Leveraging machine learning for multi-source data enrichment and analytics in air quality monitoring and crowd sensing

Mohammad Abboud

▶ **To cite this version:**

Mohammad Abboud. Leveraging machine learning for multi-source data enrichment and analytics in air quality monitoring and crowd sensing. Artificial Intelligence [cs.AI]. Université Paris-Saclay, 2023. English. NNT : 2023UPASG057 . tel-04323377

HAL Id: tel-04323377

https://theses.hal.science/tel-04323377v1

Submitted on 5 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

université
PARIS-SACLAY

Leveraging machine learning for multi-source data enrichment and analytics in air quality monitoring and crowd sensing

Application de l'apprentissage automatique à l'enrichissement et l'analyse de données multi-sources dans la surveillance de la qualité de l'air et la collecte participative

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 580 sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Informatique
Graduate School : Informatique et sciences du numérique
Référent : Université de Versailles Saint-Quentin-en-Yvelines

Thèse préparée dans l'unité de recherche **DAVID** (Université Paris-Saclay, UVSQ), sous la direction de **Karine ZEITOUNI**, Professeure, et le co-encadrement de **Yehia TAHER**, Maître de Conférence

Thèse soutenue à Versailles, le 8 Novembre 2023, par

Mohammad ABBOUD

Composition du jury

Ana-Maria Olteanu-Raimond Directrice de recherche (HDR), LASTIG, Université Gustave Eiffel, IGN-ENSG	Présidente
Thomas Devogele Professeur, Université de Tours	Rapporteur
Vasile-Marian Scuturici Professeur, INSA de Lyon	Rapporteur
Cyril Ray Maître de conférences, Institut de Recherche de l'Ecole Navale	Examinateur
Chiara Renso Chercheuse Senior, CNR Pisa	Examinatrice

THESE DE DOCTORAT
NNT : 2023UPASG057

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre: Application de l'apprentissage automatique à l'enrichissement et l'analyse de données multi-sources dans la surveillance de la qualité de l'air et la collecte participative

Mots clés: Apprentissage automatique, apprentissage profond, science de données, enrichissement de données, surveillance environnementale, données spatio-temporelles

Résumé: L'enrichissement de données à l'aide de techniques d'apprentissage automatique et profond dans le contexte de l'Internet des objets (IoT) est de plus en plus crucial dans le paysage technologique actuel. L'expansion rapide des appareils de l'IoT a amené à la génération croissante de données à partir de diverses sources telles que des capteurs, des actionneurs et des systèmes embarqués. Cependant, ces données sont souvent massives, complexes et non structurées, ce qui rend difficile leur exploitation dans l'extraction d'information pertinente et la détection ou la prédiction de situations ou d'événements. Les algorithmes d'apprentissage automatique fournissent une solution robuste pour l'extraction automatique de motifs, de tendances et de corrélations dans les données de l'IoT, augmentant ainsi leur valeur. Les systèmes IoT peuvent en effet apprendre des données historiques, s'adapter à l'évolution des paramètres et améliorer leur exploitation en utilisant l'apprentissage automatique. Cet enrichissement des données par des connaissances facilite l'analyse prédictive et améliore le processus décisionnel.

Cette thèse se place dans le contexte de la collecte participative (en anglais Mobile Crowd Sensing ou MCS) de la qualité de l'air et propose des méthodes d'enrichissement des données liées à la pollution atmosphérique en se basant sur l'apprentissage automatique. Les concentrations de pollution sont mesurées à l'aide de capteurs portatifs dans le cadre de campagnes de MCS. D'autres données proviennent de stations fixes qui mesurent la qualité de l'air sur des sites pré-définis. Nous proposons une méthode d'apprentissage automatique combinant ces deux sources de données pour estimer la pollution de l'air offrant une couverture complète. Par ailleurs, nous développons une méthode d'enrichissement des données MCS par apprentissage du micro-environnement, ce qui est important pour une analyse contextualisée de la pollution et de l'exposition individuelle. Une autre source de données, peu structurée, provient des médias sociaux. Nous nous sommes intéressés aux tweets et cherché à enrichir nos données en passant par la géolocalisation précise de tweets. Nous avons proposé un pipeline d'apprentissage permettant de géolocaliser précisément les tweets, et ainsi de détecter des événements localisés en lien avec la pollution. Cette approche intégrée d'enrichissement des données qualitatives et quantitatives améliore notre capacité à analyser et à comprendre la dynamique de la pollution atmosphérique de manière plus complète et localisée.

Dans cette thèse, nous avons adapté des techniques d'apprentissage automatique et profond pour l'enrichissement sémantique des données liées à la qualité de l'air. Toutes les approches proposées ont été appliquées à des données réelles collectées dans le cadre des projets Polluscope et GOGREEN Routes.

Title: Leveraging Machine Learning for Multi-Source Data Enrichment and Analytics in Environmental Monitoring and Crowd Sensing

Keywords: Machine learning, deep learning, data science, data enrichment, environmental monitoring, spatio-temporal data

Abstract: Data enrichment using machine and deep learning techniques in the context of the Internet of Things (IoT) has become increasingly crucial in today's technological landscape. Due to the rapid expansion of IoT devices, there is an excess of data created from diverse sources such as sensors, actuators, and embedded systems. However, this data is frequently enormous, complex, and unstructured, making it difficult to extract significant insights and make accurate conclusions. Machine learning algorithms provide a robust solution by automatically extracting patterns, trends, and correlations from IoT data, increasing its value. IoT systems may learn from prior experiences, adapt to changing settings, and enhance operations by employing machine learning. This enrichment facilitates enhanced predictive analytics and improved decision-making processes.

Our study focuses on improving the quality of both qualitative and quantitative data in the context of air pollution assessments. Regarding quantitative data, we monitor air pollution levels using sensor data acquired by Mobile Crowd Sensing (MCS). We also include data from permanent stations that assess air pollution levels. We want to detect the micro-environment within the MCS data, allowing for a more detailed understanding of pollution trends at a localized level and per participant. Furthermore, we use quantitative data from fixed stations and MCS to estimate air pollution levels in areas without fixed stations, giving comprehensive coverage. On the other hand, in the case of qualitative data, we enrich it by incorporating geolocation information from tweets. By adopting a location prediction model, we can accurately assign geolocation information to tweets, enabling us to utilize them in detecting local events, particularly those related to pollution. This integrated approach of enriching qualitative and quantitative data enhances our ability to analyze and understand air pollution dynamics more comprehensively and localized.

Within this thesis, we have adapted machine and deep learning techniques to provide semantics to our collected quantitative and qualitative data. All the proposed approaches have been applied to real-world datasets collected within Polluscope and GOGREEN Routes projects.

(وَقُلْ رَبِّ زِدْنِي عِلْمًا)

طه - ١١٤

"And say, My Lord, increase me in knowledge."

Quran, Chapter 20, Verse 114

Dedicated to My Precious Family

My Father (Hassan Abboud) & My Mother (Alia Choumar)
My Sisters (Fatima & Alaa)
My Brother (Ali)
My dear person

Mohammad @ Versailles, France
November 8th, 2023

Acknowledgments

I am immensely grateful for the support and guidance I have received throughout my educational journey, leading to the successful completion of my PhD. I would like to express my heartfelt appreciation to my university, UVSQ-Université Paris-Saclay, for providing me with the platform and resources necessary to pursue my research ambitions. Many thans to all people who have accompanied me during my PhD studies. I want to thank all my professors who taught me during learning process. I am truly grateful to UVSQ-Université Paris-Saclay and Lebanese University, my college, and my school for their unwavering support and belief in my abilities.

First and foremost, I am deeply grateful for the invaluable guidance, support, and mentorship provided by my thesis supervisors throughout the course of my PhD journey. I would like to express my sincere appreciation to Prof. Karine Zeitouni for her exceptional guidance and expertise. Her encouragement, enthusiasm, and belief in my abilities have been a constant source of motivation throughout this challenging endeavor. I would also like to extend my heartfelt gratitude to my **co-advisor** Dr. Yehia Taher for always being so helpful and motivating. His insightful feedback, attention to detail, and rigorous academic standards have challenged me to push the boundaries of my research.

I consider myself extremely lucky to have had the opportunity to work under the guidance of such amazing people. Their knowledge, passion, and commitment to my academic development have helped shape me into a competent researcher. The faith they placed in me, the numerous hours they committed to offering advice, and their genuine interest in my development were critical to the successful completion of my thesis. I'd want to offer my heartfelt appreciation to both of my thesis supervisors for their consistent support, faith in my abilities, and essential contributions to my academic and personal development.

I want to thank all of the jury members of my thesis defense, including Dr. Thomas Devogele, Dr. Vasile-Marian Scuturici, Dr. Ana-Maria Olteanu-Raimond, Dr. Chiara Renso, and Dr. Cyril Ray. I am appreciative of their time and efforts in assessing this thesis work. My special words of thanks go to Dr. Thomas Devogele and Dr. Vasile-Marian Scuturici for reviewing my dissertation.

I would like to express my sincere appreciation to my doctoral school STIC, and more broadly to Université Paris-Saclay, and the GOGREEN ROUTES project who has funded my work, under the grant agreement H2020- EU.3.5.2 No 869764. I would like to thank all the members of the GOGREEN ROUTES, MASTER, and Polluscope projects.

This thesis would not be able to reach this level without the contribution from many excellent colleagues in DAVID Lab. I extend my deepest gratitude to all of them. All my appreciation to all ADAM Team permanent and doctoral members for their invaluable feedback on my research and for always being so supportive. I would like to thank my lab mates (Zoé, Perla, Saloua, Rahma, Hadi, Saeed, Baudouin, Souheir, Hafsa, Mohammad Rihany, Alaa, and Jingwei) for always being there and for supporting me.

I would like to take a moment to express my heartfelt gratitude to my friends who have been a constant source of support and encouragement throughout my PhD journey. Whether it was lending a listening

ear during moments of doubt or celebrating the milestones achieved, their presence has been a source of strength and motivation.

Last but not least, even most importantly, I am profoundly grateful for the unwavering love, support, and encouragement extended to me by my family members in both France and Lebanon throughout my PhD journey. Their unchanging faith in my abilities, sacrifices, and unwavering presence have been a pillar of support during the ups and downs of this challenging endeavor. Despite the physical distance between us, their emotional support, words of encouragement, and prayers have provided inspiration and determination. I want also to express my heartfelt thanks to you, my dear person. Your kindness, support, and presence in my life mean the world to me. This doctoral dissertation is dedicated to them. Particularly to my father Hassan Abboud, my mother Alia Choumar, my sisters Fatima and Alaa, my brother Ali, and my dear person.

Contents

List of Figures

List of Tables

Chapter 1 - Introduction

Contents

1.1 . Background

Urbanization and climate change present a problem for urban human and environmental health. Urbanization has been a rapidly growing phenomenon globally, with more than half the world's population living in cities. Complex global challenges face European cities today, predicted to become more dramatic in the coming decades. By 2050, up to 68 % of the entire world's population will be living in urban regions[1]. However, it has already become a reality as between 65 - 75 % of Europe's population currently lives in urban areas [2], representing the third most urbanized region on our planet. Urbanization refers to the increase in the proportion of a country's population in cities and urban areas. It benefits society, including better healthcare, education, and job opportunities. However, it also presents significant challenges, including pollution, traffic congestion, and overcrowding, which can have severe health and environmental consequences.

GOGREEN Routes is a European project whose primary goal is to foster urban mental health and well-being. It is conducted in different cities, such as Versailles, Burgas, Limerick, etc.... Raising the fact that "Making nature healthy again is key to our physical and mental well-being," the GOGREEN ROUTES project will help guide the development of solutions to address the interconnected challenges of urbanization. It will draw on a transdisciplinary body of knowledge beyond nature-based solutions (NBS), ecosystem services, and urban green infrastructure approaches.

As a part of the GOGREEN Routes project, we are interested in understanding urbanization's reasons and effects since it is a crucial factor in overcoming its challenges. One of the significant challenges of urbanization is air pollution. Air pollution is caused by various factors, including industrial activities, transportation, and energy generation, which are heavily concentrated in urban areas.

Monitoring pollution levels (especially air pollution) in urban areas can help understand urbanization's effects on human beings. Those pollution measures provide an informative view of exposure to pollution in such areas. Moreover, understanding people's activities in urban and green areas can also help understand urbanization. Analyzing those aspects can help provide a clear vision of the urbanization problem. This can help decision-makers to develop natural-based solutions better to face urbanization challenges.

The world is witnessing a vast rise in IoT devices where we currently have around 10 Billion active devices. This widespread use of IoT devices is moving the world beyond the Internet of Everything in which everything –from humans to devices to animals- is connected and can be monitored. Every device connected to the Internet is generating data. Air pollution monitoring is one use case application

[1] https://ourworldindata.org/urbanization#future-urbanization

[2] https://ourworldindata.org/urbanization#share-of-populations-living-in-urban-areas

for IoT. The emerging low-cost and lightweight air quality sensor boxes have led to a paradigm shift in environmental monitoring. In such applications, IoT devices collect air quality data from the crowd [59, 97]. Besides the IoT devices, several research initiatives have used fixed air quality sensors to monitor air quality [157, 13, 52].

Mobile crowdsensing (MCS) [50], which empowers volunteers to contribute data acquired by a multi-sensor box and a mobile device to monitor large-scale phenomena, is an excellent option for increasing spatial coverage in outdoor and indoor environments. Mobile crowd sensing mainly depends on low-cost devices that may lead to impreciseness and imperfectness. Conversely, while fixed stations provide accurate measurements, implementing fixed stations with complete area coverage is challenging and expensive. Moreover, linked open data such as traffic and weather datasets can help monitor pollution.

People's opinions about pollution and urbanization also play a significant role in developing natural-based solutions. Qualitative data collected from social media platforms or surveys can provide valuable insight into people's activities in both urban and green areas. This external data source must be integrated with the monitoring system to develop effective solutions.

Overall, understanding the reasons and effects of urbanization is crucial to overcoming its challenges and promoting urban health and well-being. By utilizing ML to analyze qualitative data such as tweets, researchers can gain insights into public sentiment, identify pollution hotspots, and understand the factors contributing to air pollution. Harnessing the power of ML in monitoring air pollution and analyzing qualitative data is a transformative step toward creating healthier and more sustainable cities.

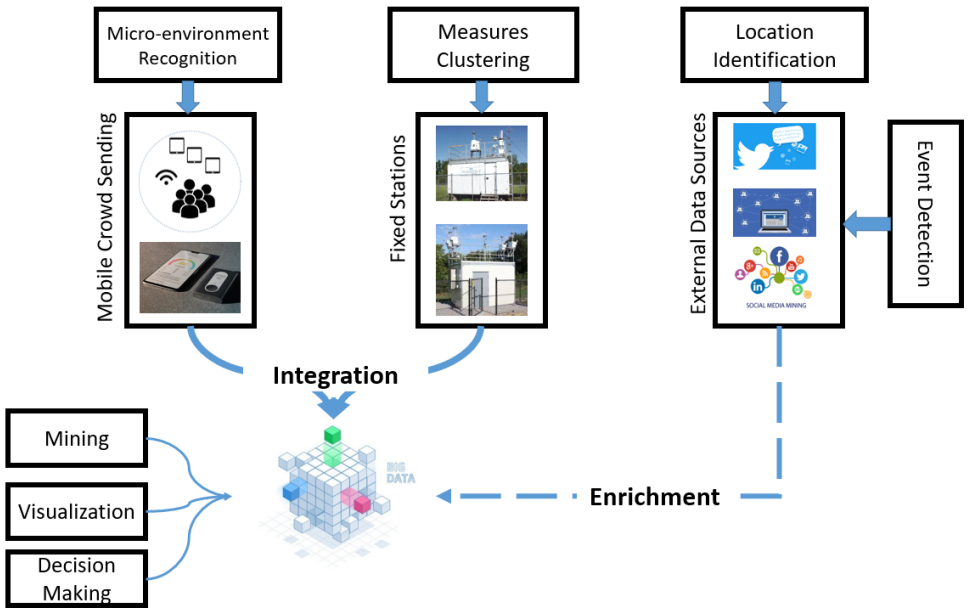


Figure 1.1: Overall approach.

1.2 . Motivation

Data enrichment from several sources has become integral to the Internet of Things (IoT) ecosystem. The goal of enriching data from several sources is to improve the quality of the obtained data by merging numerous data sources and leveraging their distinct properties. This section will go through the reasoning for enhancing data from various sources in the context of IoT.

Two types of data are available in pollution monitoring scenarios. User-centric data is to track individual exposure and the data evolution in both space and time. On the other hand, environment-centric data produces pollution heat maps from a network of fixed stations. We are motivated to collect both data types while enriching them with external sources. The goal is to develop mining methods to help decision-makers solve the urbanization challenges.

User-centric data is nowadays collected with the new paradigm called mobile crowd sensing. MCS is a mechanism that collects and shares data from sensors in mobile devices with a centralized system. MCS sensor data typically contains the sensor's location and measurements. However, because this data needs more semantic information, it is challenging to interpret the context in which it was obtained. Adding semantic information to MCS data, such as the type of activity being performed, can give more relevant insights into the data. The enhanced data may be utilized for several applications by annotating the trajectory data with context.

Conversely, fixed stations can provide the environment-centric track with the appropriate data. While fixed sensors give precise and trustworthy data, they

have limited geographical and temporal coverage. Enriching air pollution data with mobile participatory monitoring data, such as data acquired from wearable sensors or mobile apps, might increase data coverage. It is feasible to estimate air pollution levels with greater precision and resolution by combining data from stationary sensors with data from mobile sensors. This augmented data may be utilized for many purposes, including urban planning and public health.

In addition, to the environmental tracks, we are motivated to gain meaningful insights from external data sources, such as qualitative ones. Twitter is a rich source of information that can be used to gain insights into various topics, such as events or public opinion [133, 78, 158, 11, 144, 145]. However, the location data of tweets are rarely available, making it difficult to associate the data with specific locations. Adding exact location information to Twitter data can give more accurate insights into the data. It is feasible, for example, to correlate tweets with specific events or locations, such as demonstrations or concerts, by determining the actual location of tweets. This enhanced data may improve air pollution by adding context to pollutant observations.

1.3 . Objectives

We are mainly interested in three topics within the context of this thesis: (1) Enriching the crowd-sensed data by micro-environments or, in other words contextualizing the collected trajectory, since ambient air is strongly dependent on the context. (2) Integrating fixed stations and opportunistic mobile participatory monitoring data to expand the Spatiotemporal coverage. (3) Enriching our data with external data sources, we are mainly interested in Twitter data as it is a valuable source of information that helps understand urbanization and pollution exposure. Figure 1.1 shows the different data sources and the aforementioned objective of the work in this thesis.

It is critical to acknowledge that the micro-environment considerably impacts air quality in a specific place [20], and the level of pollution people are exposed to can vary dramatically within a city. As a result, there is a rising interest in performing exposure assessments that consider specific micro-environmental factors since air pollution can vary between indoor and outdoor places. The exposure to pollution at home or office is not the same as the exposure to public transport or car micro-environments. Because of the significant impact of these variables, ignoring micro-environmental effects would render data-gathering attempts worthless.

In real-world applications, annotating the micro-environmental information is exceedingly difficult since only a small percentage of participants can completely record their micro-environment. As a result, there is a strong desire to reduce the load on participants by automating the detection of micro-environmental conditions. By developing automated methods for micro-environment detection, researchers can enhance the accuracy and reliability of air quality assessments, en-

abling a more comprehensive understanding of pollution patterns and facilitating targeted interventions to improve urban air quality. Detecting such micro-environments can help better calculate the user's exposure to air pollution, and it could be attached to some recommendation systems that suggest transportation types based on daily exposure.

The second topic within this thesis is enriching fixed stations generated air pollution maps with opportunistic mobile monitoring data. Fixed stations have proven to provide accurate, reliable, and continuous temporal coverage. However, these stations need more spatial coverage since it is hard to deploy such stations everywhere. On the other hand, mobile sensors have shown the ability to overcome the lackness of fixed stations in terms of spatial coverage. Nevertheless, those sensors may provide some inaccurate measures.

Different research works were conducted to estimate air pollution based either on fixed station measures[13, 52, 128], mobile sensor measures [51, 96], or even a combination of the two data types [56, 129]. The integration of fixed station and mobile monitoring data allows for more extensive coverage of geographical areas and enables the identification of pollution hotspots that may otherwise go unnoticed. Merging fixed and mobile sensory data is not a straightforward task. Challenges such as heterogeneity, data density, data inaccuracy, and volume can result from merging both data types. In addition, all the mobile sensory data in the previous studies were collected based on targeted campaigns [96, 146, 56]. Mainly, the data collection was performed on some specified routes and hotspots. At the same time, no studies, to the best of our knowledge, have used opportunistic mobile participatory monitoring to enrich the fixed stations' data and utilize interpolation techniques on top of such resultant data.

Another critical topic is qualitative external data enrichment. Enriching data analysis with qualitative data from tweets is essential due to its wide range of applications, especially in local event detection [144, 138]. Tweets are user-generated content that can provide valuable insights into various aspects of local events, such as natural disasters, public gatherings, or community opinion. This allows for the discovery of developing patterns, the identification of possible risks or opportunities, and the capacity to respond quickly to circumstances as they unfold. Qualitative data from tweets may be automatically analyzed, classified, and filtered using machine learning and natural language processing algorithms to extract useful information about local events. This valuable data can aid various stakeholders, including emergency responders, journalists, urban planners, and policymakers, in making informed decisions and implementing effective strategies to mitigate the impact of events, enhance public safety, and promote community engagement.

Only a small portion of geotagged tweets are available. Only around 1 - 3 % of tweets contain geolocation information [126], which makes analysis a challenging task in the absence of this information. In order to address the scarcity of geotagged tweets, researchers have increasingly relied on geolocation prediction approaches

[24, 103, 89]. These innovative techniques employ a range of methodologies to infer the geographic location of a tweet, even if it lacks explicit geotagging. These techniques continue to advance and improve, enhancing the accuracy of the predictions. Nowadays, we have effective approaches that can predict the location of tweets at country and city levels. However, the attempts to predict the location with reduced distance error are still preliminary. In such scenarios as local event detection, the precise location is crucial; thus, we need to find a methodology that minimizes the distance error of the predictions to use those tweets in such applications.

Based on the mentioned topics that this thesis will handle, our main objective and goal is to develop mining methods and study multi-dimensional analysis to help decision-makers solve the urbanization challenges. Formally we can define a main objective for each mentioned topic:

- Automatic Micro-environment recognition in the context of MCS, while comparing the approach of using only environmental data versus using such data with GPS data.
- Enriching air pollution fixed stations maps with opportunistic mobile participatory monitoring to better estimate air pollution in uncovered spots.
- Predicting tweet's location to make use of these tweets in local events detection, especially those related to pollution, knowing that most tweets miss the geolocation information.

1.4 . Problem Statement & Research Questions

Monitoring air pollution using data collected from the crowd and data collected from fixed stations while enriching it with external data sources can offer many advantages. Fixed monitoring stations provide accurate measurements throughout the day, covering defined locations. Mobile crowd sensing helps expand spatial coverage, capture local variabilities, and allow participants to gain insights about their exposure to pollution. In addition, enriching collected data with external data sources provides a better understanding of the urbanization problem.

Although this technology provides several advantages compared to conventional monitoring infrastructure, it is not straightforward, as many challenges arise when using such monitoring systems.

The following are some challenges of the different topics we are dealing with:

- The missing data and data inaccuracy problems within the mobile crowd sensing collection.
- The rareness in ground truth of micro-environments annotations by the users, as it is a hard task.

- Heterogeneity among the different devices will raise integration problems.
- The hybrid monitoring system combing fixed and mobile devices result in different data collection protocols.
- Opportunistic mobile participatory data have a low contribution to map enrichment.
- External data sources may need a specific treatment to integrate them into the system.
- Rareness of available geotagged tweets.

We consider different aspects of data to achieve the objectives mentioned earlier and effectively make sense of the collected data series. Starting with data acquisition and dealing with the heterogeneity of different data sources and hybrid networks. Passing through data preprocessing to clean up the raw data and make it usable. Then, the data enrichment phase changed the collected raw data into more semantic views for efficient multidimensional data analysis. Reaching data analysis which includes learning meaningful patterns from the enriched data. All those aspects will be treated within the context of the thesis work. The solutions will be implemented and evaluated from a large-scale collection of complex data series perspectives and applied to a real scenario of opportunistic air quality monitoring.

Starting with the first topic micro-environment recognition. The problem of automatically annotating MCS data can be seen as a problem of activity recognition from rich trajectory data collected by heterogeneous sensors. The available data in such scenarios may have a lot of missing values. The low-cost sensors used may lead to some inaccurate measures. Besides, we lack the ground truth as annotating micro-environments by participants is not easy.

We also have several challenges for the second topic, estimating air pollution on top of data from fixed stations and opportunistic mobile participatory monitoring data. Opportunistic data acquired from participants performing their real-life activities have a low contribution to the pollution map. The main challenge is to utilize such data in the enrichment process. Thus, we can enhance air pollution estimation.

Moreover, for qualitative data, the rareness of available geotagged tweets makes location prediction an important task to better use such tweets. While predicting a text's country or city location becomes easier nowadays with the proposed methods, identifying the location at a finer granularity (i.e., a higher spatial resolution) is still complex.

To more accurately express our motivation to provide decision-makers insights while using data mining and analysis techniques for quantitative and qualitative data. Based on our objectives, we divide the topics into a collection of fundamental research questions we will examine, discuss, and answer during this thesis work.

- **R1. How to deal with complex, missing, heterogeneous multivariate time series data while automating the micro-environment recognition?** This research question tackles the challenge of automatically labeling MCS data to identify micro-environments based on the abundant trajectory data from different sensors. In addition, it investigates the relationship between environmental data and micro-environments, exploring the degree to which environmental data can determine the characteristics of different micro-environments.

- **R2. How to estimate air pollution levels in uncovered spots while enriching fixed stations generated maps with opportunistic Mobile Participatory Monitoring (MPM) data?** This research question discusses how can the MPM data or mobile crowd sensing (MCS) data collected opportunistically, where no targeted scope is defined, help enrich the air pollution map. Opportunistic MPM data is usually collected by participants conducting their real-life activities. In other words, it is not targeted data collection; thus, the contribution to the map can be limited. Then, using different approaches, including deterministic methods, geostatistical methods, and machine/deep learning models, to determine the best method for estimating air pollution in uncovered spots depending on the available data.

- **R3. How to predict a tweet's location more precisely at a fine granularity to minimize the distance error?** This question explores the existing methods and techniques researchers utilize to predict geolocation in social media. How to tackle the challenge posed by the limited availability of geotagged tweets, which hampers location-based analysis? Moreover, how to integrate different methods to enhance location prediction precision and minimize estimation errors in determining the actual location of tweets.

Each research question mentioned above is general and contains other minor questions. Each chapter in this dissertation will try to answer one raised question.

1.5 . Contributions

Towards the motivation to establish effective and efficient data enrichment models in the context of IoT devices, this dissertation focuses on providing different enrichment approaches on top of quantitative and qualitative data. Specifically, this thesis formulates the following major contributions to answer the aforementioned research questions:

- **C1: Implementing an End-to-End pipeline for micro-environment recognition from data collection to analysis while using the multi-view learning approach.** Presenting a comprehensive approach that focuses on preparing, processing, applying, and comparing various machine

learning algorithms for annotating and detecting micro-environments using diverse trajectory data collected from heterogeneous sensors. The work explores the potential of multivariate time series classification (MTSC) for recognizing activities and detecting micro-environments within the MCS data context. It takes into consideration the challenges posed by missing data and the heterogeneous nature of sensor inputs. Also, it advances the understanding of its applicability for analyzing micro-environments using diverse sensor data. This contribution is a joint work [2, 41] with another colleague, the extension of this work is found in [39].

- **C2: Proposing a methodology to enhance the enrichment of each group in fixed station-generated maps by clustering them, which enables the incorporation of relevant opportunistic MPM data to better estimate air pollution levels in unmonitored locations.** A novel methodology is proposed in this research to estimate air pollution levels by merging data from fixed and mobile sensors. Unlike previous studies focusing on specific routes, this research utilizes mobile crowd-sensing (MCS) data obtained from volunteers' sensor-enhanced mobile devices with GPS capabilities during their daily activities. This non-persistent data collection approach captures real-life scenarios and indoor environments, contributing to advancing air pollution estimation techniques by incorporating MCS data in a broader and more representative manner. The study aims to enhance the accuracy and coverage of air pollution maps by integrating fixed station data with MCS data. This work improves the spatial and temporal coverage of pollution estimation by identifying clusters of different fixed station data and matching them with corresponding MCS data collected at the same periods. This approach leads to the creation of enriched pollution maps that provide detailed insights into air pollution levels' variability at a higher resolution.

- **C3: Proposing an End-to-End hybrid framework that uses existing methods to predict city-level, then uses Fine-grained LocAtIon pRediction (FLAIR) algorithm to predict the fine granularity.** Through proposing geolocation prediction approaches that incorporate various factors such as text content, user profiles, and social network connections. The aim is to overcome the limitation of limited geotagged tweet data by exploring alternative approaches to geolocation prediction. By considering factors beyond geotagged information, such as the textual content of tweets, user profiles, and social network connections, the work introduces new insights and techniques for inferring tweet locations, even without explicit geotags. Combining existing and proposed work enhances the accuracy of predicting coarse and fine granularity locations for non-geotagged tweets. This contribution proposes an innovative hybrid framework integrating existing

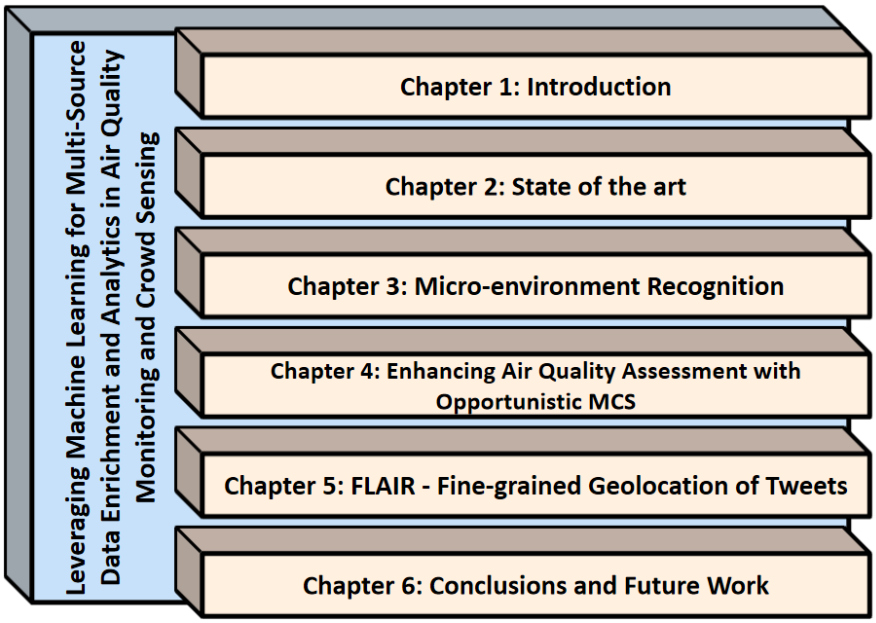


Figure 1.2: Dissertation structure organization.

geolocation prediction approaches with the Fine-grained LocAtIon pRediction (FLAIR) algorithm. By combining different methods, the framework aims to improve the accuracy of predicting the location, both at coarse and fine granularity, for tweets that do not have explicit geotags.

These are the main contributions of this thesis work. However, the following chapters break each main contribution into sub-contributions.

1.6 . Structure of the Dissertation

Figure 1.2 portrays the structure of the dissertation. Specifically, the rest of this thesis is organized as follows:

- Chapter 2 reviews existing works on quantitative and qualitative enrichment. Mainly focusing on activity and context enrichment in the context of mobile crowd sensing. Text location prediction enrichment at the coarse and fine granularity. In addition, fixed and mobile sensory data map enrichment to estimate air pollution.
- Chapter 3 focuses on raw trajectory enrichment to semantic trajectories. It investigates activity recognition on top of mobile crowd sensing to add semantic enrichment to the trajectory.
- Chapter 4 discusses the methodology of enriching fixed station data with mobile crowd-sensing data to estimate air pollution better by investigating unsupervised

and interpolation learning methods.

- Chapter 5 presents an end-to-end framework for enriching text with location information. It proposes a coarse and fine granularity text location prediction approach based on spatial and textual views.

- Chapter 6 provides general conclusions of this dissertation and highlights some future work directions.

1.7 . Publications

During this thesis work, several contributions to the research field have been achieved. Articles have been published or submitted in different venues, ranging between workshops, national conferences, international conferences, and journals.

- **Abboud, M.**, Taher, Y., Zeitouni, K., and Olteanu-Raimond, A. M. (2023). "How opportunistic mobile participatory monitoring can enhance air quality assessment?". Submitted to GeoInformatica Journal.

- **Abboud, M.**, Zeitouni, K., and Taher, Y. (2023). "Coarse to Fine Location Prediction of non-geotagged tweets". Submitted to TSAS Journal.

- **Abboud, M.**, Zeitouni, K., & Taher, Y. (2023). Enriching fixed stations air pollution monitoring with opportunistic mobile monitoring. In Big Mobility Data Analytics with EDBT 2023 (BMDA'23).

- Bouillon L, Gros V, **Abboud M**, El Hafyani H, Zeitouni K, Alage S, Languille B, Bonnaire N, Naude JM, Srairi S, Campos Y Sansano A. NO2, BC and PM Exposure of Participants in the Polluscope Autumn 2019 Campaign in the Paris Region. Toxics. 2023 Feb 23;11(3):206.

- **Abboud, M.**, Zeitouni, K., and Taher, Y. (2022, November). Fine-grained location prediction of non geo-tagged tweets: a multi-view learning approach. In Proceedings of the 5th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery (pp. 82-91).

- El Hafyani, H., **Abboud, M.**, Zuo, J., Zeitouni, K., Taher, Y., Chaix, B., and Wang, L. (2022). "Learning the micro-environment from rich trajectories in the context of mobile crowd sensing: Application to air quality monitoring." GeoInformatica, 1-44.

- H. El Hafyani, **M. Abboud** and Y. Taher. "A Microservices Based Architecture for Implementing and Automating ETL Data Pipelines for Mobile Crowdsensing Applications". In 2021 IEEE International Conference on Big Data (Big Data) (pp. 5909-5911). IEEE.

- H. El Hafyani, **M. Abboud**, J. Zuo, K. Zeitouni and Y. Taher. "Tell Me What Air You Breathe, I Tell You Where You Are". The 17th International Symposium on Spatial and Temporal Databases (SSTD'21). Accepted also in 37ème Conférence sur la Gestion de Données – Principes, Technologies et Applications 2021 (BDA'21).

- **M. Abboud**, H. El Hafyani, J. Zuo, K. Zeitouni and Y. Taher. "Micro-environment Recognition in the context of Environmental Crowdsensing". In Big Mobility Data Analytics with EDBT 2021 (BMDA'21).

- H. El Hafyani, K. Zeitouni, Y. Taher, **M. Abboud**. Leveraging change point detection for activity transition mining in the context of environmental crowdsensing. The 9th SIGKDD International Workshop on Urban Computing, San Diego, CA, 24/08/202. Accepted also as short research paper at 36ème conference sur la Gestion de Données – Principes, Technologies et Applications 2020 (BDA'20).

Chapter 2 - State of the Art

Contents

2.1 . Introduction

Enrichment through integrating quantitative and qualitative data has become a prevalent approach across various disciplines, including social sciences, public health, and environmental research. This approach combines diverse data sources to obtain a more comprehensive understanding of the phenomenon being studied.

Our current work investigates the potential of enrichment by utilizing both quantitative and qualitative data. Specifically, we focus on identifying participants' activities and contexts based on data from wearable sensors and GPS devices. We are also interested in estimating air pollution levels by leveraging data collected from stationary and mobile sensors. Alongside the quantitative data, we also explore incorporating qualitative data during enrichment.

Analyzing user mobility using GPS and sensor data is a central aspect of ubiquitous computing. Extensive research has been conducted in sensory data mining to extract valuable insights from raw data. Furthermore, employing fixed and mobile sensory data to estimate air pollution can enhance such assessments' spatial and temporal coverage. Additionally, examining qualitative data sources, such as tweets from specific regions, can contribute to a better understanding of ongoing events. This chapter provides an extensive review of state-of-the-art studies that address various related issues.

