R^2 across all configurations.

Summary of ANN Performance:

- For each pollutant, increasing the fine-tuning size led to incremental gains in accuracy, with the ANN achieving its lowest errors and highest R^2 at 50%.
- The ANN was particularly effective for SO_2 and CO, offering strong accuracy even with 10% fine-tuning.
- Despite its simpler architecture, the ANN model performed on par with or better than GRU and LSTM, demonstrating its robustness and computational efficiency.

Comparison Insights:

1. **Performance Range:** ANN maintained accuracy within the range of more complex models across pollutants and data sizes.
2. **Stability:** ANN results showed low variance across pollutant types, reinforcing its adaptability.

Pollutant	Model	10%			20%			30%			40%			50%		
		MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
PM_{10}	ANN	0.2380	0.3966	0.8750	0.2351	0.4116	0.8785	0.2064	0.3520	0.8825	0.2092	0.3669	0.8913	0.1939	0.3388	0.8877
	RNN	0.2341	0.3955	0.8758	0.2334	0.4111	0.8788	0.2078	0.3528	0.8819	0.2071	0.3670	0.8913	0.1918	0.3395	0.8873
	GRU	0.2331	0.3914	0.8783	0.2352	0.4111	0.8789	0.2070	0.3530	0.8818	0.2064	0.3671	0.8912	0.1913	0.3382	0.8881
	LSTM	0.2393	0.3940	0.8767	0.2396	0.4121	0.8783	0.2129	0.3552	0.8803	0.2116	0.3696	0.8898	0.1969	0.3395	0.8872
SO_2	ANN	0.1322	0.2305	0.9217	0.1315	0.2457	0.9274	0.0974	0.1637	0.9087	0.0961	0.1796	0.9133	0.0897	0.1639	0.9089
	RNN	0.1285	0.2296	0.9223	0.1293	0.2460	0.9273	0.0921	0.1620	0.9106	0.0943	0.1801	0.9128	0.0865	0.1633	0.9095
	GRU	0.1277	0.2266	0.9243	0.1339	0.2461	0.9272	0.0998	0.1636	0.9088	0.0957	0.1791	0.9138	0.0907	0.1637	0.9091
	LSTM	0.1317	0.2274	0.9238	0.1390	0.2474	0.9264	0.1087	0.1674	0.9045	0.1041	0.1808	0.9122	0.1021	0.1677	0.9046
NO_2	ANN	0.2593	0.3756	0.8913	0.2515	0.3741	0.8953	0.2411	0.3584	0.8945	0.2425	0.3702	0.9019	0.2216	0.3401	0.8958
	RNN	0.2581	0.3761	0.8910	0.2494	0.3739	0.8954	0.2402	0.3590	0.8941	0.2419	0.3705	0.9017	0.2203	0.3400	0.8959
	GRU	0.2540	0.3712	0.8938	0.2494	0.3702	0.8975	0.2379	0.3548	0.8966	0.2374	0.3637	0.9053	0.2197	0.3362	0.8982
	LSTM	0.2584	0.3732	0.8926	0.2562	0.3735	0.8956	0.2430	0.3570	0.8953	0.2438	0.3672	0.9034	0.2267	0.3390	0.8965
CO	ANN	0.2497	0.4886	0.9138	0.2183	0.4360	0.9140	0.1798	0.3645	0.9140	0.2046	0.4235	0.9129	0.1897	0.3941	0.9129
	RNN	0.2498	0.4913	0.9128	0.2217	0.4402	0.9124	0.1794	0.3660	0.9133	0.2038	0.4236	0.9128	0.1892	0.3947	0.9126
	GRU	0.2520	0.4941	0.9118	0.2182	0.4337	0.9149	0.1774	0.3606	0.9159	0.2024	0.4182	0.9151	0.1855	0.3881	0.9155
	LSTM	0.2555	0.4998	0.9098	0.2214	0.4361	0.9140	0.1832	0.3648	0.9139	0.2084	0.4244	0.9125	0.1914	0.3936	0.9131
O_3	ANN	0.1887	0.2884	0.9251	0.1941	0.2972	0.9248	0.2177	0.3341	0.9226	0.1894	0.2866	0.9245	0.2120	0.3182	0.9219
	RNN	0.1875	0.2879	0.9254	0.1922	0.2970	0.9249	0.2187	0.3367	0.9214	0.1881	0.2873	0.9241	0.2122	0.3191	0.9215
	GRU	0.1911	0.2868	0.9260	0.1956	0.2980	0.9243	0.2174	0.3319	0.9236	0.1886	0.2845	0.9256	0.2095	0.3156	0.9232
	LSTM	0.1957	0.2867	0.9260	0.1965	0.2961	0.9253	0.2194	0.3325	0.9234	0.1917	0.2857	0.9250	0.2103	0.3164	0.9228

Table 5.7.: Performance metrics for different pre-trained models (ANN, RNN, GRU, and LSTM) across different pollutants and fine-tuning sizes.

3. **Model Simplicity:** Given its performance and lower computational cost, the ANN is an attractive alternative to deeper and more complex architectures.

These findings support the use of ANN-based transfer learning as a reliable and efficient framework for multi-pollutant air quality forecasting, with the potential for broad applicability in environmental monitoring.

5.3.2. Discussion

This study demonstrates the effectiveness of neural transfer learning in multi-pollutant air quality forecasting. By fine-tuning a model pre-trained on $PM_{2.5}$ using limited data from other pollutants (PM_{10}, SO_2, NO_2, CO, and O_3), we observed consistent improvements in forecasting accuracy, even with as little as 10% of the target data. Performance gains were particularly notable for PM_{10} and SO_2, where transfer learning significantly outperformed models trained from scratch.

The approach not only enhances predictive accuracy but also reduces training time and computational costs. Fine-tuned models converged faster, making this method suitable for real-time and data-scarce applications. These advantages support the use of transfer learning as a practical and scalable solution for environmental monitoring.

While the results are promising, the current study is limited to a single monitoring station. Future research should investigate the generalizability of this approach across multiple sites and pollutant contexts. Further optimization of model architectures and regularization techniques could also enhance robustness.

In summary, this work highlights the potential of transfer learning to improve air quality forecasting with minimal data, offering a valuable framework for efficient, accurate, and scalable environmental decision support systems.

5.3.3. Summary and Conclusion

This study explored the application of neural transfer learning to multi-pollutant air quality forecasting within a single monitoring station. A pre-trained MLP/ANN model initially developed for $PM_{2.5}$ forecasting was fine-tuned to predict concentrations of other key pollutants—PM_{10}, SO_2, NO_2, CO, and O_3—using varying proportions of task-specific data. The objective was to improve forecasting accuracy while minimizing data requirements and computational overhead.

Extensive evaluations using standard performance metrics (MAE, MSE, RMSE, and R^2) demonstrated that the transfer learning approach consistently outperformed models trained from scratch, especially in low-data scenarios. Fine-tuned models achieved high predictive performance even with only 10–20% of the target pollutant data, highlighting the approach's robustness and data efficiency.

The findings underscore the practical benefits of using transfer learning in environmental monitoring:

- **Rapid Adaptation:** Models can quickly adapt to new pollutant targets with minimal retraining.
- **Scalability:** The framework enables efficient expansion to new pollutants or stations without the need for extensive data collection.
- **Improved Accuracy:** Transfer learning enhances predictions for pollutants with limited data by leveraging knowledge from well-studied sources.
- **Cost and Time Efficiency:** Reduced training times and lower data demands make this approach resource-efficient.

In summary, this research validates neural transfer learning with ANN/MLP as an effective strategy for multi-pollutant air quality forecasting. It offers a scalable, accurate, and computationally efficient solution for addressing key challenges in environmental monitoring. Future work may extend this framework to multi-station scenarios, incorporate additional pollutants, and further refine model architectures to enhance generalizability and operational readiness for real-world deployment.

5.4. LagEnsembleForecasting: Multi-Lag Ensemble Learning and Cross-Pollutant Transfer Learning for Comprehensive Air Quality Forecasting

One of the fundamental challenges of the current air pollution forecasting approaches is the difficulty of selecting an appropriate lagged input sequence that captures both short-term fluctuations and long-term pollutant dynamics often limits forecasting accuracy. To address

this limitations, we introduce *LagEnsembleForecasting*, a novel deep ensemble learning framework that combines four specialized deep learning models trained on distinct lagged input sequences (1h, 6h, 12h, and 24h), aggregating their outputs to produce both point forecasts and probabilistic uncertainty estimates. By integrating information across multiple temporal scales, the ensemble captures fine-grained short-term fluctuations as well as broader accumulation trends in pollutant concentrations. Initially developed for $PM_{2.5}$ forecasting, the ensemble models are subsequently fine-tuned to predict additional pollutants using transfer learning principles, maximizing knowledge reuse and avoiding retraining from scratch.

5.4.1. Proposed model and result analysis

This section presents the architecture of the proposed *LagEnsembleForecasting* framework, developed for multivariate air pollutant forecasting. The approach builds on predictive knowledge from hourly PM2.5 forecasting and transfers it to improve prediction of PM10, SO_2, CO, O_3, and NO_2 at the same station. The framework integrates multiple base models trained on distinct temporal resolutions (1h, 6h, 12h, 24h) and aggregates their outputs to enhance robustness and quantify uncertainty via mean and standard deviation estimates. Figure 5.7 illustrates the proposed methodology.

A. Pre-trained model based on *LagEnsembleForecasting* for $PM_{2.5}$ forecasting

Dataset Overview The dataset employed in this study is sourced from the Beijing Multi-Site Air Quality dataset (Section: 2.7.1), specifically focusing on the Aotizhongxin monitoring station. It contains hourly measurements of six key air pollutants: $PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, and O_3.

Data Preprocessing and Feature Engineering Effective data preparation is crucial for ensuring optimal model performance, particularly when dealing with environmental time series data that exhibits cyclical patterns and temporal dependencies. Our pre-processing pipeline addressed several key challenges inherent to air quality forecasting.

Handling Cyclical Features Environmental data contains numerous cyclical variables whose numerical representations must preserve their inherent periodicity. Wind direction

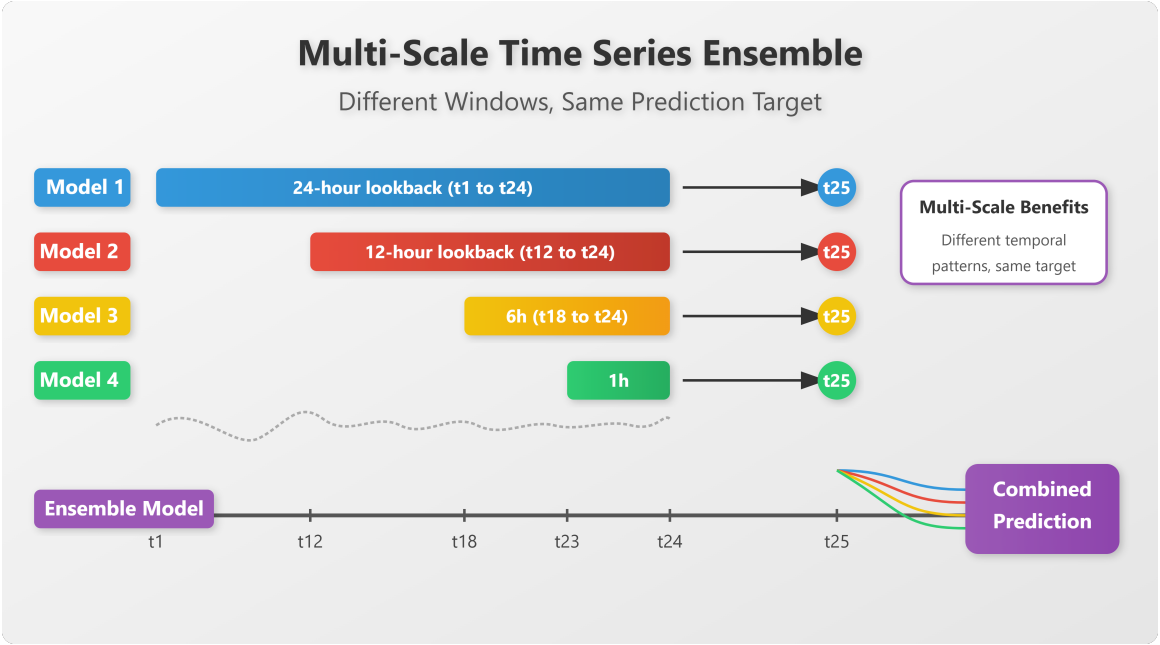


Figure 5.7.: Schematic representation of the proposed *LagEnsembleForecasting* framework.

presents a classic example where traditional encoding creates artificial boundaries between adjacent values (e.g., 359° and 0°). To address this limitation, we implemented sine-cosine transformation for all directional data.

Rather than treating wind direction as either categorical or continuous values with artificial boundaries, we decomposed each direction into its sine and cosine components. This approach maps the circular variable onto a unit circle where proximity in physical space is preserved in the feature space. The resulting orthogonal components (WD_{sin} and WD_{cos}) enable the model to recognize that North (0°) and NNW (337.5°) are adjacent directions rather than distant values, eliminating discontinuities at boundary conditions.

Temporal Feature Extraction Time-based patterns significantly influence air pollutant concentrations due to natural diurnal and seasonal variations in emissions, atmospheric chemistry, and meteorological conditions. To enable our models to capture these temporal dependencies, we employed a circular embedding approach for all time-related variables.

For monthly patterns, we transform the month index $m \in \{1, 2, ..., 12\}$ into its circular components:

$$month_{sin} = \sin\left(\frac{2\pi \cdot m}{12}\right), \quad month_{cos} = \cos\left(\frac{2\pi \cdot m}{12}\right) \tag{5.1}$$

Similarly, for hourly patterns, we encoded each hour $h \in \{0, 1, ..., 23\}$ as:

$$hour_{sin} = \sin\left(\frac{2\pi \cdot h}{24}\right), \quad hour_{cos} = \cos\left(\frac{2\pi \cdot h}{24}\right) \tag{5.2}$$

This transformation preserves the cyclical nature of temporal variables, enabling the model to recognize that hour 23 is temporally adjacent to hour 0, and December (month 12) is adjacent to January (month 1). These circular embeddings provide a continuous, boundary-free representation of time that captures both seasonal and diurnal patterns critical for air quality forecasting.