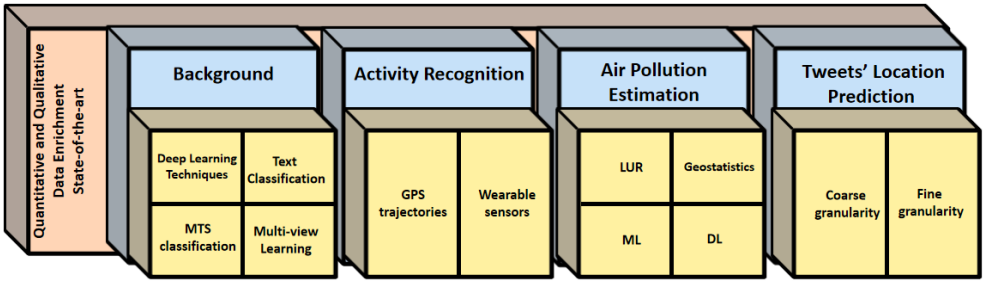


Figure 2.1: Structure of state-of-the-art chapter.

In this chapter, we commence with a background section that presents a broad overview of the machine learning and deep learning techniques employed in the thesis research. We extensively review relevant studies concerning enriched trajectory mining, encompassing activity recognition and multivariate time series classification. Additionally, we explore text location prediction approaches at both coarse and fine granularities and various methods for estimating air pollution, including Land Use Regression (LUR), Geostatistics, machine learning, and deep learning methods. Figure 2.1 illustrates the components under investigation in this chapter: Background, Activity Recognition, Tweets' Location Prediction, and Air Pollution Estimation.

2.2 . Background

In this section, we provide an in-depth examination of the fundamental machine and deep learning techniques that form the basis for the methodologies employed in this thesis. We explore a variety of methodologies, including classic machine learning algorithms and cutting-edge deep learning models like Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. Then, we present text classification, a key problem in natural language processing, where the objective is to automatically assign predetermined groups or labels to textual input. Furthermore, we dig into the field of multivariate time series classification, which is concerned with analyzing and classifying data sequences that contain numerous variables or characteristics that evolve. We examine the complexities associated with this task and methods like Recurrent Neural Networks (RNNs) and their variations, which have successfully identified temporal dependencies. Also, we introduce the idea of multi-view learning, which entails combining data from various viewpoints or sources to enhance classification performance. We aim to provide a thorough understanding of the various machine and deep learning techniques used in this thesis.

2.2.1 . Deep Learning Techniques

Deep learning techniques have revolutionized the field of machine learning, enabling powerful modeling of complex patterns and structures in data. Convolutional neural networks (CNNs) and long short-term memory (LSTM) networks have distinguished themselves as essential elements for managing many sorts of data, including sequential data and pictures, respectively. The capabilities of these models have also been further enhanced by using ConvLSTM architectures and the CNN-LSTM (a combination of CNN and LSTM). We examine various deep learning methodologies and their uses in this section.

CNN

Convolutional Neural Networks (CNNs) have made major contributions to the advancement of computer vision by enabling efficient visual input and image processing. Using convolutional filters to take advantage of spatial linkages, CNNs automatically learn hierarchical representations from incoming data. While the network's architecture allows for the inclusion of increasingly complicated representations, these filters capture regional trends and characteristics.

CNNs have succeeded in several applications, including image segmentation, object identification, and classification. LeCun et al. groundbreaking research developed the LeNet-5 [83] design, demonstrating the potency of CNNs for handwritten digit recognition. Later developments have significantly enhanced the functionality and capability of CNN models, including AlexNet [76], VGG [124], and ResNet [53].

Beyond image processing, convolutional neural networks (CNNs) have many uses. CNNs have demonstrated promising results in text categorization, sentiment analysis, and question-answering tasks [71] in the area of Natural Language Processing (NLP). Additionally, automated voice recognition (ASR) and accurate transcription are made possible by the use of CNNs [119]. CNNs have also been used for video analysis tasks, including action identification, video summarization, and video captioning, to name a few [123]. By identifying patterns in user-item interaction data, CNNs have helped recommendation systems increase the precision of customized suggestions [54]. CNNs have been applied in genomics to examine DNA sequences, detecting regulatory regions and gene expression patterns [7]. CNNs have changed medical applications, including illness diagnosis, tumor detection, and radiology image analysis [92]. Time series analysis, robotics, and autonomous systems also leverage CNNs for tasks like predicting stock prices, object recognition, and scene understanding [123, 119].

LSTM

Recurrent neural networks (RNNs) using LSTMs are made to represent sequential data and deal with the difficulties of capturing long-term relationships. Traditional RNNs frequently encounter disappearing or inflating gradients, which reduces their ability to learn from lengthy sequences. By including memory cells and gates that selectively store, update, or delete information at each time step, LSTMs get around this restriction.

Natural language processing, audio recognition, sentiment analysis, and time series forecasting are just a few of the several sequential data processing tasks that have shown LSTMs to be quite effective in the past. This sort of network was established by Hochreiter and Schmidhuber's LSTM architecture [55], which has subsequently been improved and expanded with versions including Gated Recurrent Units (GRUs) [26] and Bidirectional LSTMs (BiLSTMs) [49].

CNN-LSTM and ConvLSTM

The fusion of the CNN and LSTM architectures has produced robust models that can handle spatial and sequential data. In the CNN-LSTM model, spatial properties are extracted from input data using CNN layers and fed into LSTM layers to capture temporal relationships. Because of this combination, the model can concurrently learn high-level spatial representations and temporal dynamics.

Application areas for CNN-LSTM models include spatiotemporal analysis, action identification, and video categorization. For instance, CNN-LSTM models may train to recognize actions based on visual and motion signals retrieved from video frames while performing action identification tasks. Similarly, ConvLSTM designs expand LSTM networks by adding convolutional operations within the LSTM cells, allowing them to directly analyze spatiotemporal sequences.

ConvLSTM models have been successfully applied to video prediction, video segmentation, and anomaly detection in videos. By combining the capabilities of CNNs and LSTMs, ConvLSTM models can capture spatial and temporal dependencies in video data, facilitating accurate predictions and analysis.

Deep Learning for Spatio-temporal data

In machine learning and deep learning applications, dealing with spatiotemporal data presents key challenges. Spatio-temporal data has spatial and temporal aspects, such as movies, sensor readings, or geographical data gathered over time. Analyzing such data need models capable of capturing the intricate relationships between geographical locations and temporal events. One popular strategy is to combine Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. CNNs may extract spatial characteristics from each frame or location by adding convolutional processes, whereas LSTM layers collect temporal relationships across frames or sequences [64, 132]. This combination enables the models to learn geographically and temporally dependent patterns and dynamics.

Furthermore, 3D convolutions and spatiotemporal attention mechanisms have been developed to improve spatiotemporal data modeling. 3D convolutions provide a temporal dimension to classic 2D convolutions, allowing for direct processing of spatio-temporal volumes [132]. Spatiotemporal attention processes capture relevant geographical and temporal regions by dynamically weighting the importance of various spatial locations and time increments [136]. These approaches give more extensive representations of spatiotemporal data and have been effectively employed in various areas, including action identification, video comprehension, and environmental monitoring [141, 136].

2.2.2 . Text Classification

Text classification, an essential issue in natural language processing (NLP), involves categorizing or labeling text documents. Numerous methodologies, ranging from classic machine learning algorithms to deep learning-based approaches, have been developed to solve this challenge over the years. Transformer-based models, notably BERT (Bidirectional Encoder Representations from Transformers), have recently emerged as cutting-edge text categorization approaches due to their efficient collection of contextual information.

BERT, launched in 2018 by [33], changed NLP using transformer layouts. Transformers are deep learning models that use an attention-based mechanism to handle sequential input, such as text. As a pre-trained language model, BERT is trained on massive volumes of unlabeled text from the internet, allowing it to acquire complex representations of words and phrases. The pre-training of BERT involves two crucial steps: masked language modeling (MLM) and next sentence prediction (NSP). MLM randomly masks words in a sentence, and BERT learns to predict the masked words based on the context provided by the surrounding

words. NSP involves providing two sentences as input and training BERT to predict whether the second sentence follows the first in the original text. This pre-training process endows BERT with a strong understanding of context and semantics.

A task-specific layer is built on the pre-trained model to adjust BERT for classifying text. This layer is often a simple neural network, such as a feed-forward network or a softmax layer, that receives the final hidden states of BERT as input and predicts target labels. BERT may learn task-specific patterns and achieve exceptional performance by fine-tuning the whole model on labeled data from the specific text classification job.

Other transformer-based models, besides BERT, have shown good performance in classifying texts. GPT (Generative Pre-trained Transformer), created by [113] in 2018, is one famous example. In contrast to BERT, which is trained for bidirectional understanding, GPT is trained for autoregressive language modeling, creating text one token at a time dependent on the tokens before it. GPT may be fine-tuned for text categorization by incorporating a task-specific layer, much like BERT.

In addition to transformers, several other NLP techniques have been employed for text classification. Word embeddings are one of those techniques, which represent words as dense vectors in a continuous space, capturing semantic relationships between words. Popular word embedding models include Word2Vec, introduced by [102], GloVe, introduced by [110], and FastText, introduced by [67]. These embedding vectors are used as input to train traditional machine learning algorithms or deep learning models.

Recurrent neural networks (RNNs) are another way that processes sequential input by preserving hidden states that collect information from previous phases. Popular RNN variations for text classification include Long Short-Term Memory (LSTM) established by [55] and Gated Recurrent Unit (GRU) presented by [26]. RNNs have proved effective at modeling sequential dependencies in text, but due to the vanishing gradient problem, they may struggle with capturing long-range dependencies.

2.2.3 . Multivariate Time Series Classification

Human activity recognition falls into the problem of labeling data segments with the activity type, leading to a multivariate time series classification (MTSC) problem based on data collected by multiple sources. There is a wide range of time series classification approaches that can be classified into four categories: distance-based methods [14], feature-based methods [111], ensemble methods [46], and deep learning models [43][22][135]. The one-nearest neighbor (1-NN) classifier with different distance measures, such as Euclidean distance (ED) or dynamic time wrapping (DTW) [14], is always considered as the benchmark to give a preliminary evaluation in the MTSC problem.

Feature-based methods are based on various features learned from TS data, through which we can distinguish the differences between data and classify them.

The disadvantages of these methods lie in the complexity and weak generality of building features, which limits their versatility. This method follows the abovementioned approach, depicted in Figure 2.5.

Besides hand-engineered features, some methods use the deep neural network (DNN) to extract time series features for classification. In their survey, Fawaz *et al.* [43] review the current studies of deep learning algorithms for time series classification (TSC) and present an empirical study of the most recent DNN architectures for TSC, including convolutional neural network (CNN), recurrent neural network (RNN), echo state network (ESN), and multi-layer perceptron (MLP). Besides univariate time series, the authors tested the approaches on 12 multivariate time series datasets and gave an overview of the most successful deep learning applications.

Considering the real-life scenarios where it is difficult or expensive to obtain a large amount of labeled data for training, some studies used labeled and unlabeled data to learn the human activity, that is, semi-supervised learning (SSL) [139] on MTSC. The pioneering work by [139] proposes a semi-supervised technique for time series classification. The authors demonstrated that semi-supervised learning requires less human effort and generally achieves higher accuracy than training on limited labels. The semi-supervised model [139] is based on the self-learning concept with the one-nearest-neighbor (1-NN) classifier. First, the labeled set denoted by P (as positively labeled) is applied to train the 1-NN classifier C. Then, the unlabeled samples U are given the pseudo labels progressively based on their distance to the samples in P. After that, the enriched labeled set P allows iteratively repeating the previous step and improving the classifier. More recently, the deep learning-based models on MTSC have shown promising performance under weak supervision. For instance, Zhang et al. [150] propose a novel semi-supervised MTSC model named time series attentional prototype network (TapNet) to explore the valuable information in the unlabeled samples. TapNet projects the raw MTS data into a low-dimensional representation space. The unlabeled samples approach themselves to the class prototype in the representation space, where pseudo labels are generated by the distance-based probability allowing the model's training progressively. Moreover, the hybrid convolutional neural network (CNN) and long short-term memory (LSTM) structure adopted in TapNet allows the modeling of the variable interactions and the temporal features of MTS.

2.2.4 . Multi-View Learning

Another line of studies proposes multi-view learning to classify time series data originating from multiple sensors to recognize user activities. Garcia-Ceja et al. [46] propose a method based on multi-view learning and stacked generalization for fusing audio and accelerometer sensor data for human activity recognition using wearable devices. Each sensor's data is seen as a different "view" and combined using stacked generalization [140]. The approach trains a specific classification model over each view and an extra meta-learner using the view models as input.

The general idea of the authors is to combine data from heterogeneous types of sensors to complement each other and, thus, increase recognition accuracy.

Wang et al. [134] propose a framework based on deep learning to learn features from different aspects of the data based on features of sequence and visualization. In order to imitate the human brain, which can classify data based on visualization, the authors transform the time series into an area graph. The area graph here is used to model time series as images to apply a Convolutional Neural Network (CNN) on top of it. They use well-trained Long short-term memory with an attention mechanism (LSTM-A) neural networks and CNN with attention (CNN-A) to extract the features of time series data. LSTM-A extracts sequence features, while CNN-A extracts visual features from the time series. Then, based on the fusion of features, the authors carry out the time series classification task. Although the approach gained promising results, further performance gain was achieved by recent deep learning methods such as InceptionTime [44].

Li et al. [85] propose multi-view discriminative bilinear projections (MDBP) for multi-view MTSC. The proposed approach is a multi-view dimensionality reduction method for time series classification, which aims to extract discriminative features from multi-view MTS data. MDBP mainly projects multi-view data to a shared subspace through view-specific bilinear projections that preserve the temporal structure of MTS and learn discriminative features by incorporating a novel supervised regularization.

2.2.5 . Discussion

In brief, a thorough overview of the fundamental machine and deep learning techniques forms the basis for the strategies used in this thesis. In our discussion of text classification methods, we covered conventional machine learning algorithms and cutting-edge deep learning models like CNNs and LSTMs. These techniques have demonstrated impressive performance in extracting contextual and semantic information from textual data, allowing precise classification and analysis. We also looked into multivariate time series classification, which involves analyzing and categorizing data sequences with multiple variables that change over time. In order to capture temporal dependencies and extract discriminative features from multivariate time series data, methods like RNNs and their variations be effective.

Lastly, we introduced the idea of multi-view learning, which entails using knowledge from various viewpoints or data sources to enhance classification performance. For multi-view data, fusion-based methods and specialized deep learning architectures were discussed. The thesis aims to create fresh strategies for tackling particular difficulties in micro-environment recognition, tweet location prediction and qualitative enrichment, and air pollution estimation.

2.3 . Activity Recognition

In this section, we are interested in addressing the work related to our raised question **R1**. We are considering micro-environment recognition as an activity recognition task, thus, we will further present the state-of-art work in this field. Human activity recognition covers a wide range of applications, including intelligent home activities [8], daily human activities [148][93][25], and human mobility [34][154], to name a few. It simulates a typical machine learning scenario, and the benchmarks heavily rely on several open datasets. We evaluate a few research that used data from mobile sensing to identify activities in this section. We primarily focus on detecting activities using GPS trajectory and wearable sensor data.

2.3.1 . Activity Recognition from GPS Trajectories

Recent research has been directed towards the identification of activities using GPS-based trajectory data. As illustrated in Figure 2.2, this particular challenge revolves around annotating raw trajectory data, or its segments, with semantic labels. These labels contribute to the interpretation of trajectory data on a semantic level, providing insights into the mobility patterns of the moving object. The integration of semantic information serves various applications, including offering trip recommendations or identifying frequently visited locations by a moving object.

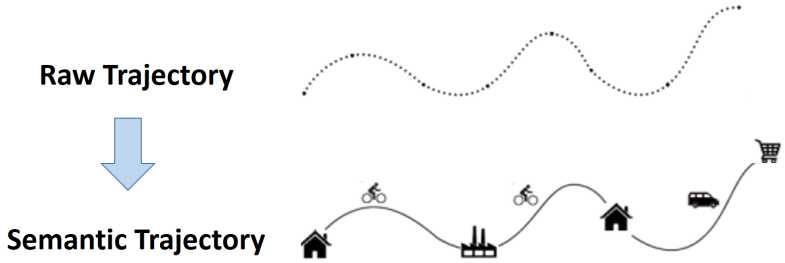


Figure 2.2: From raw trajectory data to semantic trajectory.

In his survey [153], Zheng differentiates between the application requirements of associating semantic labels with raw trajectory data or its segments, such as inferring transportation modes and identifying human activities. His proposed framework outlines three key processes for activity identification from GPS trajectory data: (1)Employ a segmentation method to partition the trajectory data. For instance, segment the trajectory data into stop and move segments and subsequently further divide the motion segments based on changes in the mode of movement. (2) Recognize characteristics for each segment (or point) within a trajectory. (3) Develop a model capable of categorizing segments (or points) based on their identified characteristics.

Figure 2.3 illustrates the overall data flow for activity recognition from GPS trajectory data. Given that a trajectory is essentially a sequence, sequence inference models like Dynamic Bayesian Network (DBN), Hidden Markov Model (HMM), and Conditional Random Field (CRF) can be employed to assimilate information from

local points (or segments) within the trajectory and capture sequential patterns between adjacent points (or segments). In a more recent survey, Mazimpaka and Timpf [100] provide an overview of generic methods for trajectory data mining and the interrelationships among them. The authors align with Zheng's methodology, noting that the majority of trajectory classification algorithms adhere to a conventional two-step approach: extracting discriminative features and then utilizing these features to train an established standard classification model.

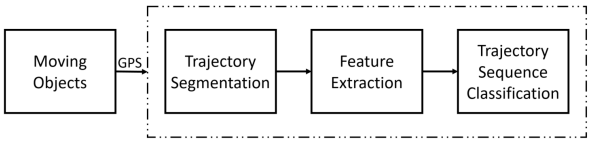


Figure 2.3: General data flow for activity recognition from GPS trajectory data.

Rehrl *et al.* [115] propose and evaluate a three-step trajectory data mining approach based on machine learning techniques. The authors first segment the trajectory into stay points clusters. After extracting 14 characteristics of each stop, they classify the detected stops into two categories: traffic-relevant and non-traffic-relevant.

In the study by Zheng *et al.* [154], leveraging GPS logs, the authors advocate for a supervised learning approach to deduce individuals' transportation modes, encompassing driving, walking, bus, and bicycle. The authors introduce a set of features designed to be resilient in varying traffic conditions, including heading change rate, stop rate, velocity change rate, and more. Subsequently, employing change point-based detection to segment the trajectory data, the authors extract features from each segment, utilizing them to train a supervised-learning model for transportation mode inference.

In their study, Etemad *et al.* [42] present a framework for predicting transportation modes exclusively based on GPS data. The authors introduce a process for generating trajectory point features and extracting trajectory segments features, incorporating factors like the bearing rate, change rate of the bearing rate, and global and local trajectory features. Subsequently, employing various machine learning algorithms such as *SVM*, *Decision Tree*, and *XGBoost*, the authors aim to classify the trajectory segments of the moving object into transportation modes, including walking, train, bus, bike, driving, among others.

The model introduced by [31] offers a travel mode inference framework utilizing convolutional neural network (CNN) schemes, allowing for the automatic extraction of high-level features from raw input. The model predicts travel mode labels using raw GPS trajectory data, encompassing walking, biking, bus, driving, and train. Notably, the CNN-based approach achieves state-of-the-art accuracy when applied to GPS data from the GeoLife dataset [155].

In a distinct domain focused on mining vessel trajectories, Kontopoulos *et al.* [73] undertake the classification of vessel trajectories in real-time streams into three activities: trawling, longlining, and underway. The trajectories are temporally segmented into intervals of 1, 2, 4, 8, 12, and 24 hours. A set of representative features, including average speed and its standard deviation, average drift and its standard deviation, and the number of turns, is generated for each activity. The authors then employ a *Random Forest* classifier, comparing its performance against three other classifiers: *Gradient Boosted Tree*, *Linear Discriminant Analysis*, and *Logistic Regression* to discern the fishing activities of the vessels.

2.3.2 . Activity Recognition from Wearable Sensors

An active area of research has emerged, driven by the aim of extracting valuable insights from data collected through ubiquitous sensors. This surge in interest is attributed to advancements in mobile devices and the availability of high-computational, compact, and cost-effective sensors. A key focus within this domain is the recognition of human activity using wearable sensors. This area holds significant promise for various applications, ranging from monitoring patients with conditions like diabetes or heart disease during their daily activities to providing contextual information for understanding pollution and other scenarios. Consequently, gaining a comprehensive understanding of activities such as running, walking, standing up, raising one's hand, and other contextual features becomes imperative for effective input in diverse application scenarios. This section delves into previous research efforts dedicated to Human Activity Recognition (HAR) utilizing wearable sensors.

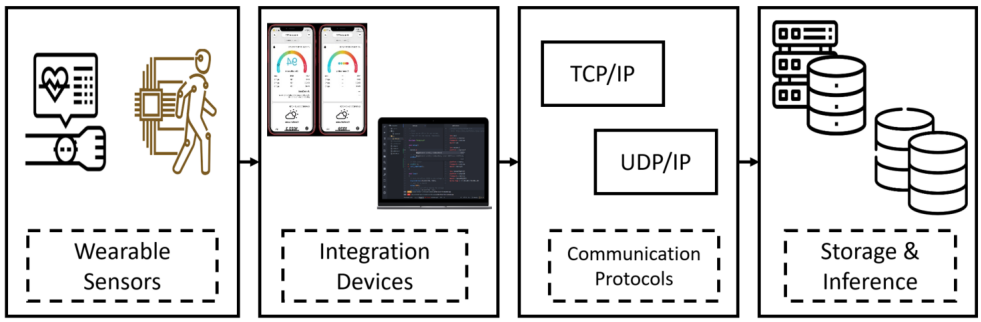


Figure 2.4: Generic data acquisition architecture for Human Activity Recognition.

In their comprehensive survey, Lara and Labrado [80] scrutinize the landscape of Human Activity Recognition (HAR) centered around wearable sensors. They propose a versatile data acquisition architecture tailored for HAR systems leveraging wearable sensors. Figure 2.4 delineates the data acquisition framework, spanning

from wearable sensors to data storage on either a local device or a remote server. The initial phase involves sensors affixed to the human body measuring pertinent information concerning specific phenomena, such as motion [60], location [29], temperature [111], ECG [63], among others. These wearable sensors establish communication with an integrated device, which could be a cellphone, PDA, laptop, or a customized embedded system. Subsequently, the gathered data from the sensors are transmitted to an application server for tasks like visualization, analysis, or real-time monitoring. Depending on the desired level of reliability, communication protocols like UDP/IP or TCP/IP can be employed. It is essential to note that not all these components are universally implemented in every HAR system, and their deployment varies based on the specific application scenario.

Figure 2.5: HAR system architecture based on wearable sensors.

Moreover, the authors introduce a generic architecture applicable to any Human Activity Recognition (HAR) system. They posit that activity recognition from wearable sensors, akin to other machine learning applications, undergoes two distinct stages: the training phase and the testing (or evaluation) phase. Figure 2.5 illustrates the typical processes involved in these stages. In the training phase, time series signals are partitioned into time windows, where feature extraction is applied, and relevant information is discerned. Subsequently, machine learning techniques are employed to build a HAR model from the dataset of extracted features. Analogously, in the testing phase, data is segmented based on a time window. The segmented data undergoes the same feature extraction process and is evaluated using the previously trained model.

In a similar approach to HAR system design, Parkka *et al.* [111] utilized various data signals from wearable sensors, such as an accelerometer, microphone, and air pressure, to classify common activities like walking, running, and cycling. Employing a 1-second segmentation technique, the authors extracted six features from the signal, including the peak frequency of up-down chest acceleration, the median of up-down chest acceleration, and the peak power of up-down chest acceleration. They applied three distinct classifiers to categorize the segments into daily activities: a custom decision tree, an automatically generated decision tree, and an artificial neural network (ANN).

In another study related to the healthcare assessment domain, Zhang and Sawchuk [148] introduce a framework based on Bag-of-Features (BoF) to construct activity recognition models using motion primitive symbols. The authors demonstrate the efficacy of their BoF-based approach in recognizing nine activity classes, including walking forward, walking left, walking right, going upstairs,

going downstairs, running forward, jumping up, sitting on a chair, and standing. On the other hand, Liu *et al.* [93] develop a dictionary of time series patterns, referred to as *shapelets*, to tackle the challenge of complex activity recognition, encompassing gestures or actions, from multiple sensors. The authors extend the concept of shapelet to represent complex activities by redefining the shapelet as a representation of the activity.

In a different approach from utilizing manually crafted features from sensor signals, Jiang *et al.* [66] suggest recognizing human physical activities based on accelerometer and gyroscope signals by automatically learning optimal features. The proposed methodology converts sensor signal sequences into images and employs Deep Convolutional Neural Networks (DCNN) to discern the most discriminative features for activity recognition, including walking, standing, and walking downstairs.

In the realm of deep learning-based approaches, Ordóñez et al. [106] introduce a framework for activity recognition incorporating convolutional and LSTM recurrent units. A key aspect of their approach is the automatic design of features without the need for expert knowledge. The authors showcase the effectiveness of their framework in inferring locomotion, postures, and gestures from wearable sensors. Additionally, Wang *et al.* [135] provide a summary of recent developments in sensor-based deep learning approaches for activity recognition.

2.3.3 . Discussion

In recent decades, there has been a growing focus on recognizing human activities. We have reviewed various approaches for detecting and categorizing activities using wearable sensors and GPS signals. Our review covers a wide range of activities, including common daily activities such as standing, walking, climbing stairs, and different forms of transportation. We have also discussed two classification systems based on GPS data and wearable sensors for recognizing activities.

In addition, we provided a large set of research proposals that employ feature extraction and classification to infer the type of activity and whether the processing is based on GPS data or wearable sensor signals. We highlighted that several types of wearable sensors are used in the HAR research proposals, such as accelerometers, microphones, and air pressure.

However, in the context of mobile crowd sensing (MCS), we avoid using the accelerometer or sound data due to privacy concerns. Additionally, previous proposals have been based on either geographical or temporal information. However, there is a need for an overall methodology that combines these aspects with real-world enriched trajectory data for a more robust inference model.

2.4 . Air Pollution Estimation

This section is dedicated to stating the existing work related to the third research question **R3**. We present a detailed overview of the various existing ap-

proaches in the literature, seeking an answer to our raised question.

Air quality assessment is crucial for monitoring and analyzing pollutant levels in the atmosphere, offering essential insights into the possible health risks and environmental consequences of air pollution. Several methodologies have been used for air quality evaluation, including chemical and physical monitoring, as well as data-driven approaches utilizing machine and deep learning techniques. This section examines the many scopes of work in air quality assessment and emphasizes the importance of machine and deep learning techniques in air pollution estimation.

Traditional air quality assessment techniques include chemical and physical monitoring. This involves the installation of physical sensors capable of measuring pollutants such as particulate matter (PM), nitrogen dioxide (NO_2), sulfur dioxide (SO_2), carbon monoxide (CO), and ozone (O_3). The collected data is analyzed statistically and analytically to calculate the air quality index (AQI) and identify probable pollution sources (U.S. Environmental Protection Agency, 2021; European Environment Agency, 2021).

Data-driven technologies, notably machine learning and deep learning, have gained importance in air quality monitoring in recent years. Machine learning models trained on historical air quality data may be used to anticipate pollution levels, find trends, and comprehend the interactions between various environmental parameters. These models help anticipate air quality and make decisions about pollution control strategies.

Deep learning focuses on training multiple-layer artificial neural networks to extract complicated patterns from difficult data. Deep learning algorithms can interpret sensory input from various sources, including satellite imaging, ground-level sensors, and meteorological data, to capture detailed correlations between variables and produce accurate forecasts of air pollution levels.

One notable example of integrating machine learning and sensor networks for air quality assessment is the CityScanner [9] project. CityScanner is a research project that blends machine learning algorithms with a network of dispersed sensors to build a complete and real-time monitoring system for air pollution. The studies use machine learning techniques to assess sensory data and predict air pollution levels within a metropolitan region. CityScanner delivers a more complete and precise knowledge of air quality dynamics by utilizing data from many sources, such as satellite images, ground-level sensors, and meteorological data. The use of machine learning in the CityScanner project demonstrates the promise of new computational approaches in transforming air quality monitoring and allowing informed pollution management decision-making.

Air quality assessment includes classic monitoring methods based on physical sensors and data-driven methods based on machine learning and deep learning techniques. While traditional methodologies give useful insights, using machine learning and deep learning in air quality assessment allows for more accurate forecasts, a better knowledge of pollution dynamics, and informed decision-making for

pollution management strategies.

The pollution estimation problem has attracted the interest of researchers over several years. It has been studied in the literature from different angles and at various scales. Meso-scale air quality modeling systems are the most common, with CHIMERE [99] and other studies [125, 57, 151]. Urban scale models that use Computational Fluid Dynamic (CFD) simulations have been proposed, but they are computationally intensive, which limits their application to a wide area [68, 69, 104, 121]. Besides these model-driven approaches, data-driven methods are trending, thanks to the growing deployment of monitoring stations, either a traditional fixed network, a denser network of low-cost fixed sensors, or low-cost mobile devices. We focus hereafter on the data-driven approaches. Moreover, we have different deployments, either dense sensors or sparse. In addition to different sensor network deployments, different types of data are collected in such applications (i.e., Air quality data, traffic data, meteorological data, etc.). Several studies proposed frameworks and systems to collect, process and analyze air pollution data and the related features.

Authors in [112] collected mobile monitoring data by installing the measurement devices on existing mobile sensor platforms. At the street level, they analyzed four atmospheric pollutants (NO2, PM1, PM2:5, and PM10). They provide a systematic guideline for processing and analyzing air pollution datasets with time sequence and geographic information. Moreover, they explore the temporal and spatial distribution of the considered pollutants and investigate the impact of various contextual factors on atmospheric pollutant concentration. The study shows the analysis conducted on the collected data and the implications and relationships of external data sources on the concentration of pollutants. The authors studied the impacts of road type, traffic signals, land use features, and low emission zones on pollution concentrations. The aforementioned results help in constructing effective air quality prediction models.

Moreover, MegaSence [114] is a Cyber-Physical System (CPS) for spatially distributed IoT-based monitoring of urban air quality. It can produce aggregated, privacy-aware maps and collected pollution data history graphs. It also provides a feedback loop in the form of personal air pollution exposure information, allowing citizens to take measures to avoid future exposure. The authors stated that MegaSense is the first end-to-end system providing coverage of air pollution exposure in different urban micro-environments to be used continuously throughout the day. It consists of two layers: the Edge and Cloud. The Edge layer is responsible for reactively receiving data from available data sources, such as sensor devices, traffic data systems, and weather information systems. It delivers advice and pollution maps to the mobile Exposure App, which provides the user with personal air pollution exposure information and district exposure maps. The Cloud layer is responsible for storing cleaned data and aggregating the crowd-sourced data while preserving participants' privacy. MegaSense provides exposure to pollution after

correlating air quality data and other external sources.

Another research [12] proposes a holistic, multi-dimensional approach to gather, monitor and analyze heterogeneous data sources of air pollutants and noise indicators into an integrated, intelligent computational system. The system will provide high-quality measurements and estimations, relying upon an underlying sensor network consisting of static and mobile sensors. The proposed system will collect data from various subjective and objective air quality and noise monitoring inputs. They proposed a spatial and temporal air-pollutant concentration estimation model based on environmental features.

However, the main objective of all studies is to expand the spatial and/or temporal coverage. This section summarizes conducted works on pollution estimation/interpolation for different types of measurements.

In recent years, several approaches have been proposed to estimate or interpolate pollution measures in areas lacking monitoring stations. Air quality estimation approaches are mainly designed for fixed stations, but the same approaches are adapted to all data collected from fixed or mobile sensors. Here comes a brief introduction to those approaches; they can be grouped into five groups (Land use regression (LUR), Dispersion models, Deterministic interpolation methods, Geostatistics, and ML/DL algorithms).

- **Land Use Regression** in short (LUR) methods that use local characteristics of the environment, such as land use features, meteorological features, etc., to find a correlation between those features and fixed station data and build a regression model.

- **Dispersion models** use mathematical formulations to characterize the atmospheric processes that disperse a pollutant emitted by a source. A dispersion model can predict concentrations at selected downwind receptor locations based on emissions and meteorological inputs.

- **Deterministic interpolation methods** calculate the value at the unknown location based on created surfaces from measured points. Inverse Distance Weighting is one of the most popular deterministic approaches, as it tries to interpolate the data at a specific location based on the weighted averages of collected data points.

- **Geostatistics** These techniques utilize statistical properties of the measured points. It is known by kriging method we have various types: simple Kriging, ordinary Kriging, etc... The main idea is to determine the spatial covariance of the collected data points. Then the derived weights from the covariance structure are used to interpolate values of un-sampled points.

- **ML/DL algorithms** Machine learning and Deep learning models try to map the input into the specific output based on features from the training set. Regression models from machine learning are used to build regression models

to interpolate data. In addition, CNN and LSTM are used to expand the spatial and temporal coverage.

2.4.1 . Land Use Regression

Habermann et al. in [52] used the LUR methods to visualize the geographical distribution of pollution concentration of NO2. They used LUR as it is based on characteristics related to the overall trends of air pollutant concentration. The authors considered NO2 measurements of the dependent features and built a LUR model based on land use, demographics, and geographical features. Statistical analysis is applied to find the correlations between each used feature and NO2. After calculating the predicted LUR-NO2 for each point, Kriging was applied to visualize the surface of the LUR model. Data were collected from 25 fixed sites, and the pollution estimation map was at 1km X 1km X 1h granularity. The results show that the model could predict almost 60% of NO2 variability, but LUR methods have some limitations that the authors mentioned in their work.

In [30], authors developed a LUR model to estimate intra-city nitrogen dioxide (NO2) exposure for a Sydney cohort. They compare those estimates from a national satellite-based LUR model (Sat-LUR) and a regional Bayesian Maximum Entropy (BME) model. NO2 and NOx were measured at 46 sites. Based on local knowledge, the sites were categorized a priori: 16 as traffic sites, 28 as urban background sites, and two as regional sites. For the LUR model, the explanatory regression variables were calculated for each geocoded address, and the estimates were made using the NO2 and NOx regression equations. Hold-out validation is considered an improvement on leave-one-out cross-validation (LOOCV) validation.

Li et al. in [87] proposed a LUR model using only routine air quality measurement data to evaluate the transferability of LUR models between nearby cities. Annual average LUR models and spatial distribution maps were developed for ambient particles of aerodynamic diameter less than or equal to 10 (PM10), PM2.5, nitrogen dioxide (NO2), and ozone (O3) in northern Taiwan in 2019. In addition, the transferability of LUR models between cities in the study area was evaluated. Supervised forward linear regression method [38, 15] was used to develop the LUR models, as it ensures that only predictor variables following the plausible direction of effect are included to maximize the predictive accuracy of the established model. The predictive performance varied greatly among air pollutants in examining the transferability of city-specific LUR models between New Taipei City, Keelung City, and Taipei City. The study highlights that the established LUR models in a city area can result in a significant estimation bias when applied to another nearby city area with similar geographic and urbanization conditions. The authors have stated the limitations of their study. The supervised forward linear regression method is not proficient in modeling extreme values. In addition, there may be complex and non-linear relationships between the explanatory variables and air pollutant concentrations. Presence of uncertainty in spatial estimations of air pollutant concentrations with limited sampling stations.

In [101], a land use regression (LUR) model was established to estimate NO2 during 2008–2011 in Shanghai. This study aims to develop a LUR model using the GIS variables for NO2 in Shanghai. To compare the model performance of this LUR model with interpolation methods based on prediction accuracy and predicted spatial variation. Also, evaluate the LUR model's temporal validation based on the ratio and the absolute difference methods. The collected data included NO2 monitoring data, population count data, road network data, land use data, industrial emissions data, and the distance to the coast. The nearest distance to the coast was calculated for each monitoring station based on the coastline shape file of Shanghai. A supervised forward regression method was used to select variables, and the results indicated that the LUR model performed better than pure geostatistical interpolation methods. The authors stated the limitations of their study. They didn't include the meteorological data; however, it is an important feature. In addition, few source-specific stations were targeting major roads and industrial sources. Their LUR model tended to reflect the average exposure of residents and was weak at the source-specific exposure assessment.

2.4.2 . Geostatistics

In [59], authors use geostatic methods on data collected from mobile sensors. Simple Kriging, ordinary Kriging, and kriging with external drift are applied. The advantage of the kriging approach is that it does not require external data. It just uses the available measurements. In this study, low-cost mobile sensors are deployed on top of trams (OpenSense [5]). Several experiments were conducted to compare Kriging and deterministic methods such as IDW. The results show the superiority of these methods. Geostatistics methods do not require external data, but machine learning methods have shown better performance when combining different data types to estimate pollution.

In [95], authors developed microscale variables of the urban environment, including Point of Interest (POI) data, Google Street View (GSV) imagery, and satellite-based measures of urban form to use them as features to various pollution estimation models. The idea is to combine the traditional predictor and microscale variables to enhance the models' performance. Different modeling approaches have been adopted, such as Geostatistics and Machine learning (Stepwise Regression + Kriging, Partial Least Square + Kriging, and Machine Learning + Kriging). The authors found that the microscale variables may be a valuable substitute for traditional variables. For example, models using the microscale variables alone performed similarly to models using the traditional variables.

[36] presents statistical approaches to fuse the data from fixed and mobile sensors for air quality monitoring. Authors in this work adapt mobile sensing data collection due to the wider coverage compared to traditional fixed monitoring stations. This work focuses on how this wider coverage can be used to estimate pollution at arbitrary points by exploiting mobile sensors' spatial and temporal coverage in combination with the accuracy of fixed stations. They aim to get

subsequent pollution maps for discrete points in time that incorporate as much information as possible to get the best accuracy. They focused on spatial coverage and applied kriging interpolation, which is to estimate a value at a specific location by computing a weighted average of the known values in the neighborhood of that location. Moreover, they consider confidence by referring to fixed stations to tune the error values that could be generated by mobile sampling.

In [131], authors adapt geostatistical methods to predict PM10 Concentration in Malaysian Cities. Spatial interpolation models and geostatistical analysis such as ordinary Kriging (OK), universal Kriging (UK), and Inverse Distance Weighting (IDW) were used in this study to predict and assess the distribution of PM10 to other regions. Compared to the other techniques, the kriging method is extensively applied for the geostatistical distribution of air pollution due to its high performance. In this study, the authors use two kriging methods: ordinary and universal. The OK is the most widely utilized kriging method that predicts a point value at a location by calculating the weighted mean for surrounding data. The weight derived from the first data estimated could compute all other data. On the other hand, the UK method also called the spatial smoothing model, is mainly utilized for data with a significant spatial trend. The UK method also describes the model residuals' spatial autocorrelation. Moreover, the used Inverse Distance Weighting (IDW) gives preference to data or point values closer to each other. In IDW local influence of the measured points is assumed to reduce with distance and points more relative to the target region will have a higher weight. The study findings were based on the ten monitoring stations mounted around urban and industrial areas. The OK method showed better prediction results than IDW and UK. However, the authors stated that it is hard to conclude which models are more reliable for air pollution prediction. The over or under-prediction by the kriging methods may be subjected to these models' inability to interpolate data that are not statistically stationary.

2.4.3 . Machine and Deep Learning Models

ML Models

Guo et al. proposed a high-resolution Urban air quality mapping for multiple pollutants [51]. The authors propose a method to address the challenges of high-resolution air quality mapping by combining dense networks and machine learning techniques. This study was conducted depending on a dense monitoring network. Guo et al. took advantage of the emerging micro-station monitoring systems with multiple sensors. They used data from 448 micro-stations deployed in Lanzhou city to infer the distribution of citywide air quality at 500m X 500m X 1h. In addition, to the sensor data, they used land use and meteorological data. XGBoost algorithm was adapted on top of the collected data to estimate pollution at different grids. This work can predict the pollution concentration at fine granularities, but the monitoring phase was based on a dense network data collection.

Zheng et al. proposed a semi-supervised approach with temporal and spatial models to estimate pollution in [157]. The approach is based on a co-training framework that consists of an ANN for the spatial features and a linear chain conditional random field (CFR) handling the temporal features. They built a model on top of historical and real-time data, combined with multiple heterogeneous data such as traffic data, meteorology data, POIs (point-of-interest), etc. Air quality measurements, POIs, and road networks feed the spatial model. At the same time, the temporal model takes as an input the traffic, mobility, and weather features. The proposed approach shows a high precision when compared to other classical approaches.

Zhang et al. proposed machine learning regression models to predict real-time localized air quality in [146]. Multiple static sensors beside IoT mobile sensors of the same type are deployed to effectively monitor air quality. Authors developed machine learning regression models to estimate pollution. The gradient boosting model is the most responsive to sudden changes, while SVR and RFR are good at finding the overall trends. The results show that the hybrid network has better outcomes for all selected dates.

In [88], authors adapted mobile sampling low-cost sensors and machine learning to map urban air quality in Seoul, Korea. They collected data by conducting three weeks of campaigns across five routes with ten volunteers sharing seven AirBeams, a low-cost, smartphone-based particle counter. In contrast, geospatial data were extracted from OpenStreetMap. They applied three statistical approaches to constructing the LUR model: linear regression, random forest, and stacked ensemble. The main aim of this study was to deploy multiple units of the smartphone-based particle counter 'AirBeam' to measure and model street-level urban air quality in Seoul, South Korea. A location with limited fixed regulatory monitoring sites relative to the high population and diverse urban environments. The collected air pollution data, together with an openly available and crowd-sourced geographical data source OpenStreetMap (OSM), were then used to construct LUR models using linear regression and machine learning methods. Notable differences between morning, evening, and night were also observed across the five routes, and the LUR model was sensitive to different segment lengths and buffer radiuses.

In [90], authors estimate PM2.5 concentration using the machine learning RF-XGBoost model. Here the authors use a random forest regressor to fill the missing aerosol optical depth (AOD). Satellite-based AOD, with several spatial and temporal resolutions generated from various sensors using different algorithms, is an extremely useful and indicative variable to assess PM2.5 exposure. The authors then adapted the XGBoost model on top of interpolated AOD values and other features such as meteorological and transport data. XGBoost model was used to calculate feature importance using all variables with potential influence on surface PM2.5 to preliminarily fit the model and calculate the contribution of each variable. The results show the importance of the AOD feature in estimating PM2.5. The

benefits of the RF-XGBoost model are the following. (1) The spatial resolution is 1 km X 1 km, showing more detailed information on the spatial distribution of PM2.5 concentrations. (2) The values of PM2.5 concentrations are more accurate compared with observations from ground stations averaged by monthly and annual means. However, the limitations are that (1) the RF-XGBoost model did not consider the representation of physics, chemistry, and transport within the atmosphere; (2) the RF-XGBoost model did not evaluate the uncertainty of PM2.5 concentrations; (3) the RF-XGBoost model is underestimating PM2.5 in high pollution days and overestimating for low values, similar to most of the PM2.5 estimation models based on machine learning.

DL Models

Authors in [13] implemented a deep learning solution to predict PM2.5 concentration in Beijing, China. Their approach is based on CNN-LSTM neural network to expand the spatiotemporal coverage. They use historical data on pollutants combined with meteorological data and concentrations of PM2.5 from nearby stations. Combining the convolutional neural network and the Long-short memory network helps extract the Spatiotemporal characteristics. This research uses data from fixed air quality monitoring stations from 12 sites. They conducted some spatial analysis and feature correlation steps to find the best feature set to train the model. The proposed approach was evaluated against different deep learning methods such as LSTM, GRU, Bi-LSTM, etc., and shows its superiority. Their work mainly focused on predicting the next PM2.5 concentration and not estimating or interpolating missing values. Although in their work, the authors used CNN-LSTM, one of the best networks to expand spatial and temporal coverage, they used only fixed stations and other features in the model.

In [56], authors introduced HazeEst, a machine learning-based metropolitan air pollution estimation from fixed and mobile sensors. This approach combines sparse fixed stations with dense mobile sensor data to estimate the air pollution surface hourly. Besides the air pollution features, they use some temporal and spatial features. They try to merge fixed and mobile data by averaging the measurements collected from mobile sensors hourly. Several regression methods were implemented, such as SVR, DTR, RFR, etc. . .

A space-time learning network was proposed in [128] denoted as a Multi-AP learning network. Multi-AP estimates pixel-wise pollution based on fixed-station measures combined with land use, traffic, and meteorology features. The authors group their features into three micro, meso, and macro views. Multi-AP simulates multiple pollutants, PM2.5, PM10, and O3, using a fully convolutional network (FCN). They define their micro-view as grid-unit traffic, land use, and other local features. Meso-view features are defined for each given grid unit by referring to generalized features among neighboring space-time units. While macro-view refers to pollution measurements from monitoring sites surrounding the study area.

Several experiments were conducted using different combinations of features that show the superiority of the Multi-AP network when compared to other approaches. The authors mentioned some questions and challenges, such as data constraints, seasonality, and model extension, which are still there.

In [17], Cassard et al. propose an air quality prediction engine to predict PM2.5 and PM10 concentrations in the United States. Hundreds of official monitoring stations and more than 4000 fixed low-cost sensors feed this engine. Authors use the road network and traffic data besides the fixed and low-cost sensors. The features built are based on the five closest official monitoring stations, the five closest low-cost sensors (using the last 16 measurements), and the road and traffic features. A convolutional layer is adapted for the low-cost sensors, and then all features are flattened and combined, then passed to a fully connected layer. Three prediction models are considered either relying only on official stations, only on low-cost sensors, or the combination that shows better results. Using official monitoring stations with high-quality measures with low-cost sensors can enhance pollution estimation, but both low-cost and official stations are fixed sites. Thus, they may lack spatial coverage.

The authors in [97] proposed a deep autoencoder model for pollution map recovery. The idea is to separate the process of pollution generation and data sampling by using an encoder, decoder, and sampling imitator. These components work together to recover the spatiotemporal pollution map. ConvLSTM structure was adapted inside the decoder based on a previous work [96]. This approach uses data from mobile sensors without utilizing other features.

Song et al. proposed a Deep-Maps approach [129] to estimate PM2.5 measures at a granularity of 1km x 1km x 1h. The mobile sensor network can offer high resolution and dense coverage. Thus combining mobile sensors' data with fixed stations' data helps expand spatial coverage. Deep-Maps is a machine learning framework that adapts a gradient-boosting decision tree on top of local features such as land use and meteorological data; neighboring features that capture the spatiotemporal correlations among urban features; and the macro features denote pollution measurements from sites outside the study area.

In [23], authors introduce a generic neural attention model, named ADAIN (Attentional Deep Air quality Inference Network), for spatially fine-grained urban air quality inference. They adapted neural networks to model heterogenous data in a unified way and learned complex feature interactions. The authors use air quality data, road network data, meteorological data, point-of-interest, etc. Both monitoring station records and urban data are leveraged, and essential features correlated with air quality are extracted.

[82] presents a Spatiotemporal Deep Learning Model for citywide air pollution interpolation and prediction. Authors propose using the Convolutional Long Short-Term Memory (ConvLSTM) model [122], a combination of Convolutional Neural Networks and Long Short-Term Memory, which automatically manipulates both

the spatial and temporal features of the data. The core idea is to transform the air pollution data into sequences of images that leverage the ConvLSTM model to simultaneously interpolate and predict air quality for the entire city. In this conducted work, authors use data collected between 2015 and 2017, including hourly air pollution data from 39 monitoring stations, hourly meteorological data from 28 observation stations, hourly traffic volume data for about 145 main roads, and hourly average driving speed in more than 4000-speed surveying points in Seoul, Korea. Urban air pollution has both spatial and temporal characteristics. Therefore, to efficiently predict air pollution anywhere (interpolation) and at any time (forecasting), the authors proposed ConvLSTM. To apply the ConvLSTM model, they use gray-scale images as 2D input tensors with MxN dimension. The input tensors are air pollution values, the combination of air pollution, and other influential factors at the exact location. ConvLSTM model achieved the best RMSE among different baselines.

CNN-LSTM is also proposed in [149]. The authors proposed a hybrid CNN-LSTM framework named Deep-Air for fine-grained air pollution estimation and forecast in metropolitan cities. It provides fine-grained citywide air pollution estimation and station-wide forecast. It exploits domain-specific features (including Air Pollution, Weather, Urban Morphology, Transport, and Time-sensitive features) with a hybrid CNN-LSTM structure to capture the spatiotemporal features and 1 $\times$ 1 convolution layers to enhance the learning of temporal and spatial interaction. Hybrid deep learning models that combine CNN and LSTM have been extensively used for spatial-temporal data [137]. However, the authors proposed a framework that utilizes a spatial model to generate a high-level representation at each time step and learns the temporal correlation of these representations through a temporal model. Two deep-learning models were developed separately for fine-grained air pollution estimation and air pollution forecast. Historical air pollutant data are only available at air pollution monitoring stations for fine-grained air pollution estimation. In contrast, the estimation model aims to estimate the air pollutant concentrations at all locations, including those without air pollution monitoring stations. AirRes component consisted of four residual units, each with two 3 $\times$ 3 convolution layers with batch normalization and ReLU activation function. A 1 $\times$ 1 convolution layer was added between the two residual units. For the LSTM component, the length of the model input (past hourly observations) was set to 48 (hours), and the number of LSTM layers was assigned to one or two. The model was compared to five baseline models on the Hong Kong and Beijing datasets. Results show that the Deep-AIR framework has performed best in fine-grained air pollution estimation and forecast.

2.4.4 . Discussion

The pollution estimation problem has attracted the interest of researchers over several years. It has been intensively studied in the literature from different angles and at various scales. Initially, the work was focused on estimating pollution mea-

sures depending on fixed station data, but also those approaches are applied to estimate pollution from mobile sensor data. In addition, some researchers are combining fixed and mobile sensor data measures to expand the spatial and temporal coverage. Our review covers a wide range of pollution estimation techniques such as Geostatistics, LUR, and machine and deep learning models. The approaches were applied to data collected from fixed and/or mobile sensors, along with using other features related to air pollution, such as meteorological and traffic data.

All the existing approaches in the literature have conducted targeted data collection campaigns. The collected data was directed on either specific roads or all the collected data in outdoor places. However, we don't have such targeted campaigns in the context of mobile crowd sensing/ mobile participatory monitoring (MCS/MPM). Participants will collect data freely while conducting their real-life activities. Hence the contribution of such mobile sensory data to the enrichment of outdoor places on the map is low. Then the challenge here is to enrich the fixed stations' data with the mobile participatory monitoring data to better estimate air pollution.

2.5 . Tweets' Location Prediction

Tweets' location prediction has been an active area of research within the last decade. This section details most of the existing literature work related to the second research question in our thesis **R2**. Social media platforms have experienced a massive boost over the last decade. Users can connect with others through these platforms, develop online friendships, and share real-life events with them. Twitter is one of the most popular social media platforms. It allows for establishing non-mutual friendships. Besides the textual content, tweets contain meta-data describing the creation time, mentions, attachments, location if exits, and many other attributes.

Knowing the exact location of tweets can help in monitoring the real world. Many applications can benefit from geotagged text information; a few to list natural disaster and crime detection [133, 78], health care management [158], marketing recommendation systems [11]. Event detection systems [144, 145, 138], etc. However, unfortunately, only 1 to 3 % of tweets contain geotagging information [126], which makes analysis a hard task in the absence of such data.

Considering the high-importance insights one can extract from geotagged social media data, we are witnessing an increasing interest from both academic and industrial parties in the problem of tweet location prediction. Many existing strategies have been investigated in the study on location prediction from the tweet and social media content, and we will discuss some.

In [152], authors introduced an overall picture of the different families of location predictions performed on Twitter data. In their work, they have illustrated the different types of inference that can take place depending on the tweet content,

the Twitter network, and the tweet context and meta-data. They grouped the prediction types into three groups:

- **Home location** refers to Twitter users' long-term residential addresses. Home locations may be represented at different levels of granularity. Generally, there are three categories of home location granularity: (Administrative region, geographic grids, and geographical coordinates). For this type, the ground truth home locations may be collected from users' self-declared profiles, or an aggregation of the attached geotags with users' tweets is considered their ground truth home locations.

- **Tweet location** is where a tweet is posted. Tweet locations are generally based on geotags of tweets. Due to the original views of tweet locations, point-of-interests (POIs in short) or coordinates are broadly adopted as representations of tweet locations instead of administrative regions or grids.

- **Mentioned location** refers to the names of some places in tweet contents. It facilitates a better understanding of tweet contents and benefits applications like location recommendation and disaster & disease management.

The authors stated that all the mentioned location prediction types could use the Twitter network, Twitter content, Twitter context, or different combinations as input features. Moreover, the authors defined the evaluation metrics that could be categorized as distance-based or token-based. In the distance-based, locations are represented by their geographical coordinates. Token-based metrics treat locations as discrete symbols, e.g., country, city, grid, POI. Usually, for distance-based metrics, a distance error is defined, and the metrics are the mean, median, and mean squared error distance. While for the token-based evaluation, precision, recall, accuracy, and ranking-based accuracy are used.

We have classified the reviewed work into two categories: coarse and fine granularity based on the specified output, such as city, neighborhood, or the reported distance error between the real and anticipated locations.

2.5.1 . Coarse Granularity

In [74], the authors proposed an approach relying on a Language Model (LM) built by calculating term occurrence probabilities from processing a massive amount of geotagged items of a training set. The authors present a geotagging approach for estimating the locations alluded to by text annotations based on refined language models learned from massive corpora of social media annotations. The proposed method relies on an LM built by calculating term occurrence probabilities from processing an enormous amount of geotagged items of a training set. The LM is refined through feature selection and weighting. Terms are ranked and filtered based on accuracy, spatial entropy, and locality. The location estimation system employs two more steps (multiple grids and similarity search) to achieve accurate

location estimation. This model has shown promising results but depends on the trained language model. Authors have trained the language model on Flickr Images, and when they applied it to tweets, it did not show the same results. This approach is dependent on the training dataset, and it cannot be used for any piece of text.

Chi et al. [24] proposed an algorithm to predict the location of Twitter users and tweets. The algorithm utilizes a multinomial Naive Bayes classifier, using a textual feature set that includes a combination of location indicative words (LIW), city/country names, #hashtags, and mentions, which are automatically learned from an extensive collection of Twitter data. Moreover, the authors study the effects of various feature sets. They applied a pre-processing step and a feature selection step. The authors have tried different combinations of features to find the best variety. Their results show they still have a high distance error, while monitoring and event detection applications require more precise location prediction.

In [103], the author proposed a BiLSTM neural regression model that can identify the linguistic intricacies of a tweet to predict the location. Such linguistic attributes can provide a regional approximation of tweet origins. A double regression approach is adopted to identify the latitude and longitude separately for a given text to determine the location of a given tweet. This paper presents a method to predict a user's location using a neural model trained solely on the tweets' text content without external knowledge sources. This allows the model to generalize more easily to new domains and languages. For this purpose, the authors use a Bi-LSTM. They used TF-IDF weighting to focus on highly relevant tokens in the text and used the FastText model to calculate embedding. Then two bidirectional LSTM layers are used, one for latitude training and one for longitude. This work is considered a regression as the aim is to assign the latitude and longitude of the text, and the results reported show that we still have a noticeable distance error.

Izbicki et al. [62] proposed a method for geolocating tweets in any language. This work introduces the Unicode Convolutional Neural Network (UnicodeCNN) for analyzing text written in any language. UnicodeCNN does not require the language to be known in advance, allows the language to change mid-sentence arbitrarily, and is robust to the misspellings and grammatical mistakes commonly found in social media. The Unicode CNN generates its features in four stages: a character encoder, convolutional layers, a language estimator, and a feature mixing layer. UnicodeCNN generates features directly from the Unicode characters in the input text. The authors use a character encoder that converts the input text into a binary matrix. Those encodings are passed as the input to a series of six temporal convolutional layers. After passing the information to the convolutional layers, the softmax layer predicts the tweet's country of origin. A Mixture of von Mises-Fisher distributions is used to predict the GPS coordinates. This approach reported a high accuracy in predicting the location at the country and city levels, but it fails to predict with high precision at finer granularities.

In [37], authors utilized millions of Twitter posts and end-users' domain expertise to build deep neural network models using natural language processing (NLP) techniques to predict the geolocation of non-geo-tagged Tweet posts. The authors aim to predict Twitter posts' geolocation at a granular level, such as neighborhood, zip code, and longitude with latitude values. They collaborate with urban planning analysts to assess our modeling task's success through an iterative modeling pipeline. They also incorporate their valuable domain expertise in modeling decisions to refine model performance. Their contribution was to provide a novel text modeling approach informed with feedback to predict the geolocation information. Different levels of granularity are covered in this work. They initially considered their problem a regression problem, and they utilized a 1-dimensional convolution-based CNN for the sentence classification model to predict the longitude and latitude. However, depending on the experts' feedback, they found that the models predicting precise locations show less than satisfactory results. Hence, they retrieve the zip code of the predictions using reverse geocoding. Moreover, they trained a model to predict the neighborhood following the same pipeline as the previous model. The reported accuracy for predicting the coordinates was very low. However, the accuracy improved at the neighborhood and zip code granularities. The measured accuracy is at 30 miles, considered a wide range, while the aim is to minimize the distance error between the actual and predicted locations.

In [98], authors aim to predict the geolocation of real-time tweets at the city level collected for 30 days by using a combination of convolutional neural network and a bidirectional long short-term memory by extracting features within the tweets and features associated with the tweets. They have proposed a model to solve the problem of geolocation prediction of Tweets by combining two neural networks, CNN and BiLSTM. The intention of combining these two deep learning techniques is to benefit from the advantages of CNN and BiLSTM architecture. On the one hand, CNN can utilize its multilayer perceptron structure to extract high-level features in the text and can absorb complex and non-linear mapping relationships from the text. While LSTMs generally take advantage of their ability to capture long-term dependencies between the text. They chose BiLSTM as it is known to solve the problem of gradient disappearance or explosion, which may occur in RNN. Moreover, BiLSTM provides additional training by scanning the data two times, from left to right and right to left, thus, extracting the semantics of a word in the context of the information preceding and succeeding it. Then location-specific features can be extracted easily by aggregating these two deep-learning techniques. The three attributes used to perform the prediction task are the screen name, tweet text, and user profile location. The output of the two models is then combined in a flattened layer, and max pooling is applied before passing it to a fully connected layer. The results show that the accuracy of the proposed approach outperformed the basslines' accuracy.

Kinsella et al. created a language model of locations in [72]. They model

locations at varying levels of granularity, from zip code to country level. For each location, they estimate a distribution of terms associated with the location. Then they estimate the probability that a tweet was issued from a given location by sampling from the term distribution for that location. Then rank the locations by the probability that they "generated" the tweet. They use Bayesian inversion and Kullback-Leibler (KL) divergence methods for ranking locations. The results show acceptable results, especially for the zip code prediction.

In [94], the authors proposed a Hidden-Markov-based model to integrate tweet contents and user movements for geotagging. They propose a framework integrating content and user movement for better geotagging performance. Raising the following contributions: (1) Introduce movement patterns of users into geotagging; (2) Propose a Hidden Markov Model to integrate language model and user movements for geotagging. The proposed model considers a user's home location to better represent the transition probability for users of different home cities. The states of the Hidden Markov Model are the city-level locations of users, and the state observations are tweets. The state (city) is not directly visible, but the observation (tweet) is visible. Therefore, the sequence of tweets generated gives some information about the sequence of cities. Compared with related works, the improvement in error distances demonstrates that even for the incorrectly estimated cases, the proposed model can locate the tweet with a closer city to the actual location of the tweet due to the benefits from the usage of patterns of user movements.

Galal and Elkorany [45] study the relationships between geolocation information published by users at different times. This geolocation information was used to model users' interests and behavior to enhance the prediction of user locations. Authors extract semantic features such as topics of interest and location categories from this information to overcome the sparsity of data. The proposed framework aims to predict the category of the user's current location. To enhance the prediction of the user's current location, they propose a context-based model that integrates content-based and location-based attributes to investigate the relationship between the published posts and the user's check-in behavior and their variation over time. The proposed framework can be divided into two major components: the first is responsible for identifying and modeling users' context, such as locations and topics. In contrast, the second component is the prediction engine. They used categories from Foursquare to identify and model users' context and reduced them to 23 place categories. At the same time, prediction utilized both tweets' posting time and user topics as features for the prediction model. They used Naïve Bayes (NB) and k-Nearest Neighbor (kNN) as classical models. The reported results show that their proposed framework significantly outperformed the baseline prediction method on all classification problems when increasing the number of predicted location categories.

2.5.2 . Fine Granularity

In [89], authors proposed a Convolutional Neural Network (CNN) architecture for geotagging tweets to landmarks based on the text in tweets and other meta information, such as posting time and source. The main contribution of this work includes proposing an algorithm for geotagging tweets to landmarks. This algorithm requires tweet text representing word embedding, source, creation time, and user location. The intuition behind posting time and source is that given specific landmarks, people are more likely to post tweets at particular timings and using specific sources. The authors utilize CNN as various works have effectively utilized CNN for text classification tasks relating to sentiment analysis. The proposed model took a tweet of n words and represented it as an embedding vector. Other features like posting time and source are encoded as one-hot. The CNN network maps the input into pre-defined lists of POIs. The proposed model was evaluated against various baselines and reported promising results. However, this approach can work only when we have landmarks. Thus, if we do not have a landmark in some places, we cannot geolocate the tweet.

Ozdikis et al. proposed a kernel density-based location prediction method for tweets based on the geographical probability distribution of their terms over a region in [107]. Probabilities are calculated using Kernel Density Estimation (KDE). Here the bandwidth of the kernel function for each term is determined separately according to the location indicativeness of the term. Thus, combining its terms' probability distributions predicts the tweet's location. The authors' main contributions are (1) to investigate the use of kernel density estimators to analyze geographical distributions of terms in tweets and propose a fine-grained location prediction method based on integrated densities of terms. (2) Relying on statistical techniques to obtain term-specific KDE settings based on location indicativeness of the terms without requiring parameter tuning. (3) Presenting a weighing method for combining probability distributions to obtain higher prediction accuracies. This approach shows an improvement in accuracy at 5 km but fails to predict with the same accuracy at finer granularities.

An end-to-end neural network to predict the geolocation of a tweet was proposed in [81]. The proposed model is language-independent and requires six features the tweet text and other meta-data information. The proposed network can automatically learn different regions' location-indicative words and activity patterns. The model contains a text network to learn text representation, three RBF networks for (creation time, UTC offset, and account creation time), a time-zone embedding, and a Convolutional network for user-defined location. The tensors of different networks are concatenated and passed to the output layer. They used a character-level recurrent convolutional network for the text network for the tweet message. The reported results show that combining all features has better accuracy than using each alone. Conversely, in [127], the authors have followed the same architecture of the proposed model but replaced the character-level recurrent network with the *Word2Vec* embedding. Although **DeepGeo2** has increased the

accuracy of location predictions, we still need predictions at a higher accuracy for monitoring applications. Moreover, some of the presented features in this paper are no more available on Twitter, such as the time zone and UTC offset.

In EDGE [58], the authors cast the geolocation problem as a neutral network optimization problem by learning probabilistic generative models. This work presents a tweet geolocation prediction framework, EDGE (Entity-Diffusion Gaussian Ensemble), which delivers accurate and highly interpretable predictions without requiring additional contextual information, such as user profile and location history. EDGE consists of three primary parts entity embedding extraction, attention aggregation, and mixture distribution learning. EDGE has two distinctive features: (1) the inference builds upon mining the correlation between non-geo-indicative entities and geo-indicative entities by diffusing their semantic embeddings over the constructed graph neural network, and (2) each prediction result is returned as a Gaussian mixture rather than specific geographical coordinates. In their approach, authors propose entity2vec to extract embedding for named entities appearing in tweets instead of treating them as a composition of independent words. Moreover, they developed an entity diffusion mechanism to capture the correlation between non-geo-indicative and geo-indicative entities. Also, an attention mechanism is designed to weigh the importance of the entities that co-occur in the same tweet while extracting the spatially smoothed embedding for each tweet. Finally, EDGE is trained end-to-end to maximize the likelihood that geotagged tweets are located in their associated locations.

Gonzalez et al. [47, 109] proposed a majority voting method that compares the similarity between non-geotagged and geotagged tweets. In this work, they adopted a weighted majority voting algorithm for the problem of fine-grained geolocalisation of tweets. They estimated the geographical location of a given non-geo-tagged tweet by collecting the geolocation votes of the geotagged tweets most similar regarding their contents to that tweet. The weights of the votes were calculated based on the source's credibility. Their approach consists of three steps. Creating a grid to divide the geographical area into squares. Obtaining the Top-N content based on similar geotagged tweets to non-geo-tagged using different retrieval models. Combining evidence from the Top-N tweets by adopting a weighted majority voting algorithm. In their approach, they extract the credibility from the tweet's user to use it in the majority voting. This model shows acceptable results when compared to other baselines.

Authors in [27] proposed a fine-grained tweet geolocation ranking approach to link tweets to venues. The approach depends on users' location history data and tweet posting time, as these features serve as additional contextual information for geolocation. In their previous work [84, 86], authors link tweets to the specific venues from which they are posted, e.g., a restaurant. However, in this work, they cast fine-grained geolocation as a ranking problem. They rank venues such that high-ranking venues are more likely to be posting venues. They use users'

history and tweeting time since the problem is challenging as tweets are short and may not contain any location names or location-indicative words. Posting time accounts for venue popularity at the time of day. Kernel density estimation is used to estimate the probability of a venue at a given time. For Location history, they recap that users are spatially focused in that he is more likely to visit venues spatially near any of their previously visited venues. They also use the frequency of words in a vocabulary to model the tweet content. Finally, they maximize the Mean Reciprocal Rank metric to learn the ranking. The proposed model can achieve a significant improvement in ranking accuracy over baselines.

In [28], the authors have extended the same ranking approach by adding location, user, and peer signals. The absence of information on the user's activity regions or hangout places makes fine-grained geolocation more challenging. Thus, they propose several models that leverage three types of signals from locations, users, and peers. They exploit location signals from words that are indicative of locations. A location-indicative weighting scheme is proposed to capture this. Moreover, they exploit user signals from each user's content history to enrich the limited content of users' tweets targeted for geolocation. The intuition is that the user's other tweets may have been from the test venue or related venues, thus providing informative words. In addition, they exploit the signals from peer users with similar content history and, therefore, potentially similar visitation behavior to the test tweets' users. They tried different combinations in their experiments, and the results reported that the best model combined all three aspects.

In [84], authors present a systematic approach to increasing the number of geotagged tweets by predicting the fine-grained location of each tweet using a multi-source and multi-model-based inference framework. They build probabilistic models for locations using unstructured short messages tightly coupled with their semantic locations. To achieve the tight coupling between text and location, they used Foursquare - a popular location-centric social network, as a source for building these probabilistic models. Each tweet is considered by its words in the textual content. They have extracted the point of interest (POIs) from Foursquare to build more accurate and dependable probabilistic models for locations. They propose a 3-step technique (Filtering-Ranking-Validating) for predicting the location of the tweets at fine-grained resolution. In the filtering step, they develop a set of filters that can remove those location-neutral tweets which may not be related to any location. This effort enables filtering out as many location-neutral tweets as possible to minimize the noise level and improve the accuracy of the location prediction model. In the ranking step, candidate locations for each tweet are determined using one of the three ranking techniques: standard machine learning approaches, naive Bayes model, or TFIDF value. Once the top-ranked location is assigned to the tweet, in the validating step, they utilize a classification-based prediction validation method to accurately predict the location where the tweet was written. They build the classification model based on tweets containing geotag

information. The proposed framework shows promising results when compared to other baselines.

Cao et al. [16] presented a method for providing a ranked list of geolocated venues for a non-geotagged tweet, simultaneously indicating the venue name and the geolocation at a very fine-grained granularity. In the proposed method for Venue Inference for Tweets (VIT), they construct a heterogeneous social network to analyze the embedded social relations and leverage available but limited geographic data to estimate the geolocated venue of tweets. Authors attempt to infer the location of a tweet as a geolocated venue, exploring content-based features and features extracted from a user's friendship network to infer both location and venue name. However, noisy geotags and text ambiguity are two challenging problems. The authors proposed a single-trained model for ranking. The authors consider analyzing the social activities embedded in their constructed heterogeneous social network and leveraging available but limited geographic data to infer tweet venues. They exploit the social network in different paths. The Ego Path directly relates a user's tweets to venues. The Friend Path connects a user's tweets to venues through their friends. Interest Path expands the relationship between tweets and venues through Foursquare categories. Finally, the Text Path models the content words tweeted about venues, unlike conventional approaches focusing on text processing for content analysis. The SVM classifier was trained to classify whether the link between a tweet and a venue is positive or negative. The experiments show that the proposed model with all features combined outperforms any single feature type and can achieve very good performance.

Authors in [61] proposed a density estimation method for geolocation. They propose the convolutional mixture density network (CMDN), which uses text data to estimate the mixture model parameters. The authors utilize a density-based approach as it can accommodate the representation of multiple output data, whereas the regression-based approach cannot. The estimated density appends the estimation reliability for each tweet as the likelihood value. The estimated density provides a high likelihood for reliable estimation and vice versa. The proposed CMDN extracts valuable features using a convolutional neural network architecture and converts these features to mixture density parameters. Convolutional Mixture Density Network (CMDN) is the extension of the Mixture Density Network. In contrast to the regression approach that directly represents the output values, CMDN can accommodate more complex information as the probability distribution. The CMDN estimates the parameters of the Gaussian mixture model. Overall results show that the proposed model CMDN provides the lowest median error distance and acceptable results compared to other approaches.

Dredze et al. [35] studied the cyclical temporal effects on Twitter geolocation accuracy. In this work, authors focus on the temporal effects stating that time is relevant when geolocation is needed for a single tweet. Usually, tweets written in the morning might be in different locations (at home), while those written

during the day might be (at work). This information is often ignored but can provide important clues about a tweet's location. The authors utilize a supervised learning approach, training a multi-class classifier to identify the city of a tweet. All the input features of this model are extracted from a single tweet. After preprocessing and tokenizing the tweet, they extract unigrams and bigrams from the text. Profile locations are extracted from users' profiles. They include in their features also the Time-zone and UTC offset and the creation time of the tweet. For training, the used vowpal wabbit [6] is a linear classifier trained using stochastic gradient descent with adaptive, individual learning rates. After performing their experiments, they have the following findings: (1) Geolocation accuracy is cyclical, varying significantly with time. (2) While access to massive training data improves accuracy, these effects are largely lost when models are deployed on new tweets due to new users and duplicate tweets. (3) Periodically updating geolocation models, even with data available from the free Twitter API, can largely supplant massive training datasets.

2.5.3 . Discussion

The problem of text and tweet location prediction has received much attention from researchers over the years. Various approaches and methods have been proposed to address this issue. However, most of these methods have succeeded in predicting tweets' city and country levels while failing to predict location at finer granularities with high accuracy. This is a significant limitation as finer-grained location information can be valuable in many applications, including disaster management, event detection, and public health surveillance.

Existing text location prediction methods do not consider the relation between non-geotagged tweets and other geotagged ones in the same spatial area. This is a missed opportunity since non-geotagged tweets can complement geotagged tweets, providing additional information about the location.

A text location approach that minimizes the distance error between real and predicted locations is needed to overcome these limitations. This requires considering tweets' spatial and temporal correlation in a specific region. Tweet location prediction is still an open research problem. To benefit from tweets in monitoring and event detection systems, we need a precise location approach that can provide accurate, fine-grained location information.

Chapter 3 - Micro-environment Recognition

Contents

3.1 . Introduction

This chapter presents a joint effort between myself and my colleague Hafsa, as detailed in our joint publication titled "Learning the micro-environment from rich trajectories in the context of mobile crowd sensing: Application to air quality monitoring," published in GeoInformatica Journal [41] which was extended publication to our approach [2]. The work encapsulated in this chapter represents a collective exploration into Micro-environment recognition in the context of environmental crowd-sensing. I will begin by acknowledging the foundational aspects of our joint venture before transitioning to a more focused examination of my individual contributions. Through this chapter, I aim to elucidate and build upon the findings presented in our collaborative work, shedding light on my distinct contributions to the broader research landscape, mainly derived from [2], and the extension of this work is discussed in details in Hafsa's thesis [39].