Missing Value Imputation Air quality monitoring networks frequently experience sensor failures, maintenance periods, or communication issues, resulting in missing measurements. To maintain data continuity without introducing artificial patterns, we employed a bidirectional temporal imputation strategy. This approach prioritizes temporal proximity for missing value estimation through a sequential two-pass algorithm:

1. *Forward propagation*: Each missing value is estimated using the most recent valid observation, preserving the influence of immediate historical conditions.

2. *Backward propagation*: Any remaining gaps (typically at dataset beginnings) are filled by propagating future valid observations backward.

This bidirectional approach preserves local temporal patterns while ensuring complete data coverage, respecting the time-series nature of environmental monitoring data.

Feature Normalization Environmental datasets often exhibit considerable scale differences between variables. For example, atmospheric pressure (PRES) is typically measured around 1000 hPa, while temperature (TEMP) ranges between $-10°C$ and $40°C$. Such disparities can lead to model instability, slower convergence, or bias during training, where larger magnitude features dominate gradient updates.

To harmonize feature scales, we applied Min-Max normalization (Equation: 2.84) to all continuous variables, rescaling each feature individually to the $[0, 1]$ range as illustrated in Section B.

B. *LagEnsembleForecasting* Architecture and Model Development

To establish a robust foundation for $PM_{2.5}$ forecasting and to lay the groundwork for the proposed *LagEnsembleForecasting* framework, several pre-trained base models were developed, each utilizing different input lag sequences corresponding to 1-hour, 6-hour, 12-hour, and 24-hour historical windows. While the forecasting target remained consistent across all models, the architectures were specifically tailored to the nature of each lag setting, enabling the models to capture either short-term or long-term temporal dependencies with varying degrees of complexity.

The model architecture is carefully designed as follows:

- For short-term 1-hour forecasting, a lightweight LSTM model was used, composed of a single LSTM layer followed by a dense layer and a dropout layer.
- For 6-hour lag sequences, a deeper LSTM network was adopted, stacking two LSTM layers to better model the longer dependencies present in the input sequence.
- For 12-hour and 24-hour lag inputs, a hybrid CNN-LSTM model was introduced. A 1D convolutional layer was used to efficiently capture local temporal features, followed by stacked LSTM layers to model sequential dependencies at a broader scale.

A summary of the architectures is provided in Table 5.8.

Model Name	Input Lag Sequence	Architecture Design	Key Components
1h Lag Model	1 hour	Shallow LSTM model	1 LSTM layer, Dense, Dropout
6h Lag Model	6 hours	Stacked LSTM model	2 LSTM layers, Dense, Dropout
12h Lag Model	12 hours	Hybrid CNN-LSTM model	Conv1D, MaxPooling, 2 LSTM layers, Dense
24h Lag Model	24 hours	Hybrid CNN-LSTM model	Conv1D, MaxPooling, 2 LSTM layers, Dense

Table 5.8.: Summary of base model architectures for different input lags.

C. Pre-trained Model Training and Evaluation

The forecasting performance of these models was assessed using key evaluation metrics, namely Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2).

To further capitalize on the complementary strengths of the individual models trained on different lag sequences, an ensemble strategy was developed, forming the core of the

LagEnsembleForecasting architecture. By aggregating the diverse predictions generated at multiple temporal scales, the ensemble approach aims to integrate both short-term fluctuations and long-term trends, thereby enhancing the overall robustness, stability, and accuracy of $PM_{2.5}$ forecasts.

Table 5.9 summarizes the performance of the individual lag-based models and the ensemble model. The ensemble *LagEnsembleForecasting* consistently achieved the best performance, with an RMSE of 19.7710, an MAE of 11.4910, and an R^2 of 0.9548. Among the individual models, those using longer lag sequences (12-hour and 24-hour windows) outperformed models with shorter lag histories (1-hour and 6-hour), although none surpassed the ensemble model. These results indicate that combining predictions based on different historical depths helps capture a broader range of temporal dependencies, leading to more accurate and reliable forecasts.

Model Type	RMSE	MAE	R^2
LagEnsembleForecasting	**19.7710**	**11.4910**	**0.9548**
1h Lag Model	22.5222	13.8979	0.9413
6h Lag Model	22.5401	15.5545	0.9412
12h Lag Model	20.8160	13.1539	0.9499
24h Lag Model	20.8312	12.5036	0.9498

Table 5.9.: Overall performance comparison of $PM_{2.5}$ forecast models.

To further quantify the advantages of the ensemble *LagEnsembleForecasting* approach, Table 5.10 presents the relative performance improvements of the ensemble model over each individual lag-based model. The *LagEnsembleForecasting* demonstrated substantial reductions in RMSE, achieving improvements of 12.20% and 12.27% over the 1-hour and 6-hour lag models, respectively. MAE reductions were even more pronounced, particularly for the 6-hour model (26.12% improvement). In terms of R^2, gains ranged from 0.52% to 1.44%, reflecting enhanced model explanatory power when combining multiple historical perspectives.

Beyond the numerical evaluation, Figure 5.8 provides a visual comparison between the true $PM_{2.5}$ values and the *LagEnsembleForecasting* model's predictions for the first 100 samples, including uncertainty bands representing ± 1 standard deviation. The ensemble closely tracks

Lag Window	RMSE Improvement (%)	MAE Improvement (%)	R^2 Improvement (%)
1h Lag Model	12.20	17.33	1.43
6h Lag Model	12.27	26.12	1.44
12h Lag Model	5.02	12.64	0.52
24h Lag Model	5.08	8.11	0.53

Table 5.10.: Performance improvement of *LagEnsembleForecasting* model over individual lag-based models.

the real $PM_{2.5}$ trends across different fluctuation regimes. The relatively narrow uncertainty intervals for most periods indicate high model confidence, although slight underestimation is observed during certain extreme pollution peaks.

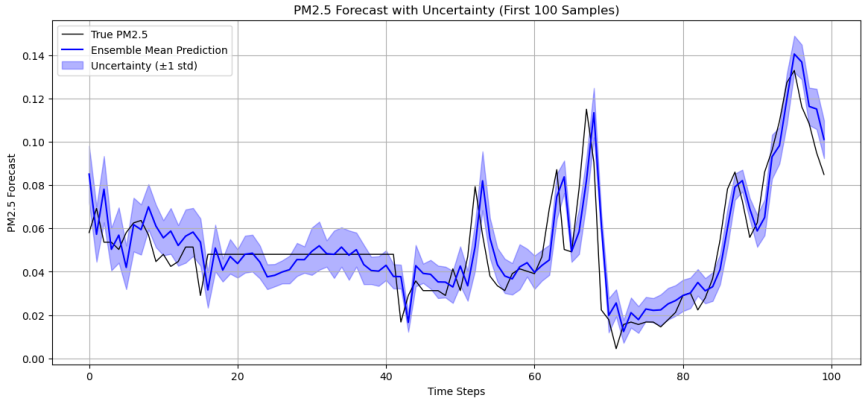


Figure 5.8.: *LagEnsembleForecasting* prediction and associated uncertainty compared to true $PM_{2.5}$ values (first 100 samples).

Overall, these results validate the effectiveness of the proposed ensemble learning strategy based on diverse input lag sequences. By leveraging multiple historical windows and explicitly modeling predictive uncertainty, the ensemble framework significantly enhances the accuracy, robustness, and interpretability of $PM_{2.5}$ forecasts.

D. Forecasting of Other Air Pollutants Based on the Pre-trained Model

Building on the successful training and evaluation of the pre-trained base *LagEnsemble-Forecasting* model for $PM_{2.5}$ forecasting, its transferability to the prediction of other air

pollutants was systematically explored. Fine-tuning was employed to adapt the pre-trained models to forecast concentrations of PM_{10}, SO_2, CO, O_3, and NO_2. The primary objective was to assess whether leveraging the temporal and meteorological representations learned from $PM_{2.5}$ data could enhance predictive performance compared to training models from scratch for each pollutant independently.

For this evaluation, pollutant-specific models were fine-tuned using historical data from the same monitoring station, ensuring consistent environmental conditions. To simulate realistic scenarios of data scarcity, only 20% of the available training data for each pollutant was utilized during fine-tuning. Corresponding models trained entirely from scratch—without any pre-training—served as baselines for comparison.

Two important adjustments were made during the fine-tuning phase to optimize the adaptation of the pre-trained models: (i) a smaller learning rate was adopted, specifically set to one-tenth of the initial learning rate used for training the base models, allowing for cautious and incremental updates to the model weights. This strategy helped preserve the beneficial representations learned during the initial training on the $PM_{2.5}$ dataset while minimizing the risk of overfitting to the limited fine-tuning data. (ii) Additionally, the number of training epochs for fine-tuning was reduced by half compared to the base model training, using only 50 epochs instead of 100.

PM_{10} Forecasting The fine-tuning approach demonstrated notable improvements in PM_{10} prediction performance. As shown in Table 5.11, the fine-tuned *LagEnsembleForecasting* model achieved an RMSE of 31.9671 and an R^2 of 0.8963, outperforming the baseline model trained from scratch (RMSE = 33.9706, R^2 = 0.8828). These results indicate that fine-tuning not only reduced the overall prediction error but also enhanced the model's ability to explain variance in the PM_{10} concentration trends.

Model	Fine-tuned (Improvement)			From Scratch		
	RMSE	MAE	R^2	RMSE	MAE	R^2
LagEnsembleForecasting	31.9671 (+5.89%)	17.9895 (+9.64%)	0.8963 (+1.53%)	33.9706	19.9117	0.8828
1h Lag Model	32.6997 (+18.77%)	18.3630 (+33.51%)	0.8914 (+6.69%)	40.2522	27.6259	0.8355
6h Lag Model	32.0459 (+4.98%)	17.9664 (+11.03%)	0.8957 (+1.26%)	33.7293	20.1928	0.8845
12h Lag Model	32.5380 (+5.87%)	18.6456 (+8.15%)	0.8925 (+1.57%)	34.5656	20.2973	0.8787
24h Lag Model	32.3879 (+11.15%)	18.7025 (+19.80%)	0.8935 (+3.28%)	36.4541	23.3302	0.8651

Table 5.11.: Performance comparison for PM_{10} forecasting (fine-tuned vs from scratch).

SO_2 Forecasting Fine-tuning also substantially enhanced the SO_2 forecasting performance. Table 5.12 shows that the fine-tuned *LagEnsembleForecasting* model achieved an RMSE of 4.8946 and an R^2 of 0.9199, compared to an RMSE of 5.9232 and an R^2 of 0.8828 for the baseline model. These results represent significant gains, particularly in capturing the variability of SO_2 concentrations.

Model	Fine-tuned (Improvement)			From Scratch		
	RMSE	MAE	R^2	RMSE	MAE	R^2
LagEnsembleForecasting	4.8946 (+17.36%)	2.8677 (+29.91%)	0.9199 (+4.21%)	5.9232	4.0912	0.8828
1h Lag Model	5.1602 (+9.80%)	2.9916 (+18.75%)	0.9110 (+2.29%)	5.7214	3.6822	0.8906
6h Lag Model	4.9053 (+11.92%)	2.8298 (+19.68%)	0.9196 (+2.60%)	5.5698	3.5224	0.8963
12h Lag Model	5.0685 (+26.51%)	3.1069 (+37.31%)	0.9142 (+8.70%)	6.8975	4.9555	0.8410
24h Lag Model	5.0756 (+44.46%)	3.0684 (+57.16%)	0.9139 (+26.83%)	9.1394	7.1620	0.7209

Table 5.12.: Performance comparison for SO_2 forecasting (fine-tuned vs from scratch).

CO Forecasting The predictive performance for CO concentrations, summarized in Table 5.13, also benefited from the fine-tuning approach. The *LagEnsembleForecasting* model achieved a reduced RMSE of 388.1997 and an increased R^2 of 0.9218, compared to 405.6656 and 0.9146, respectively, for the model trained from scratch.

Model	Fine-tuned (Improvement)			From Scratch		
	RMSE	MAE	R^2	RMSE	MAE	R^2
LagEnsembleForecasting	388.1997 (+4.31%)	185.0608 (+13.40%)	0.9218 (+0.78%)	405.6656	213.6611	0.9146
1h Lag Model	405.7879 (+5.21%)	194.6102 (+19.43%)	0.9146 (+1.07%)	428.1425	241.5813	0.9049
6h Lag Model	397.7637 (+7.99%)	186.7120 (+14.58%)	0.9179 (+1.64%)	432.2130	218.5825	0.9031
12h Lag Model	390.9671 (+11.61%)	192.3781 (+24.04%)	0.9207 (+2.47%)	442.4078	253.2832	0.8985
24h Lag Model	397.9321 (+6.75%)	195.2837 (+20.57%)	0.9179 (+1.36%)	426.7219	245.8722	0.9055

Table 5.13.: Performance comparison for CO forecasting (fine-tuned vs from scratch).

O_3 Forecasting The benefits of fine-tuning are also evident in O_3 forecasting. As presented in Table 5.14, the fine-tuned *LagEnsembleForecasting* model achieved an RMSE of 13.3708 and an R^2 of 0.9477, compared to an RMSE of 14.2935 and an R^2 of 0.9403 for the model trained from scratch. This represents a reduction in prediction error of approximately 6.45% and a meaningful improvement in model explanatory power.

Model	Fine-tuned (Improvement)			From Scratch		
	RMSE	MAE	R^2	RMSE	MAE	R^2
LagEnsembleForecasting	13.3708 (+6.45%)	8.6093 (+11.89%)	0.9477 (+0.79%)	14.2935	9.7730	0.9403
1h Lag Model	13.7215 (+1.07%)	8.8721 (+6.58%)	0.9450 (+0.13%)	13.8701	9.4950	0.9438
6h Lag Model	13.5974 (+1.22%)	8.8404 (+2.88%)	0.9460 (+0.15%)	13.7657	9.1032	0.9446
12h Lag Model	13.9137 (+11.37%)	9.0146 (+16.69%)	0.9434 (+1.67%)	15.7010	10.8189	0.9279
24h Lag Model	13.5661 (+18.23%)	8.9471 (+24.58%)	0.9462 (+2.90%)	16.5925	11.8634	0.9195

Table 5.14.: Performance comparison for O_3 forecasting (fine-tuned vs from scratch).

NO_2 Forecasting The improvement trend continues for NO_2 forecasting. As summarized in Table 5.15, the fine-tuned *LagEnsembleForecasting* model attained an RMSE of 395.7345 and an R^2 of 0.9140, showing better performance than the from-scratch model (RMSE = 423.3294, R^2 = 0.9016).

Model	Fine-tuned (Improvement)			From Scratch		
	RMSE	MAE	R^2	RMSE	MAE	R^2
LagEnsembleForecasting	395.7345 (+6.52%)	258.7563 (+9.83%)	0.9140 (+1.38%)	423.3294	287.0196	0.9016
1h Lag Model	404.3940 (+17.66%)	263.7077 (+25.47%)	0.9102 (+4.90%)	490.9885	353.7058	0.8676
6h Lag Model	396.9733 (+4.22%)	260.2061 (+8.18%)	0.9135 (+0.86%)	414.4160	283.3895	0.9057
12h Lag Model	404.5316 (+9.29%)	271.4678 (+11.46%)	0.9101 (+2.16%)	445.9710	306.5778	0.8908
24h Lag Model	408.8325 (+10.30%)	272.2324 (+15.49%)	0.9082 (+2.53%)	455.7736	322.1671	0.8859

Table 5.15.: Performance comparison for NO_2 forecasting (fine-tuned vs from scratch).

Overall, the experimental results demonstrate the substantial benefits of fine-tuning pre-trained $PM_{2.5}$ models for forecasting other key air pollutants. Across PM_{10}, SO_2, CO, O_3, and NO_2, the fine-tuned models consistently outperformed the models trained from scratch, achieving lower prediction errors and higher R^2 values.

These findings validate the effectiveness of transfer learning in enhancing multi-pollutant forecasting performance, particularly under data scarcity conditions. They also highlight the potential of the proposed *LagEnsembleForecasting* framework for deployment in real-world air quality monitoring and decision-support systems, where accurate and reliable multi-pollutant forecasting is essential.

Building upon these promising results, the next section explores potential extensions of the framework, including its generalization across multiple monitoring sites, pollutant categories, and broader spatiotemporal scales.

5.4.2. Discussion

This study establishes the substantial effectiveness of the proposed *LagEnsembleForecasting* framework for comprehensive air quality forecasting through its innovative integration of multi-temporal perspectives and advanced ensemble learning strategies. The framework's ability to aggregate predictions from specialized models trained on distinct historical lag sequences (1h, 6h, 12h, and 24h) represents a significant methodological advancement that addresses fundamental limitations in existing air quality prediction approaches.

Multi-Scale Temporal Learning and Ensemble Performance

The ensemble methodology demonstrated superior predictive performance by strategically leveraging the complementary strengths inherent in different temporal scales. Short-lag models (1h, 6h) excel at capturing immediate atmospheric fluctuations and rapid emission changes, while longer-lag models (12h, 24h) effectively model broader meteorological patterns and pollutant accumulation dynamics. This multi-scale integration proved particularly valuable for pollutants exhibiting complex temporal behaviors—such as NO_2 with its pronounced diurnal variability driven by traffic patterns, and SO_2 with its slower background variations influenced by industrial emissions and atmospheric chemistry.

The consistent superiority of the *LagEnsembleForecasting* framework across all evaluated pollutants (PM_{10}, SO_2, CO, O_3, and NO_2) validates the hypothesis that temporal diversity enhances model generalization. The ensemble achieved substantial improvements over individual base models, with RMSE reductions ranging from 5.02% to 12.27% and R^2 improvements of 0.52% to 1.44%, demonstrating that the fusion of multi-temporal information significantly enhances the model's capacity to adapt to varying pollutant dynamics.

Uncertainty Quantification and Predictive Reliability

A critical innovation of this framework lies in its natural provision of uncertainty quantification through ensemble variance estimation. The generation of confidence intervals (mean $\pm$ standard deviation) addresses a significant gap in current air quality forecasting systems, where point predictions without uncertainty estimates limit practical applicability for risk assessment and decision-making processes.

Robustness and Operational Advantages

The ensemble architecture inherently provides enhanced robustness by mitigating individual model vulnerabilities to temporal anomalies, data noise, and missing patterns. While single-lag models may exhibit sensitivity to specific temporal features or data irregularities, the

ensemble approach distributes prediction responsibility across multiple temporal perspectives, resulting in more stable and reliable forecasts. This robustness is crucial for real-world deployment scenarios where air quality monitoring networks often experience sensor failures, communication disruptions, or data quality issues.

Furthermore, the ensemble framework's compatibility with transfer learning strategies represents a significant operational advantage. The preservation of diverse temporal representations during fine-tuning ensures that the ensemble maintains its generalization capabilities when adapted to new pollutants, enabling efficient knowledge transfer without compromising predictive performance or uncertainty quantification accuracy.

Implications for Multi-Pollutant Forecasting Systems

The demonstrated effectiveness of cross-pollutant transfer learning within the ensemble framework has important implications for scalable air quality monitoring systems. The ability to leverage knowledge learned from $PM_{2.5}$ forecasting to enhance predictions for other pollutants addresses practical constraints in data-scarce environments and reduces computational overhead associated with training independent models for each pollutant.

This capability is particularly valuable for emerging monitoring networks or regions with limited historical data, where the proposed framework can bootstrap reliable forecasting capabilities by transferring knowledge from well-established monitoring stations or pollutants with extensive historical records.

Future Directions and Broader Impact

The *LagEnsembleForecasting* framework establishes a foundation for several promising research directions. Future work could explore the integration of additional temporal scales, incorporation of spatial information for multi-station ensemble forecasting, and adaptation to other environmental monitoring applications beyond air quality. The framework's uncertainty quantification capabilities also open opportunities for developing more sophisticated decision-support systems that can optimize public health interventions based on prediction confidence levels.

In conclusion, this study demonstrates that the combination of multi-lag ensemble learning with cross-pollutant transfer learning represents a significant advancement toward building more reliable, interpretable, and scalable air quality forecasting systems. By providing both high-fidelity predictions and associated uncertainty estimates, the proposed framework addresses critical gaps in current methodologies and offers a robust foundation for operational air quality management and public health protection.

5.4.3. Summary and Conclusion

This research presents a comprehensive investigation into the transformative potential of combining multi-lag ensemble learning with cross-pollutant transfer learning for advanced air quality forecasting. The proposed *LagEnsembleForecasting* framework addresses two critical limitations in current air pollution prediction methodologies: the challenge of capturing multi-scale temporal dependencies and the inefficiency of developing independent models for each pollutant.

Methodological Contributions and Technical Achievements The study introduces a novel ensemble architecture that integrates four specialized deep learning models, each optimized for distinct temporal horizons (1h, 6h, 12h, and 24h), enabling comprehensive capture of both immediate atmospheric fluctuations and longer-term pollutant accumulation patterns. Initially developed for $PM_{2.5}$ forecasting, this ensemble framework was systematically adapted through transfer learning to predict concentrations of five additional pollutants—PM_{10}, SO_2, NO_2, CO, and O_3—demonstrating remarkable versatility and knowledge transferability across different atmospheric species.

Comprehensive evaluations using rigorous performance metrics (MAE, MSE, RMSE, and R^2) consistently demonstrated the superiority of the transfer learning approach over traditional from-scratch training methodologies. Most notably, the framework achieved exceptional predictive performance even when fine-tuned with only 20% of target pollutant data, establishing its effectiveness in data-constrained environments typical of emerging monitoring networks or newly introduced pollutant measurements.

Practical Implications and Operational Advantages The research findings reveal several transformative benefits for operational air quality monitoring systems:

Enhanced Predictive Capabilities: The multi-lag ensemble approach captures complementary temporal patterns that individual models cannot detect, resulting in substantial improvements in forecast accuracy and reliability. The framework's uncertainty quantification capabilities provide critical confidence estimates essential for risk assessment and decision-making processes.

Unprecedented Data Efficiency: Transfer learning enables rapid model adaptation to new pollutant targets with minimal data requirements, reducing the typical 2-3 year data

collection period needed for reliable forecasting model development to mere weeks or months.

Computational and Economic Efficiency: The framework significantly reduces training computational overhead while maintaining superior performance, making advanced air quality forecasting accessible to resource-constrained monitoring agencies and developing regions.

Scalable Architecture: The modular design facilitates seamless expansion to additional pollutants, monitoring stations, or temporal resolutions without requiring fundamental architectural modifications or extensive retraining procedures.

Operational Robustness: The ensemble approach inherently provides resilience against individual model failures, data anomalies, and temporal irregularities common in real-world monitoring environments.

Scientific Significance and Broader Impact This work establishes a new paradigm for environmental monitoring that bridges fundamental atmospheric science with advanced machine learning methodologies. The demonstration that knowledge learned from one pollutant can effectively enhance predictions for chemically and behaviorally distinct species opens new avenues for understanding inter-pollutant relationships and atmospheric dynamics.

The framework's uncertainty quantification capabilities address a critical gap in current air quality forecasting systems, enabling more informed public health advisories, emergency response protocols, and regulatory decision-making. This is particularly valuable for protecting vulnerable populations and optimizing resource allocation during pollution episodes.

Future Research Directions and Implementation Pathways The success of this research establishes several promising directions for future development. Immediate extensions could incorporate spatial information for multi-station ensemble forecasting, integrate additional meteorological variables or emission source data, and explore applications to other environmental monitoring domains such as water quality or soil contamination assessment.

Long-term research opportunities include developing adaptive learning mechanisms that continuously update model parameters as new data becomes available, investigating the framework's applicability to climate change scenarios with shifting pollutant patterns, and creating specialized modules for extreme pollution event prediction and early warning systems.

Concluding Remarks This research validates the transformative potential of combining advanced ensemble learning with transfer learning principles for air quality forecasting. The *LagEnsembleForecasting* framework represents a significant methodological advancement that addresses fundamental challenges in environmental monitoring while providing practical solutions for operational deployment.

By offering accurate, reliable, and uncertainty-aware predictions with minimal data requirements, this framework democratizes access to advanced air quality forecasting capabilities and establishes a robust foundation for next-generation environmental monitoring systems. The demonstrated effectiveness across multiple pollutants and temporal scales positions this approach as a cornerstone technology for protecting public health and supporting evidence-based environmental management in an era of increasing atmospheric complexity and monitoring demands.

5.5. Enhancing Air Quality Forecasting Accuracy at Multiple Temporal Resolutions Through Deep Transfer Learning

5.5.1. Proposed Models and Result Analysis

This section presents the design, implementation, and evaluation of neural network models developed for forecasting hourly $PM_{2.5}$ concentrations, and their extension to coarser temporal resolutions (6-hour, 12-hour, and daily) using transfer learning. The proposed approach builds on knowledge captured from hourly predictions and adapts it to improve accuracy and efficiency in higher-resolution forecasts. This approach enables efficient model adaptation across different temporal scales while maintaining forecasting accuracy. Figure 5.9 illustrates the methodology framework proposed in this study.

A. Pre-trained Models for Hourly $PM_{2.5}$ Forecasting

Dataset Description The study utilizes the Beijing $PM_{2.5}$ dataset (Section: 2.7.2), comprising hourly air quality data and meteorological variables recorded from 2010 to 2014, totaling 43,824 entries.

Data Preprocessing and Feature Engineering The $PM_{2.5}$ column contains approximately 4.71% missing values as illustrated in Table 2.5, Section 2.7.2. To address this, we

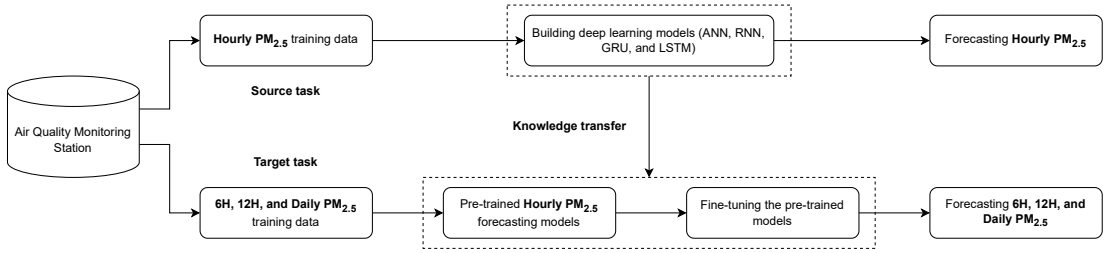


Figure 5.9.: Methodology of the proposed transfer learning approach for different temporal resolutions.

applied k-nearest neighbors (kNN) imputation (Section: 2.3.6), which leverages correlations between environmental variables (e.g., temperature, pressure, wind speed) to estimate missing values. Prior to imputation, all features were standardized using standardization formula (Equation: 2.85)

Categorical features, such as wind direction (CBWD), were encoded using one-hot encoding. Additionally, autocorrelation and partial autocorrelation (ACF/PACF) plots (Figure 5.10) were used to determine the optimal lag input, with PACF suggesting a strong correlation at lag-1, which was used in all models.

Neural Architectures and Hyperparameter Optimization We developed and compared four deep learning models: ANN, RNN, GRU, and LSTM. Hyperparameters were optimized using Optuna (Section: 2.6.2), with results summarized in Table 5.16.

B. Training and Evaluation of Pre-trained Models

The models were trained on 60% of the data, validated on 20%, and tested on the remaining 20%. Training used the Adam optimizer and MSE loss. Figure 5.11 shows the convergence of training and validation losses. Performance metrics for hourly $PM_{2.5}$ predictions are presented in Table 5.17.

We further validated the models using 5-fold TimeSeriesSplit cross-validation (Table 5.18), confirming the models' generalizability and robustness.