Our primary goal in this chapter is to focus on the first main objective in this thesis, which is automating the micro-environment detection. By focusing our efforts on this goal, we want to investigate and clarify the multidimensional character of the stated research question **R1**. We endeavor to deliver complete and insightful answers to **R1** by careful analysis, rigorous study, and the use of relevant approaches, therefore significantly contributing to the current body of knowledge in our field. Hence, in this chapter, we present our first contribution **C1**, i.e., proposing an end-to-end micro-environment recognition pipeline handling steps of data collection, pre-processing, learning through multi-view approach, and visualization.

The rest of this chapter is organized as follows: The following section presents background about MCS and activity recognition, what the problems are, and we break down our contribution **C1** into sub-contributions within this work. Section 3.2 states the problem of micro-environment recognition formally. Then, we present the detailed approach of multi-view learning in section 3.3. Section 3.4 presents the following methodology and the detailed pipeline implementation with the explanation. Section 3.5 shows the findings of experiments conducted on a real-world dataset, and section 3.6 shows the ability of model generalization. In 3.7, we discuss the results and show the effectiveness of our model. Finally, the last section 3.8 summarizes our work and draws our perspectives.

3.1.1 . Background

Nowadays, the Internet of Things (IoT) heavily relies on advanced sensor technologies to establish a connection between the physical realm and information systems. Notably, with the widespread adoption of GPS, various mobile sensors gather extensive data from the surrounding environment and human activities. This data is typically presented as geo-referenced time series, essentially trajectories enriched with multiple measures. A novel paradigm, Mobile Crowd Sensing (MCS) [50], has emerged, empowering volunteers to contribute data acquired by

their personal, sensor-enhanced mobile devices. An illustrative example of this paradigm is Polluscope[1], a French project implemented in Île-de-France (i.e., the Paris region). Polluscope serves as a representative case study for MCS, striving to continually gain insights into individual exposure to pollution in diverse settings (both indoor and outdoor). This endeavor enriches the conventional monitoring system with data collected by the crowd. Participants, contributing voluntarily, utilize sensor kits and mobile devices to collect and transmit air quality (AQ) measurements, along with corresponding GPS coordinates. This enables participants to gain personalized insights into their exposure to pollution indoors or outdoors and with heightened granularity along their trajectories. Consequently, it facilitates the capture of local variations and pollution concentrations, contingent on participants' locations, commonly referred to as *micro-environments*.

It is worth mentioning that the micro-environment significantly influences air quality, and individual exposure to pollution is similarly affected. Consequently, a substantial interest is in incorporating micro-environment awareness into exposure analysis. Knowing such information is essential to interpreting pollution levels or concentrations. Ignoring the micro-environment would make the data collection useless precisely because of the influence of the micro-environment. However, the micro-environment annotation is the most difficult information to collect in a real-life application. Users are not supposed to annotate their changes in real life. Even though, in data collection campaigns, only very few participants thoroughly annotate their micro-environment. Therefore, there is considerable interest in alleviating the burden on participants by implementing automatic methods for micro-environment detection.

3.1.2 . Problem Statement & Related Works

Human activity recognition (HAR) has garnered significant attention from the scientific community in recent years, encompassing activities in daily human life and human mobility. In Section 2.3, we delved into relevant literature on HAR. Micro-environment recognition emerges as a subset within the broader HAR context [148, 93, 25, 154].

However, with the advent of Mobile Crowd Sensing (MCS), the focus has shifted towards integrating the characteristics of both data sources and incorporating them into the activity recognition framework. This chapter addresses the semantic enrichment of MCS data derived from multi-sensors and GPS tracks. The goal is to augment contextual information in the data using multivariate time series classification and stop & move detection techniques within trajectory analyses.

The challenge of automatically annotating Mobile Crowd Sensing (MCS) data can be framed as an activity recognition problem, leveraging rich trajectory data collected from diverse sensors. Activity recognition has been extensively explored in various research studies, with Yu Zheng's survey [153] offering a systematic

[1] http://polluscope.uvsq.fr

overview of trajectory mining methods. Despite the plethora of trajectory data mining approaches, a comprehensive method that integrates multiple sensor data beyond GPS remains absent. Conversely, incorporating various sensory data implies the potential use of multivariate time series classification (MTSC) for activity recognition. While MTSC has demonstrated exceptional performance in specific domains [117], its efficacy with heterogeneous sensors, such as environmental data, is uncertain. Firstly, including diverse data types may introduce missing data issues, particularly when specific sensors cease operation. Therefore, there is a demand for a model capable of characterizing micro-environments even in missing dimensions. Secondly, the extent to which environmental data can accurately define micro-environments remains an unexplored aspect requiring further investigation.

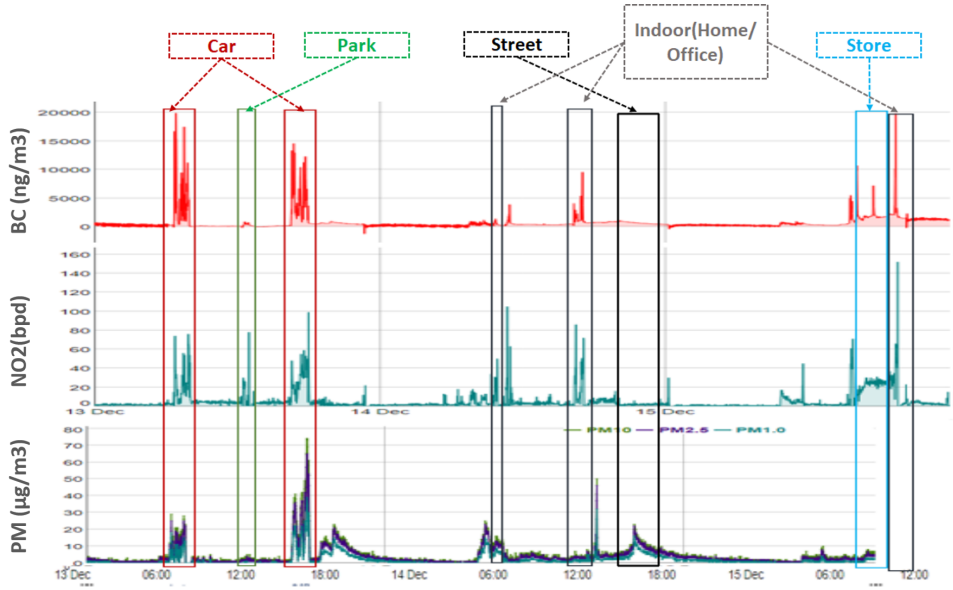


Figure 3.1: Inter-sensor and micro-environment correlations.

Indeed, upon visually examining the data, we discerned a discernible pattern within micro-environments. Furthermore, an inter-sensor correlation becomes apparent, particularly concerning the micro-environment. In Figure 3.1, the progression of three dimensions (Black Carbon (BC), NO2, and Particulate Matters (PM)) is illustrated alongside micro-environment identification. The figure illustrates that BC and NO2 exhibit congruent shapes and statistical characteristics within the micro-environment labeled "car." Notably, a correlation exists among the three dimensions throughout the timeline, implying that fluctuations in one dimension are mirrored by the other two.

3.1.3 . Proposition and Contributions

The idea we promote is to utilize a wisely chosen annotated dataset to train a model on the acquired rich trajectories (composed of environmental and mobility

dimensions) as predictors of the micro-environment. We hypothesize that the multivariate time series collected by the MCS campaigns not only depends on the micro-environment but could be a proxy of it. The question that arises now is how to automate the micro-environment recognition within the presence of complex, heterogeneous, and missing dimensions in multivariate time series data.

In this chapter, we evaluate different approaches and provide a framework dedicated to the preparation, the application, and the comparison of different machine learning algorithms. Precisely, we make the following contributions derived from **C1** contribution:

- Identifying the problem of micro-environment recognition in the MCS context.
- Demonstrating that AQ determines the type of micro-environment.
- Proposing an ML approach based on multi-view learning for the recognition of micro-environment.
- Conducting extensive experiments in a real scenario setting and comparing with baselines, which shows the effectiveness of our proposed approaches.

3.2 . Problem Formalization

The following questions and answers establish a formal definition of our problem.

3.2.1 . What are rich trajectories ?

Rich trajectories encompass more than just GPS coordinates; they are trajectories enriched with a spectrum of continuous measures or attributes. These additional measures extend beyond location data and can include variables like air quality measurements, temperature, humidity, and other environmental parameters. These trajectories are typically gathered through mobile sensors and offer intricate details about an individual's movements and the environmental conditions surrounding them. For instance, in the context of air quality, a rich trajectory (semantic trajectory) might include not only the geographical path of an individual but also continuous data points related to air pollution levels at different locations and times. This comprehensive data facilitates a nuanced understanding of how the environment influences an individual's exposure to various factors. To elaborate further on this concept, we begin with the definition of time series.

Definition 3.2.1. (Univariate Time Series). A univariate time series is a sequence $U = [(t_1, v_1), ..., (t_l, v_l)]$ where l is the length of U and for $i = 1...l$, $t_i \in T$ is a timestamp from a time domain T and $v_i \in D$ is a scalar value of a domain D.

Univariate time series refers to a type of time series data that consists of a single sequence of observations recorded over time. In other words, it is a set of data where each observation is a single value, and the data is collected at regular intervals, such as daily, weekly, monthly, or yearly

Example 3.2.1. Environmental sensor measurements such as temperature constitute a univariate time series.

Definition 3.2.2. (Multivariate Time Series). A multivariate time series MV is defined as $MV = (U_1, U_2, ..., U_i, ..., U_n)$ where U_i is a univariate time series for dimension D_i, and $i = 1, ..., n$.

Multivariate time series refer to time series data that involve more than one variable changing over time. In other words, it's a collection of time series data where each observation consists of multiple measurements taken at the same time.

Example 3.2.2. Environmental sensor measurements such as temperature, humidity and NO2 constitute a 3-Dimensional time series.

Definition 3.2.3. (Trajectory). In general, a trajectory refers to the path that an object or a system takes through time and space. In the context of data analysis and machine learning, a trajectory typically refers to a sequence of observations or measurements of a system or process over time.

In our work we define a trajectory T as a multivariate time series with two or three dimensions for the spatial position.

Example 3.2.3. A multivariate time series with latitude and longitude as dimensions represent a trajectory.

Definition 3.2.4. (Rich Trajectory). A rich trajectory RT is defined as a multivariate time series where a subset of the dimensions D_i where $i \in [1, ..., n]$ constitutes a spatial position, plus additional non-spatial information.

Example 3.2.4. A GPS trajectory data of a moving object associated with environmental sensor measures such as temperature, humidity and NO2 is a typical example of a rich trajectory.

3.2.2 . What is micro-environment recognition ?

Within the realm of Mobile Crowd Sensing (MCS), micro-environment recognition involves the identification and comprehension of the physical features and attributes in the immediate vicinity of a mobile device or a cluster of devices. This recognition can be accomplished through diverse sensing techniques, including the utilization of built-in sensors on mobile devices, gathering data from external sensors in the environment. Following data collection, processing, and analysis, often

employing machine learning or other data analysis techniques, valuable insights about the micro-environment can be extracted.

First, we define a trajectory segmentation, then we introduce the annotated version of rich trajectories before defining the target problem of micro-environment recognition learning.

Definition 3.2.5. (Rich Trajectory Segment). A rich trajectory segment RTS is defined as a sub-sequence of contiguous vectors of RT between j and k $(1 \leq j \leq k \leq l)$. So, $RTS = RT(j,k) = (U'_1, U'_2, ..., U'_i, ..., U'_n)$ where $U'_i = [(t_{ij}, v_{ij}), ..., (t_{lk}, v_{lk})]$, and $\forall 1 \leq i \leq n$.

Example 3.2.5. A one hour trajectory constitutes a rich trajectory segment of rich trajectory data.

Definition 3.2.6. (Trajectory Segmentation). Given a trajectory or a rich trajectory as input, trajectory segmentation is a process that splits it into non overlapping trajectory segments.

Example 3.2.6. Splitting trajectory data of a moving object into hourly segments represent a one form of trajectory segmentation.

An annotated rich trajectory is defined as a sequence of trajectory segments along with annotations that belong to a predefined list of categories. Formally:

Definition 3.2.7. (Annotated Rich Trajectory). An annotated rich trajectory ART is defined as a sequence of couples $ART = [(RT(1, i_1), a_1), (RT(i_1, i_2), a_2), ..., (RT(i_j, i_{j+1}), a_{j+1}), ..., (RT(i_p, l), a_{p+1})]$, where $RT(i_j, i_{j+1})$ are rich trajectory segments RTS between j and $j+1$, $a_k \in A$, and A is a discrete domain.

Example 3.2.7. Rich trajectory segments enriched with contextual information such as the whereabouts of a moving object represent an annotated rich trajectory.

In this work, annotations describe the micro-environment of the participant. In this work, micro-environments can either be an indoor space (e.g. home, office, restaurant, etc.), outdoor space (e.g. street, park, etc.) or a transportation mode (e.g. metro, bus, car, etc.). The micro-environment recognition question relates to the problem of segmenting data and assigning a label to each segment by combining every available data.

Definition 3.2.8. (Micro-environment Recognition). Given a rich trajectory RT as input, micro-environment recognition is a process that outputs the corresponding annotated rich trajectory ART.

Definition 3.2.9. (Micro-environment Recognition Learning). Given a set of annotated rich trajectories, train a model where the rich trajectory segments are the predictors, and the annotations constitute the class labels.

Why is this information of the micro-environment important ? The annotation details associated with a mobile object enhance the comprehension of a rich trajectory on a semantic level, and the utility of this information varies according to the application scenario. For instance, when examining people's trajectories, semantic information proves valuable for pinpointing frequently visited locations by the mobile object, thereby facilitating trip recommendations [156]. In the context of recognizing daily human activities through wearable sensors, numerous application domains exist, encompassing pervasive healthcare [147] [65] and monitoring athletic endeavors [75]. In these applications, annotations detailing daily activities, such as walking, standing up, or raising a hand, play a pivotal role in providing feedback relevant to the specific application scenario.

In our context, the specifics of annotations, referring to micro-environments, play a crucial role in offering insights into personal exposure to pollution. This correlation is directly tied to individuals' habits and the locations where they spend their time. For instance, if a person is highly exposed in their home during cooking time without much room ventilation, it would be time for them to revisit their habits and start ventilating the room when cooking. Therefore, the information of micro-environment is necessary to correctly interpret the collected AQ data, get insight on the individual exposure, and for a participant, adapt her behavior to reduce his/her exposure.

3.2.3 . How can micro-environments be recognised ?

Micro-environments are primarily defined by both temporal attributes, such as air quality (AQ) measures, and spatial characteristics. Existing works on activity recognition often focus on either geographical or temporal information. However, a comprehensive methodological approach for integrating these diverse aspects within real-world complex trajectory data is currently lacking. This combination may lead to a more robust detection model rather than the usage of a single attribute, and it needs to be investigated. In this chapter I will focus on micro-environment recognition using air quality measures, however, the extension of this work is described in my colleague's thesis [39].

To employ every available facet of the rich trajectories, the design of the micro-environment recognition model needs to integrate two layers: a geographic layer and a multivariate time series layer. I will discuss the second layer throughout this chapter. We leverage multivariate time series to detect the exact label of segments (e.g. *home*, *office*, *bike*, *metro*, *park*, etc.). Typically, this challenge prompts the adoption of a multivariate time series (MTS) classification approach, where air quality (AQ) data serves as the input, and the identified micro-environments act as the output. However, Mobile Crowd Sensing (MCS) data is marked by its heterogeneous nature, meaning that the data originates from various sensor readings. Notably, some sensors may be offline, resulting in periods of missing data that pose challenges for MTS classification. Therefore, employing MTS classification in this context is not straightforward. It becomes imperative to devise a model capable

of effectively combining data from diverse sensors, demonstrating proficiency in classification even when one or more dimensions are absent.

Moreover, in real-world scenarios, issues like imbalanced data arise. For example, we note that the prevalent labels are "home" and "work" due to the substantial time individuals spend in these locations. Consequently, the model tends to erroneously categorize the majority of segments as either "home" or "work" given their predominance in the dataset.

3.3 . Multi-view Learning Model

In this section, we present the multi-view learning approach with stacked generalization. We followed the proposal of Garcia-Ceja *et al.* [46], however, we changed the original multi-view stacking generalization approach to best fit for solving our problem.

It is expected to encounter applications that utilize diverse sensor types, such as accelerometers and gyroscopes, for activity recognition. Handling this challenge often involves extracting features from each sensor and aggregating them to construct the ultimate classification model. However, this approach has limitations as each sensor possesses distinct statistical properties. Thus, the concept of multi-view stacking arises to integrate data from various sensors with heterogeneous characteristics effectively.

The multi-view paradigm consists of learning a model based on the different views of the data. The key idea is to consider each source of data independently and fuse them with *stacked generalization* (also called *stacking*), which is a type of ensemble method [159] for combining multiple learners.

The overall process is described as follows:

1. The first step consists of defining the first-level learner and *meta learner*.
2. Train the first-level learner on each view of the original data.
3. Predict the labels of each view using the first-level learner. Each view will produce a vector with associated prediction probabilities (In the original approach, they use the probabilities of available classes and add them among the available views. In our approach, we use the probability of the prediction, which means that for each view, we will have one probability denoting the weight of this classification.).
4. Form a new matrix by column binding the prediction vectors and the actual labels. This matrix forms the new training data D' for the meta-learner.
5. Train the meta-learner with D'.
6. Generate the final multi-view stacking model.

Table 3.1: An example of the new generated dataset D'.

First-Level Learners						**Associated Prediction Probabilities**						**True Label**
l_1	l_2	...	l_i	...	l_n	p_1	p_2	...	p_i	...	p_n	y

From a conceptual perspective, assuming Y_{it} represents a dimension of the n-dimensional time series $Y_t = (Y_{1t}, Y_{2t}, ..., Y_{it}, ..., Y_{nt})$, each view V_i, where $V = (V_1, V_2, ..., V_i, ..., V_n)$ constitutes a dimension Y_{it} within the multivariate time series Y_t. Consequently, the number of views aligns with the number of dimensions.

The first-level learner accepts the values from each view as input. Subsequently, each view generates its own predicted labels, accompanied by prediction probabilities presented in the form $[l_i, p_1, p_2, ..., p_j, ..., p_k, y]$. Here, l_i signifies the predicted label of the initial-level learner i, p_j denotes the associated prediction probability for each class j among the k possible classes, and y represents the actual label. In our model, we take from each view the predicted label l_i and the weight of this prediction p_i (denoting the probability of the predicted class).

A new dataset D' is then created by column binding the output of each view and the actual labels. We remind that these outputs consist of the predicted labels and the associated prediction probabilities for each of the k possible classes. Thus D' has the form shown in Table 3.1, where l_i is the predicted label of the first-level learner i, p_i is the probability of this prediction, and y is the actual label.

After generating a new dataset D', a second-level classifier, or *meta-learner*, is trained over D' through ensemble learning [159, 2]. This approach allows us to preserve the statistical properties of each view and learn the classes of the instances with a significant improvement in classification accuracy.

Figure 3.2 describes the multi-view approach architecture. It refers to combining insights and predictions from diverse data sources or views to enhance the accuracy and robustness of predictive models. The flexibility of multi-view stacking lies in its ability to accommodate various types of first-level learners, which is one of its notable advantages. This allows us to employ a wide range of algorithms, each suited to the unique characteristics of the integrated data sources. The prediction label and its associated probability of each learner will be combined as shown in this table to generate a new dataset D'. Then, a meta-leaner is trained on the generated dataset to output the final label.

3.4 . Micro-environment recognition model

This section provides an overview of our proposed framework for micro-environment recognition in the context of MCS.

Figure 3.3 provides a panorama of the steps to achieve the micro-environment recognition objective. It shows a roadmap from the derivation of air quality and trajectory data (i.e., step 1) to data preparation (i.e., step 2), which produces data

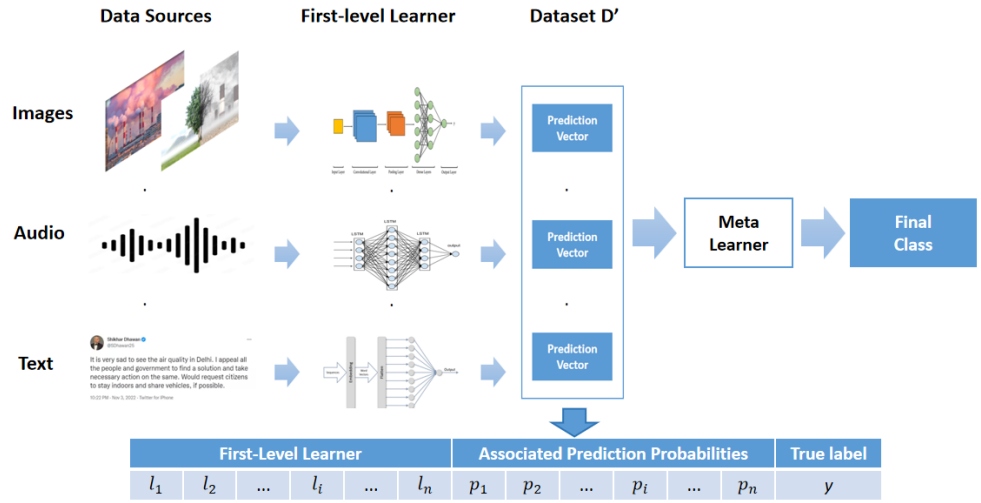


Figure 3.2: Multi-view approach Architecture.

ready to be consumed by a univariate time series classification model (e.g., kNN-DTW, LSTM, random forest, decision tree, etc.) (i.e., step 3). The univariate time series classification outputs constitute a new data set (i.e., step 4), which serves as an input for a meta-learner (i.e., step 5). The meta-learner produces the final classification results. In the following sections, we discuss data collection preparation and model training, the hybrid approach is described in detail in [39]. It is necessary to mention that the red dashed lines represent the hybrid approach.

3.4.1 . Data Collection

The essence of our micro-environment recognition approach is encapsulated in a structured sequence of steps as shown in 3.3. The journey begins with data collection. Over the course of three campaigns, more than a hundred participants were enlisted to gather environmental measurements and geo-location data continuously for a week, 24 hours a day, as they went about their daily routines. Each participant carried a multi-sensor box and a tablet with a GPS chipset. The sensors captured time-annotated measurements, including Particulate Matter (PM1.0, PM10, PM2.5), nitrogen dioxide (NO_2), Black Carbon (BC), as well as Temperature and Relative Humidity. Simultaneously, the tablet recorded participants' geo-locations and provided a means for micro-environment annotation through a self-reporting mobile app. Consequently, participants reported every transition to a micro-environment, such as home, office, park, restaurant, and so on.

3.4.2 . Data Preparation

The second step is the data pre-processing, which includes data de-noising, imputation, segmentation, and class balancing.

Initially, a significant challenge arises from the inherent noise in most sensor data, including irrelevant measurements that deviate from the actual conditions.

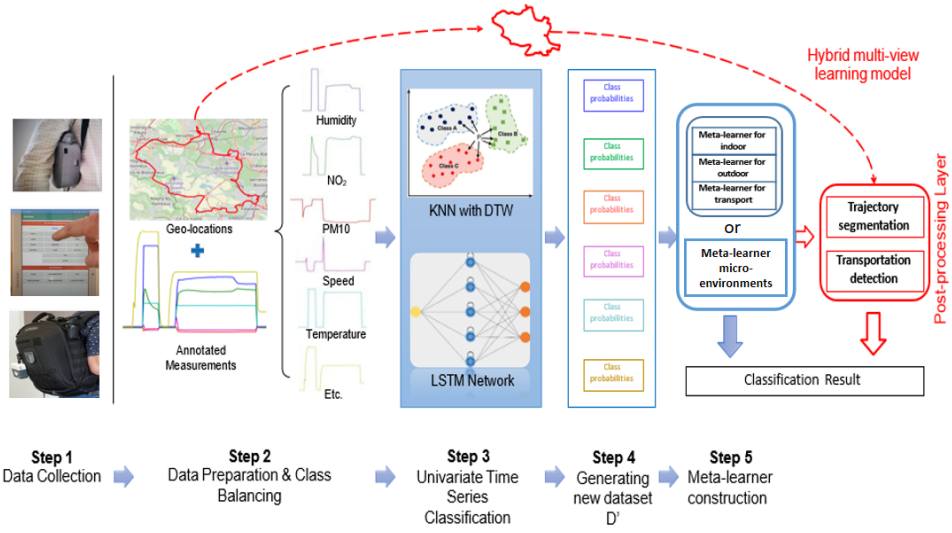


Figure 3.3: Overview of the Micro-Environment Recognition Process.

Despite maintaining a continuous focus on sensor data quality throughout the project, noise was observed in both GPS data (attributable to signal loss) and air quality data. This observation was made through a meticulous evaluation before data selection and periodic qualification during the campaign, as outlined in [79]. Unlike the sensors for climatic parameters, GPS and air quality sensors manifest these defects. Consequently, a de-noising process is implemented for GPS and air quality data. Specifically, we differentiate between peaks and artifacts by relying on expert judgment.

Next, the acquired sensory data often exhibits incompleteness due to device errors or communication issues, resulting in missing values at certain timestamps. We establish a threshold of ten consecutive missing steps for the imputation process to address this. We conduct data imputation on intervals with missing values that do not surpass 10 minutes (i.e., 10 steps). Specifically, new values are inferred using the linear interpolation approach based on the non-missing temporal neighbors; in other words, these new values are interpolated through a linear function of the two temporal ends of the missing values.

At a global level, the highest quality sample of annotated data is chosen as the baseline for validating the micro-environment recognition process. The objective is to extend the applicability of micro-environment recognition to all participants' data by utilizing a model derived from a dataset of superior quality.

Finally, micro-environment recognition confronts the challenge of class imbalance. Typically, individuals spend a significant portion of their time indoors at home or in the office. A dataset is considered imbalanced when the representation of classification categories is not uniform, as is the case in our study. Consequently, due to this imbalance (with "home" being the majority class, followed by "office"),

achieving a high accuracy value in classification is highly probable. The classifier will likely predominantly assign the majority class to nearly every data segment, thereby overlooking the minority classes. This results in an overall high accuracy that may not reflect the classifier's performance. Consequently, re-sampling and data augmentation are commonly employed techniques to address this issue.

Regarding data re-sampling, the prevalent methods involve random oversampling of minority classes and random under-sampling of majority classes. However, the random oversampling approach often introduces duplicates to stabilize the training process, limiting the exploration of valuable information within the data. Consequently, alternative methods consider synthesizing new samples for the minority class. One such approach is the Synthetic Minority Oversampling Technique (SMOTE) [21], which involves under-sampling the majority class and over-sampling the minority class based on K-nearest neighbors. SMOTE selects samples close to the feature space and generates synthetic samples in their proximity. This process can be repeated to generate as many synthetic examples for the minority class as needed.

In the realm of data augmentation, Generative Adversarial Network (GAN) [48] has demonstrated promising performance across diverse types of data, utilizing existing data more effectively compared to re-sampling techniques. Within the time series domain, Time series Generative Adversarial Networks (TimeGAN or TGAN) [143] have emerged as a recent proposal designed to generate realistic time series data while considering temporal dependencies. However, practical implementation often encounters challenges in converging the adversarial training process, especially when dealing with very limited samples [10], as is the case in our context.

Therefore, we integrate both data re-sampling and data augmentation approaches. Initially, we employ SMOTE to under-sample the majority classes and moderately over-sample the minority classes. Subsequently, we utilize the TimeGAN network to generate additional samples within the minority classes. Figures 3.4 and 3.5 depict the data distributions before and after class balancing, respectively.

3.4.3 . Multi-View Learning Model Application

Our proposal involves learning participants' micro-environments from multivariate time series (MTS) using a two-stage model based on multi-view learning. The classification model comprises training a first-level learner on each view (as illustrated in step 3 in Figure 3.3) and subsequently training a meta-learner (as depicted in step 5 in Figure 3.3) to amalgamate the output from each view, thereby enhancing the overall accuracy of the classification. As previously mentioned, the number of views aligns with the number of dimensions. For example, each dimension is treated as a separate view in a multivariate time series with four dimensions—temperature, humidity, speed, and NO2. Consequently, the multi-view learning model considers four distinct views.

In step 3, the first-level learner (e.g., *kNN*, LSTM, random forest, decision

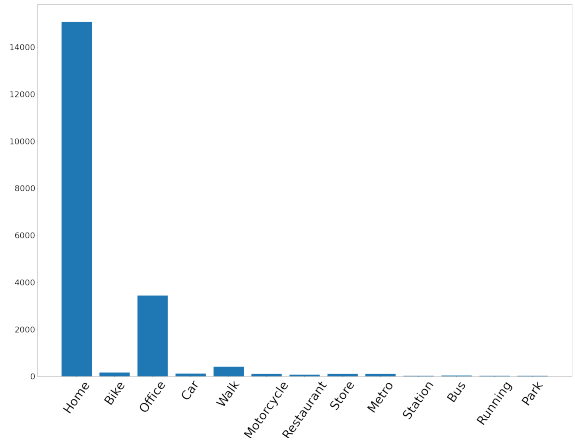


Figure 3.4: Distribution of data over classes before class balancing.

tree, etc.) takes the values of the time series data from each view as input and produces, for each view, a vector as described in 3.3. Consider the example of the multivariate time series with four dimensions—temperature, humidity, speed, and NO2—and examine the output of the first-level learners. Suppose we aim to classify the MTS into three classes: indoor, outdoor, and transport, assuming the actual label is indoor. The temperature view generates its predicted label (e.g., indoor) and associated prediction probabilities in the following format: $[l_{temperature} = indoor, p_{indoor} = 0.6, p_{outdoor} = 0.2, p_{transport} = 0.2, y = indoor]$. Similarly, the remaining three dimensions generate their predicted labels with corresponding probabilities in this structure: $[l_{humidity} = indoor, p_{indoor} = 0.7, p_{outdoor} = 0.1, p_{transport} = 0.2, y = indoor]$, $[l_{speed} = outdoor, p_{indoor} = 0.4, p_{outdoor} = 0.5, p_{transport} = 0.1, y = indoor]$, $[l_{NO2} = transport, p_{indoor} = 0.2, p_{outdoor} = 0.2, p_{transport} = 0.6, y = indoor]$.

In step 4, we generate a new dataset D' by column binding the output of the first-level learner and the true label as shown in Table 3.1, where l_i is the predicted label of the first-level learner i, p_i is the probability of this prediction, and y is the true label. Continuing with the same example of the four-dimensional MTS above, the feature structure of the generated dataset would be in the structure shown in Table 3.2.

Table 3.2: A concrete example of the new generated dataset D'.

First-Level Learners				**Associated Prediction Probabilities**				True Label
temperature	humidity	speed	NO2	temperature	humidity	speed	NO2	
indoor	indoor	outdoor	transport	0.6	0.7	0.5	0.6	indoor

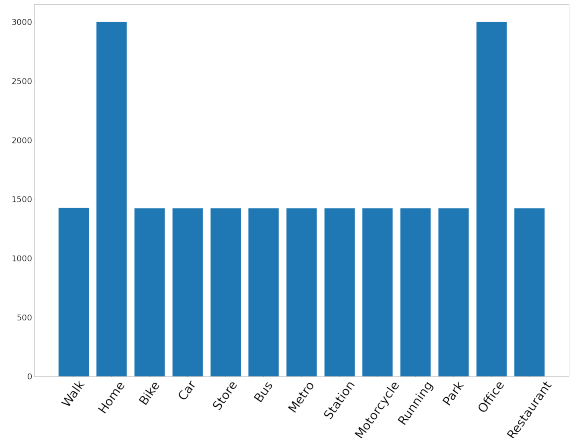


Figure 3.5: Distribution of data over classes after class balancing.

In step 5, following the generation of a new dataset D', a *meta-learner* is trained on D'. To illustrate, considering the example above, the second-level learner (e.g., *Random Forest*) takes the generated features—each view's detected label along with its corresponding probability—as input and generates the final detected label. For instance, using D' as shown in Table 3.2, the meta-learner takes as input the label produced by the "temperature" view (i.e., indoor) and its associated prediction probability (i.e., 0.6), along with the labels and their corresponding probabilities from the other three views (i.e., humidity, speed, and NO2). Thus, the input to the meta-learner has the following structure: [indoor, indoor, outdoor, transport, 0.6, 0.7, 0.5, 0.6], ultimately producing the final label (e.g., indoor) through the combination of labels and their associated prediction probabilities. Particularly, we have 13 micro-environments to detect; they are grouped into three categories: indoor, outdoor, and transport. Thus, we followed two approaches for training the meta-learner: a 2-step classification model that detects the category in the first prediction and then discriminates between micro-environments within each category or training a meta-learner to discriminate between the micro-environments directly.

One of the advantages of multi-view learning is its versatility in first and second-level learners' choices. One can flexibly substitute classifier choices between kNN, LSTM, random forest decision tree, or any other classifier [40]. In this work, we opt for *Random Forest* classifier for the first as well as meta-learners since it has shown high performance when applied in the human activity recognition domain [46].

3.4.4 . Hybrid Multi-view Learning Model

The multi-view learning model records some limitations, and we need further improvement. The red dashed line in figure 3.3 shows the post-processing phase,

Table 3.3: General characteristics of the two campaigns VGP and RECORD.

Campaign	Number of participants	Measurement period	Sensor's wearing time
VGP	15	October 2019	7 days
	12	November 2019	
	09	November 2019	
	15	December 2019	
	12	December 2019	
RECORD	13	January - March 2020	7 days

which is detailed with its experiments in Hafsa's thesis [39].

3.5 . Experiments and Results

The experiments are carried out in different environments. The multi-view learning model was implemented in Python 3.6 using scikit-learn 0.23.2 and tslearn [130]. The deep-learning models (MLSTM-FCN [70], TapNet [150]) were trained on a single Tesla V100 GPU of 32 Go memory with CUDA 10.2, using respectively Keras 2.2.4 and PyTorch 1.2.0.

3.5.1 . Experimental Settings

We assess the proposed models using real-life data from the Polluscope project in these experiments. In Polluscope, three data collection campaigns have been conducted, covering the whole study area (i.e., Paris region). A total of 103 volunteers actively participated in the one-week data collection phase. Participants were equipped with kits comprising air pollution sensors and tablets featuring GPS chipsets. The sensors recorded time-annotated concentrations of various pollutants, including Particulate Matters (PM1.0, PM10, PM2.5), Nitrogen dioxide (NO2), Black Carbon (BC), temperature, and relative humidity, at one-minute intervals. Simultaneously, the tablet served to geolocate participants and allowed them to input their time micro-environment through a dedicated Android app. Additionally, the speed dimension was derived from the geo-locational data.

In total, 13 activities (i.e., micro-environment to recognize) are considered in this study, which can be organized into three categories:

- Indoor environment: *home*, *office*, *restaurant*, *store*, *station*
- Outdoor environment: *park*, *walk*, *run*, *bike*
- Transport environment: *metro*, *car*, *bus*, *motorcycle*

In our prior work ([3]), we discussed the annotation tool used, namely an Android app installed on tablets. In this current study, data enrichment involves not only the aforementioned tool but also incorporates data from an additional tool,

Table 3.4: Average time spent per micro-environment.

Micro-environment	Stay duration (in minutes)
Office	446
Bus	13
Home	899
Station	4
Store	24
Motorcycle	20
Metro	17
Park	76
Restaurant	46
Running	76
Car	29
Bike	50
Walk	12

TripBuilder Web [19], and meticulous human control of participant annotations within the third campaign known as RECORD [18]. Consequently, the data derived from RECORD is deemed more reliable compared to our previously utilized data from the second campaign, VGP. General characteristics of the two campaigns, VGP and RECORD, are presented in Table 3.3. The dataset is composed of 8 dimensions, comprising more than 1 million rows per dimension, with an average of 82,071 rows per participant. We carefully divide the collected data into two-thirds for training and one-third for testing, ensuring that the data for each participant remains grouped in either the training or testing set. Cross-validation with "repeated stratified k-fold" is employed to further split the training set into training and validation sets. Model performance is evaluated on the testing set to assess overall effectiveness.