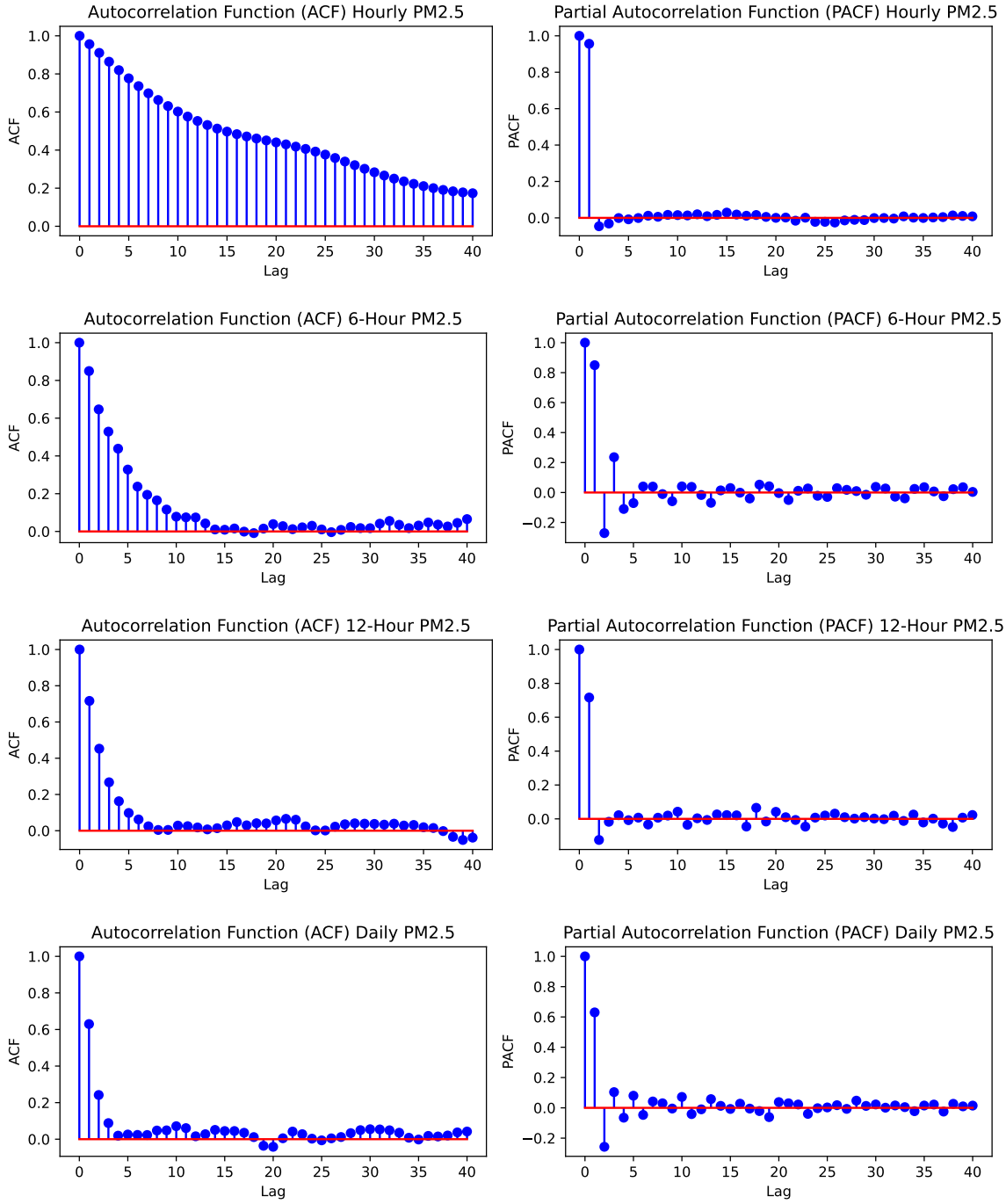


Figure 5.10.: ACF and PACF plots of hourly, 6-hour, 12-hour, and daily data.

Hyperparameter	Search Space	ANN	RNN	GRU	LSTM
Number of Layers	{1, 2}	1	1	1	1
Hidden Units	{8, 16, 32, 64}	32	32	64	64
Dropout Rate	{0, 0.1, 0.2, 0.3}	0.1	0.1	0.1	0.1
Learning Rate	{1e-5, 1e-4, 1e-3, 1e-2, 1e-1}	1e-3	1e-3	1e-3	1e-3
Batch Size	{8, 16, 32, 64}	32	32	32	32
Regularization Type	{none, L1, L2}	none	none	none	none
Regularization Strength	{1e-4, 1e-3, 1e-2, 1e-1, 1}	0	0	0	0
Epochs	Fixed	50	50	50	50

Table 5.16.: Hyperparameter optimization results for different neural network architectures.

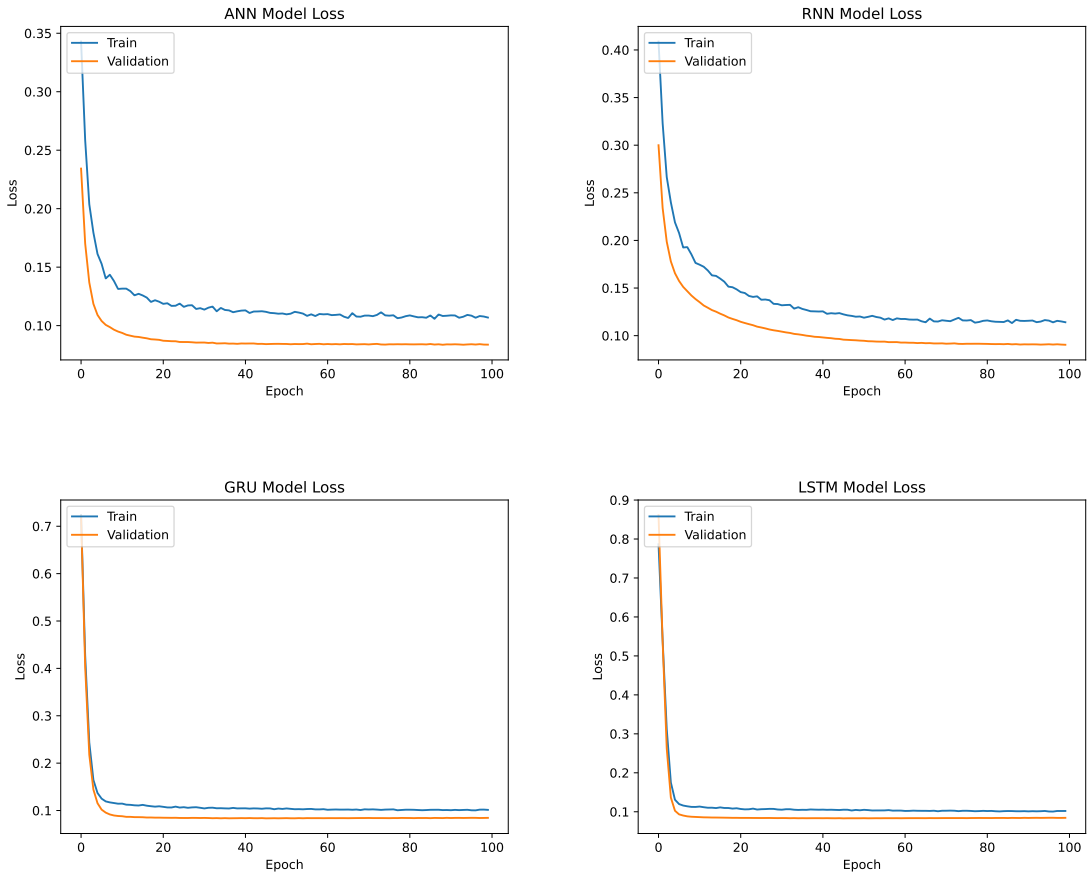


Figure 5.11.: Training and validation loss curves of ANN, RNN, GRU, and LSTM.

Model	MAE	MSE	RMSE	R^2
ANN/MLP	0.1451	0.0658	0.2566	0.9420
Simple RNN	0.1449	0.0659	0.2567	0.9419
GRU	0.1449	0.0661	0.2572	0.9417
LSTM	0.1446	0.0663	0.2575	0.9416

Table 5.17.: Performance metrics of the base (pre-trained) models for hourly $PM_{2.5}$ forecasting.

C. Transfer Learning for Higher Temporal Resolutions

Building on the pre-trained hourly models, we applied transfer learning to forecast $PM_{2.5}$ concentrations at 6-hour, 12-hour, and daily intervals. The pre-trained models were fine-tuned using various data proportions (10% to 50%), and compared to models trained from scratch.

Tables 5.19, 5.20, and 5.21 provide a detailed comparison of forecasting models performance for different temporal resolutions, 6-hour, 12-hour, and daily respectively. It contrasts two approaches: fine-tuning the pre-trained hourly $PM_{2.5}$ models with data sizes ranging from 10% to 50% and training new models from scratch. The comparison is based on metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2).

Performance Summary Key findings include:

1. **Limited Data Efficiency:**
   - Significant improvements were achieved using only 10% of target data.
   - LSTM and GRU showed the largest performance gains, especially in daily predictions (R^2 improvements up to 242.87% and 182.17%, respectively).
2. **Model-Specific Trends:**
   - LSTM and GRU consistently outperformed ANN and RNN across all resolutions.

Model	Fold	MSE	MAE	RMSE	R^2
ANN	1	0.1176	0.2001	0.3429	0.9119
	2	0.1015	0.1762	0.3185	0.9073
	3	0.0937	0.1624	0.3062	0.9276
	4	0.0577	0.1415	0.2403	0.9417
	5	0.0493	0.1255	0.2220	0.9358
	Average	0.0840	0.1611	0.2860	0.9249
RNN	1	0.1178	0.2005	0.3432	0.9117
	2	0.1007	0.1749	0.3173	0.9080
	3	0.0929	0.1597	0.3048	0.9282
	4	0.0578	0.1407	0.2403	0.9417
	5	0.0499	0.1271	0.2233	0.9350
	Average	0.0838	0.1606	0.2858	0.9249
GRU	1	0.1324	0.2059	0.3639	0.9008
	2	0.0993	0.1731	0.3152	0.9093
	3	0.0935	0.1609	0.3058	0.9277
	4	0.0579	0.1417	0.2406	0.9416
	5	0.0488	0.1235	0.2208	0.9364
	Average	0.0864	0.1610	0.2893	0.9232
LSTM	1	0.1280	0.2032	0.3578	0.9041
	2	0.1009	0.1744	0.3176	0.9079
	3	0.0933	0.1612	0.3054	0.9279
	4	0.0583	0.1423	0.2415	0.9411
	5	0.0489	0.1239	0.2212	0.9362
	Average	0.0859	0.1610	0.2887	0.9234

Table 5.18.: Performance metrics of different models with 5-fold cross-validation.

Model	Split	From-Scratch Results			Fine-Tuned Results (Gain)		
		MAE	RMSE	R^2	MAE	RMSE	R^2
ANN	0.1	0.4828	0.7107	0.6985	0.4316 (+**10.61%**)	0.6521 (+**8.24%**)	0.7462 (+**6.83%**)
	0.2	0.3598	0.5281	0.7338	0.3363 (+**6.53%**)	0.5056 (+**4.26%**)	0.7561 (+**3.04%**)
	0.3	0.3570	0.5275	0.7355	0.3360 (+**5.88%**)	0.5088 (+**3.54%**)	0.7539 (+**2.50%**)
	0.4	0.3435	0.5116	0.7464	0.3273 (+**4.71%**)	0.4973 (+**2.80%**)	0.7603 (+**1.86%**)
	0.5	0.3461	0.5206	0.7538	0.3317 (+**4.16%**)	0.5078 (+**2.46%**)	0.7657 (+**1.58%**)
RNN	0.1	0.4846	0.7122	0.6973	0.4343 (+**10.38%**)	0.6542 (+**8.14%**)	0.7446 (+**6.78%**)
	0.2	0.3605	0.5286	0.7334	0.3389 (+**5.99%**)	0.5061 (+**4.26%**)	0.7555 (+**3.01%**)
	0.3	0.3572	0.5274	0.7356	0.3386 (+**5.21%**)	0.5107 (+**3.17%**)	0.7521 (+**2.24%**)
	0.4	0.3433	0.5113	0.7466	0.3291 (+**4.14%**)	0.4991 (+**2.39%**)	0.7586 (+**1.61%**)
	0.5	0.3457	0.5201	0.7543	0.3331 (+**3.64%**)	0.5093 (+**2.08%**)	0.7643 (+**1.33%**)
GRU	0.1	0.5706	0.8557	0.5630	0.4290 (+**24.82%**)	0.6477 (+**24.31%**)	0.7496 (+**33.14%**)
	0.2	0.3624	0.5411	0.7206	0.3346 (+**7.67%**)	0.5022 (+**7.19%**)	0.7593 (+**5.37%**)
	0.3	0.3478	0.5204	0.7426	0.3352 (+**3.62%**)	0.5067 (+**2.63%**)	0.7560 (+**1.80%**)
	0.4	0.3356	0.5035	0.7543	0.3262 (+**2.80%**)	0.4943 (+**1.83%**)	0.7632 (+**1.18%**)
	0.5	0.3387	0.5122	0.7617	0.3305 (+**2.42%**)	0.5060 (+**1.21%**)	0.7674 (+**0.75%**)
LSTM	0.1	0.6563	0.9232	0.4913	0.4309 (+**34.35%**)	0.6511 (+**29.47%**)	0.7470 (+**52.04%**)
	0.2	0.3721	0.5488	0.7125	0.3361 (+**9.67%**)	0.5045 (+**8.07%**)	0.7571 (+**6.26%**)
	0.3	0.3498	0.5229	0.7401	0.3362 (+**3.89%**)	0.5075 (+**2.95%**)	0.7552 (+**2.04%**)
	0.4	0.3345	0.5038	0.7540	0.3274 (+**2.12%**)	0.4964 (+**1.47%**)	0.7612 (+**0.95%**)
	0.5	0.3380	0.5138	0.7602	0.3316 (+**1.89%**)	0.5080 (+**1.13%**)	0.7656 (+**0.71%**)

Table 5.19.: Performance comparison of different models across various split ratios, including fine-tuning results for 6-hour forecasting.