Taking into account the temporal nature of the data, we segment the collected data into samples of a maximum length of 5 minutes. Typically, individuals spend a significant portion of their time indoors, necessitating consideration of outdoor activities with shorter durations compared to indoor activities. For instance, the average time spent in a "station" is approximately 4 minutes, as indicated in Table 3.4, illustrating the average time spent per micro-environment. Participants tend to allocate more time to certain micro-environments, such as "home" and "office," than to others like "walk," "metro," "store," etc. As depicted in Figure 3.4, the distribution of data samples is highly imbalanced across different classes, resulting in suboptimal classification performance, particularly for the minority classes. Specifically, the model tends to optimize the global loss error, biased towards the majority classes, while neglecting the minority ones. Consequently, the obtained accuracy

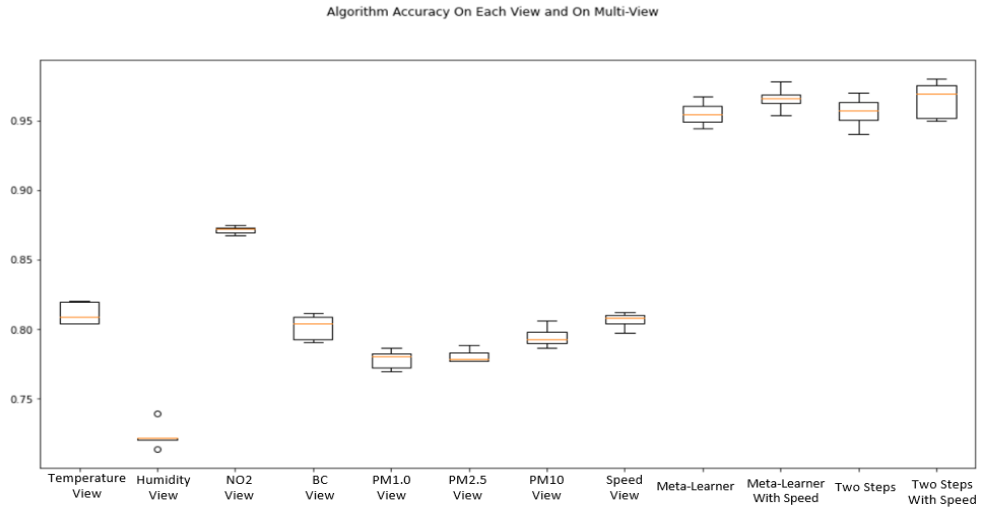


Figure 3.6: Accuracy among different views.

is not a reliable metric for evaluating actual model performance. To address this issue, we rebalanced the classes through data re-sampling and data augmentation, as detailed in Section 3.4.2 during the data preprocessing stage. Figures 3.4 and 3.5 illustrate the data distributions before and after class balancing, respectively.

3.5.2 . Experimental Design

In this section, we assess our fundamental multi-view learning model without incorporating the post-processing layer. Taking into account the mobility information present in our data, our experiments are conducted on datasets both with and without the integration of the *speed* variable. To comprehensively evaluate the significance of mobility information, we introduce and assess a two-step approach. In the first step, we distinguish between *indoor*, *outdoor*, and *transport* micro-environments, followed by a refinement step to learn a more specific class.

3.5.3 . Model Performance

In this section, we detail the experimental results of the multi-view learning model without integrating the post-processing layer. First, we evaluate the first-level learners on each single view and the multi-view learner on the global view. We evaluate as well the multi-view learner when applying the two-step approach which learns firstly the coarse-grained classes (i.e., *indoor*, *outdoor* and *transport*) then refine them into more specific classes (e.g., *home*, *park*, *metro*, etc.). Then, we compare the multi-view learner with MLSTM-FCN [70], the state-of-the-art on Multivariate Time Series Classification.

As mentioned in Section 3.4.3, the micro-environment recognition can be formulated as a Multivariate Time Series Classification (MTSC) problem, and the multi-view learner combines the predictions of each independent view (i.e., dimen-

Table 3.5: Performance of Multi-view Learner (with/out speed)

class	Without Speed			With Speed		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Walk	0.96	0.86	**0.91**	0.95	0.87	**0.91**
Bus	0.99	0.96	**0.98**	0.98	0.97	**0.98**
Office	0.96	0.88	0.92	0.97	0.92	**0.95**
Restaurant	0.97	0.97	0.97	0.99	0.97	**0.98**
Home	0.87	0.97	0.92	0.90	0.99	**0.94**
Bike	0.92	0.97	0.94	0.96	0.99	**0.97**
Car	0.99	0.98	0.98	0.99	0.99	**0.99**
Store	0.94	0.93	0.94	0.96	0.96	**0.96**
Metro	0.96	0.93	0.94	0.98	0.95	**0.96**
Station	0.98	0.96	0.97	0.99	0.97	**0.98**
Motorcycle	0.99	0.99	**0.99**	0.99	0.99	**0.99**
Running	0.99	0.99	**0.99**	0.99	0.99	**0.99**
Park	0.99	0.98	0.98	0.99	0.98	**0.99**

sion) from the first-level learners to get the final classification results. In Figure 3.6, we report the accuracy of the first-level learners over the different views, as well as the multi-view learner and the two-step approach with and without considering the mobility (i.e., speed) dimension. Globally, the results suggest that the multi-view learner shows comparable performance, with or without adopting the two-step approach. Integrating the *speed* dimension helps slightly improve the performance of the multi-view learner. We observe that the first-level learners usually have low accuracy performance, which is not surprising as the incomplete local information is not enough to train a reliable model. By combining the local information from different views, the multi-view learner can improve the model accuracy significantly.

To know how our multi-view learner performs compared to the state-of-the-art work, we select MLSTM-FCN [70], a powerful deep learning model for Multivariate Time Series Classification. We show as well the detailed evaluation results when applying the two-step approach. Since MLSTM-FCN requires enormous computational resources for parameter optimization, we train the model on GPU. In contrast, our multi-view-based approaches are trained on a normal CPU with less requirement on computational resources. For each of the models, we study the impact of using or not the mobility data and report the performance in terms of *precision*, *recall*, and *F1 score*.

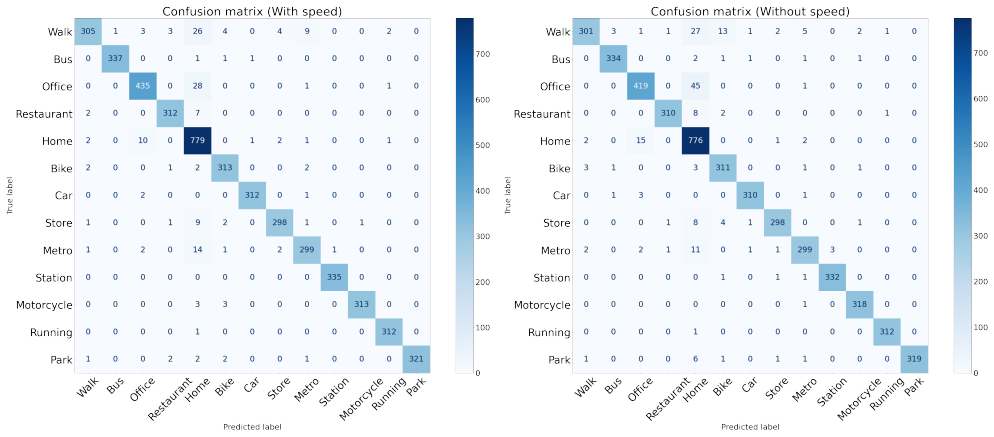
The detailed results are grouped in Table 3.5, 3.6, and 3.7. Globally, the three models have comparable results before and after adding mobility. While MLSTM shows slightly better performance than the two-step model, the latter outperforms the multi-view model. Looking at the F1-score, the out-performance of MLSTM, compared to the two-step model, does not go beyond 3 point (e.g. 0.96 and 0.99

Table 3.6: Performance of MLSTM-FCN (with/out speed)

class	Without Speed			With Speed		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Walk	0.98	0.95	**0.96**	0.94	0.97	**0.96**
Bus	1.0	1.0	**1.0**	1.0	1.0	**1.0**
Office	0.97	0.95	**0.96**	0.96	0.94	0.95
Restaurant	1.0	1.0	**1.0**	1.0	1.0	**1.0**
Home	0.97	0.97	**0.97**	0.98	0.97	**0.97**
Bike	0.98	1.0	**0.99**	0.98	1.0	**0.99**
Car	0.99	1.0	**1.0**	0.98	1.0	0.99
Store	0.99	1.0	**0.99**	0.99	1.0	**0.99**
Metro	0.99	1.0	**0.99**	1.0	0.97	**0.99**
Station	0.99	1.0	**1.0**	1.0	1.0	**1.0**
Motorcycle	1.0	1.0	**1.0**	1.0	1.0	**1.0**
Running	1.0	1.0	**1.0**	0.99	1.0	0.99
Park	1.0	1.0	**1.0**	1.0	1.0	**1.0**

Table 3.7: Performance of Multi-view Learner (2-step approach with/out speed)

class	Without Speed			With Speed		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Walk	0.93	0.97	**0.95**	0.95	0.96	**0.95**
Bus	0.99	0.99	**0.99**	0.99	0.99	**0.99**
Office	0.97	0.92	**0.94**	0.97	0.91	**0.94**
Restaurant	0.99	0.98	**0.98**	0.99	0.98	**0.98**
Home	0.93	0.97	**0.95**	0.93	0.98	**0.95**
Bike	0.97	0.96	0.96	0.97	0.97	**0.97**
Car	0.98	0.99	**0.99**	0.99	0.99	**0.99**
Store	0.98	0.97	**0.97**	0.98	0.96	**0.97**
Metro	0.98	0.97	**0.98**	0.98	0.98	**0.98**
Station	1.0	1.0	**1.0**	0.99	1.0	0.99
Motorcycle	0.99	0.98	0.98	0.99	0.99	**0.99**
Running	0.98	0.98	**0.98**	0.98	0.98	**0.98**
Park	0.99	0.96	0.97	0.99	0.97	**0.98**



(a) With Speed (b) Without Speed

Figure 3.7: Multi-view Approach Confusion Matrix

for the class bike) before adding mobility, and 2 points (e.g. 0.97 and 0.99 for the class store) after adding mobility, whereas the difference between the two-steps model and the multi-view model does not exceed 4 points (e.g. 0.91 and 0.95 for the class walk) before and after adding mobility.

As for our multi-view learner, when integrating the speed dimension for model training, we observe an improvement in the model's performance, particularly the F1-score, while the performance of MLSTM-FCN does not improve or even deteriorates. Figure 3.7 shows the confusion matrix of multi-view approach. Figure 3.7a reports the confusion matrix with the presence of the mobility dimension (i.e. speed), while figure 3.7b corresponds to the confusion matrix of the model with the absence of mobility dimension. We notice that the model can easily discriminate between the "indoor", "outdoor" and "transport" activities, but it cannot perfectly distinguish between the micro-environments inside each category. For example, even though some of the samples in the "home" micro-environment are falsely predicted as "restaurant" or "office", the three micro-environments, "home", "office" and "restaurant" can be classified as indoor. Thereby, we introduced a grouping step before recognizing the micro-environment. In this step we classify the sample into either an "indoor", "outdoor", or "transport" environment. Based on the classification result, a model will be specialized for each indoor, outdoor or transport micro-environments. Table 3.7 shows the results of the added step.

We apply four basic MTSC models:

- MVB: our proposed Basic Multi-View learning model.
- MV-2steps: our proposed Basic Multi-View learning model with two-step classification as shown in Section 3.5.3.
- MLSTM-FCN [70]: a powerful deep learning model for Multivariate Time

Table 3.8: The description of various model variants

Model	Description
MVB	Basic multi-view model
MV-2steps	Multi-view model having 2 steps, first discriminate between indoor/outdoor/transport and then classify the micro-environment.
MLSTMB	Basic MLSTM-FCN model.
KNN-DTWB	Basic KNN-DTW model.

Series Classification.

- KNN-DTW [14]: the most popular benchmark for Time Series Classification which adopts the K-nearest neighbor (K-NN) classifier with dynamic time wrapping (DTW) distance.

As the models are trained on different hardware environments (e.g., MLSTM-FCN is trained on a GPU, which is ten times faster than running on CPU), it is unfair to compare them in terms of efficiency. However, according to the recent study [118], the deep learning-based models usually require more computational resources than classic data mining approaches; the lazy classifiers (e.g., KNN-DTW) are much slower than the tree-based classifier (e.g., Random Forest) due to the costly distance computations (e.g., DTW). As the first-level learner and meta-learner in our multi-view learning model are based on Random Forest, thus the model training and prediction are quite efficient compared to other models. The model variants are described in table 3.8.

To validate our approach against state-of-art approaches, we used the trained model to predict the micro-environment of our real MCS data and we adopted the post-processing techniques on the results. Here, we show the global accuracy comparison between the models. Tables 3.9 report the accuracy of our models versus the state-of-art approaches. The *NaN* in the results of MLSTM and KNN models indicates that no complete data is collected, thus, the models are not applicable. More precisely, some variables are missing during the data collection process. However, the multi-view-based models succeed all to detect the micro-environment even some dimensions are missing. Moreover, our approach shows its superiority when compared to the existing approaches.

3.6 . Model Generalization

Table 3.9: Performance comparison of various models

Participant ID	MVB	MV-2steps	MLSTMB	KNN-DTWB
988088403	88.2	**89.2**	63.5	85.8
988231648	**90.9**	90.7	NaN	NaN
982228564	**94.3**	93.8	23.6	74.8
986002161	**91.0**	89.5	NaN	NaN
986939872	**83.3**	82.2	34.4	72.0
988335737	**60.9**	59.2	NaN	NaN
986174566	92.0	**92.5**	10.8	76.7
986884172	**85.9**	85.7	NaN	NaN
986938604	**98.6**	98.4	37.6	91.4
985935431	90.7	**90.8**	NaN	NaN
987014104	**98.1**	97.6	38.5	89.8
82119412	**96.8**	95.9	10.0	66.0
983602168	89.8	**89.9**	NaN	NaN
Overall Accuracy	**91.33**	91.0	31.14	83.48

In practice, we should consider the model generalization on unseen data, which allows evaluating the model in more complex scenarios. We have used the multi-view model (which have been trained over RECORD campaign data) to classify data that have never been seen by it before. We opt for the VGP campaign data, which was collected during a different time period from RECORD, to prove the generalization ability of the proposed model. We have plotted the predictions versus the declared activities (which is not guaranteed to be accurate, not all participants thoroughly annotated their data). Figure 3.8 shows the plot of declared versus predicted micro-environments. For this participant (i.e. participant 9999988), we trust his/her annotations, so we can notice that the model has performed well. While for figure 3.9, as we don't have the real ground truth, we can see that the model's predictions are more reliable than the annotations. For instance, the participant in the plot has declared three times staying outdoors in the middle of the night (i.e. 24, 25 and 26th of October 2019), which is very unlikely to be true. Some other participants may completely forget to annotate the change of micro-environment, so the declared annotations are indeed imperfect.

3.7 . Discussions & Perspectives

The multi-view learner adopted in this paper is composed of the base learner (i.e., Random Forest) and the meta-learner (i.e., Random Forest), which has greatly improved the performance compared to the single kNN-DTW classifier. The objective of this paper is not to propose the best classifier for MTS classification, but to provide an insight that the multi-view learner is capable of coordinating effectively

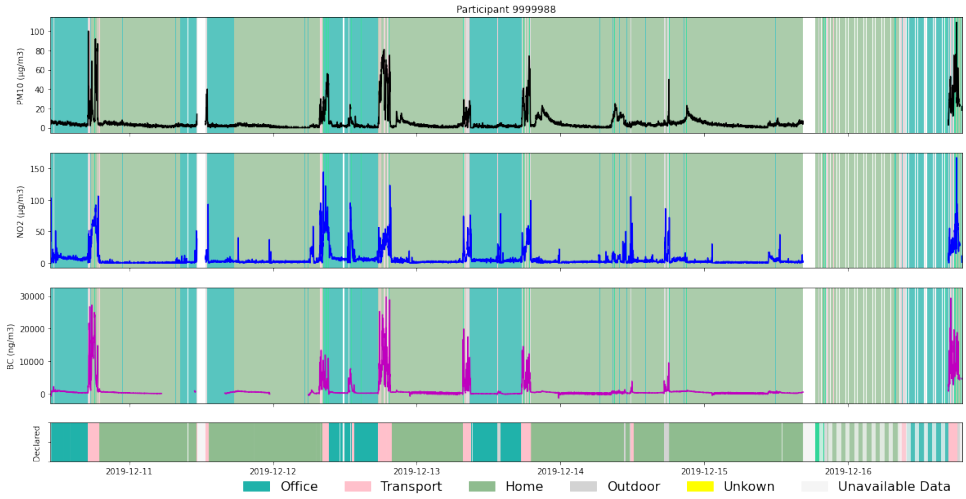


Figure 3.8: Predictions of VGP campaign for participant 9999988.

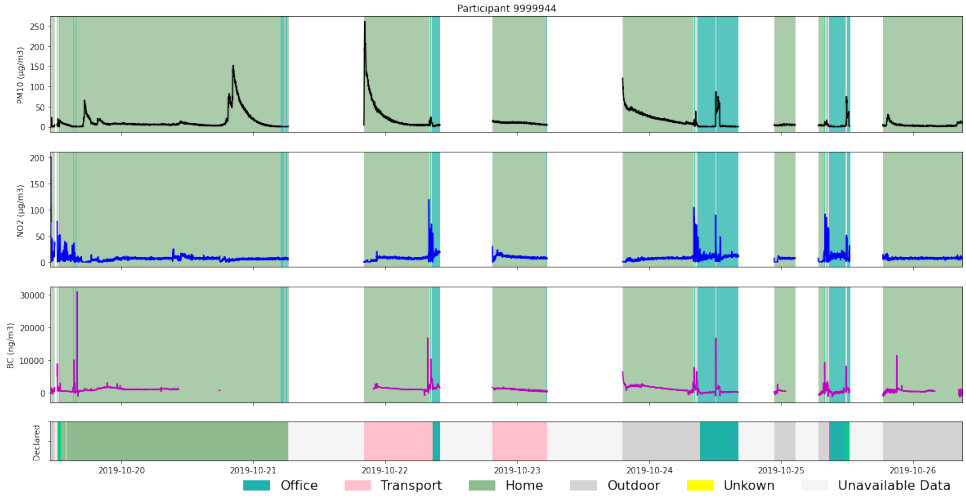


Figure 3.9: Predictions of VGP campaign for participant 9999944.

the information from different variables and achieving more reliable performance than a single base learner. Moreover, the results of the grouping approach which is based on the multi-view approach confirms that there is a clear signature for each micro-environment, thus we can have an effective prediction with this approach. Moreover, the multi-view approach offers the reusability of the first-level learners, and allows using different classifiers and combinations for the first-level learners. Multi-view model doesn't require a special hardware such as GPU for training Neural Networks (i.e. MLSTM-FCN). In addition, it does not require a long execution time for classification as other classifiers such as KNN-DTW do. Besides, using the multi-view approach allows the prediction of micro-environments in the absence of some dimensions in the data. Another advantage of using the multi-view approach is that its meta-learner is trained on out-of-fold predictions thus the model will not over fit.

Nevertheless, the kNN-DTW is considered as the baseline for MTS classification and is widely outpaced by the advanced approaches such as Shapelets [142, 161, 160] or the frequent patterns [105]. Essentially, the kNN-DTW captures the global feature based on the distance measure between the entire sequences, while the local features (e.g., the frequent patterns [105], the interval features [32], Shapelets [142], etc.) are more appropriate when a specific pattern characterizes a class. More specifically, a combination of features extracted from different domains may dramatically improve the performance of the base learner [91]. Therefore, one of the perspectives consists of the **optimization** of the base learner and the exploration of the **explainability** of the multi-view learner on both the feature interpretation and the variable importance for building the classifier. For this reason, we have removed the NO2 and BC dimensions to show their importance for some classes. Table 3.10 shows the precision, recall, and F1 score for MVB while removing some dimensions (NO2 and BC) compared to the MVB model containing all dimensions. The comparison shows that the F1-score of the MVB model for all classes is greater than that model without NO2 and BC. Except for *Running* class which is only one point difference. This comparison shows the importance and role of those dimensions (NO2 and BC) in micro-environment prediction. We have chosen to remove NO2 and BC because depending on figure 3.6, these 2 dimensions have the highest accuracy compared to other dimensions. The visual representation of Shapelets make them good candidates for such improvement.

3.8 . Conclusion

The current focus on mobility sensors has heightened interest in activity recognition among researchers. Micro-environment recognition, crucial in projects like Polluscope within the realm of MCS, enables the analysis of an individual's air pollution exposure in relation to their micro-environment. Our study's key insight suggests that environmental observations can, to some extent, characterize

Table 3.10: Performance of MVB without NO2 and BC VS. MVB

class	MVB without NO2 and BC			MVB		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Walk	0.74	0.76	0.75	0.82	0.79	**0.80**
Bus	0.58	0.54	0.56	0.85	0.64	**0.73**
Office	0.78	0.74	0.76	0.86	0.85	**0.85**
Restaurant	0.43	0.60	**0.50**	0.42	0.60	**0.50**
Home	0.92	0.93	0.93	0.95	0.95	**0.95**
Bike	0.49	0.47	0.48	0.57	0.61	**0.59**
Car	0.34	0.15	0.20	0.51	0.18	**0.27**
Store	0.54	0.57	0.55	0.61	0.61	**0.61**
Metro	0.52	0.60	0.55	0.62	0.70	**0.66**
Station	0.10	0.12	0.11	0.16	0.17	**0.16**
Motorcycle	0.25	0.07	0.11	0.33	0.08	**0.12**
Running	0.32	0.61	**0.42**	0.30	0.61	0.40
Park	0.26	0.89	0.41	0.32	0.86	**0.47**

the micro-environment. Notably, the model's high accuracy implies the potential for automated micro-environment detection, alleviating the need for participant self-reporting. Integrating the mobility feature as a time series marginally improves accuracy, with a moderate gain. Consequently, micro-environment characterization remains viable even in the absence of the speed dimension.

Employing various approaches and learners, we conducted a comprehensive experimental study showcasing the efficacy of the multi-view approach for time series classification, even when some dimensions are missing. Comparison with baseline classifiers, MLSTM-FCN and kNN-DTW, further underscores the strength of our proposed methodology.

Chapter 4 - Enhancing Air Quality Assessment with Opportunistic MPM

Contents

4.1 . Introduction

In this chapter, we are motivated by the second objective in this dissertation. The main aim is to estimate air pollution levels in uncovered area while leveraging fixed and opportunistic mobile participatory monitoring data.

The deteriorating air quality in urban areas, particularly in developing countries, has led to increased attention being paid to the issue. Daily reports of air pollution are essential to effectively manage public health risks. Pollution estimation has become crucial to expanding spatial and temporal coverage and estimating pollution levels at different locations. The emergence of low-cost sensors has enabled high-resolution data collection, either in fixed or mobile settings, and various approaches have been proposed to estimate air pollution using this technology. The objective of this study is to enhance the data from fixed stations by incorporating opportunistic mobile participatory monitoring (MPM) data. We aim to address the raised research question **R2** which is in short: How can we augment fixed station data through MPM? Hence this chapter presents the second contribution of this thesis **C2**, i.e. enriching maps with fixed and opportunistic mobile monitoring data to estimate pollution. To overcome the limited availability of MPM data, we leverage existing data collected during periods when the pollution maps align with those observed by the fixed stations. The combined fixed and mobile data is then subjected to interpolation methods to generate more accurate pollution maps. The effectiveness of our approach is demonstrated by experiments conducted on a real-life dataset.

The rest of this chapter is organized as follows: The following section presents background about monitoring and estimation approaches, what are the problems, and the contributions of this work. Section 4.2 presents the methodology and the detailed explanation of our approach. Section 4.3 details the implementation part and section 4.4 shows the findings of experiments conducted on a real-world dataset. Section 4.5 . Finally, the last section 4.6 summarizes our work and draws our perspectives.

4.1.1 . Background

Air pollution has become one of the major concerns of the 21st century, especially in densely populated urban areas. The combination of urbanization and climate change poses a significant threat to the health of urban populations and the environment. The impact of air pollution on human health and the environment has been well-documented, including respiratory and cardiovascular diseases, reduced life expectancy, and ecological damage. By 2050, up to 70% of the global population is projected to reside in urban areas, with 75% of Europeans already living in cities. This trend presents a range of interconnected challenges that impact social, economic, and environmental infrastructures, with deteriorating air quality being a particular concern, especially in developing nations.

Virtually everyone on Earth is breathing polluted air. Indeed, according to the

World Health Organization (WHO), 99% of the world's population lives in places where air quality exceeds internationally approved limits [1]. WHO's estimates show that around 7 million premature deaths per year are attributable to the combined effect of ambient and household air pollution.

The significance of air pollution monitoring has risen in recent years due to its ability to generate the Air Quality (AQ) index for the region under consideration. Air pollution monitoring can be highly beneficial by aiding policymakers in devising more effective strategies to tackle pollution-induced urbanization challenges.

Monitoring and estimating air pollution in uncovered spots is essential to take adequate measures to reduce air pollution. Air pollution monitoring involves the measurement of pollutants in the atmosphere, such as particulate matter, nitrogen oxides, sulfur dioxide, carbon monoxide, and ozone. This information can help identify areas with high pollution levels and determine the sources of pollution. With the advancement of technology, air pollution monitoring has become more efficient, accurate, and cost-effective.

4.1.2 . Problem Statement

Different air pollution monitoring approaches include fixed stations, low-cost fixed sensors, and mobile sensors [23, 129, 17]. Each monitoring approach has advantages and limitations, and selecting a monitoring approach depends on the specific needs of the study or monitoring program.

Air pollution monitoring has extensively relied on fixed stations for the last three decades to generate the AQ pollution index. These stations typically record the hourly average of pollution levels in a specific region. Regrettably, the deployment of such stations is financially demanding, and their maintenance is also a significant concern, leading to limited coverage.

On the other hand, low-cost fixed sensors are cheaper and easier to install than fixed stations. They can be placed in various locations, such as street lamps or buildings, and provide real-time air pollution data. However, their accuracy can be limited, and they may only measure some pollutants of interest.

Researchers have shown recent interest in using air quality mobile sensing as an alternative method for measuring air pollution [88]. Mobile sensors, such as vehicles equipped with sensors, can capture air pollution data in specific areas or along transportation routes. They can provide high spatial resolution data but may not capture long-term trends or variations in air pollution. Mobile sensors for air quality are cost-effective and offer high-resolution pollution measurements while being deployed in high densities, as noted by [77] and [88]. However, calibration is typically necessary for such sensors.

Fixed stations can produce precise measurements but fall short regarding spatial coverage. Conversely, mobile sensing can expand spatial coverage but may also yield some imprecise measurements. Additionally, fixed stations generally maintain continuous temporal coverage at specific locations, while mobile sensors may not have steady temporal coverage at specific locations, and typically last for a brief

period of time.

Air pollution estimation consists in predicting air pollution concentrations at locations without monitoring equipment. This approach is beneficial for regions where monitoring stations are limited or nonexistent.

To overcome the limitations of individual air pollution monitoring approaches, combining different approaches can provide a more comprehensive understanding of air pollution levels in uncovered spots. Researchers have utilized fixed stations and mobile sensor data to estimate pollution maps. Some studies have relied exclusively on fixed stations [13, 52, 128], while others have applied air pollution estimation methods used in fixed stations to low-cost mobile sensor data [51, 96]. However, recent research proposes combining data from fixed and mobile sensors [56, 129]. Prior studies that integrate fixed and mobile sensor data or solely rely on mobile sensing typically involve targeted campaigns focused on specific routes or deploying sensors on buses or trams following fixed paths. These studies raise several unresolved questions. Firstly, what are the most effective deterministic methods, geostatistical methods, or machine/deep learning models? Secondly, what features should be considered during the pollution estimation process? Lastly, how should we address the challenges of merging data from fixed and mobile sensors, considering the differences in their resolution and spatiotemporal coverage?

However, existing approaches work only for data collected through targeted and synchronized campaigns. Such approaches do not consider opportunistic data acquired from participants performing their real-life activities at different times and places. Nowadays, the concept of opportunistic mobile sensing is rapidly spreading. Smartphones can capture location, motion, environmental and health parameters, etc. In our study, we are trying to use opportunistic mobile data along with fixed sensor data to estimate pollution in uncovered spots. The main problem is the scarcity of opportunistic mobile data matching the fixed sensor measurements, leading to a low enrichment of such opportunistic data to the pollution maps.

4.1.3 . Summary of Related Work

Researchers have shown interest in the problem of estimating pollution for several years [9]. The problem has been examined in the literature from various perspectives and scales. This problem has been studied in meso-scale [99, 125, 57], urban-scale [68, 69, 121], and beside these approaches we have the data driven approaches that rely on collected sensory data. Data-driven methods [12, 114, 112] have become popular due to the increased use of monitoring stations, including traditional fixed networks, denser networks of low-cost fixed sensors, and low-cost mobile devices. We are interested in data-driven approaches in our work.

Over the years, numerous techniques have been suggested for approximating or interpolating pollution levels in areas without monitoring stations. Although air quality estimation methods are typically intended for stationary sites, they can also be modified to accommodate information obtained from mobile and stationary sensors. These techniques can be divided into five categories: Land Use Regression

(LUR), Dispersion Models, Deterministic Interpolation Methods, Geostatistics, and ML/DL Algorithms.

These approaches can be deployed for fixed stations data, mobile sensors data, and even the combination of both data sources. Existing approaches in the literature that use fixed and/or mobile data have typically conducted targeted data collection campaigns on specific roads or outdoor places. However, this work aims to use MCS data to enhance fixed stations' data without relying on directed or outdoor data collection campaigns. In this chapter we are using opportunistic MPM data to enrich air pollution maps generated on top of fixed sensors data, then the enriched maps will be used to estimate the air pollution in un-monitored areas.

4.1.4 . Proposition and Contributions

In our work within the GoGreen Routes project we propose a framework allowing fixed station data enrichment with opportunistic mobile crowd-sensing data (i.e., low-cost hand-held sensors that collect data opportunistically from the crowd) to expand the spatiotemporal coverage of air pollution monitoring. Our research hypothesis is that combining these data sources makes it possible to define enriched maps that capture the spatio-temporal variability of air pollution at a higher resolution than using each source/approach separately.

This work presents a novel approach to assessing air pollution concentrations using data from fixed and opportunistic mobile sensors. Our methodology leverages a mobile crowd-sensing (MCS) approach. MCS [50], is a new paradigm that harnesses data acquired by volunteers using sensor-enhanced mobile devices with GPS capabilities while carrying out their daily routines, resulting in non-persistent data collection and limited outdoor data samples as most activities are indoors. Unlike the existing approaches, this schema's main issue is coping with the scarcity of such opportunistic data in the outdoor environment in the enrichment task. From one hand, MCS data do not have a steady temporal coverage, and from another hand the amount of instantaneous data collected outdoors remains very low.

Our research question centers on the possibility to still utilize MCS/MPM[1] data to supplement fixed station data for better estimation of air pollution across the city.

Using deep learning methods, we will then use these enriched maps to quantify air pollution concentrations in uncovered spots. Deep learning methods have shown promise in predicting air pollution concentrations by learning the underlying patterns and relationships.

In order to address the challenge of limited MPM data availability, we merge the MPM data corresponding to similar general pollution conditions. Once the MPM

[1]Please note that MPM (opportunistic mobile participatory monitoring) and MCS (opportunistic mobile crowd sensing) are the same. For the rest of this chapter, we will use MPM to refer to opportunistic mobile monitoring.

data is aligned with the fixed station map in the same time interval, we look for similar conditions at different periods and harness the MPM data collected in these periods. To do so, we identify clusters of different fixed station data, match them with MPM data at corresponding times, and combine them to generate more data samples and improve the pollution map. This method results in enhanced pollution estimation.

The proposed study has several sub-contributions to air pollution monitoring and estimation derived from **C2** contribution.

- Propose a method to combine fixed station data with mobile participatory monitoring data. We can create enriched maps that capture the spatiotemporal variability of air pollution in uncovered spots. This approach provides a more comprehensive understanding of air pollution levels than individual monitoring approaches and can be used to identify pollution hotspots and sources.
- Using deep learning methods to estimate air pollution levels in uncovered spots, we can expand the spatiotemporal coverage of air pollution monitoring.
- Validating our approach on top of real-life datasets from two cities in France and the USA: Versailles and Chicago.

4.2 . Methodology

This section will present our proposed methodology for enhancing fixed station measures with data obtained through mobile crowd sensing or low-cost sensors. We may have very few samples from various outdoor locations when collecting opportunistic mobile crowd-sensing data. Our proposed solution addresses this data scarcity, allowing us to utilize MPM data and fixed station measures to estimate air pollution.

Air pollution levels can vary significantly from one place to another and may change rapidly due to various factors such as meteorological conditions, traffic, and land use. Despite these differences, it is possible to group these changes into clusters that reflect pollution levels during specific periods.

This intuition guided our method to overcome MPM data scarcity. Hence, we hypothesize that fixed station measures within the same pollution cluster could share similar MPM data. To test this hypothesis, we will cluster fixed station measurements and use the matching dates and times to identify relevant MPM data. We will then use the union of these MPM data to enrich the pollution maps and estimate pollution using an interpolation or a prediction algorithm with a larger input sample.

The approach involves clustering the air pollution levels based on fixed station data, merging these clusters with MPM data, and applying interpolation using

Convolutional Neural Networks and Long Short-Term Memory (CNN-LSTM) to estimate the values at uncovered places.

Algorithm 1: Pollution estimation using Fixed and MPM data

Input: *Hourly Fixed Stations measures*, *MPM measures*
Output: *Enriched pollution estimation map*

1 Split the area of interest into cells using a grid view.
2 Create different snapshots of pollution maps based on **hourly** fixed station measures.
3 Apply a clustering algorithm to group those snapshots into clusters having the same pollution levels.
4 Select the timestamps of measures within each cluster.
5 Calculate the hourly average of MPM data.
6 Select hourly average MPM data matching the timestamps extracted from each cluster.
7 Enrich fixed station snapshot with the available average MPM measures sharing the same timestamps.
8 Adapt an estimation approach to interpolate values in the remaining uncovered spots on top of the enriched map.

The methodology for enriching air pollution fixed station data with data collected from mobile crowd sensing or low-cost fixed stations involves clustering, data enrichment, and interpolation. Our approach is detailed in Algorithm 1, which outlines the following steps. First, identify an area of interest to estimate air pollution in that area, and split it into cells using a grid view (Step 1). Then, the available fixed station measures are assigned to their corresponding cells in the map. After that, creating hourly map snapshots based on the pollution levels measured by the fixed stations (Step 2). Clustering the created snapshots is the next step. The algorithm groups the air pollution levels based on the similarity of their values. At this stage, we consider snapshots of different timestamps altogether without distinguishing between rush hours or working and holidays. However, the rush hours or holidays may eventually belong to the same cluster if the corresponding pollution maps are similar. This clustering aims to identify the timestamps sharing the same pollution conditions (Step 3). In the next step, we keep track of timestamps to merge them with MPM data to produce enriched maps (Step 4). Table 4.1 shows an example of the snapshots representing the measurements of different fixed stations at a timestamp T_i. For instance, the first snapshot corresponds to the pollution map at period T_1 with sensor values (13.3, 16.2, ..., 12.1, 10.9) for the fixed sensors in order S_1, S_2, ..., S_{k-1}, and S_k. The values here represent the same type of measures (e.g., PM2.5); thus, they have the same scale and range. Therefore, we simply used the Euclidean distance between the vectors of sensor values in K-means.

The algorithm output is K clusters where each cluster groups together pollution maps based on their similarity (i.e., the similarity of their fixed sensor vector values). The corresponding timestamps are also returned to match the timestamps of mobile sensors. These steps are described in figure 4.1.