Model	Split	From-Scratch Results			Fine-Tuned Results (Gain)		
		MAE	RMSE	R^2	MAE	RMSE	R^2
ANN	0.1	0.6247	0.8932	0.5399	0.5763 (+**7.75%**)	0.8311 (+**6.95%**)	0.6016 (+**11.43%**)
	0.2	0.4743	0.6696	0.5776	0.4468 (+**5.80%**)	0.6447 (+**3.72%**)	0.6084 (+**5.34%**)
	0.3	0.4765	0.6733	0.5682	0.4484 (+**5.90%**)	0.6498 (+**3.49%**)	0.5978 (+**5.21%**)
	0.4	0.4616	0.6603	0.5792	0.4397 (+**4.74%**)	0.6422 (+**2.74%**)	0.6020 (+**3.93%**)
	0.5	0.4756	0.6830	0.5806	0.4533 (+**4.69%**)	0.6629 (+**2.94%**)	0.6048 (+**4.17%**)
RNN	0.1	0.6254	0.8938	0.5392	0.5767 (+**7.79%**)	0.8319 (+**6.93%**)	0.6008 (+**11.42%**)
	0.2	0.4746	0.6698	0.5773	0.4478 (+**5.65%**)	0.6446 (+**3.76%**)	0.6085 (+**5.40%**)
	0.3	0.4766	0.6733	0.5683	0.4480 (+**6.00%**)	0.6492 (+**3.58%**)	0.5985 (+**5.31%**)
	0.4	0.4612	0.6601	0.5795	0.4405 (+**4.49%**)	0.6426 (+**2.65%**)	0.6016 (+**3.81%**)
	0.5	0.4755	0.6830	0.5806	0.4534 (+**4.65%**)	0.6622 (+**3.04%**)	0.6057 (+**4.32%**)
GRU	0.1	0.7682	1.0852	0.3207	0.5707 (+**25.71%**)	0.8237 (+**24.10%**)	0.6087 (+**89.80%**)
	0.2	0.5193	0.7325	0.4944	0.4413 (+**15.02%**)	0.6373 (+**13.00%**)	0.6173 (+**24.86%**)
	0.3	0.4805	0.6803	0.5591	0.4428 (+**7.85%**)	0.6429 (+**5.50%**)	0.6063 (+**8.44%**)
	0.4	0.4538	0.6541	0.5872	0.4348 (+**4.19%**)	0.6343 (+**3.03%**)	0.6117 (+**4.17%**)
	0.5	0.4638	0.6710	0.5952	0.4479 (+**3.43%**)	0.6565 (+**2.16%**)	0.6125 (+**2.90%**)
LSTM	0.1	0.8238	1.1355	0.2564	0.5729 (+**30.46%**)	0.8256 (+**27.29%**)	0.6068 (+**136.66%**)
	0.2	0.5811	0.7926	0.4082	0.4436 (+**23.66%**)	0.6394 (+**19.33%**)	0.6148 (+**50.61%**)
	0.3	0.5092	0.7037	0.5283	0.4446 (+**12.69%**)	0.6444 (+**8.43%**)	0.6045 (+**14.42%**)
	0.4	0.4528	0.6497	0.5926	0.4363 (+**3.64%**)	0.6361 (+**2.09%**)	0.6095 (+**2.85%**)
	0.5	0.4602	0.6667	0.6004	0.4488 (+**2.48%**)	0.6574 (+**1.39%**)	0.6114 (+**1.83%**)

Table 5.20.: Performance comparison of different models across various split ratios, including fine-tuning results for 12-hour forecasting.

Model	Split	From-Scratch Results			Fine-Tuned Results (Gain)		
		MAE	RMSE	R-squared	MAE	RMSE	R^2
ANN	0.1	0.7818	1.0833	0.3525	0.7393 (+**5.44%**)	1.0293 (+**4.99%**)	0.4155 (+**17.87%**)
	0.2	0.5920	0.8053	0.3978	0.5674 (+**4.15%**)	0.7837 (+**2.68%**)	0.4297 (+**8.02%**)
	0.3	0.5853	0.7999	0.3801	0.5670 (+**3.13%**)	0.7812 (+**2.34%**)	0.4088 (+**7.55%**)
	0.4	0.5767	0.7913	0.3885	0.5614 (+**2.65%**)	0.7727 (+**2.35%**)	0.4169 (+**7.31%**)
	0.5	0.5959	0.8194	0.3944	0.5772 (+**3.14%**)	0.7988 (+**2.51%**)	0.4244 (+**7.61%**)
RNN	0.1	0.7818	1.0834	0.3524	0.7384 (+**5.55%**)	1.0288 (+**5.04%**)	0.4160 (+**18.05%**)
	0.2	0.5921	0.8054	0.3977	0.5678 (+**4.10%**)	0.7840 (+**2.66%**)	0.4293 (+**7.94%**)
	0.3	0.5854	0.7999	0.3801	0.5680 (+**2.97%**)	0.7827 (+**2.15%**)	0.4065 (+**6.94%**)
	0.4	0.5768	0.7912	0.3886	0.5608 (+**2.77%**)	0.7728 (+**2.32%**)	0.4167 (+**7.23%**)
	0.5	0.5958	0.8193	0.3945	0.5766 (+**3.22%**)	0.7986 (+**2.53%**)	0.4247 (+**7.65%**)
GRU	0.1	0.8937	1.2385	0.1537	0.7284 (+**18.50%**)	1.0131 (+**18.20%**)	0.4337 (+**182.17%**)
	0.2	0.6663	0.8972	0.2525	0.5611 (+**15.79%**)	0.7727 (+**13.88%**)	0.4456 (+**76.48%**)
	0.3	0.6365	0.8489	0.3018	0.5613 (+**11.81%**)	0.7717 (+**9.09%**)	0.4231 (+**40.19%**)
	0.4	0.5885	0.7982	0.3778	0.5527 (+**6.09%**)	0.7600 (+**4.79%**)	0.4359 (+**15.38%**)
	0.5	0.5934	0.8148	0.4012	0.5676 (+**4.35%**)	0.7859 (+**3.55%**)	0.4428 (+**10.37%**)
LSTM	0.1	0.9145	1.2590	0.1255	0.7301 (+**20.16%**)	1.0161 (+**19.29%**)	0.4303 (+**242.87%**)
	0.2	0.7017	0.9377	0.1835	0.5621 (+**19.90%**)	0.7743 (+**17.43%**)	0.4433 (+**141.58%**)
	0.3	0.6743	0.8936	0.2263	0.5619 (+**16.67%**)	0.7728 (+**13.52%**)	0.4215 (+**86.24%**)
	0.4	0.6192	0.8247	0.3357	0.5544 (+**10.46%**)	0.7618 (+**7.63%**)	0.4333 (+**29.07%**)
	0.5	0.6090	0.8215	0.3912	0.5695 (+**6.49%**)	0.7876 (+**4.13%**)	0.4405 (+**12.60%**)

Table 5.21.: Performance comparison of different models across various split ratios, including fine-tuning results for daily forecasting.

- ANN and RNN displayed stable, though more moderate, improvements.

3. **Temporal Horizon Insights:**
   - Longer-term predictions (daily) benefited most from transfer learning.
   - Intermediate gains were observed for 12-hour forecasts.
   - 6-hour forecasts showed modest but consistent improvements.
4. **Practical Implications:**
   - Transfer learning offers an efficient approach in data-scarce environments.
   - Enables scalable deployment of air quality forecasting models across varying temporal requirements.

These results demonstrate the versatility and effectiveness of transfer learning in environmental forecasting tasks, particularly under limited data conditions. The consistent improvement across models and horizons underscores the practical relevance of this approach for operational air quality monitoring systems.

5.5.2. Discussion

The application of transfer learning across different temporal resolutions in air quality forecasting represents a notable advancement. Our findings show significant improvements in forecasting accuracy, with R^2 gains of up to 242.87% for LSTM models and 182.17% for GRU models at the daily prediction horizon. These substantial improvements—especially when using only 10% of the data for fine-tuning—highlight the potential of this approach for enhancing environmental monitoring systems. By adapting models pre-trained for hourly $PM_{2.5}$ forecasting to longer temporal scales (6-hour, 12-hour, and daily), this method demonstrates clear benefits in both predictive performance and computational efficiency.

A key outcome of this study is the demonstrated effectiveness of transfer learning in improving forecasting accuracy across different temporal resolutions. Fine-tuning pre-trained models consistently outperformed training from scratch, particularly in data-scarce settings. These improvements were observed across various model architectures and time scales.

A. Experimental Analysis Patterns

The experimental analysis uncovered several important patterns in temporal resolution adaptation:

- **Performance with Limited Data:**
  - At 10% fine-tuning size, models achieved substantial improvements:
    - Daily predictions: LSTM (R^2 +242.87%), GRU (R^2 +182.17%)
    - 12-hour predictions: GRU (MAE +18.50%), LSTM (MAE +20.16%)
    - 6-hour predictions: LSTM (MAE +34.35%), GRU (MAE +24.82%)
  - MAE reductions across all time scales, with 6–8% improvements for most models
  - Particularly strong performance observed for longer-term (daily) predictions
- **Architectural Considerations:**
  - LSTM and GRU models delivered the most notable performance gains:
    - LSTM: Peak MAE improvements of 34.35% (6-hour), 20.16% (12-hour and daily)
    - GRU: Consistent gains across all scales with peak MAE reductions of 24.82% (6-hour), 18.50% (12-hour and daily)
  - ANN and RNN models showed stable but more moderate improvements:
    - ANN: Steady 5–10% gains across all time scales
    - RNN: Similar trend with slightly more variability
- **Temporal Resolution Patterns:**
  - Daily forecasts showed the greatest relative benefit from transfer learning
  - 6-hour forecasts yielded consistent but smaller improvements
  - 12-hour forecasts exhibited intermediate performance gains

B. Practical Implications

The demonstrated effectiveness of transfer learning across time scales has several practical benefits for air quality monitoring systems:

1. **Resource Efficiency:**
    - High accuracy achieved with only 10% of fine-tuning data
    - Strong R^2 improvements for daily predictions (50–242%)
    - Stable MAE improvements of 3–20% with minimal data
    - Lower computational costs than training from scratch
    - Faster deployment of models for new temporal resolutions
2. **Operational Flexibility:**
    - Models can quickly adapt to different prediction horizons
    - Maintained accuracy across multiple time scales
    - Efficient use of computational and data resources depending on forecasting needs

These findings are particularly relevant for real-world applications where collecting data at longer time intervals can be difficult or expensive. The ability to reuse hourly-trained models for accurate forecasting at larger temporal resolutions marks a significant step forward in scalable and efficient air quality monitoring.

C. Limitations and Future Work

Current Limitations:

- The analysis is limited to single-station temporal adaptation
- Performance with larger fine-tuning sizes (above 30%) varies and requires further investigation
- Differences in performance across architectures (e.g., LSTM vs. ANN) warrant deeper analysis
- Alternative fine-tuning strategies were not exhaustively explored
- More analysis is needed on how temporal resolution affects model architecture performance

Future Research Directions:

- Extension to multi-station and multi-regional transfer learning
- Development of adaptive fine-tuning strategies optimized for different temporal resolutions
- Exploration of hybrid or ensemble architectures for multi-resolution forecasting
- Study of seasonal variations and their influence on model transferability
- Investigation into optimal fine-tuning size selection approaches
- Analysis of model robustness under varying pollution level scenarios

In conclusion, this study confirms the effectiveness of transfer learning for adapting air quality forecasting models to different temporal resolutions. This approach is especially valuable in data-limited situations and supports the development of scalable, accurate, and resource-efficient forecasting systems. These findings contribute to improving environmental monitoring practices and offer new directions for the use of AI in addressing air pollution challenges.

5.5.3. Summary and Conclusion

This study explores the application of transfer learning in environmental research, focusing on the task of forecasting air pollutant concentrations across different temporal resolutions—from hourly to 6-hour, 12-hour, and daily intervals. We developed and adapted several pre-trained neural network models (ANN/MLP, RNN, GRU, and LSTM), originally trained to predict hourly $PM_{2.5}$ levels, for forecasting at larger temporal scales. By fine-tuning these models using varying amounts of target resolution data, we aimed to enhance both predictive performance and computational efficiency. The study also investigates how the size of the fine-tuning dataset influences model accuracy, providing practical recommendations for different pollutants and temporal resolutions.

The performance of the models is evaluated using standard metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2). Results demonstrate that transfer learning significantly boosts the accuracy of forecasts across different temporal resolutions and fine-tuning conditions.

In summary, this research underlines the effectiveness of transfer learning in improving air quality forecasting for larger temporal scales. Adapting pre-trained models to new tasks proves to be a powerful strategy for addressing the challenges of forecasting air pollution. Key benefits of this approach include:

1. **Rapid Adaptation to New Data:** Models can quickly adapt to new temporal resolutions or changing environmental conditions with minimal retraining, helping maintain system responsiveness.

2. **Scalability and Flexibility:** Pre-trained models support scalable solutions by enabling the extension to higher temporal resolutions without the need for extensive new data collection.

3. **Improved Accuracy with Limited Data:** Transfer learning improves forecasting performance even when target data is scarce, by reusing knowledge from related models and datasets.

4. **Reduced Cost and Time:** This method minimizes the need for large training datasets and reduces computation time and resources, making it suitable for real-time and large-scale applications.

Overall, transfer learning offers an efficient, flexible, and accurate approach to air quality monitoring, supporting better decision-making and public health protection. Building on these findings, we are currently collaborating with the French air quality observatory to evaluate our models on national datasets. Future work includes expanding the system to cover additional pollutants, deploying it across more monitoring stations and regions, and further refining the models to improve their robustness. This research contributes to the advancement of air pollution forecasting and highlights the potential of transfer learning in tackling complex environmental issues.

5.6. Summary and Prospects

This research establishes transfer learning as a transformative paradigm for air quality forecasting, successfully addressing critical challenges in environmental monitoring through three interconnected studies. The work demonstrates that knowledge learned from $PM_{2.5}$

forecasting can be effectively transferred to predict other pollutants across multiple atmospheric species, achieving substantial performance improvements with minimal target data requirements. The novel *LagEnsembleForecasting* framework addresses the fundamental challenge of temporal scale selection by integrating multiple lag-specific models, resulting in enhanced prediction accuracy while providing natural uncertainty quantification through ensemble variance. Additionally, the successful adaptation of models across temporal resolutions demonstrates the versatility of transfer learning in addressing diverse forecasting horizons from short-term to long-term predictions.

The practical implications of this work are substantial, offering immediate benefits for operational air quality monitoring systems worldwide. The demonstrated data and computational efficiency democratizes access to advanced forecasting capabilities, enabling rapid deployment of reliable prediction systems in data-scarce environments and resource-constrained regions. The frameworks significantly reduce typical model development timelines while providing uncertainty quantification capabilities that support more informed decision-making for public health advisories and emergency response protocols. These contributions address real-world challenges in environmental monitoring, making sophisticated air quality forecasting accessible to emerging monitoring networks and developing countries where traditional approaches would be prohibitively resource-intensive.

Future research should focus on extending these frameworks to multi-station networks for spatial transfer learning, incorporating additional pollutants and meteorological variables, and developing real-time adaptive systems that continuously update based on evolving environmental conditions. The integration of physics-informed constraints could enhance model interpretability and ensure physical consistency, while operational deployment across diverse geographical regions would validate the global applicability of these approaches. This research establishes a robust foundation for next-generation environmental monitoring systems that combine scientific rigor with practical efficiency, contributing to improved air quality management and public health protection in an era of increasing environmental challenges.

6. Conclusions and Perspectives

6.1. Conclusion

This thesis has proposed and validated innovative approaches for advancing AI-driven environmental monitoring, with a particular focus on intelligent air quality indicators as part of a broader **water-air-soil** strategic framework. By integrating machine learning, granular computing, explainable AI, and transfer learning techniques, the research addresses some of the most pressing limitations in current air quality monitoring systems—namely, the lack of interpretability, insufficient cross-pollutant and cross-temporal modeling, and inadequate particulate matter toxicity assessment.

Through this work, we move toward a paradigm of **environmental intelligence**—a multidisciplinary approach that combines data-driven methodologies with domain expertise to enable proactive, interpretable, and context-aware environmental decision-making.

6.1.1. Summary of Key Contributions

- **Predictive Modeling of PM Toxicity Thresholds:** Developed machine learning models capable of accurately predicting toxicity thresholds for particulate matter, leveraging physico-chemical and exposure characteristics to support more scalable and timely health risk assessments.
- **Transfer Learning for Air Quality Forecasting:** Established deep learning transfer strategies to enable cross-pollutant and multi-temporal knowledge sharing, thereby improving prediction performance in data-scarce scenarios.
- **Granular Computing for Explainable Classification:** Designed interpretable rule-based classification models through granular computing, bridging the gap between high predictive accuracy and stakeholder trust, particularly in regulatory and governance contexts.

- **Foundations for Multi-Agent Environmental Monitoring Systems:** Laid the groundwork for integrating air, water, and soil intelligence into a distributed multi-agent system capable of supporting holistic environmental assessment and adaptive policy-making.

6.1.2. Outlook and Future Work

While this thesis focuses primarily on air quality, the conceptual and methodological advances it presents open several promising avenues for future research:

- **Extension to Water and Soil Domains:** Future work will expand the AI-based framework to incorporate indicators from water and soil pollution, fulfilling the full potential of the **water-air-soil** ecosystem strategy.
- **Deployment in Real-Time Monitoring Scenarios:** Translating the developed models into real-time environmental monitoring systems will facilitate early warnings and dynamic intervention strategies in urban and industrial settings.
- **Strengthening Generalization and Adaptability:** Enhancing the adaptability of models through advanced transfer learning and domain adaptation will improve their robustness across varying climatic, geographical, and socio-political contexts.
- **Policy Integration and Decision Support:** Continued collaboration with policy-makers and environmental agencies will support the incorporation of interpretable AI insights into regulatory frameworks and sustainable development agendas.
- **Toxicity Forecasting and Explainable AI:** As a short-term perspective, a collaborative project with Atmo Hauts-de-France is planned to collect PM toxicity data. This will enable the development of advanced forecasting models that treat toxicity as a time series problem. Furthermore, we will integrate explainable AI (XAI) techniques to provide human-understandable explanations and interpretations for predicted toxicity levels.

6.1.3. Final Reflections

By addressing key methodological and domain-specific challenges—such as data fragmentation, model opacity, and limited cross-domain integration—this thesis contributes to the

growing field of environmental intelligence. It offers not only technical advancements but also conceptual clarity and practical tools to bridge the gap between environmental data and actionable knowledge. Ultimately, this research represents a meaningful step toward building intelligent, interpretable, and integrated systems for environmental monitoring, public health protection, and sustainable governance.

Bibliography

Adel A Abdel-Rahman. 2008. "On the atmospheric dispersion and Gaussian plume model." In: *Proceedings of the 2nd International Conference on Waste Management, Water Pollution, Air Pollution, Indoor Climate, Corfu, Greece*. Vol. 26 (cit. on p. 139).

MSK Abhilash, Amrita Thakur, Deepa Gupta, and B Sreevidya. 2018. "Time series analysis of air pollution in Bengaluru using ARIMA model." In: *Ambient Communications and Computer Systems: RACCCS 2017*. Springer, 413–426 (cit. on p. 139).

Archit Aggarwal, Tanupriya Choudhary, and Praveen Kumar. 2017. "A fuzzy interface system for determining Air Quality Index." In: *2017 International conference on infocom technologies and unmanned systems (trends and future directions)(ICTUS)*. IEEE, 786–790 (cit. on p. 141).

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. "Optuna: A next-generation hyperparameter optimization framework." In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2623–2631 (cit. on p. 84).

Mniai Ayoub, Tamouh Abdelhamid, and Jebari Khalid. 2025. "Granular computing framework for credit card fraud detection." *Alexandria Engineering Journal*, 121, 387–401 (cit. on p. 27).

Kosala Bandara, Christoph Bergmeir, and Spencer Smyl. 2020. "Forecasting time series with LSTM: A review." *International Journal of Forecasting*, 36, 3, 1181–1196 (cit. on p. 45).

Andrzej Bargiela and Witold Pedrycz. 2003. *Granular Computing: An Introduction*. Springer. DOI: 10.1007/978-1-4615-1033-8 (cit. on pp. 21, 25, 26, 29).

Andrzej Bargiela and Witold Pedrycz. 2006. "The roots of granular computing." In: *2006 IEEE International Conference on Granular Computing*. IEEE, 806–809 (cit. on p. 23).

Rashid Behzadidoost, Farnaz Mahan, and Habib Izadkhah. 2024. "Granular computing-based deep learning for text classification." *Information Sciences*, 652, 119746 (cit. on p. 28).

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 1994. "Learning long-term dependencies with gradient descent is difficult." *IEEE Transactions on Neural Networks*, 5, 2, 157–166 (cit. on p. 66).

James Bergstra and Yoshua Bengio. 2012. "Random search for hyper-parameter optimization." *Journal of Machine Learning Research*, 13, 2, 281–305 (cit. on p. 84).

James Bergstra, Daniel Yamins, and David Cox. 2013. "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures." In: *International conference on machine learning*. PMLR, 115–123 (cit. on p. 84).

James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. "Algorithms for hyper-parameter optimization." In: *Advances in Neural Information Processing Systems*, 2546–2554 (cit. on p. 84).

Bertrand Bessagnet, Amir Hodzic, Robert Vautard, Matthias Beekmann, Sylvain Cheinet, Cécile Honoré, Catherine Liousse, and Laurence Rouil. 2004. "Aerosol modeling with CHIMERE—preliminary evaluation at the continental scale." *Atmospheric environment*, 38, 18, 2803–2817 (cit. on p. 139).

James C Bezdek, Chris Coray, Robert Gunderson, and James Watson. 1981. "Detection and characterization of cluster substructure i. linear structure: Fuzzy c-lines." *SIAM Journal on Applied Mathematics*, 40, 2, 339–357 (cit. on p. 33).

Fabio Biancofiore et al.. 2017. "Recursive neural network model for analysis and forecast of PM10 and PM2. 5." *Atmospheric Pollution Research*, 8, 4, 652–659 (cit. on p. 140).

Francis S Binkowski and Shawn J Roselle. 2003. "Models-3 Community Multiscale Air Quality (CMAQ) model aerosol component 1. Model description." *Journal of geophysical research: Atmospheres*, 108, D6 (cit. on p. 139).

Gianluca Bontempi, Souhaib Ben Taieb, and Yann-Aël Le Borgne. 2012. "Machine learning strategies for time series forecasting." In: *European Big Data Management and Analytics Summer School*. Springer, 62–77 (cit. on p. 45).

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons (cit. on pp. 39, 40, 44).

Aslı Bozdağ, Yeşim Dokuz, and Öznur Begüm Gökçek. 2020. "Spatial prediction of PM10 concentration using machine learning algorithms in Ankara, Turkey." *Environmental Pollution*, 263, 114635 (cit. on p. 139).

Leo Breiman. 2001. "Random forests." *Machine learning*, 45, 5–32 (cit. on p. 52).

Boris Fresnel Cachon et al.. 2014. "Proinflammatory effects and oxidative stress within human bronchial epithelial cells exposed to atmospheric particulate matter (PM2. 5 and PM> 2.5) collected from Cotonou, Benin." *Environmental pollution*, 185, 340–351 (cit. on p. 101).

Mauro Castelli, Fabiana Martins Clemente, Aleš Popovič, Sara Silva, and Leonardo Vanneschi. 2020. "A machine learning approach to predict air quality in California." *Complexity*, 2020 (cit. on p. 140).

Asha B Chelani, CV Chalapati Rao, KM Phadke, and MZ Hasan. 2002. "Prediction of sulphur dioxide concentration using artificial neural networks." *Environmental Modelling & Software*, 17, 2, 159–166 (cit. on p. 140).

Yi-Chun Chen, Chia-Hua Lin, Shih-Chun Candice Lung, Ku-Fan Chen, Wen-Cheng Vincent Wang, Cheng-Tai Chou, and Chia-Hsiang Lai. 2019. "Environmental concentration of spray paint particulate matters causes pulmonary dysfunction in human normal bronchial epithelial BEAS-2B cell." *Process Safety and Environmental Protection*, 126, 250–258 (cit. on p. 101).

Song Chen. 2015. *Beijing PM2.5*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5JS49. (2015) (cit. on p. 88).

Song Chen. 2017. *Beijing Multi-Site Air Quality*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5RK (2017) (cit. on p. 88).

Tianqi Chen and Carlos Guestrin. 2016. "Xgboost: A scalable tree boosting system." In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794 (cit. on p. 54).

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. "Learning phrase representations using RNN encoder-decoder for statistical machine translation." In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1724–1734 (cit. on p. 67).

Giorgio Corani and Mauro Scanagatta. 2016. "Air pollution prediction via multi-label classification." *Environmental modelling & software*, 80, 259–264 (cit. on p. 141).

Corinna Cortes and Vladimir Vapnik. 1995. "Support-vector networks." *Machine learning*, 20, 273–297 (cit. on p. 48).

Thomas Cover and Peter Hart. 1967. "Nearest neighbor pattern classification." *IEEE transactions on information theory*, 13, 1, 21–27 (cit. on p. 57).

Harsh S Dhiman, Dipankar Deb, and Josep M Guerrero. 2019. "Hybrid machine intelligent SVR variants for wind forecasting and ramp events." *Renewable and Sustainable Energy Reviews*, 108, 369–379 (cit. on p. 45).

Aditya Dhole, Ishan Ambekar, Gaurav Gunjan, and Shilpa Sonawani. 2021. "An ensemble approach to multi-source transfer learning for air quality prediction." In: *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*. IEEE, 70–77 (cit. on pp. 169, 170).

David A Dickey and Wayne A Fuller. 1979. "Distribution of the estimators for autoregressive time series with a unit root." *Journal of the American statistical association*, 74, 366a, 427–431 (cit. on p. 40).

Finale Doshi-Velez and Been Kim. 2017. "Towards a rigorous science of interpretable machine learning." *arXiv preprint arXiv:1702.08608* (cit. on p. 124).

Jianbang Du, Fengxiang Qiao, Pan Lu, and Lei Yu. 2022. "Forecasting ground-level ozone concentration levels using machine learning." *Resources, Conservation and Recycling*, 184, 106380 (cit. on p. 3).

Shengdong Du, Tianrui Li, Yan Yang, and Shi-Jinn Horng. 2019. "Deep air quality forecasting using hybrid deep learning framework." *IEEE Transactions on Knowledge and Data Engineering*, 33, 6, 2412–2424 (cit. on p. 140).

Grzegorz Dudek. 2022. "A comprehensive study of random forest for short-term load forecasting." *Energies*, 15, 20, 7547 (cit. on p. 45).

Jeffrey L. Elman. 1990. "Finding structure in time." *Cognitive Science*, 14, 2, 179–211 (cit. on p. 64).

Beytullah Eren, İpek Aksangür, and Caner Erden. 2023. "Predicting next hour fine particulate matter (PM2. 5) in the Istanbul Metropolitan City using deep learning algorithms with time windowing strategy." *Urban Climate*, 48, 101418 (cit. on p. 140).

Matthias Feurer and Frank Hutter. 2019. "Hyperparameter optimization." *Automated Machine Learning*, 3–33 (cit. on p. 84).

Valentin Flunkert, David Salinas, and Jan Gasthaus. 2017. "DeepAR: Probabilistic forecasting with autoregressive recurrent networks." In: *International Conference on Learning Representations* (cit. on p. 45).

Iat Hang Fong, Tengyue Li, Simon Fong, Raymond K Wong, and Antonio J Tallon-Ballesteros. 2020. "Predicting concentration levels of air pollutants by transfer learning and recurrent neural network." *Knowledge-Based Systems*, 192, 105622 (cit. on pp. 169, 170).

Joao Gama, Indre Zliobaite, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. "A survey on concept drift adaptation." *ACM computing surveys (CSUR)*, 46, 4, 1–37 (cit. on p. 45).

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. 2016. "Domain-adversarial training of neural networks." *Journal of machine learning research*, 17, 59, 1–35 (cit. on p. 78).

Jeremy M Gernand. 2013. "Quantitative Risk Assessment of the Pulmonary Toxicity of Nanoparticles by Machine-Learning-Enabled Meta-Analysis." Ph.D. Dissertation. Carnegie Mellon University (cit. on p. 102).

Jeremy M Gernand and Elizabeth A Casman. 2014. "Machine learning for nanomaterial toxicity risk assessment." *IEEE intelligent systems*, 29, 3, 84–88 (cit. on p. 102).

Behzad Ghiasi, Roohollah Noori, Hossein Sheikhian, Amin Zeynolabedin, Yuanbin Sun, Changhyun Jun, Mohamed Hamouda, Sayed M Bateni, and Soroush Abolfathi. 2022. "Uncertainty quantification of granular computing-neural network model for prediction of pollutant longitudinal dispersion coefficient in aquatic streams." *Scientific reports*, 12, 1, 4610 (cit. on p. 142).

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. 2016. *Deep learning*. 2. Vol. 1. MIT press Cambridge (cit. on pp. 59, 64).

Allan D Gordon. 1987. "A review of hierarchical classification." *Journal of the Royal Statistical Society: Series A (General)*, 150, 2, 119–137 (cit. on p. 28).

Ranjana Waman Gore and Deepa S Deshpande. 2017. "An approach for classification of health risks based on air quality levels." In: *2017 1st International Conference on Intelligent Systems and Information Management (ICISIM)*. IEEE, 58–61 (cit. on p. 141).