Time Periods	Vectors of Sensor Values
T_1 = 2023-06-15 18:00:00	(13.3, 16.2, ... , 12.1, 10.9)
.	.
T_i = 2023-06-30 14:00:00	(3.4, 2.2, ... , 1.1, 0.9)
.	.
T_n = 2023-07-14 08:00:00	(6.3, 8.2, ... , 10.1, 5.7)

Table 4.1: Fixed Stations' Snapshots Example

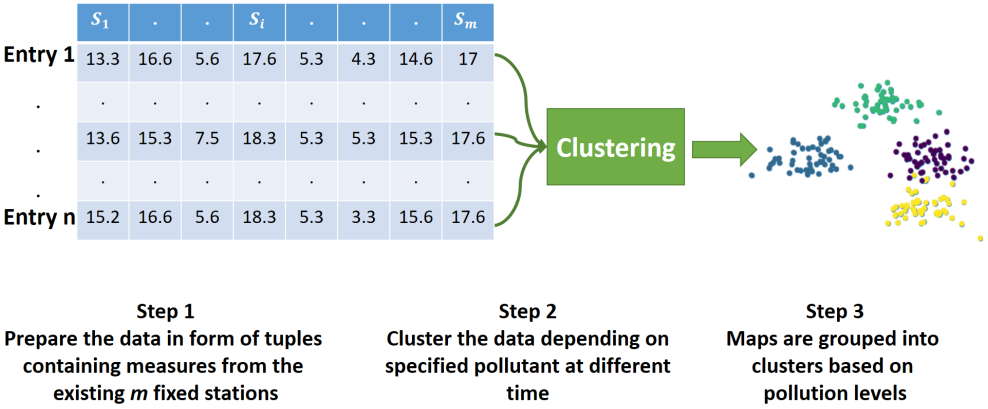


Figure 4.1: Clustering Fixed Stations Data

The next step in the methodology is to enrich the data collected from fixed stations with MPM data collected. To do this, we use the date-time values of the measures in each cluster to merge those measures with the mobile crowd sensing or low-cost fixed stations data. We begin by calculating the hourly average of MPM data to have unified timestamps. Then, merging is performed using the date-time values as the common identifier. We add available MPM data to unmonitored cells. Cells containing fixed station measures are not changed. The result is enriched clusters containing data from fixed stations and MPM measures (steps 5 - 7). Figure 4.2 describes the previous steps.

The final step in the methodology is to use interpolation to estimate the air pollution levels at uncovered places. Interpolation is a technique used to estimate the value of a variable at a point that is not explicitly measured. Based on the available features, we can select the appropriate technique and perform interpolation to estimate pollution levels at uncovered spots (Step 8). As shown in figure 4.3, the selected model inputs the enriched maps and outputs the estimated map.

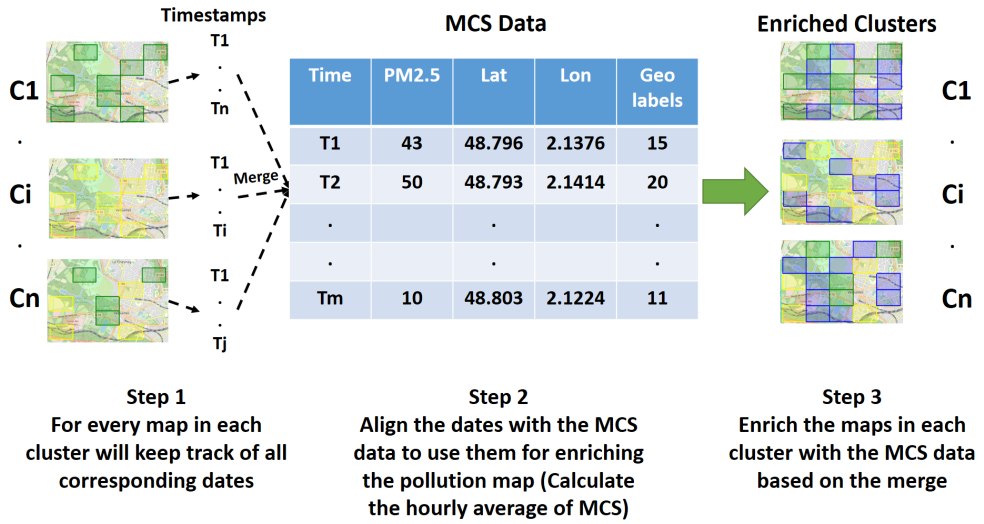


Figure 4.2: Enriching the representative map with MPM data

Our methodology uses unsupervised machine learning algorithms for clustering and interpolation methods for air pollution estimation. Our approach can be used to create air pollution maps that provide a comprehensive view of air pollution levels in a given area. The maps can be used to identify areas with high pollution levels.

4.3 . Implementation

This section details the implementation pipeline of our air pollution enrichment and estimation approach. Figure 4.4 shows the implementation pipeline that includes several parts: data collection, data preprocessing, data enrichment, and air pollution estimation.

4.3.1 . Data Collection

This subsection will focus on data collection from two study areas, Versailles and Chicago. We collect data from fixed, low-cost, and mobile sensors.

In **Versailles city**, we collected data from 14 June until 16 July 2023. The collection includes data from 6 fixed stations spread over different spots in Versailles. We used the AtmoTube Pro sensor for opportunistic MPM data to collect air pollution measures. With the help of ten volunteers, we collected around 4000 minutes of outdoor records. For fixed stations, we had around 700 hourly average records. The study area in this experiment only covers Versailles city with an area around 27 km^2.

Fixed sensors were deployed by eLichens[2] as part of the GoGreen Routes project in Versailles. These sensors monitor air quality and provide data for analysis. Fixed

[2]https://www.elichens.com/

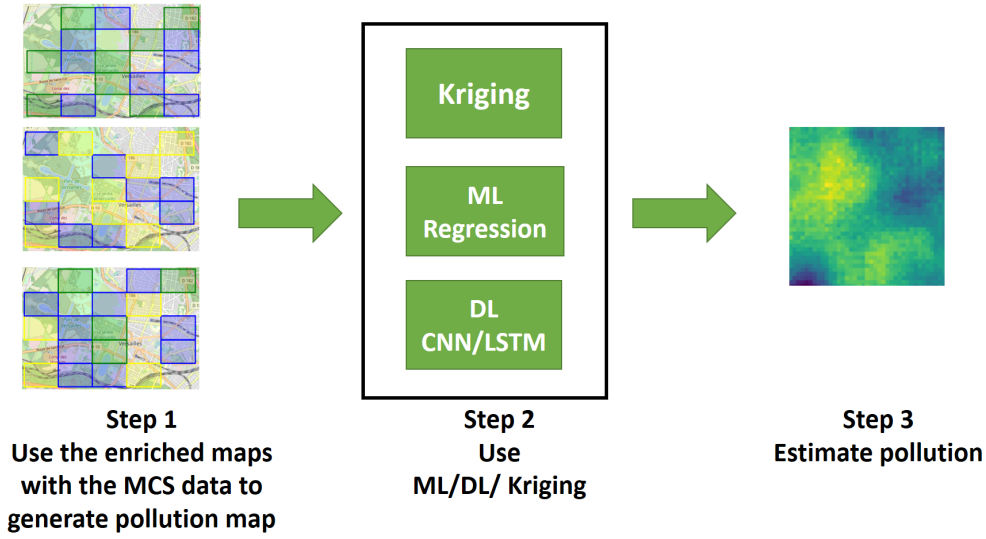


Figure 4.3: Approach for Pollution Estimation

stations provide reliable data over a longer period and are suitable for long-term air quality monitoring. These sensors measure of particulate matter ($PM1.0, PM2.5$, and $PM10$), as well as estimates of Nitrogen dioxide NO_2, Ozone O_3, temperature, and humidity. They provide hourly aggregations, resulting in one representative record per station for each timestamp.

MPM data is collected with the help of ten volunteers. They used AtmoTube Pro sensor which collects $PM1.0$, $PM10$, $PM2.5$, $VOCs$, temperature, and relative humidity. The collection was performed opportunistically, as the participants conducted real-life activities. The data collected from mobile crowdsensing is valuable as it provides a high spatial resolution of air quality data.

To generalize our approach, in the study's second phase, we seek out public datasets with community-based data collection from Aircasting[3] and OpenAQ[4] specifically in **Chicago city**.

OpenAQ is a platform that provides data from various sources, including fixed stations and low-cost sensors. The data from OpenAQ provides a more comprehensive picture of air quality by combining data from different sources. This platform helps compare data from different sources and identify patterns in air quality over time. Using the provided API[5], we collected reference grade measures (fixed stations) and low-cost fixed station measures.

On the other side, Aircasting is a platform that provides data from low-cost sensors, which are small, portable, and easy to install. These sensors measure carbon monoxide, nitrogen dioxide, and particulate matter. The data from Air-

[3]https://www.habitatmap.org/aircasting
[4]https://openaq.org/
[5]https://docs.openaq.org/docs/about-api

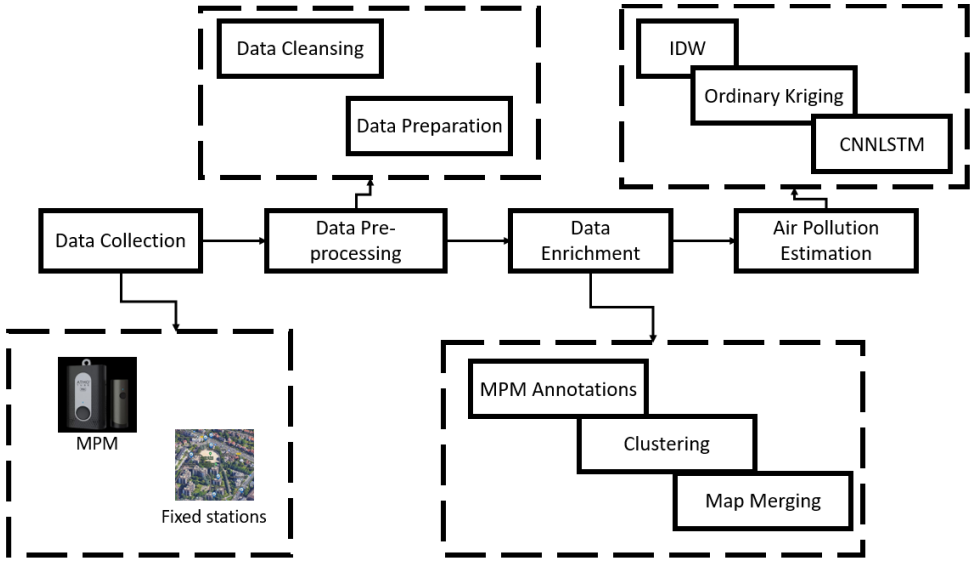


Figure 4.4: Implementation Pipeline

casting provides valuable information for monitoring air quality in real time. Using AirCasting API[6], we were able to acquire mobile participatory data.

We collected fixed stations, low-cost sensors, and mobile sensor data within a bounding box of 288 km^2 in Chicago. The data collection took place between October and December. The fixed stations produced roughly 1304 hourly average records. For the low-cost fixed sensors, we have 40575 minutes of data. At the same time, the length of MPM data was originally 368276 records at the seconds' timescale, which results in 2515 minutes of data after averaging the measures.

For the sake of simplicity, we restricted our experiments to PM2.5, which is the most available in both fixed and mobile sensors. However, our method can be applied to any environmental measure.

4.3.2 . Data Pre-processing

Preprocessing and data preparation are essential steps in data analysis as they ensure the data is clean, organized, and ready for analysis. This study followed a series of steps to preprocess and prepare the data for analysis.

First, we selected an area of interest for our study. This area could be a city, a neighborhood, or any other geographical region we wanted to analyze. Once we had selected the area of interest, we split it into cells using a grid view. We used a specified granularity of either 1km x 1km or 500m x 500m, depending on the level of detail we wanted in our analysis. This step allowed us to analyze air quality data at a more granular level and identify hotspots or areas of concern.

For MPM data, we filtered the GPS data with the help of scikit-mobility [108]. GPS data often contain noise and outliers that can affect the accuracy of the data.

[6]https://github.com/HabitatMap/AirCasting

With the help of scikit mobility library functionalities we performed data cleansing, by filtering data to remove noise and spikes. This library is designed for mobile data analysis and provides various data cleaning and preprocessing functions.

The MPM data and low-cost sensor data usually have high-frequency sampling rates. On the contrary, the fixed stations provide an hourly average of pollution levels. Thus, we calculated the hourly averages for mobile participatory monitoring data and low-cost sensor data to have a unified sampling rate. This step allowed us to identify air quality patterns over time and compare air quality across different times of day or days of the week.

Finally, after ensuring all the data was cleaned and relevant, we assigned the collected data to their proper cells in the map using the GPS coordinates.

4.3.3 . Data Enrichment

Data enrichment is a crucial step in air pollution analysis. This section provides a more detailed description of the data enrichment process used in our analysis.

We used the data collected from the fixed stations to create clusters of different pollution levels. We grouped the pollution levels based on the fixed stations' data, which allowed us to better understand the spatial distribution of pollution levels over time. This clustering was done using unsupervised machine-learning techniques such as k-means. For each timestamp, we used the vector of air pollution levels from all available fixed stations as the input to the clustering model. The output of this model is the different clusters, where each cluster represents different periods with the same pollution conditions.

After clustering the fixed-station pollution maps, MPM data are merged with the cluster that matches their acquisition time and propagated to the whole validity periods in that cluster. This increases the enrichment power of the MPM data while maintaining their relevance. The output of this step is to augment the data spatial coverage in the same way in each cluster. The resulting coverage can be different from one cluster to another.

We can then integrate this data with the air quality data using timestamp and location as the common variables. This provides valuable insights into the factors contributing to air pollution and informs the development of effective air quality management strategies.

4.3.4 . Air Pollution Estimation

Air pollution estimation estimates the concentration of pollutants in the air at a given location and time. This section discusses interpolation using traditional and deep learning methods (CNN-LSTM). We are using sensory data, and the estimation is conducted on top of pollution maps generated from fixed stations and enriched using opportunistic MPM.

This work proposes an approach to estimate air pollution using sensory data only. We use the enriched maps created in the previous step **??** to do this. We apply three methods for interpolation: Inverse distance weighting (IDW), ordinary

kriging, and CNN-LSTM to expand the spatial and temporal coverage.

IDW is a simple and commonly used method for interpolation. It works by assigning a weight to each observation based on its distance to the location is estimated. The weights are then used to calculate the estimated value at the location of interest. The closer the observation is to the location of interest, the higher the weight assigned to that observation.

Ordinary kriging is another interpolation method that considers the spatial autocorrelation of the data. This method assumes that the spatial correlation between the observations decreases with increasing distance between them. It uses this information to estimate the value at the location of interest based on the values of the neighboring observations.

CNN-LSTM is a more complex method that combines convolutional neural networks (CNN) and long short-term memory (LSTM) networks. CNNs are commonly used for image processing, while LSTMs are used for sequence modeling. In this case, we represent each cell by the n nearest stations and their distances. Figure 4.5 describes the architecture of our CNN-LSTM model. The CNN-LSTM network is then trained to learn the spatiotemporal patterns in the data and estimate the value at the location of interest. In this approach, each cell is represented by the n nearest stations and their distances. The model's output is the air pollution level at the specified cell.

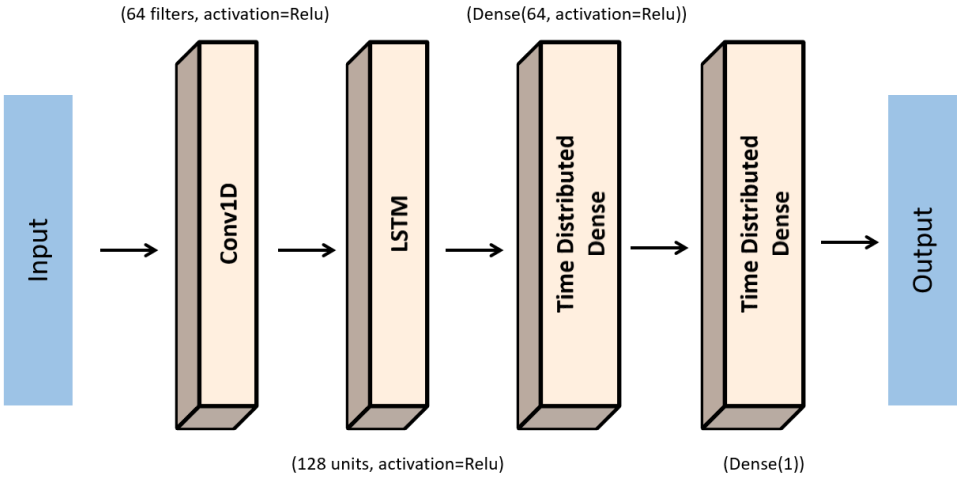


Figure 4.5: CNN-LSTM Architecture

4.4 . Experiments and Results

In order to validate our approach, we conducted experiments on two different datasets. The first dataset was collected in Versailles, France, and the second was collected from Chicago, USA. Our approach was applied in all experiments

equally, starting with preprocessing and data preparation steps, enriching, and finally, estimation.

Our work aimed to determine the optimal number of clusters (K) in the K-means algorithm for our experiments. We evaluated several commonly used methods to determine K, including the Elbow method, Calinski-Harabasz Index, Davies Bouldin Index, and Silhouette Score.

The elbow method is commonly used for selecting K in K-means clustering. This method involves plotting the within-cluster sum of squares (WCSS) against the number of clusters and selecting the K value at which the decrease rate in WCSS slows down, resulting in an "elbow" shape. The optimal value of K is often chosen at the "elbow" point, where the decrease in WCSS slows down.

The Calinski-Harabasz Index is another method for selecting K in K-means clustering. This method calculates the ratio of the between-cluster variance to the within-cluster variance for each potential value of K. The optimal value of K is often chosen to maximize this ratio.

The Davies Bouldin Index is a measure of cluster separation and compactness. This method calculates the average similarity between each cluster and its most similar cluster and chooses the value of K that minimizes this average similarity.

Finally, the Silhouette Score measures how well each data point fits into its assigned cluster. This method calculates a score for each data point based on the distance between the data point and other data points in its cluster and the distance between it and neighboring clusters. The optimal value of K is often chosen to maximize the average Silhouette Score across all data points.

After applying the mentioned methods to our data, we chose the best K value determined by most methods as our experiment cluster numbers. As for the parameter settings in spatial interpolation methods, that is, the power of distance weight and the variogram, we found through experimentation that linear distance weighting where p = 1 for IDW and the linear variogram for Ordinary Kriging performs best in terms of mean absolute error and mean squared error. Therefore, we applied them in the following experiments.

4.4.1 . Versailles Experiment

The experiment was carried out on real-life data collected in Versailles City as described in 4.3.1.

Firstly, we loaded data from all available stations, precisely the PM2.5 dimension. Secondly, we removed all missing values and kept only records with measurements from all available stations. Finally, we normalized the data using min-max normalization. Once the data was preprocessed and prepared, we utilized K-means clustering to partition it into distinct clusters. Using the different approaches for choosing the best K for the clustering, we set $K = 3$ in our experiment, forming 3 clusters. Figure 4.6 shows the mean of each station per the three clusters. The records with low pollution levels correspond to cluster 0, those with medium pollution levels were grouped in cluster 2, and those with high pollution levels fall in

cluster 1.

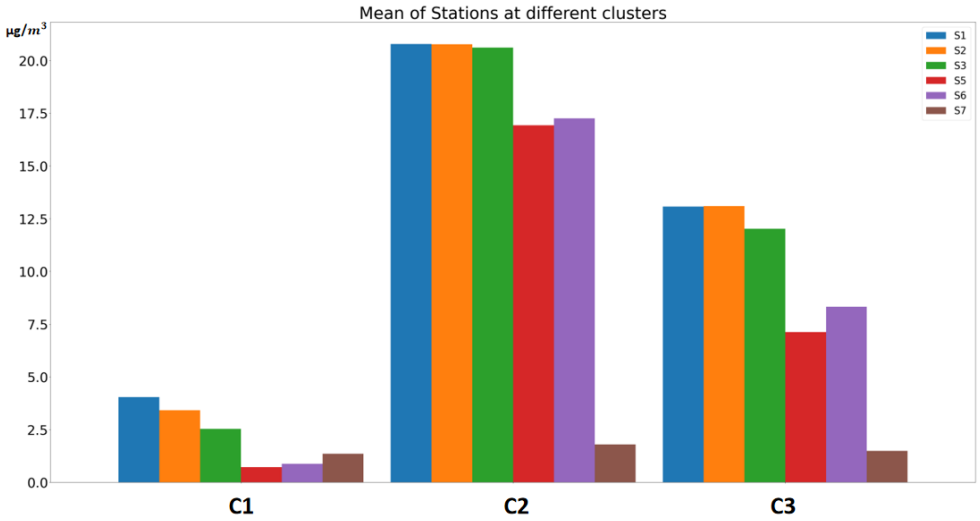


Figure 4.6: Mean of fixed stations per clusters - Versailles

Then, we split the map into two granularities to enable a more detailed analysis of our data. The first granularity was set to *1km x 1km*, while the second is *500m x 500m*. The stations were distributed over five cells using the 1km x 1km granularity. While with 500m x 500m granularity, they were spread over six cells. We preprocess and prepare the MPM data and assign it to the proper cells.

For CNN-LSTM, we used a feature vector that included the values of the three nearest neighbors having stations and the distance between the current cell and the nearest neighbors having stations. Specifically, for each cell in the map, we calculated the distance to the nearest neighbors with stations and included their corresponding values in the feature vector.

We use leave-one-out validation (cell containing fixed station "ground truth"), where we try to interpolate the cell's value having the fixed stations, as it is considered the ground truth. Mean absolute error (MAE) and root mean squared error (RMSE) are used as metrics for validation.

The distribution of the fixed stations allows experimentation for the comparison between using only fixed stations' maps or using the enriched maps based on our approach. The first part reports the results using only fixed station measures; the second part presents the results of merging both data types based on the proposed approach.

Figure 4.7 shows the plots of a sample from each cluster at the 1km x 1km granularity while using only fixed station measures to generate the pollution maps. While figure 4.8 shows the plots at a finer granularity (500m x 500m).

Table 4.2 reports the results of MAE and RMSE for the interpolation using only fixed stations at different granularities. Figures 4.9 and 4.10 show the plots of

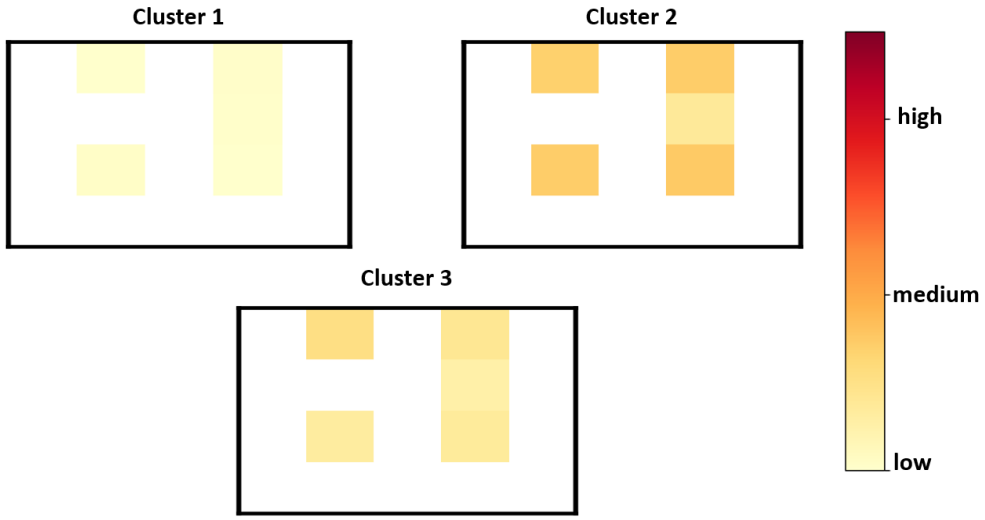


Figure 4.7: Fixed Stations' Maps 1km x 1km Granularity - Versailles

Figure 4.8: Fixed Stations' Maps 500m x 500m Granularity - Versailles

applying IDW and Ordinary Kriging on random samples in the dataset for 1km x 1km and 500m x 500m granularity respectively. On the other side, figures 4.11 and 4.12 plots the estimation using the CNN-LSTM method for the two granularities 1km x 1km and 500m x 500m respectively. The plots correspond to different samples from different clusters.

After enriching the pollution maps with the opportunistic MPM data following the proposed approach, we repeated the same experiments. Figure 4.13 shows the plots of a sample from each cluster at the 1km x 1km granularity after enriching the pollution maps with the opportunistic MPM data. While figure 4.14 shows the

	1km x 1km		*500m x 500m*	
	MAE	RMSE	MAE	RMSE
IDW	2.95	4.34	6.45	9.30
Ordinary Kriging	2.85	4.22	6.20	9.09
CNN-LSTM	2.95	3.75	6.36	9.41

Table 4.2: MAE and RMSE (Fixed Stations) - Versailles

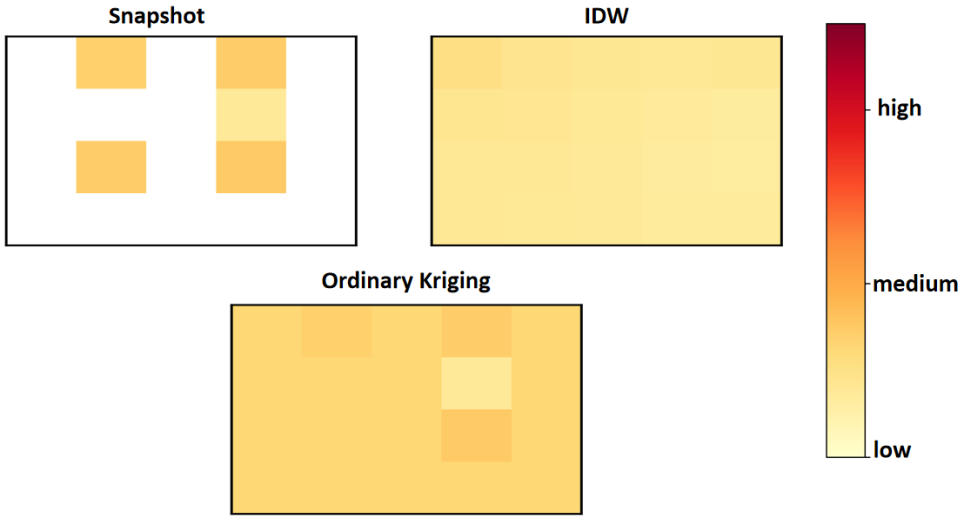


Figure 4.9: IDW and Ordinary Kriging 1km x 1km (Fixed Stations) - Versailles (Cluster 2)

plots at a finer granularity (500m x 500m).

Table 4.3 reports the results of MAE and RMSE for the interpolation using enriched maps at different granularities. Figures 4.15 and 4.16 show the plots of applying IDW and Ordinary Kriging on random samples in the dataset for 1km x 1km and 500m x 500m granularity respectively. On the other side, figures 4.17 and 4.18 plots the estimation using the CNN-LSTM method for the two granularities 1km x 1km and 500m x 500m respectively. The plots correspond to different samples from different clusters.

4.4.2 . Chicago Experiments

In order to validate our approach, we applied our methodology to data collected from open datasets such as OpenAQ and Aircasting as described in 5.4.1 section.

This experiment has two types of opportunistic sensing data besides the fixed station measures. We have low-cost fixed sensors that provide measures at a specific place but only sometimes provide measures. In addition, we have the opportunistic MPM data as in the previous experiment.

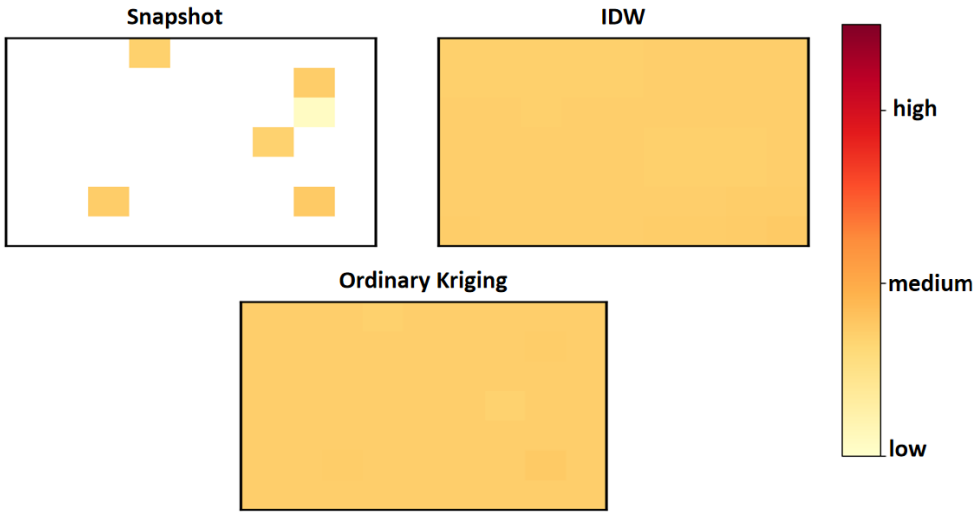


Figure 4.10: IDW and Ordinary Kriging 500m x 500m (Fixed Stations) - Versailles (Cluster 2)

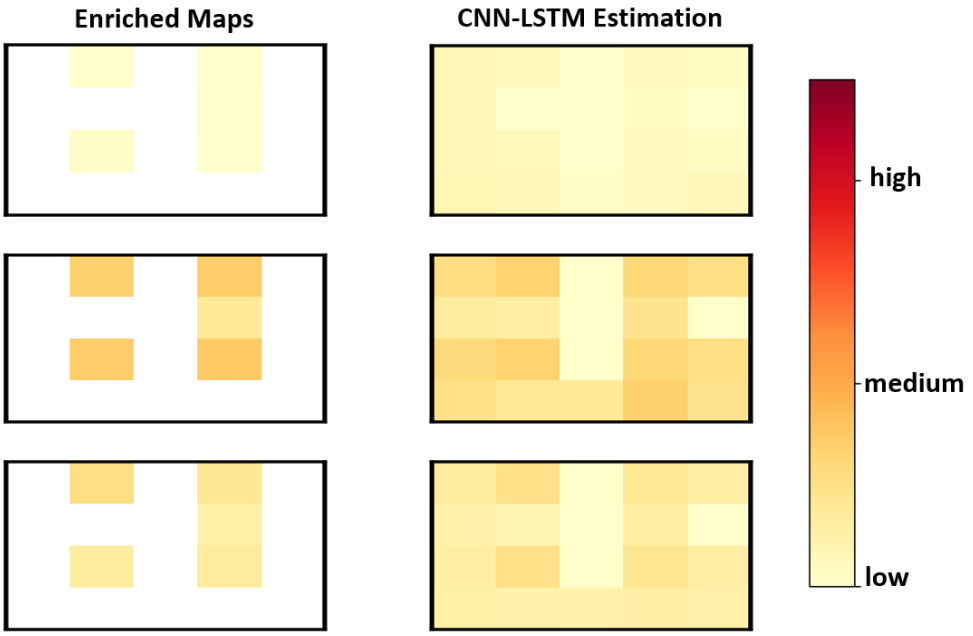


Figure 4.11: CNN-LSTM 1km x 1km (Fixed Stations) - Versailles

Once the data was preprocessed and prepared, we utilized K-means clustering to partition it into distinct clusters. For each record, three measures were associated with the three reference grade stations in the area of interest. Using the different approaches for choosing the best K for the clustering, we set $K = 4$ in our experiment, forming 4 clusters. Figure 4.19 shows the mean of each station per

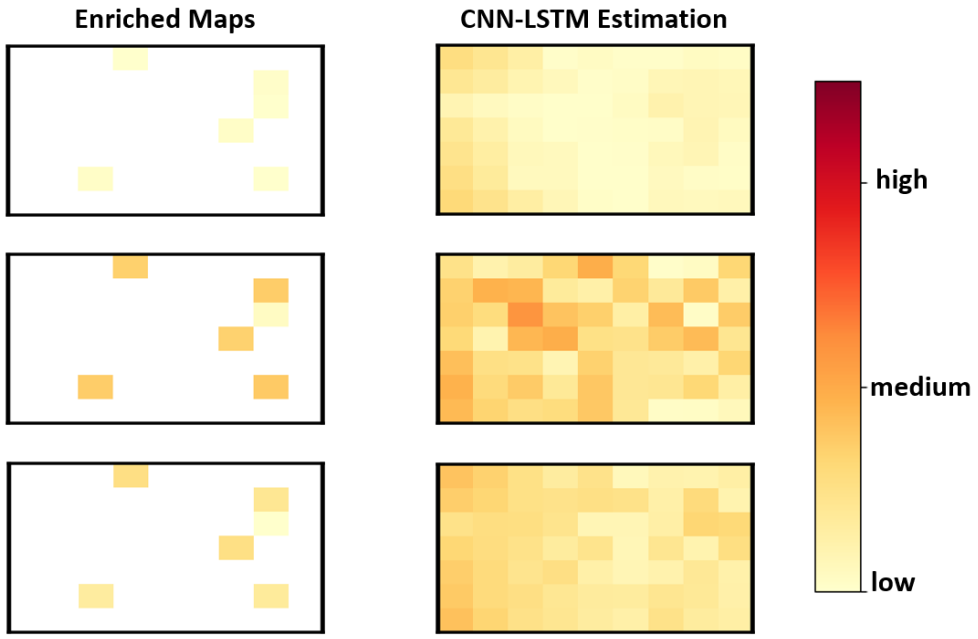


Figure 4.12: CNN-LSTM 500m x 500m (Fixed Stations) - Versailles

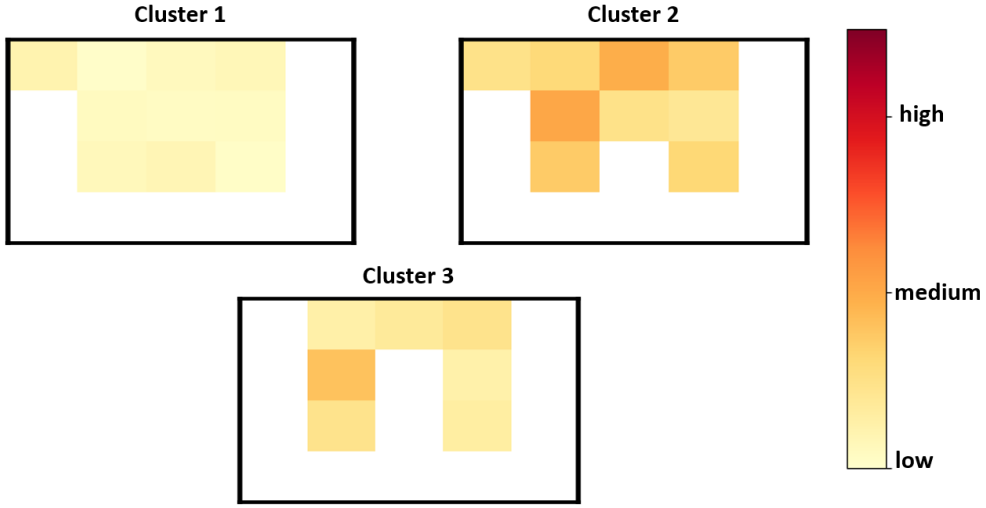


Figure 4.13: Enriched Maps 1km x 1km Granularity - Versailles

the four clusters. Records with low pollution levels correspond to cluster 2, records between low and medium pollution levels are grouped in cluster 1, records with medium pollution levels correspond to cluster 3, and records with high pollution levels fall in cluster 0.

We applied the same settings as the previous experiment for the opportunistic data. We first selected the PM2.5 dimension from the preprocessed data. Then,

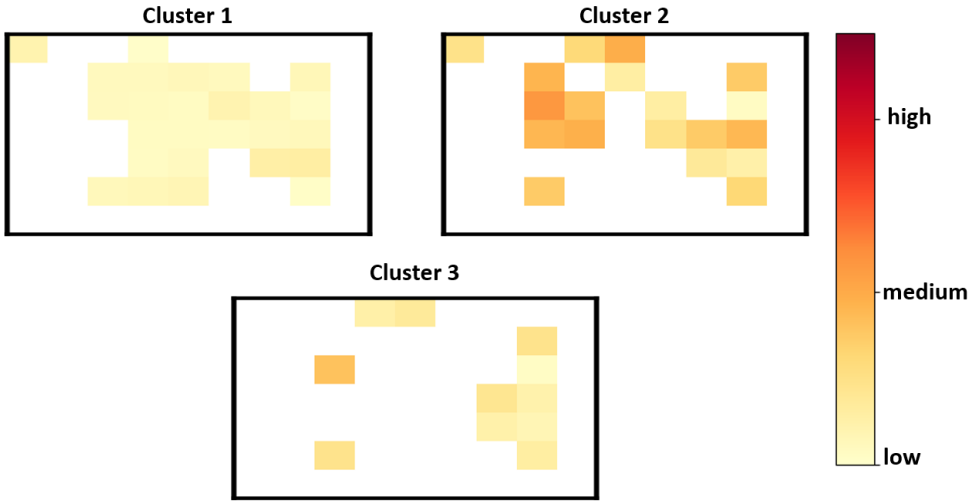


Figure 4.14: Enriched Maps 500m x 500m Granularity - Versailles

	1km x 1km		*500m x 500m*	
	MAE	RMSE	MAE	RMSE
IDW	0.63	0.65	5.25	7.34
Ordinary Kriging	1.03	1.22	5.75	7.82
CNN-LSTM	0.20	0.39	3.24	5.51

Table 4.3: MAE and RMSE - Versailles

we split the map into two granularities to enable a more detailed analysis of our data. The first granularity was set to *1km x 1km*, while the second is *500m x 500m*.

Figure 4.20 shows the plots of a sample from each cluster at the 1km x 1km granularity after we enriched the fixed stations' data with MPM data. While figure 4.21 shows the plots at a finer granularity (500m x 500m).

We applied the same methods in experiments conducted in Versailles 4.4.1. Also, the same validation metrics, MAE and RMSE, apply here. The validation is also performed using leave-one-out validation.

Table 4.4 reports the results of MAE and RMSE for the different splits. The results show a significant improvement in using CNN-LSTM to estimate pollution levels. Figures 4.22 and 4.23 show the plots of applying IDW and Ordinary Kriging on random samples in the dataset for 1km x 1km and 500m x 500m granularity respectively. On the other side, figures 4.24 and 4.25 plots the estimation using the CNN-LSTM method for the two granularities 1km x 1km and 500m x 500m respectively. The plots correspond to different samples from different clusters.

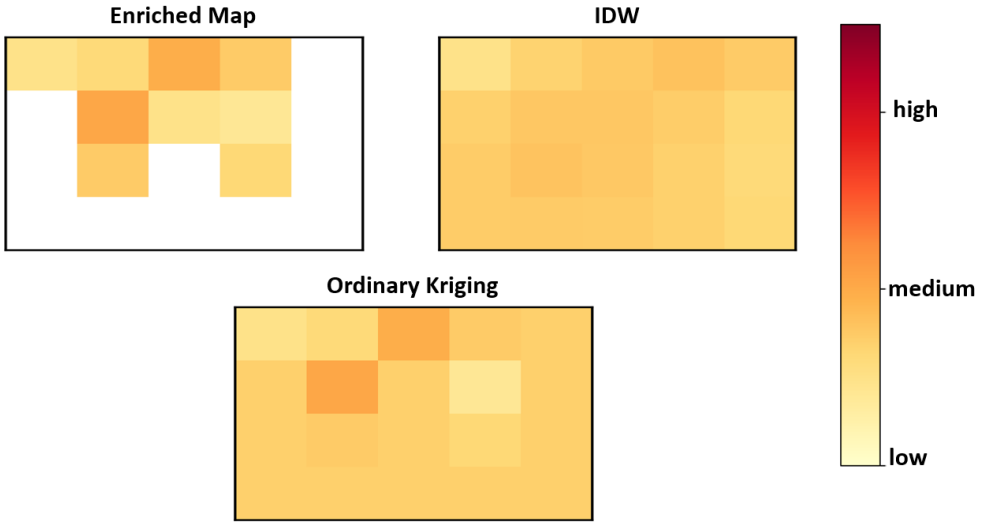


Figure 4.15: IDW and Ordinary Kriging 1km x 1km - Versailles (Cluster 2)

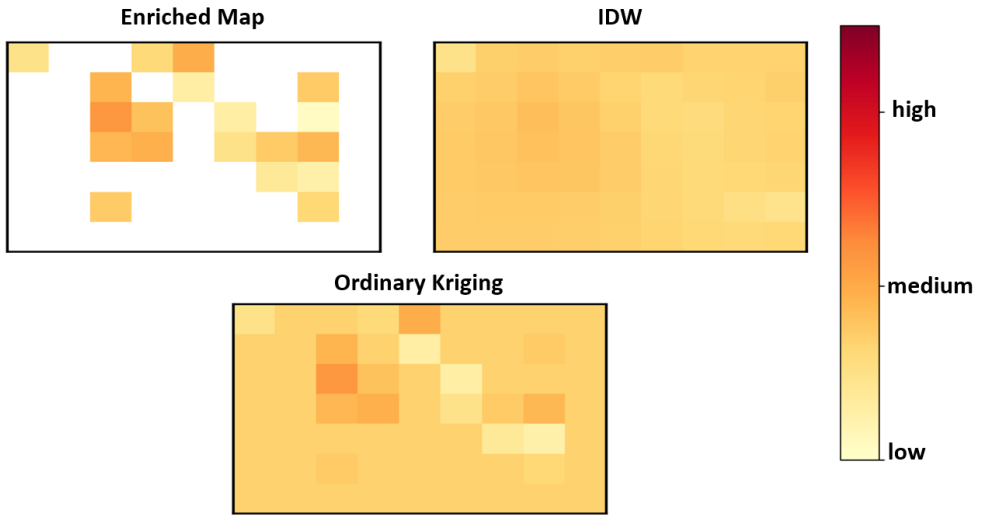


Figure 4.16: IDW and Ordinary Kriging 500m x 500m - Versailles (Cluster 2)

4.5 . Discussion

Our primary goal in this study is to expand the spatiotemporal coverage. We are enhancing air monitoring fixed stations by incorporating mobile sensor data collected from the public. Previous projects have typically conducted targeted mobile sensing campaigns in specific areas or along particular paths. In contrast, our study utilizes opportunistic MPM data to supplement fixed station data.

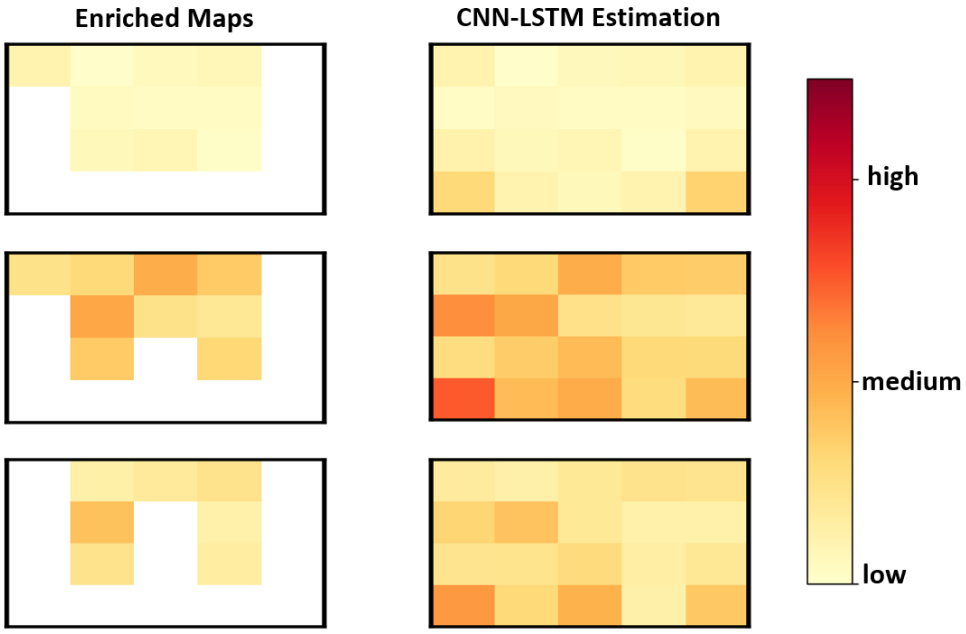


Figure 4.17: CNN-LSTM 1km x 1km - Versailles

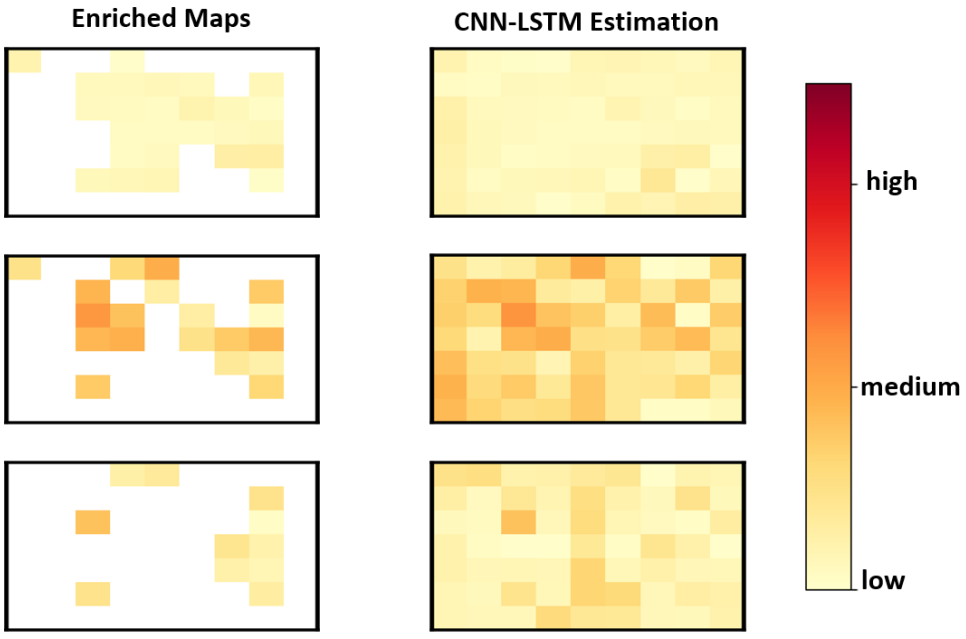


Figure 4.18: CNN-LSTM 500m x 500m - Versailles

Our initial challenge was to figure out how to combine opportunistic MPM data with fixed-station data for estimating air pollution. We hypothesized that periods of air pollution where fixed stations' measurements fall within the same cluster could share similar MPM data. We clustered the fixed stations' data to

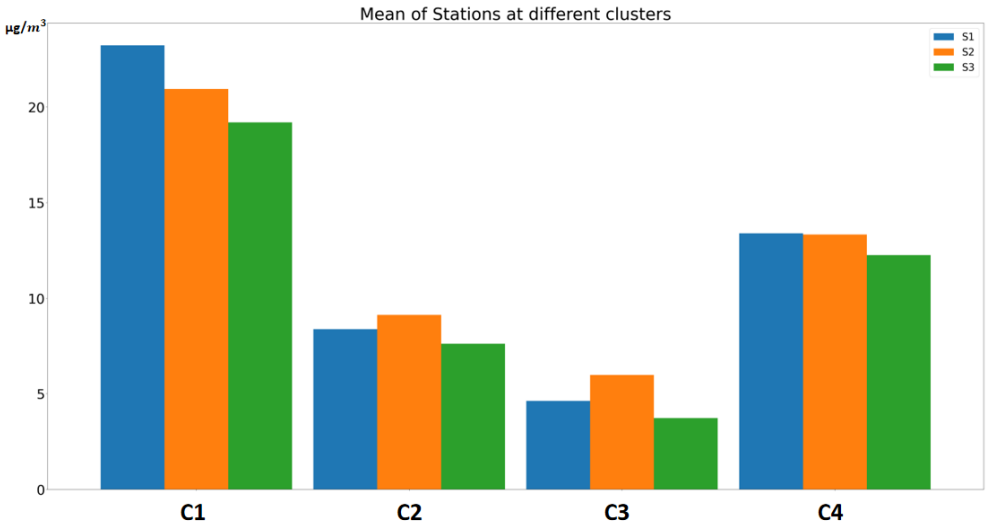


Figure 4.19: Mean of fixed stations per clusters - Chicago

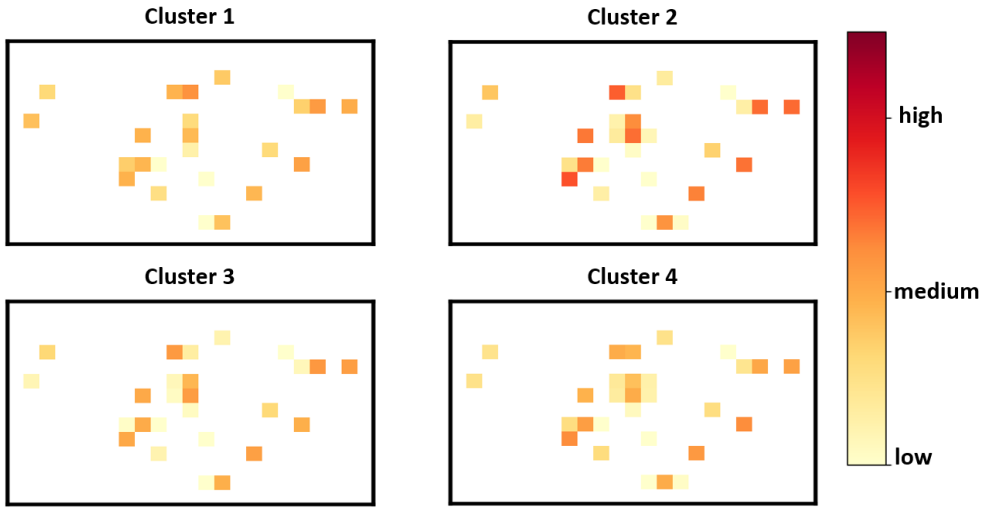


Figure 4.20: Enriched Maps 1km x 1km Granularity - Chicago

test our hypothesis and merged them with MPM data. Our experiments confirmed the validity of our hypothesis, and this methodology could improve the accuracy of fixed station data.

This section will analyze and interpret the experiments' results of estimating air pollution using fixed and opportunistic Mobile Participatory Monitoring (MPM) data. The experiments aimed to evaluate the effectiveness of the proposed approach of enriching the fixed stations generated air pollution maps with opportunistic MPM data. We used two basic interpolation techniques, Inverse Distance Weighting (IDW) and Ordinary Kriging; we then utilized CNN-LSTM to

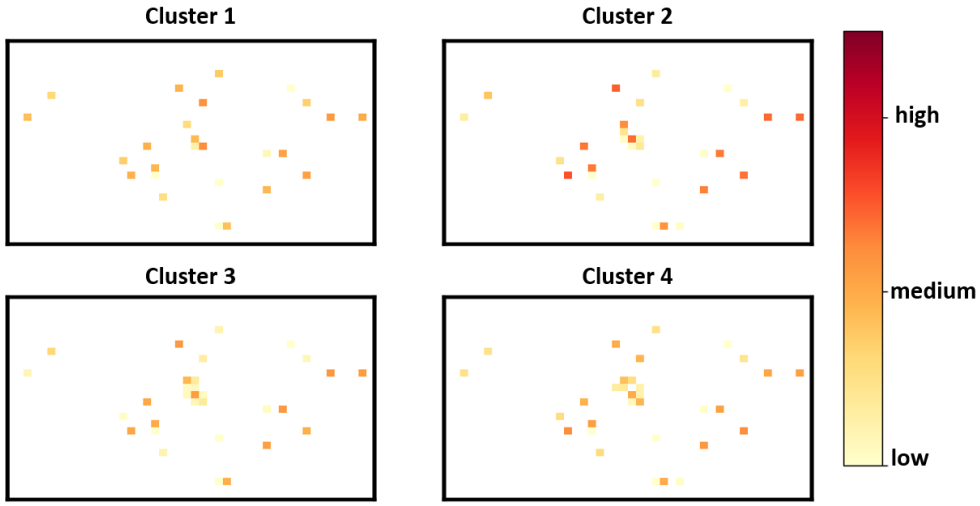


Figure 4.21: Enriched Maps 500m x 500m Granularity - Chicago

	1km x 1km		*500m x 500m*	
	MAE	RMSE	MAE	RMSE
IDW	7.381	8.211	6.307	6.893
Ordinary Kriging	7.979	8.651	7.389	7.993
CNN-LSTM	0.793	0.917	0.804	1.017

Table 4.4: MAE and RMSE - Chicago

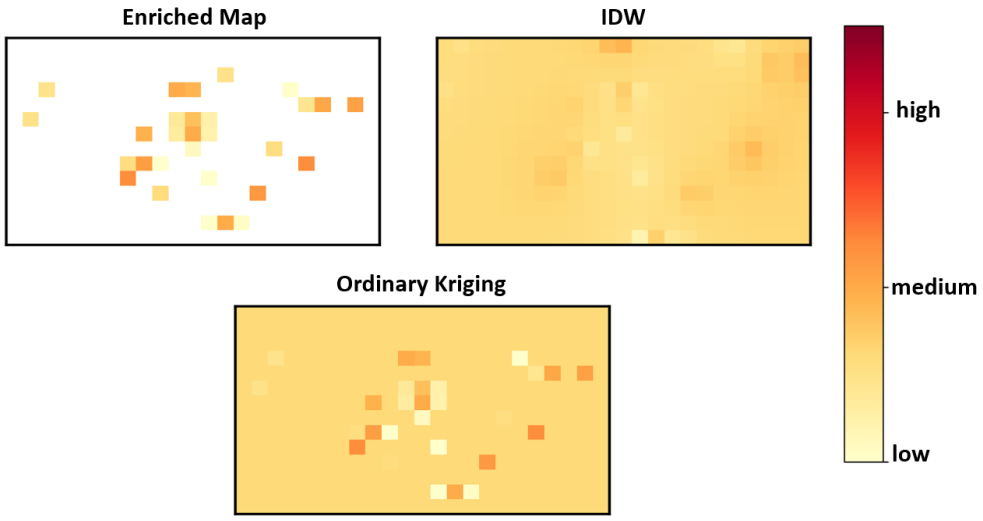


Figure 4.22: IDW and Ordinary Kriging 1km x 1km - Chicago (Cluster 4)

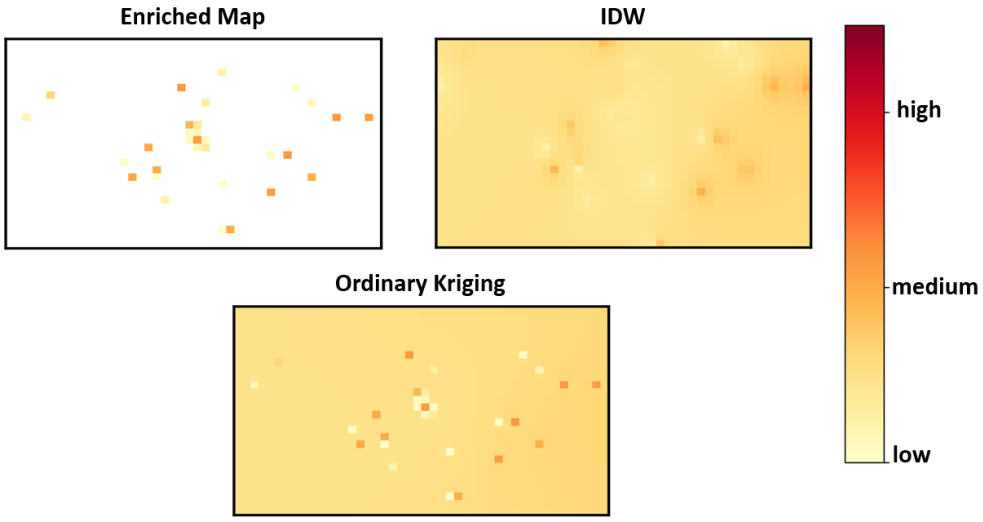


Figure 4.23: IDW and Ordinary Kriging 500m x 500m - Chicago (Cluster 4)

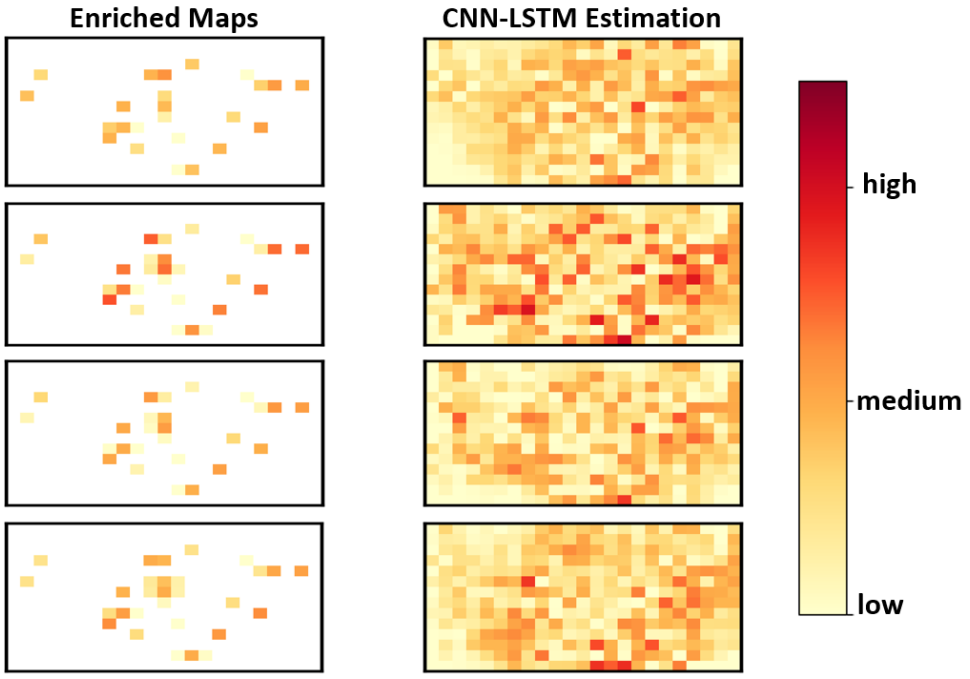


Figure 4.24: CNN-LSTM 1km x 1km - Chicago

enhance the estimation accuracy.

In **Versailles experiment**, we compared the results using fixed stations alone for estimation versus the combination of fixed stations and opportunistic MPM data following the proposed approach. The results interpreted in the experiment section

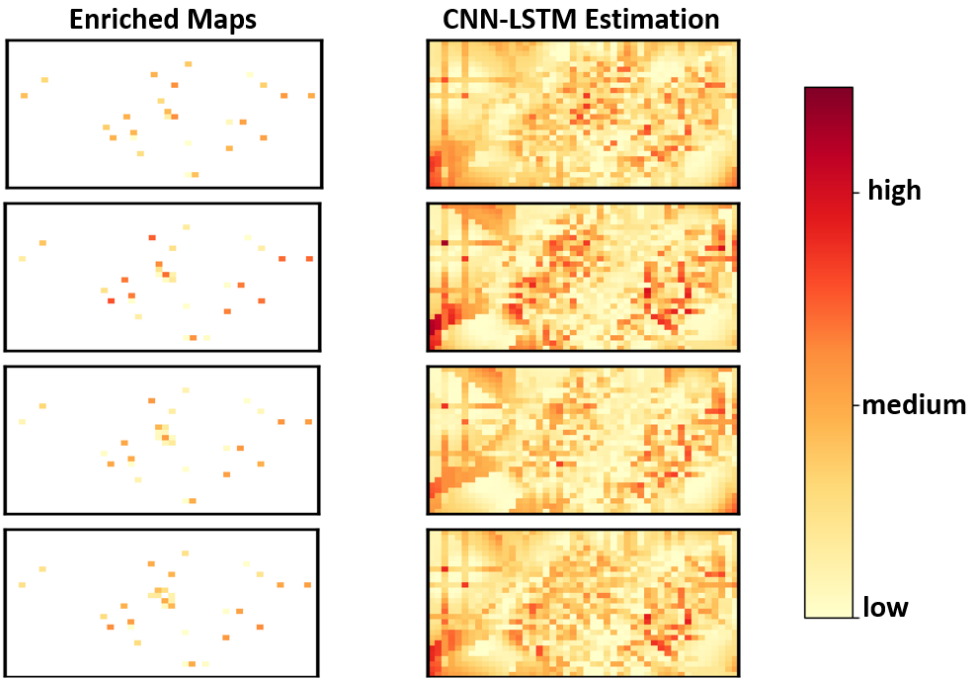


Figure 4.25: CNN-LSTM 500m x 500m - Chicago

under Versailles experiments 4.4.1 show the efficiency of our proposed approach. When using the enriched air pollution maps to estimate air pollution, we had better results in terms of MAE and RMSE using all the interpolation techniques. Table 4.2 and table 4.3 summarizes the error of air pollution estimation using different techniques in Versailles city while using only fixed stations data and while using the combination of fixed stations and opportunistic MPM data respectively.

In table 4.2, we notice that the MAE and RMSE for all the methods are similar. Even using advanced techniques such as CNN-LSTM, we still had a high MAE and RMSE. Hence, the model couldn't learn the correct patterns while using only fixed stations. On the other side, in table 4.3, MAE and RMSE decreased significantly for all the used methods, and CNN-LSTM has shown the best results among the used techniques.

Moreover, we used data from open datasets such as OpenAQ and Aircasting to better validate our approach **Chicago experiment**. We ran our approach on top of the available data in the area of interest. The reported results in section 4.4.2 illustrate the effectiveness of our approach in better-estimating air pollution when following the proposed approach. Results in table 4.4 show acceptable results in terms of MAE and RMSE, especially for the CNN-LSTM model.

Table 4.5 summarizes the Mobile Participatory Monitoring data statistics in the two experiments for 1KM x 1KM granularity. Min MPM and Max MPM columns refer to the minimum and maximum number of sensors available in a time period. The percentiles show that we have a very low contribution of MPM data to the

	Min MPM	Max MPM	50 - Percentile MPM	75 - Percentile MPM	90 - Percentile MPM	Coverage Before Enrichment	Coverage After Enrichment
Versailles Experiment	0	6	1	2	2	41.6 %	58.3 - 83.3 %
Chicago Experiment	0	24	18	19	19	1.39 %	8.3 - 10.1 %

Table 4.5: Monitoring Coverage Before and After Enrichment

map. Coverage Before Enrichment column shows the percentage of the monitored area. Coverage After Enrichment shows the percentage of the monitored area after we applied our enrichment process. Each cluster has a percentage as the number of MPM sensors varies between clusters. In *Versailles Experiment*, the coverage after enrichment is 83.3%, 75%, and 58.3% for clusters one, two, and three, respectively. While for *Chicago Experiment* 8.3%, 9.7%, 9%, 10.1% are the coverage percentages in clusters one, two, three, and four, respectively.

Moreover, figure 4.26 plots the map of the monitoring coverage before and after enrichment for one time period of the maps in Versailles. Black squares represent the original averaged MPM data collected at that time. Red squares denote the averaged values of the additional MPM data corresponding to one cluster (here cluster 1). Therefore, the enriched map is obtained by the union of the black and red dots. It's noteworthy that if we have MPM and fixed stations data in one cell we take the measurements of the fixed stations only as they are more precise. For instance, the green pin shows a fixed sensor which data prevail all MPM data in the same grid cell. The map coverage before enrichment in this example was around 35%, reaching around 80% after enrichment. It is clear that using our approach, we can expand the spatiotemporal coverage. The output of the enrichment phase is an enriched map with better observations. Thus, we can have better estimation when applying interpolation.

The experiments' results indicate that the proposed method performs well even when using only sensory data. This finding is valuable, suggesting that the approach can be utilized in areas with limited access to extensive supplementary data.

IDW and Ordinary Kriging may have exhibited suboptimal results when the density of observations varies significantly across various sites. When certain places have many measurements while others have none, IDW and Ordinary Kriging tend to over-smooth the interpolated values, resulting in an unsatisfactory depiction of local variations and sudden changes in the data. Advanced interpolation techniques, such as machine and deep learning-based approaches that account for the spatial characteristics of the data and the underlying processes, may yield more accurate and reliable results. For example, the CNN model can handle the spatial characteristics of the data. As for the temporal characteristics, recurrent neural networks such as LSTM can perform well. That's why we have used the CNN-LSTM interpolation to handle both spatial and temporal characteristics of the data

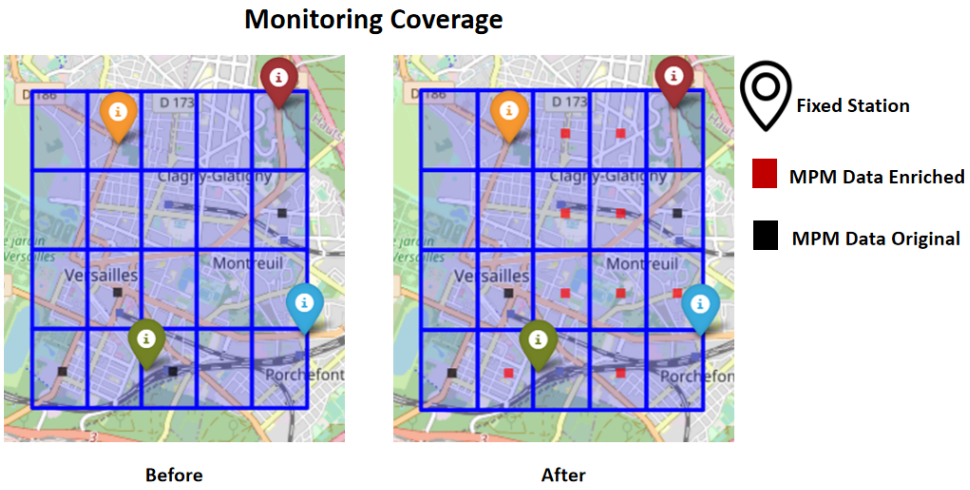


Figure 4.26: Monitoring Coverage Before and After Enrichment

and have a better estimation of pollution. However, even while using advanced techniques, we still have some concerns. Indeed, in some areas where no observations are found, the model tends to overestimate values, such as in the left corner of the plots in figure 4.25. This suggests to investigate the explainability of these models. A possible solution to work around the estimation near the border is to retain only the results within the convex hull of the dataset.

Moreover, there are still opportunities for further improvement. Augmenting the approach with additional features can result in better estimation. We believe that enrichment with air pollution features such as land use, traffic, and meteorological data has significant improvements. These features can give more insights and provide valuable context into the factors influencing air pollution levels. Integrating such data could refine the accuracy of the estimation and provide more comprehensive predictions.

The experiments show that the proposed method successfully estimates air pollution measurements, especially when incorporating opportunistic MPM data and leveraging deep learning models like CNN-LSTM. The technique shows promise for further improvement in air quality estimate with potential advancements by adding new significant features.

4.6 . Conclusion

Air pollution monitoring using fixed stations and low-cost and mobile sensors has been a trendy topic over the last few years. Air quality is a permanent concern in urban areas, as improving the air quality index can help face urbanization challenges. Several studies tried to interpolate pollution measures from fixed stations, mobile sensors, or a combination. They use different methods and may require

additional features.

In this study, we present our approach, which combines fixed station data with mobile participatory sensing (MPM) data collected by individuals during their daily activities rather than at specific outdoor locations. This type of data collection presents a challenge, as only 10% of the time is spent outdoors, resulting in a scarcity of MPM data. The mobile sensing data, in our case, have different characteristics than the data used in previous approaches. MPM data is not collected initially to monitor air pollution outdoors and in specific places. We were interested in using the opportunistic MPM data in enriching fixed stations' data to better estimate air pollution in uncovered spots.

The primary objective was to leverage the opportunistic MPM data and utilize it to enhance the fixed stations' measures to generate enriched air pollution maps to estimate air pollution better. We applied clustering based on the fixed station measurements to group periods with similar pollution maps to achieve this. We hypothesized that these clusters could provide a representation of the overall air pollution conditions within specific periods. After that, we merge the relevant opportunistic MPM data with fixed station data. Select the MPM data corresponding to the identified periods within each cluster. This process results in enriched air pollution maps for the different clusters providing a more comprehensive and detailed picture of pollution levels across the study area. Then the final step is to utilize the enriched maps in the air pollution estimation process. We leveraged different interpolation techniques for this process. Inverse Distance Weighting (IDW), Ordinary kriging, and CNN-LSTM are used for the estimation task.

Our approach has been validated using three real-world datasets from Versailles City and Chicago. The reported results show our proposed approach's applicability and efficiency in estimating air pollution in unmonitored spots. For now, we have validated the feasibility of our approach using sensory data. However, we believe involving more air pollution-related features such as land use, meteorological, and traffic features can significantly improve the estimation performance, especially when utilizing deep learning models such as the CNN-LSTM model.

Chapter 5 - FLAIR - Fine-grained Geolocation of Tweets

Contents

5.1 . Introduction

The primary focus of this chapter lies on our third objective throughout this thesis, which serves as the central pursuit of our work in this chapter. The main goal is to identify tweets' locations to better use them in applications such as local event detection. By dedicating this chapter to the mentioned objective, we aim to effectively address the research question **R3**, which has been formulated to guide our investigation. This chapter presents our third contribution, denoted as **C3**, designed to provide unique and valuable insights that expand the existing knowledge base in our field. Within this chapter, our work focuses on developing a hybrid End-to-End framework that uses existing methods to predict city-level, then uses a Fine-grained LocAtIon pRediction (FLAIR) algorithm to predict the fine granularity.

This chapter is organized as follows: The following section presents background highlights, the motivation, and the goal to present a tweet's location identification approach. Section 5.2 presents an overview of our fine location prediction model at a specific city, then details the procedure. In 5.3, we present the methodology and the detailed explanation of our end-to-end framework combing existing approaches for coarse granularity location prediction and $FLAIR$ for fine granularity prediction. Section 5.4 details the implementation part, and section 5.5 states the results of applying our approach to a real-world dataset. Section 5.6 is a discussion about our findings. Finally, the last section 5.7 restates our objectives and summarizes the work.

5.1.1 . Background

Twitter is one of the most popular social media platforms. It allows for establishing non-mutual friendships. One type of social media content that has gained significant attention in recent years is geotagged tweets. Geotagged tweets contain location information, allowing users to share their location. Twitter has its API, the *Twitter API*[1] that allows researchers and developers to crawl and analyze tweets. Tweets contain, in addition to the textual posts, links to media posted and other useful metadata for analysis.

With the popularity of GPS-enabled mobile devices, Twitter users can share their location voluntarily when tweeting. This location information can be used in various ways, including monitoring real-life activities. Furthermore, they can mention points of interest (POIs) in their tweets, such as names of restaurants, cities, tourist spots, etc.

Many applications can benefit from geotagged text information, such as natural disaster and crime detection [133, 78], health care management [158], marketing recommendation systems [11], and event detection systems [144, 145], to name a few.

[1] https://developer.twitter.com/en/docs/twitter-api

Researchers have used geotagged tweets to study the spread of diseases and monitor outbreaks. By analyzing the location information in tweets, researchers can track the movement of individuals and identify areas where disease transmission is likely to occur. This information can help public health officials respond to outbreaks more effectively. Geotagged tweets have also been used to monitor traffic and transportation. By analyzing the location information in tweets, researchers can identify areas with high traffic congestion and monitor the movement of vehicles. In addition to these uses, geotagged tweets have been used to monitor real-time events such as protests and natural disasters. By analyzing the location information in tweets, researchers can track the movement of individuals and identify areas where events are occurring.

Overall, geotagged tweets have become an essential tool for monitoring real-life activities. Researchers can gain valuable insights into various areas by analyzing the location information in tweets. As social media continues to grow and evolve, geotagged tweets will likely become an even more critical source of information for researchers and analysts.

While geotagged tweets have proven to be a valuable source of information for monitoring real-life activities, their availability is often limited. This is because only a small percentage of tweets are geotagged. Unfortunately, only 1 to 3 % of tweets contain geotagging information [126], which makes analysis a complex task in the absence of such data. The rarity of available geotagged tweets can make it difficult for researchers to study the location of tweets. However, geolocation prediction approaches offer a way to overcome this limitation.

As a result, researchers have turned to geolocation prediction approaches as a way to overcome the rarity of available geotagged tweets. Geolocation prediction approaches use various techniques to infer a tweet's location, even if it is not geotagged. These techniques include using the text of the tweet, the user's profile information, and social network connections.