Marian B Gorzałczany. 1987. "A method of inference in approximate reasoning based on interval-valued fuzzy sets." *Fuzzy sets and systems*, 21, 1, 1–17 (cit. on p. 33).

"Granular Computing: An International Journal." *Granular Computing*, 1, 1, 1–3. Editorial: Inaugural Issue. DOI: 10.1007/s41066-015-0001-2 (cit. on p. 25).

David Gunning. 2017. "Explainable artificial intelligence (XAI)." *Defense Advanced Research Projects Agency (DARPA), nd Web* (cit. on p. 124).

Faqih Hamami and Iqbal Ahmad Dahlan. 2022. "Air Quality Classification in Urban Environment using Machine Learning Approach." In: *IOP Conference Series: Earth and Environmental Science*. Vol. 986. IOP Publishing, 012004 (cit. on p. 141).

Faqih Hamami and Inayatul Fithriyah. 2020. "Classification of air pollution levels using artificial neural network." In: *2020 International Conference on Information Technology Systems and Innovation (ICITSI)*. IEEE, 217–220 (cit. on p. 141).

Mohd Anul Haq. 2022. "SMOTEDNN: A novel model for air pollution forecasting and AQI classification." *Computers, Materials & Continua*, 71, 1 (cit. on p. 141).

Haibo He and Edwardo A Garcia. 2009. "Learning from imbalanced data." *IEEE Transactions on knowledge and data engineering*, 21, 9, 1263–1284 (cit. on p. 119).

Pradeep Hewage, Ardhendu Behera, Marcello Trovati, Ella Pereira, Morteza Ghahremani, Francesco Palmieri, and Yonghuai Liu. 2020. "Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station." *Soft Computing*, 24, 16453–16482 (cit. on p. 45).

Martin Hitziger, Roberto Esposito, Massimo Canali, Maurizio Aragrande, Barbara Häsler, and Simon R Rüegg. 2018. "Knowledge integration in One Health policy formulation, implementation and evaluation." *Bulletin of the World Health Organization*, 96, 3, 211 (cit. on p. 5).

Jerry R Hobbs. 1990. "Granularity." In: *Readings in qualitative reasoning about physical systems*. Elsevier, 542–545 (cit. on pp. 26, 29).

Sepp Hochreiter and Jürgen Schmidhuber. 1997. "Long short-term memory." *Neural Computation*, 9, 8, 1735–1780 (cit. on pp. 66, 69).

David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. 2013. *Applied logistic regression*. John Wiley & Sons (cit. on p. 46).

Rob J Hyndman and George Athanasopoulos. 2018. *Forecasting: principles and practice*. OTexts (cit. on pp. 39, 45).

Gabriel Idakwo, Joseph Luttrell, Minjun Chen, Huixiao Hong, Zhaoxian Zhou, Ping Gong, and Chaoyang Zhang. 2018. "A review on machine learning methods for in silico toxicity prediction." *Journal of Environmental Science and Health, Part C*, 36, 4, 169–191 (cit. on p. 102).

IEEE International Conference on Granular Computing. Beijing, China, July 25-27, 2005. IEEE, (2005) (cit. on p. 25).

Anil K Jain, M Narasimha Murty, and Patrick J Flynn. 1999. "Data clustering: a review." *ACM computing surveys (CSUR)*, 31, 3, 264–323 (cit. on p. 33).

Iny Jhun, Brent A Coull, Joel Schwartz, Bryan Hubbell, and Petros Koutrakis. 2015. "The impact of weather changes on air quality and health in the United States in 1994–2012." *Environmental research letters*, 10, 8, 084009 (cit. on p. 140).

Frank J Kelly and Julia C Fussell. 2012. "Size, source and chemical composition as determinants of toxicity attributable to ambient particulate matter." *Atmospheric environment*, 60, 504–526 (cit. on p. 96).

Fateme Khamespanah, Mahmoud Reza Delavar, Hadis Samadi Alinia, and Mehdi Zare. 2013. "Granular computing and Dempster–Shafer integration in seismic vulnerability assessment." *Intelligent Systems for Crisis Management: Geo-information for Disaster Management (Gi4DM) 2012*, 147–158 (cit. on pp. 37, 142).

Ki-Hyun Kim, Ehsanul Kabir, and Shamin Kabir. 2015. "A review on the human health impact of airborne particulate matter." *Environment international*, 74, 136–143 (cit. on p. 96).

Ven Jyn Kok and Chee Seng Chan. 2016. "GrCS: Granular computing-based crowd segmentation." *IEEE transactions on Cybernetics*, 47, 5, 1157–1168 (cit. on p. 26).

Sotiris Kotsiantis, Dimitris Kanellopoulos, Panayiotis Pintelas, et al.. 2006. "Handling imbalanced datasets: A review." *GESTS international transactions on computer science and engineering*, 30, 1, 25–36 (cit. on p. 119).

Kattariya Kujaroentavon, Supapom Kiattisin, Adisom Leelasantitham, and Sotarat Thammaboosadee. 2014. "Air quality classification in Thailand based on decision tree." In: *The 7th 2014 Biomedical Engineering International Conference*. IEEE, 1–5 (cit. on p. 141).

Anikender Kumar and Piyush Goyal. 2013. "Forecasting of air quality index in Delhi using neural network based on principal component analysis." *Pure and Applied Geophysics*, 170, 711–722 (cit. on p. 140).

Anikender Kumar and Pramila Goyal. 2011. "Forecasting of air quality in Delhi using principal component regression technique." *Atmospheric Pollution Research*, 2, 4, 436–444 (cit. on p. 139).

Ujjwal Kumar and VK Jain. 2010. "ARIMA forecasting of ambient air pollutants (O 3, NO, NO 2 and CO)." *Stochastic Environmental Research and Risk Assessment*, 24, 751–760 (cit. on p. 139).

Quoc V Le, Navdeep Jaitly, and Geoffrey E Hinton. 2015. "A simple way to initialize recurrent networks of rectified linear units." *arXiv preprint arXiv:1504.00941* (cit. on p. 67).

B Leclercq et al.. 2018. "Air pollution-derived PM2. 5 impairs mitochondrial function in healthy and chronic obstructive pulmonary diseased human bronchial epithelial cells." *Environmental pollution*, 243, 1434–1449 (cit. on p. 101).

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. "Deep learning." *nature*, 521, 7553, 436–444 (cit. on pp. 28, 59).

Thomas MT Lei, Stanley CW Ng, and Shirley WI Siu. 2023. "Application of ANN, XGBoost, and Other ML Methods to Forecast Air Quality in Macau." *Sustainability*, 15, 6, 5341 (cit. on p. 140).

Wenlin Li and Xuchu Jiang. 2023. "Prediction of air pollutant concentrations based on TCN-BiLSTM-DMAttention with STL decomposition." *Scientific Reports*, 13, 1, 4665 (cit. on p. 45).

Decui Liang and Dun Liu. 2015. "Deriving three-way decisions from intuitionistic fuzzy decision-theoretic rough sets." *Information Sciences*, 300, 28–48 (cit. on p. 28).

TY Lin. 1997. "Granular computing: From rough sets and neighborhood systems to information granulation and computing in words." In: *European congress on intelligent techniques and soft computing*. Vol. 8. Citeseer, 12 (cit. on p. 25).

Hui Liu, Haiping Wu, Xinwei Lv, Zhiren Ren, Min Liu, Yanfei Li, and Huipeng Shi. 2019. "An intelligent hybrid model for air pollutant concentrations forecasting: Case of Beijing in China." *Sustainable Cities and Society*, 47, 101471 (cit. on p. 140).

Jianguo Liu et al.. 2007. "Complexity of coupled human and natural systems." *science*, 317, 5844, 1513–1516 (cit. on p. 1).

Scott M Lundberg and Su-In Lee. 2017. "A unified approach to interpreting model predictions." *Advances in Neural Information Processing Systems*, 30 (cit. on p. 124).

Jinghui Ma, Zhongqi Yu, Yuanhao Qu, Jianming Xu, Yu Cao, et al.. 2020. "Application of the XGBoost machine learning method in PM2. 5 prediction: A case study of Shanghai." *Aerosol and Air Quality Research*, 20, 1, 128–138 (cit. on p. 140).

Jun Ma, Jack CP Cheng, Changqing Lin, Yi Tan, and Jingcheng Zhang. 2019. "Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques." *Atmospheric Environment*, 214, 116885 (cit. on pp. 169–171).

Jun Ma, Zheng Li, Jack CP Cheng, Yuexiong Ding, Changqing Lin, and Zherui Xu. 2020. "Air quality prediction at new stations using spatially transferred bi-directional long short-term memory network." *Science of The Total Environment*, 705, 135771 (cit. on pp. 169, 170).

Wei Ma, Zibing Yuan, Alexis KH Lau, Long Wang, Chenghao Liao, and Yongbo Zhang. 2022. "Optimized neural network for daily-scale ozone prediction based on transfer learning." *Science of the Total Environment*, 827, 154279 (cit. on pp. 169–171).

Zhanfei Ma, Bisheng Wang, Wenli Luo, Jing Jiang, Dongxiang Liu, Hui Wei, and HaoYe Luo. 2024. "Air pollutant prediction model based on transfer learning two-stage attention mechanism." *Scientific Reports*, 14, 1, 7385 (cit. on p. 170).

Balasubbareddy Mallala, Azka Ihtesham Uddin Ahmed, Sastry V Pamidi, Md Omar Faruque, and Rajasekhar Reddy. 2025. "Forecasting global sustainable energy from renewable sources using random forest algorithm." *Results in Engineering*, 25, 103789 (cit. on p. 45).

R Mangayarkarasi, C Vanmathi, Mohammad Zubair Khan, Abdulfattah Noorwali, Rachit Jain, and Priyansh Agarwal. 2021. "COVID19: Forecasting Air Quality Index and Particulate Matter (PM2. 5)." *Computers, Materials & Continua*, 67, 3 (cit. on p. 141).

CA McHugh, DJ Carruthers, and HA Edmunds. 1997. "ADMS–Urban: an air quality management system for traffic, domestic and industrial pollution." *International Journal of Environment and Pollution*, 8, 3-6, 666–674 (cit. on p. 139).

Laurent Menut et al.. 2013. "CHIMERE 2013: a model for regional atmospheric composition modelling." *Geoscientific model development*, 6, 4, 981–1028 (cit. on p. 139).

Saber Moazami, Roohollah Noori, Bahman Jabbarian Amiri, Bijan Yeganeh, Sadegh Partani, and Salman Safavi. 2016. "Reliable prediction of carbon monoxide using developed support vector machine." *Atmospheric Pollution Research*, 7, 3, 412–418 (cit. on p. 139).

Ramon E Moore. 1966. *Interval analysis*. Prentice-Hall (cit. on pp. 21, 24, 33).

Jianjun Ni, Yan Chen, Yu Gu, Xiaolong Fang, and Pengfei Shi. 2022. "An improved hybrid transfer learning-based deep learning model for PM2. 5 concentration prediction." *Applied Sciences*, 12, 7, 3597 (cit. on pp. 169, 170).

Harri Niska, Teri Hiltunen, Ari Karppinen, Juhani Ruuskanen, and Mikko Kolehmainen. 2004. "Evolving the neural network model for forecasting air pollution time series." *Engineering Applications of Artificial Intelligence*, 17, 2, 159–167 (cit. on p. 140).

Xinyi Niu, Steven Sai Hang Ho, Kin Fai Ho, Yu Huang, Jian Sun, Qiyuan Wang, Yaqing Zhou, Zhuzi Zhao, and Junji Cao. 2017. "Atmospheric levels and cytotoxicity of polycyclic aromatic hydrocarbons and oxygenated-PAHs in PM2. 5 in the Beijing-Tianjin-Hebei region." *Environmental pollution*, 231, 1075–1084 (cit. on p. 101).

Roohollah Noori, Behzad Ghiasi, Hossien Sheikhian, and Jan Franklin Adamowski. 2017. "Estimation of the dispersion coefficient in natural rivers using a granular computing model." *Journal of Hydraulic Engineering*, 143, 5, 04017001 (cit. on p. 142).

Roohollah Noori, Gholamali Hoshyaripour, Khosro Ashrafi, and Babak Nadjar Araabi. 2010. "Uncertainty analysis of developed ANN and ANFIS models in prediction of carbon monoxide daily concentration." *Atmospheric Environment*, 44, 4, 476–482 (cit. on p. 140).

Roohollah Noori, Hossien Sheikhian, Farhad Hooshyaripor, Ali Naghikhani, Jan Franklin Adamowski, and Behzad Ghiasi. 2017. "Granular computing for prediction of scour below spillways." *Water Resources Management*, 31, 313–326 (cit. on p. 142).

Stanislaw Osowski and Konrad Garanty. 2007. "Forecasting of the daily meteorological pollution using wavelets and support vector machine." *Engineering Applications of Artificial Intelligence*, 20, 6, 745–755 (cit. on p. 140).

Unjin Pak, Jun Ma, Unsok Ryu, Kwangchol Ryom, U Juhyok, Kyongsok Pak, and Chanil Pak. 2020. "Deep learning-based PM2. 5 prediction considering the spatiotemporal correlations: A case study of Beijing, China." *Science of the Total Environment*, 699, 133561 (cit. on p. 3).

Sinno Jialin Pan and Qiang Yang. 2010. "A Survey on Transfer Learning." *IEEE Transactions on Knowledge and Data Engineering*, 22, 10, 1345–1359. DOI: 10.1109/TKDE.2009.191 (cit. on p. 72).

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. "On the difficulty of training recurrent neural networks." In: *International Conference on Machine Learning*. PMLR, 1310–1318 (cit. on p. 67).

Zdzisław Pawlak. 1982. "Rough sets." *International journal of computer & information sciences*, 11, 341–356 (cit. on pp. 21, 24, 32).

Witold Pedrycz. 2005. *Knowledge-based clustering: from data to information granules.* John Wiley & Sons (cit. on p. 30).

Witold Pedrycz. 2011. "Information granules and their use in schemes of knowledge management." *Scientia Iranica*, 18, 3, 602–610 (cit. on p. 23).

Witold Pedrycz. 2018. *Granular computing: analysis and design of intelligent systems.* CRC press (cit. on pp. 21, 23, 26, 28, 30, 31).

Witold Pedrycz and Shyi-Ming Chen. 2011. *Granular computing and intelligent systems: design with information granules of higher order and higher type.* Vol. 13. Springer Science & Business Media (cit. on p. 29).

Witold Pedrycz, Andrzej Skowron, and Vladik Kreinovich. 2008. *Handbook of granular computing.* John Wiley & Sons (cit. on pp. 25, 31).

Witold Pedrycz and George Vukovich. 2001. "Granular neural networks." *Neurocomputing*, 36, 1-4, 205–224 (cit. on p. 28).

Efren Perez Santin, Raquel Rodriguez Solana, Mariano Gonzalez Garcia, Maria Del Mar Garcia Suarez, Gerardo David Blanco Diaz, Maria Dolores Cima Cabal, Jose Manuel Moreno Rojas, and Jose Ignacio Lopez Sanchez. 2021. "Toxicity prediction based on artificial intelligence: A multidisciplinary overview." *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 11, 5, e1516 (cit. on p. 102).

Sanjay Purushotham, Wilka Carvalho, Tanachat Nilanon, and Yan Liu. 2017. "Variational recurrent adversarial deep domain adaptation." In: *International conference on learning representations* (cit. on p. 80).

Yuhua Qian, Jiye Liang, Witold Pedrycz, and Chuangyin Dang. 2010. "Positive approximation: an accelerator for attribute reduction in rough set theory." *Artificial intelligence*, 174, 9-10, 597–618 (cit. on p. 30).

J. Ross Quinlan. 1986. "Induction of decision trees." *Machine learning*, 1, 81–106 (cit. on p. 50).

Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. 2019. "Transfusion: Understanding transfer learning for medical imaging." *Advances in neural information processing systems*, 32 (cit. on p. 79).

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "'Why should I trust you?': Explaining the predictions of any classifier." In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144 (cit. on p. 124).

Agnieszka Rorat, Caroline Lanier, Jean-Philippe Jaeg, Damien Cuny, Annabelle Deram, and Ludivine Canivet. 2020. "Between physicochemical characterization of atmospheric particles and their impact on human health-challenges and perspectives (RECORD project)." In: *Atmso'Fair (Webinair), 23-24 juin 2020* (cit. on p. 96).

Seyyed Meysam Rozehkhani and Farnaz Mahan. 2022. "VM consolidation improvement approach using heuristics granular rules in cloud computing environment." *Information Sciences*, 596, 15–29 (cit. on pp. 37, 142).

Seyyed Meysam Rozehkhani and Maryam Mohammadzad. 2022. "Rule Extraction for Screening of COVID-19 Disease Using Granular Computing Approach." *Computational and Mathematical Methods in Medicine*, 2022 (cit. on pp. 37, 142).

Sebastian Ruder. 2019. "Neural transfer learning for natural language processing." Ph.D. Dissertation. NUI Galway (cit. on p. 72).

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. "Learning representations by back-propagating errors." *nature*, 323, 6088, 533–536 (cit. on p. 61).

Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. 2016. "Progressive neural networks." *arXiv preprint arXiv:1606.04671* (cit. on p. 80).

Hadis Samadi Alinia and MR Delavar. 2011. "Tehran's seismic vulnerability classification using granular computing approach." *Applied Geomatics*, 3, 229–240 (cit. on pp. 37, 142).

S Saminathan and C Malathy. 2023. "Ensemble-based classification approach for PM2. 5 concentration forecasting using meteorological data." *Frontiers in big Data*, 6, 1175259 (cit. on p. 141).

Matteo Sangiorgio and Giorgio Guariso. 2024. "Transfer learning in environmental data-driven models: A study of ozone forecast in the Alpine region." *Environmental Modelling & Software*, 177, 106048 (cit. on p. 170).

Jingyi Shao, Amanda J Wheeler, Ling Chen, Bo Strandberg, Andrea Hinwood, Fay H Johnston, and Graeme R Zosky. 2018. "The pro-inflammatory effects of particulate matter on epithelial cells are associated with elemental composition." *Chemosphere*, 202, 530–537 (cit. on p. 101).

Hossein Sheikhian, Mahmoud Reza Delavar, and Alfred Stein. 2017. "A GIS-based multi-criteria seismic vulnerability assessment using the integration of granular computing rule extraction and artificial neural networks." *Transactions in GIS*, 21, 6, 1237–1259 (cit. on pp. 37, 142).

Th Slini, K Karatzas, and A Papadopoulos. 2002. "Regression analysis and urban air quality forecasting: An application for the city of Athens." *Global Nest*, 4, 2-3, 153–162 (cit. on p. 139).

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. "Practical Bayesian optimization of machine learning algorithms." In: *Advances in Neural Information Processing Systems*, 2951–2959 (cit. on p. 84).

Mingli Song, Yan Li, and Witold Pedrycz. 2023. "Time series prediction with granular neural networks." *Neurocomputing*, 546, 126328 (cit. on pp. 27, 28).

Mingli Song and Xinyu Zhao. 2024. "Granular neural networks learning for time series prediction under a federated scenario." *Granular Computing*, 9, 3, 68 (cit. on p. 28).

Bambang Sugiarto and Rika Sustika. 2016. "Data classification for air quality on wireless sensor network monitoring system using decision tree algorithm." In: *2016 2nd International Conference on Science and Technology-Computer (ICST)*. IEEE, 172–176 (cit. on p. 141).

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. 2013. "On the importance of initialization and momentum in deep learning." In: *International conference on machine learning*. PMLR, 1139–1147 (cit. on p. 66).

Souhaib Ben Taieb and Rob J Hyndman. 2014. "A gradient boosting approach to the Kaggle load forecasting competition." *International journal of forecasting*, 30, 2, 382–394 (cit. on p. 45).

Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. "A survey on deep transfer learning." In: *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*. Springer, 270–279 (cit. on p. 72).

Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. "Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations." In: *Proceedings of the 14th ACM conference on recommender systems*, 269–278 (cit. on p. 80).

Antipas T Teologo, Elmer P Dadios, Renann G Baldovino, Romano Q Neyra, and Irister M Javel. 2018. "Air quality index (AQI) classification using CO and NO 2 pollutants: a fuzzy-based approach." In: *TENCON 2018-2018 IEEE Region 10 Conference*. IEEE, 0194–0198 (cit. on p. 141).

United States Environmental Protection Agency. 2023. *Particulate Matter (PM) Basics*. (2023). https://www.epa.gov/pm-pollution/particulate-matter-pm-basics (cit. on p. 97).

Thomas Unterthiner, Andreas Mayr, Günter Klambauer, and Sepp Hochreiter. 2015. "Toxicity prediction using deep learning." *arXiv preprint arXiv:1503.01445* (cit. on p. 102).

DANEL Vincent. 2019. "Airborne particulate matter and their health effects." *Encyclopédie de l'environnement*, 1–11. https://www.encyclopedie-environnement.org/en/health/airborne-particulate-health-effects/ (cit. on p. 97).

Andy H Vo, Terry R Van Vleet, Rishi R Gupta, Michael J Liguori, and Mohan S Rao. 2019. "An overview of machine learning and big data for drug toxicity evaluation." *Chemical research in toxicology*, 33, 1, 20–37 (cit. on p. 102).

Guanglei Wang et al.. 2019. "High-content analysis of particulate matters-induced oxidative stress and organelle dysfunction in vitro." *Toxicology in Vitro*, 59, 263–274 (cit. on p. 101).

Junshan Wang and Guojie Song. 2018. "A deep spatial-temporal ensemble model for air quality prediction." *Neurocomputing*, 314, 198–206 (cit. on p. 140).

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. 2016. "A survey of transfer learning." *Journal of Big data*, 3, 1–40 (cit. on p. 72).

Paul J Werbos. 1990. "Backpropagation through time: what it does and how to do it." *Proceedings of the IEEE*, 78, 10, 1550–1560 (cit. on p. 66).

WHO. 2022. *World Health Organization Ambient (outdoor) air pollution*. Retrieved Dec. 19, 2022 from https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health (cit. on pp. 1, 5).

Ronald J Williams and Jing Peng. 1990. "An efficient gradient-based algorithm for on-line training of recurrent network trajectories." *Neural computation*, 2, 4, 490–501 (cit. on p. 67).

Garrett Wilson and Diane J Cook. 2020. "A survey of unsupervised deep domain adaptation." *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11, 5, 1–46 (cit. on p. 80).

World Bank. 2023. *The Global Health Cost of PM2.5 Air Pollution: A Case for Action Beyond 2021*. Tech. rep. International Bank for Reconstruction and Development/The World Bank. World Bank Group, Washington, DC. https://openknowledge.worldbank.org/handle/10986/36501 (cit. on p. 1).

Vibha Yadav, Amit Kumar Yadav, Vedant Singh, and Tej Singh. 2024. "Artificial neural network an innovative approach in air pollutant prediction for environmental applications: A review." *Results in Engineering*, 102305 (cit. on p. 45).

Junzi Yang, Ajune Wanis Ismail, Yingying Li, Limin Zhang, and Fazliaty Edora Fadzli. 2023. "Transfer Learning-driven hourly PM2. 5 Prediction Based on a Modified Hybrid Deep Learning." *IEEE Access* (cit. on p. 170).

Bozhi Yao, Guang Ling, Feng Liu, and Ming-Feng Ge. 2024. "Multi-source variational mode transfer learning for enhanced PM2. 5 concentration forecasting at data-limited monitoring stations." *Expert Systems with Applications*, 238, 121714 (cit. on p. 170).

JT Yao and YY Yao. 2002. "Induction of classification rules by granular computing." In: *Rough Sets and Current Trends in Computing: Third International Conference, RSCTC 2002 Malvern, PA, USA, October 14–16, 2002 Proceedings 3*. Springer, 331–338 (cit. on pp. 35, 37).

Yiyu Yao. 2004. "A partition model of granular computing." In: *Transactions on Rough Sets I: James F. Peters-Andrzej Skowron, Editors-in-Chief*. Springer, 232–253 (cit. on pp. 25, 35).

Yiyu Yao. 2007. "The art of granular computing." In: *Rough Sets and Intelligent Systems Paradigms: International Conference, RSEISP 2007, Warsaw, Poland, June 28-30, 2007. Proceedings 1*. Springer, 101–112 (cit. on p. 29).

Yiyu Yao. 2008. "Granular computing: past, present and future." In: *2008 IEEE International Conference on Granular Computing*. IEEE, 80–85 (cit. on pp. 21, 24, 26, 30).

Yiyu Yao. 2009. "Three-way decision: An interpretation of rules in rough set theory." In: *International Conference on Rough Sets and Knowledge Technology*. Springer, 642–649 (cit. on p. 27).

Yiyu Yao. 2010. "Three-way decisions with probabilistic rough sets." *Information sciences*, 180, 3, 341–353 (cit. on pp. 27, 33, 134).

Yiyu Yao. 2011. "The superiority of three-way decisions in probabilistic rough set models." *Information Sciences*, 181, 6, 1080–1096 (cit. on p. 27).

Yiyu Yao. 2012. "An outline of a theory of three-way decisions." In: *International conference on rough sets and current trends in computing*. Springer, 1–17 (cit. on p. 134).

Yiyu Yao. 2013. "Granular computing and sequential three-way decisions." In: *International Conference on Rough Sets and Knowledge Technology*. Springer, 16–27 (cit. on p. 27).

Yiyu Yao. 2016. "Three-way decisions and cognitive computing." *Cognitive Computation*, 8, 4, 543–554 (cit. on p. 27).

Yiyu Yao. 2018. "Three-way decision and granular computing." *International Journal of Approximate Reasoning*, 103, 107–123 (cit. on p. 134).

YY Yao. 2000. "Granular Computing: basic issues and possible solutions" (cit. on p. 21).

YY Yao. 2006. "Three perspectives of granular computing." *Journal of Nanchang Institute of Technology*, 25, 2, 16–21 (cit. on pp. 23, 26, 28, 30).

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. "How transferable are features in deep neural networks?" *Advances in neural information processing systems*, 27 (cit. on p. 76).

Lotfi A Zadeh. 1996. "Fuzzy logic, neural networks, and soft computing." In: *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*. World Scientific, 775–782 (cit. on pp. 28, 29).

Lotfi A Zadeh. 1997. "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic." *Fuzzy sets and systems*, 90, 2, 111–127 (cit. on pp. 21, 25, 26).

Lotfi Asker Zadeh. 1965. "Fuzzy sets." *Information and control*, 8, 3, 338–353 (cit. on pp. 21, 24, 31).

Lotfi Asker Zadeh. 1975. "The concept of a linguistic variable and its application to approximate reasoning—I." *Information sciences*, 8, 3, 199–249 (cit. on pp. 24, 32).

Nur'atiah Zaini, Lee Woen Ean, Ali Najah Ahmed, Marlinda Abdul Malek, and Ming Fai Chow. 2022. "PM2. 5 forecasting for an urban area based on deep learning and decomposition method." *Scientific Reports*, 12, 1, 17565 (cit. on p. 140).

Zhou Zang, Yushan Guo, Yize Jiang, Chen Zuo, Dan Li, Wenzhong Shi, and Xing Yan. 2021. "Tree-based ensemble deep learning model for spatiotemporal surface ozone (O3) prediction and interpretation." *International Journal of Applied Earth Observation and Geoinformation*, 103, 102516 (cit. on p. 3).

Bo Zhang and Ling Zhang. 1992. *Theory and applications of problem solving*. Elsevier Science Inc. (cit. on p. 24).

Dong Zhang, Hanhan Li, Xiao-San Luo, Weijie Huang, Yuting Pang, Jinshan Yang, Mingwei Tang, Tariq Mehmood, and Zhen Zhao. 2022. "Toxicity assessment and heavy metal components of inhalable particulate matters (PM2. 5 & PM10) during a dust storm invading the city." *Process Safety and Environmental Protection*, 162, 859–866 (cit. on p. 101).

G Peter Zhang. 2003. "Time series forecasting using a hybrid ARIMA and neural network model." *Neurocomputing*, 50, 159–175 (cit. on p. 45).

Ling Zhang and Bo Zhang. 2004. "The quotient space theory of problem solving." *Fundamenta Informaticae*, 59, 2-3, 287–298 (cit. on p. 21).

Xiaosong Zhao, Rui Zhang, Jheng-Long Wu, and Pei-Chann Chang. 2018. "A Deep Recurrent Neural Network for Air Quality Classification." *J. Inf. Hiding Multim. Signal Process.*, 9, 2, 346–354 (cit. on p. 141).

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2020. "A comprehensive survey on transfer learning." *Proceedings of the IEEE*, 109, 1, 43–76 (cit. on pp. 72, 74).