5.1.2 . Problem Statement

The tweet location prediction problem gained the interest of researchers over the last few years. Researchers working in such fields have tried different approaches to geotag tweets. The evolution of text mining techniques and natural language processing methods helped in improving text localization and geotagging. Many research initiatives have addressed the problem of location prediction from tweets. In [152] location prediction problem is categorized either as:

- *Home location prediction* refers to Twitter users' long-term residential addresses.
- *Tweet location prediction*, which means the place where a tweet is posted
- *Mentioned Location prediction* as users may mention the names of some locations in tweet contents.

Some researchers have proposed convolutional deep neural networks [89] to learn Location Indicative Words (LIW) from word embedding. Others proposed a text network [81] consisting of Bidirectional LSTM to find out the geolocation of text. In addition, some proposed a Unicode network [62] with character encoding to find the location of text in any language. While in [107], they have used kernel densities to estimate text location. Moreover, the location prediction problem is considered a classification problem when the aim is to assign a discrete label, *e.g.,* a cell in the grid map, or a regression problem when the aim is to predict the geolocation coordinates. Those research initiatives have tried to reduce the distance error between actual and predicted locations. They have reported good results, but still, their accuracy is reported at wide ranges such as 5 km, 10 km, and 20 km, which might be sufficient for identifying the city of a tweet accurately [58, 62], but fails to geolocate tweets/text more precisely.

While predicting a text's country or city location becomes easier nowadays with the proposed methods, identifying the location at a finer granularity (i.e., a higher spatial resolution) is still hard. In our previous work, $FLAIR$ [4], a "Fine-grained LocAtIon pRediction" algorithm of non-geotagged tweets based on multi-view learning and natural language processing, we could predict the fine-grained location of a tweet given its city label. This, however, assumes the city to be known in advance.

In this work, we address the limitations of both approaches by combining them. We propose an approach from coarse to fine-grained location prediction using the combination of existing approaches and $FLAIR$. In other words, we have two fold prediction approach. The first fold is predicting the city of a tweet. The second fold is to predict the precise location within the predicted city.

Several studies have used spatial models either by finding the landmarks to geotag the tweets or learning some location indicative words (LIW) such as in [89, 107, 24, 127]. On the other hand some researchers have used textual models to derive the location such as in [58, 62, 103, 74]. In our work, we tried to combine both assumptions to increase the accuracy and better geotag the tweets.

5.1.3 . Summary of Related Work

Considering the high-importance insights one can extract from geotagged social media data, we are witnessing an increasing interest from academic and industrial parties in the problem of tweet location prediction. Many existing strategies have been investigated in the study on location prediction from tweets and social media content. This research topic is an active topic that concerns different fields, as many applications require such information.

We are interested in the tweet location prediction in this work, which aims to identify where a tweet/text was posted. The prediction task can be considered a regression problem when the output is the geo-coordinates or a classification problem when the tweet/text is assigned to a pre-defined label.

Several approaches have been investigated in the literature discussed in sec-

tion 2.4. Those approaches tackle the location prediction problem at different granularities, the coarse [74, 103, 62] and fine granularity [89, 107, 58]. However, the reported results have high distance errors between the anticipated and real locations. Those approaches main success in predicting the city and country level of tweets while failing at predicting location at a fine granularity with high accuracy. Most of the work focuses on location-indicative words and word embedding while not considering the relation between non-geotagged tweets and other geotagged ones in the same spatial area. In order to benefit from tweets in monitoring and event detection systems, we should have their precise locations. Hence, we need a text location approach that minimizes the distance error between real and predicted locations. This chapter addresses tweet location prediction with an end-to-end framework for predicting a tweet's coarse and fine granularity based on the combination of existing approaches.

5.1.4 . Proposition and Contributions

This work presents a hybrid framework for predicting a tweet's coarse and fine granularity location. We are considering 2-fold tweet location prediction. We will adapt any existing approach for the city-level prediction as they proved to have better accuracy in coarse than fine granularities. However, we proposed $FLAIR$ for the second fold. $FLAIR$ is city dependent; thus, we will have a specific model per each city or region predicted by the city-level predictor. Our proposed end-to-end geolocation approach has the potential to provide valuable insights into real-life activities that occur at a fine-grained level.

Precisely, $FLAIR$ performs as follows. Given a grid map, we predict the grid cell where a non-geotagged tweet was posted, thus reducing our location prediction problem to a classification problem. This classification method also **considers the location-related text** by using different natural language processing tools. Thus **our goal is to reduce the distance error to the actual location as much as possible**.

Formally, we present the following sub-contributions derived from the main contribution **C3**:

- We propose a hybrid framework for tweet location prediction for coarse and fine granularities.
- We propose $FLAIR$, a novel algorithm requiring minimal features to achieve a fine-grain tweet location prediction within a pre-defined region.
- We combine two prediction models by adapting multi-view learning: the one based on POIs matching (spatial model) and the other using text similarity (textual model).
- We further optimize the spatial model relying only on the geotagged tweets and show their importance in the prediction process.

- We evaluate the existing city-level prediction approaches on a balanced dataset collected from different locations worldwide.
- We implement and evaluate $FLAIR$ on two real-life datasets collected from Twitter and compare the results with baselines and the most similar approaches in the state-of-the-art, which clearly shows the advantages of $FLAIR$ in terms of accuracy and spatial resolution.
- We highlight the added value of combining both approaches.

5.2 . $FLAIR$

$FLAIR$ is a fine-grained location prediction approach for a given region. It requires minimal features compared to existing approaches, and is built on top of two factors. The first is based on the assumption that a user talking about a certain location will likely mention it in a tweet. The second factor considered is that tweets originating from the same location may be very relevant to one another because they may be discussing the same event. Based on those aspects, our work considers spatial and textual models. For the spatial model, we have used two methods: matching the POIs with the location entities of the tweets or learning the cell-indicative words in the map. While for the textual model, we compute the text-similarity of non-geotagged tweets with other geotagged ones to identify the location. Finally, a multi-view model is adopted to combine their results and maximize the accuracy.

Our solution adopts the multi-view learning approach, considering two views to learn from. Each tweet is modeled in two different ways. We extract all possible entities in tweet text to be validated by the spatial classifier. Moreover, we calculate the text embedding of each tweet to compare its similarity with others. Then we use the stacking generalization approach to combine the results and have the final prediction.

$FLAIR$ has been implemented and validated on real-life tweets data collected within the context of GOGREEN ROUTES[2] project. The goal of this H2020 project is to guide cities in identifying and developing nature-based solutions to urbanization challenges by fostering urban mental health and well-being. Modeling people's activities and events in different areas is key to understanding the urbanization challenges. This task can be easier with geotagged social media (GTSM) data.

5.2.1 . Overview

$FLAIR$ steps are introduced in the following methodology part. Given a pre-defined and well-known region of interest, we are interested in minimizing the distance error between the anticipated and the actual locations of the tweets, as

[2]https://gogreenroutes.eu/

the aim is to utilize such predictions for event and activity detection. Predicting tweet location at the country or city level is now possible, while predicting a precise location within a specified region is still complex. We address this challenge and evaluate the trade-off between prediction accuracy and spatial resolution.

A tweet contains mainly the tweet's text, a link to media if it exists, mentions and hashtags, and metadata. Metadata attached to the tweet are informative data, such as id, the creation time of the post, geographical coordinates (if it exists), language, likes, user id, etc.

We consider a tweet T defined by its text, creation time, and location T:< X, C, L > where X denotes the text, C denotes the creation time, and L denotes the location. The overall objective is to predict L for the tweets while minimizing the distance error between L and their actual location as much as possible.

Since Twitter users usually mention the point of interest in their tweets, we will detect the mentioned locations and match them with POIs found in the region. Moreover, tweets originating from the same place are highly probable to be relevant to each other. For instance, if there is a football match, most of the tweets in the region of the stadium will be talking about the match. Depending on those facts, we are proposing a model that considers POIs mentioned in tweet text on the one hand and, on the other hand, another model that finds tweet text similarity with other geotagged tweets. We consider each classifier here as an independent view. Thus we adopt the multi-view learner to estimate the final location of a tweet.

Algorithm 2 states the steps of $FLAIR$ to predict the location of a tweet. First, we must identify an area of interest to split its map into cells given a granularity (Line 1). Those cells are considered our labels in the classification problem. We find all the points of interest (POIs) in this region to assign them to their corresponding cells in the map (Line 2). Next, we train a spatial model using the POIs collected data (Line 3). Also, we collect all the geotagged tweets within the specified region and assign labels to them based on their locations (Line 4). Then, we merge tweets within each cell based on their similarity to form clusters of events (Line 5). We exclude global events at this phase to keep only local ones (Line 6). After that, we compute each non-geotagged tweet's text similarity with the events' clusters formed in the previous step. The label of the most similar cluster will be assigned as the tweet label (Line 7). At the same time, we check if there are POIs mentioned in the tweet text. Those POIs are matched to the POIs found in the map cells, and a second label is attached to the tweet (Line 8). If a tweet neither matches any geolocated tweets cluster nor matches a POI in the grid cells, we assign it the user's location if it exists (Line 9). Finally, we use a stack generalization approach that considers both views, the spatial view (*"POIs Matching"*) and textual view (*"Sentence similarity"*), and uses their predictions to predict the final grid cell label (Line 10).

5.2.2 . Learning Steps

Algorithm 2: *FLAIR* Methodology

Input: X, C, *granularity*
Output: L

1 Split the map of the pre-defined region of interest into cells using a grid view based on the given granularity.
2 Find the PoIs in each grid cell, and assign them cell labels.
3 Train a spatial model based on POIs to predict cell numbers.
4 Collect all geotagged tweets in the region, and assign them the appropriate cell labels.
5 Merge similar tweets in the same cell to form events clusters, depending on their creation time C and their text X.
6 Rank the clusters to identify true local events.
7 Calculate text similarity between non-geotagged tweets and identified clusters.
8 Match POIs mentioned in tweets with POIs found on the cells of the grid map.
9 If the tweet's text does not match the two criteria above, assign the user home location if it exists.
10 Use a multi-view learning approach to predict the final cell number (L).

This section details our tweet location prediction approach based on a multi-view learning approach and using different natural language processing tools. Figure 5.1 shows different components of our location prediction approach. Our approach mainly depends on the text. In addition, we only use the creation time of the tweet from the metadata. We consider two views: the spatial model is considered the first view, the textual model is the second view, and the multi-view model combines the results of the previous views. The dashed arrows represent the training phase of our multi-view learning model, while the lines correspond to the classification phase. *POIs data* and *Geotagged* tweets are used to train the spatial and textual models, respectively. The spatial model and textual model are considered the first-level learners. In turn, those models will predict the tweet's location and probability. The predictions and probabilities are merged to represent the feature vector as shown in table 5.2 in the new dataset D'. Finally, a meta-learner is trained on top of D' to give the final prediction. For classification, *Non-Geotagged tweets* are the input data for the model. All first-level classifiers will validate them. The predictions and probabilities will be the input of the meta-learner, which will predict the tweet's location.

We assign labels for both types of collected data, Twitter data and POIs data. Using a grid view, we have split the map (*i.e.* the area-of-interest or the bounding box) into cells. A number identifies each cell in the grid view, and this number

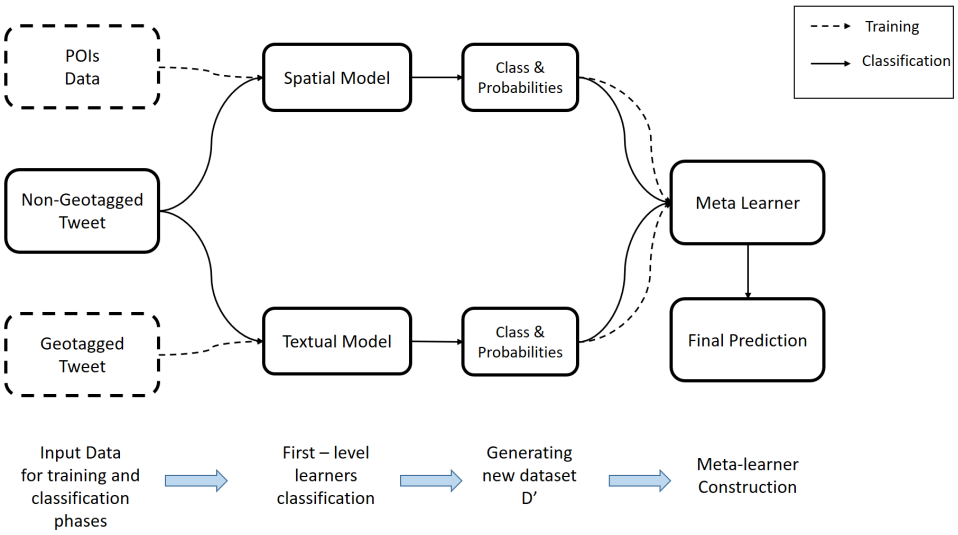


Figure 5.1: Multi-view Learning Approach

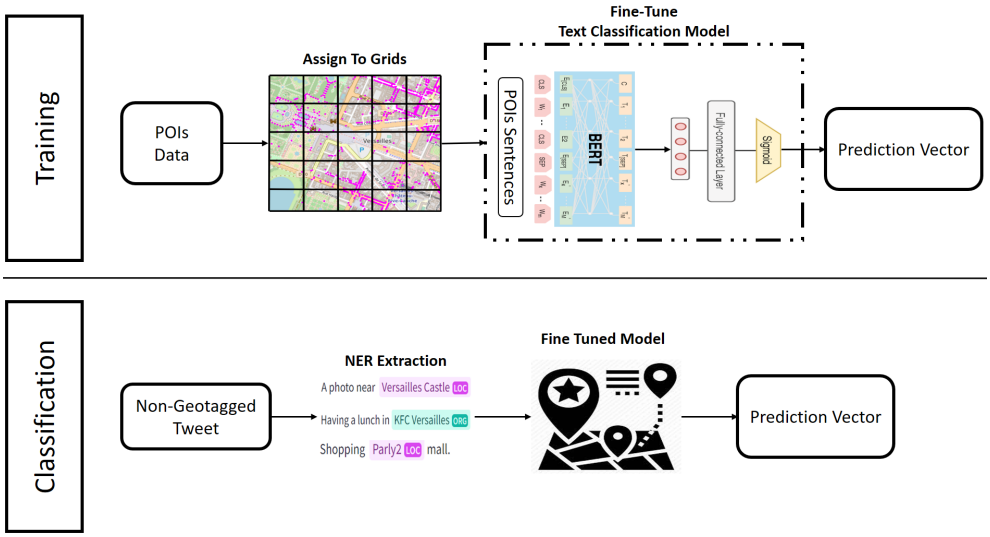


Figure 5.2: Spatial Model - POIs dataset

will be the representative label. Using the latitude and longitude geographical coordinates present in POIs and Twitter data, we will match each record by its corresponding cell and assign the cell number to each record as its label.

Figure 5.2 zoom in on the details of the spatial model. We have a training phase demonstrating the steps performed to train this model, and the classification phase shows how this model is used. First of all, this model is trained using POIs collected data. We form sentences from the POIs data (*i.e.* place name, road name, city, country...) to generate a dataset that will train the text classification model. The sentences are generated by concatenating the name of the POI and

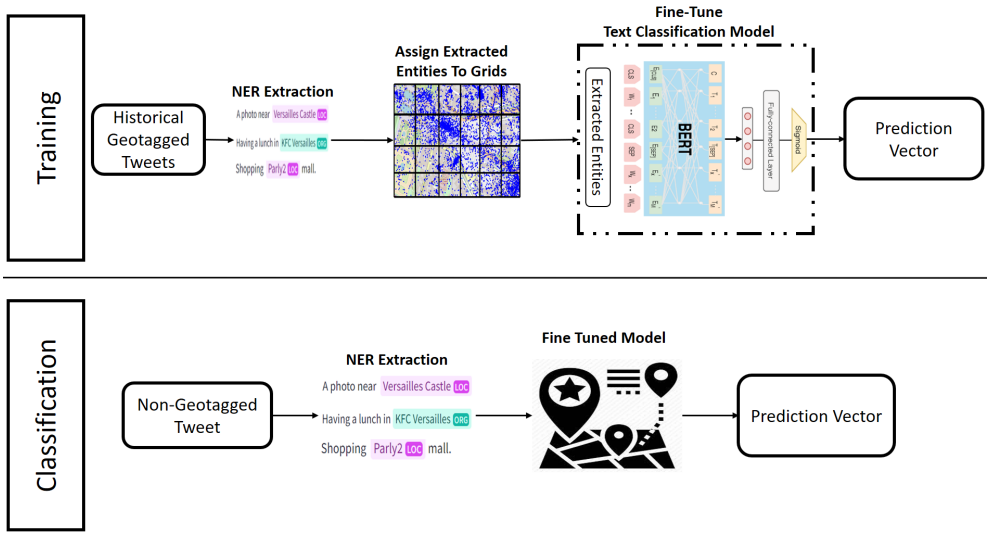


Figure 5.3: Adjusted Spatial Model - Extracted Entities

name	lat	lon	amenity	highway	place	city	street	landuse	shop	operator	Public Transport	country
CIC	48.796	2.136	bank	NaN	NaN	Versailles	Rue des Chantiers	NaN	NaN	CIC	NaN	France
Pharmacie Porchefontaine	48.796	2.154	pharmacy	NaN	NaN	Versailles	Rue Coste	NaN	NaN	NaN	NaN	France
Versailles Montreuil	48.803	2.153	post_office	NaN	NaN	Versailles	Rue Champ Lagarde	NaN	NaN	La Poste	NaN	France

Table 5.1: POI Dataset

the name of the street, the city, and the country with a space separator. Table 5.1 presents the different attributes in the POI dataset. For example, for the *CIC Bank* POI which has the following attributes in Open Street Map (OSM): (name: CIC, amenity: bank, highway: NaN, place: NaN, street: "Rue des Chantiers", landuse: NaN, operator: CIC,...., city: Versailles, country: France), we will form the following sentence: "CIC Rue des Chantiers Versailles France". The process of training the model is independent of the tweets. We have fine-tuned a text classification model to predict the cell number in a map. Fine-tuning is a known task in the world of NLP, and it is tuning the model to predict outputs depending on a given dataset. We have used an already pre-trained BERT model distilbert-base-uncased[3][120], which was trained on a vast dataset. The dataset is tokenized to be used by the model for training. Tokenizer will tokenize the inputs by converting tokens to the corresponding ids in the model vocabulary, generating other inputs required by the model (*i.e., attention mask*). The labels of those inputs are the corresponding cell numbers in the grid map.

For the classification part, for each tweet, NER (Named Entity Recognition) is applied to extract relevant location and organization entities. We will generate

[3]https://huggingface.co/distilbert-base-uncased

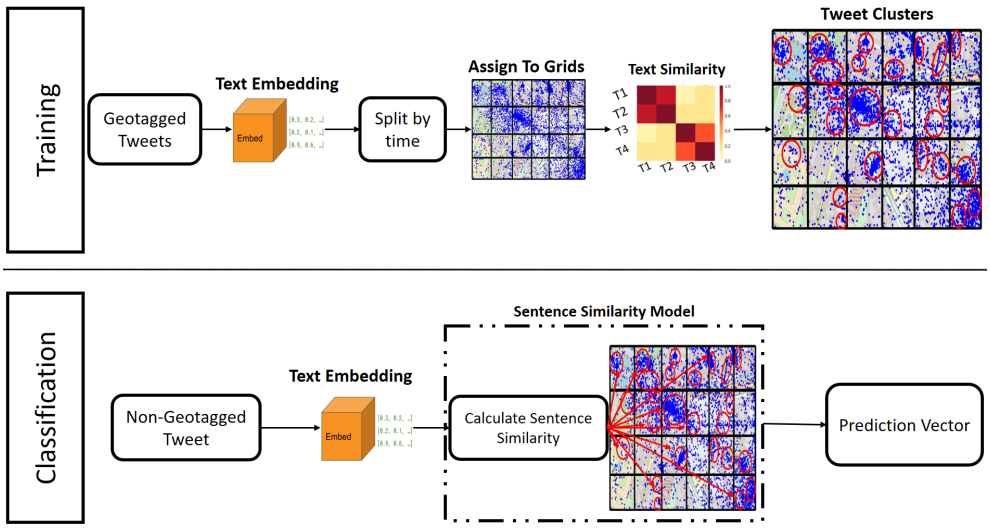


Figure 5.4: Textual Model

sentences from the recognized entities, following the same process followed with POIs. The generated sentences are the input data for the fine-tuned model. The fine-tuned spatial model will predict the tweet's location based on the recognized entities present in the tweet. Its output is a vector of each class's prediction and corresponding probabilities. As this model is not trained on tweet data, thus it can be used to predict the location of any text. At the same time, training the model using POIs data can raise some problems caused by the dataset of Twitter since we may usually find errors and mistakes in the tweet's text. In addition, some can have a dialect and a special way of tweeting. Those aspects can drop the accuracy of the tuned model.

The POIs dataset retrieved from Open Street Map (OSM) is usually in the mother language of the country. However, tweets may mention locations in other languages. For example, the POIs dataset in Greece is in Greek, while the tourists usually mention the locations in English. In Greece, the locations are written in both languages, but when retrieving those POIs from OSM, we will get only those written in Greek. Moreover, using the POIs dataset can decrease the performance of the spatial model in practice. The spatial model is trained on a dataset (POIs dataset) and validated on another dataset (locations extracted from tweets).

To address the abovementioned limitations, we proposed adjusting the spatial model to cope with the low accuracy problem. As mentioned, the main problem is the difference between the form of POIs data and the way of mentioning those data in tweets. To solve this problem, we proposed learning the location words of each cell from the historical geotagged tweets. Figure 5.3 shows the adjustment performed on our spatial model. In the training phase, instead of using data collected from POIs, we will perform NER on top of historical geotagged tweets to extract each tweet's location and organization entities. After assigning the entities

to the grid map, the entities will be identified by the labels. Then as done previously, we will fine-tune a text classification model to predict the cell number. We keep the same procedure for the classification phase as in the previous approach. The spatial model can now learn the entities usually mentioned in different cells. In other words, the model will learn the cell-indicative words.

Figure 5.4 describes the training and classification phases of the textual model. The idea behind the textual model (sentence similarity approach) is to use existing geotagged tweets. We based our work on the fact that it is highly probable to have tweets discussing the same topic originating from the same place. Thus, we are trying to identify the location of non-geotagged tweets based on their topics' similarity with geotagged tweets.

The topic should not be a global event because global events can be discussed anywhere. We need to identify global events as a first step. To identify those events, we can check the trending topics on Twitter on a specific day and in a specific country. Then all tweets related to those trending topics will not contribute to the work of this model. As shown in the training phase of figure 5.4, we split the geotagged tweets into groups depending on their creation time, and then each geotagged tweet will be assigned to its proper cell in the grid map based on its coordinates. Beforehand text embedding is calculated, and each tweet will be represented as an embedding vector. Text similarity among the tweets is calculated to form a group of relevant tweets (i.e., discussing the same topic) at each grid cell. The classification phase describes how this model will be used. We compute each non-geotagged tweet's similarity with geotagged ones taking place simultaneously. Finally, the tweet will be assigned to the grid cell that maximizes its similarity with the geotagged tweets. Again, this model's output is a prediction vector consisting of a predicted class and a vector of class probabilities.

The predictions of both spatial and textual models will be a vector. In both figures 5.2,5.3, and 5.4, the model's output is the prediction vector. The prediction vectors after that are concatenated to generate a new dataset D' as shown in table 5.2.

As we aim to enhance the prediction's accuracy and use the two methods proposed to find the location, we need to combine the results of both models. Given a specific input, we consider each model an independent view for predicting the location. The aim is to adopt a multi-view learning approach to combine the results and maximize accuracy. We adopt a multi-view learning approach inspired by the stacking generalization approach [3, 46]. The idea here is to generate a new dataset based on predictions of both models and their probabilities and train a meta-model on top of them.

Table 5.2 shows an abstraction of the feature vector of the new dataset D'. The feature vector will contain the prediction of the different views denoted in the table as the first-level learners, along with the probability of each learner (*i.e.* in other words, we can describe it as the weight of this prediction), and the valid label

will be the actual cell number of each tweet. A Random Forest classifier is used on this new dataset to train the meta-learner.

Table 5.2: An example of the newly generated dataset D'.

First-Level Learners						**Prediction Probabilities**						**True Label**
l_1	l_2	...	l_i	...	l_n	p_1	p_2	...	p_i	...	p_n	y

5.3 . Hybrid Geolocation Approach

5.3.1 . From Coarse to Fine Prediction

This section presents our proposed framework to predict a tweet's coarse and fine location. Our approach consists of two main components: a city-level prediction model and a fine-grained prediction model.

For the first fold, the city-level prediction model is based on previous approaches that use machine learning algorithms and crowdsourcing to predict the city-level location of tweets. Any existing approach that showed its effectiveness in predicting city-level labels can be used. **DeepGeo2**, Geotagging to landmarks, UnicodeCNN, or any other model can be adapted. Those approaches are designed initially to predict the precise location, however, when they are used to predict the region or city, they show better performance and accuracy. Usually, all the existing models take as input the tweet text, meta-data, and some additional information of users and other extra information. The output of the model is the city label. The appropriate $FLAIR$ model will be used to find the precise location depending on the predicted city label.

The second fold is the $FLAIR$ model, which predicts the location at a fine granularity given a specific region. Based on the predicted city label, we load the appropriate FLAIR model, which is trained to predict the cell in the grid map corresponding to the fine-grained location of the tweet. It takes as an input the tweet text and its creation time, providing additional temporal information that can aid in predicting the location.

Figure 5.5 shows the the architecture of the proposed framework to predict tweet location at coarse then fine granularity. This architecture shows the feasibility of our approach as this approach has been implemented and validated on real-life data. Algorithm 3 formally describes the steps performed to get the fine-grained granularity passing by the coarse granularity prediction and then passing the results to the appropriate FLAIR model.

5.3.2 . Illustrative Example

Table 5.3 shows a sample of unlabeled tweets. Each tweet is coupled with other meta information such as the timezone, user-identified location, and many

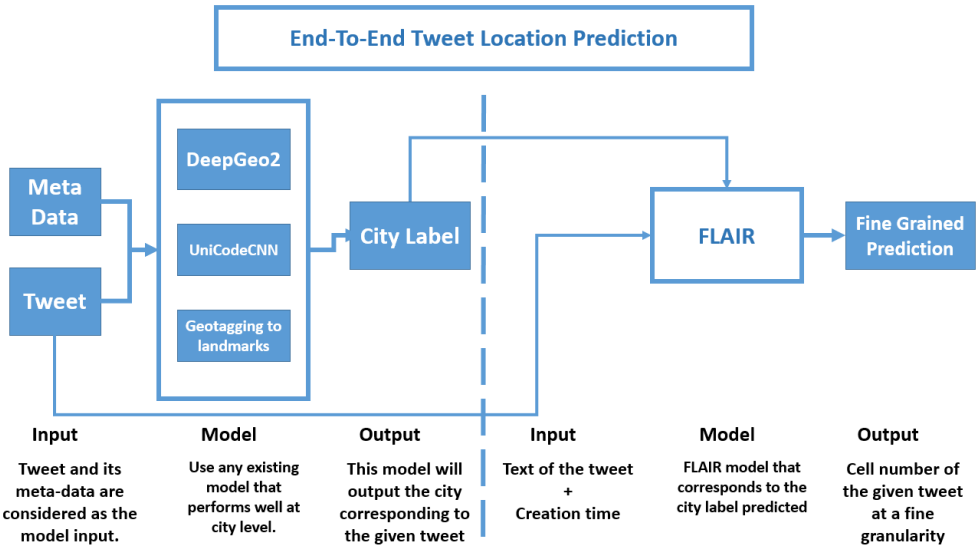


Figure 5.5: End-to-end Location Prediction Framework Architecture

Algorithm 3: End-to-End Approach

Input: Tweet, Additional information (user's data, meta-data)
Output: L

1 Tweet and data pre-processing and cleansing.
2 City-level prediction using one of the existing approaches.
3 Save the pre-processed tweet with the predicted label.
4 Load the appropriate $FLAIR$ model for the predicted city.
5 Use $FLAIR$(X,C,granularity) to predict the Location L.

other meta-data. Some rows can have missing values, such as the user location or timezone; this should be handled by the model.

Let's assume we are using the **DeepGeo2** model. Then it will take all the information present in table 5.3 as the input of the first fold of classification. The output will be the city label, for example, "Versailles", "Paris", "Santorini", etc.

Table 5.4 shows the tweets predicted to originate from Versailles. We will load the $FLAIR$ model trained on data from the Versailles region for those tweets. Figure 5.6 illustrates a case example of how the proposed solution will predict the location of tweets. Tweets in table 5.4 refer to figure 5.6, so tweet 1 in the figure is tweet number 1 in the table.

As shown in figure 5.6, the map is split into cells using the grid view. The POIs are identified, and the geotagged tweets are grouped into clusters based on their text similarity. To predict the location of non-geotagged tweets, we apply steps 6 through 9 in algorithm 2. For the first tweet, we can identify a location entity in the tweet text which is *"KFC"*, and the tweet is irrelevant to any of the geotagged

Tweet number	Text	...	User Defined Location	Timezone
1	Just a photo in Santorini near the beach...	...	Germany	GMT+2
2	Amazing Sunset at Santorini	...	France	GMT+2
3	Attending Pollution monitoring conference in Versailles. #GOGREEN	...	Italy	GMT+1
...	...	...	...	...
n-2	Ohh. Traffic jam again near Versailles castle.	...	France	GMT+1
n-1	Ici c'est Paris.	...	USA	GMT+1
n	I am happy today!	...	France	GMT+1

Table 5.3: Example of Unlabeled Tweets

clusters. Thus it will be assigned the label 1 as its cell prediction (here, the POI "KFC" is found in cell number 1). The second tweet talks about pollution, and at the same time, many geotagged tweets talk about the same topic at a specific place (cell number 6). Even though the tweet did not mention any POI in the text, depending on sentence similarity, we can assign it to cell number 6 in the grid (we assume that cell number 6 is the cell where we have a conference about pollution). For the third tweet, we have a mentioned POI; in addition, we have a cluster of geotagged tweets (here the tweets are about a car accident) that match the same topic of the third tweet, then we use the multi-view learning approach to assign cell 13 as the prediction. The case of the fourth tweet is typically the same as the second tweet. While for the fifth tweet in the table, we cannot predict the location. Here, we can assign the user's home location if it exists. Usually, tweets like tweet number five in the table are not informative for monitoring applications or detecting local events and activities.

Tweet number	Text
1	Having lunch at KFC!
2	Pollution monitoring conference. #GOGREEN
3	Ohh. Traffic jam again near Versailles castle.
4	Car accident near our home...
5	I am happy today!

Table 5.4: Example of Tweets with City Label (Versailles)

5.4 . Implementation

This section details the implementation pipeline of our location prediction approach. Figure 5.7 shows the implementation pipeline that includes five parts: data collection, data pre-processing, city-level model, data enrichment, and $FLAIR$

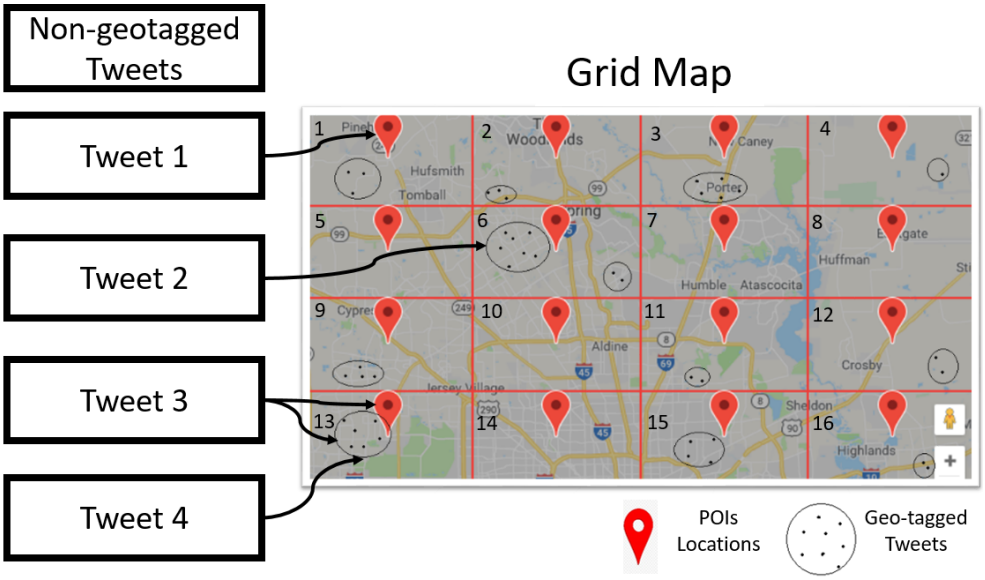


Figure 5.6: Tweet Location Prediction

model [4].

5.4.1 . Data Collection

We must acquire a real-life dataset of geotagged tweets and places of interest (POIs). We may extract the fine-grained location labels to construct and validate the proposed geolocation prediction approach for tweets. We describe our approach for data collection for Twitter and POIs in this subsection.

We used Twitter's API through a Researcher account to collect geotagged tweets. We collected tweets from a variety of locations across the world. We collected two datasets of tweets. The first dataset is collected to train and validate the city-level prediction part. In contrast, the second dataset is used to train and validate $FLAIR$ in two locations of interest *"Versailles"* and *"Santorini"*.

For the city-level prediction part, we collected data from different places worldwide. The data was collected in some US cities, such as Colorado, New Jersey, and New York. Also, we have collected tweets in cities from the UK, such as London, New Castle, and Manchester. In addition, we collected tweets in cities in France, such as Paris, Marseille, and Versailles. Another set of tweets was collected in Brussels. Also, we collected tweets from Santorini in Greece. In our collection process, we tried to acquire data from countries that use the same language, such as US and UK, on the one hand, and France and Belgium, on the other hand. Moreover, we added tweets from Santorini, which is a touristic spot that does not have a specific language for tweeting.

On the other side, for the $FLAIR$ part, we are interested in the tweets origi-

[4]https://github.com/MohammadAbboud96/FLAIR

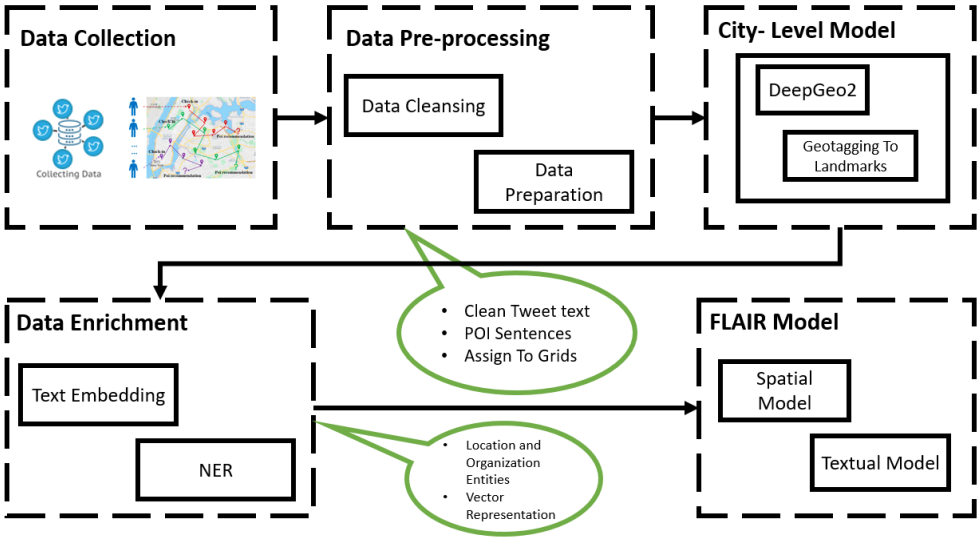


Figure 5.7: Implementation Pipeline

nating from two places *"Versailles"* and *"Santorini"*. Hence, we defined a bounding box for each region and collected historical tweets within these areas.

In the two collections scenarios, we were interested only in tweets; thus, we filtered out all the retweets. In addition to the tweet text, we collected the meta-data associated with each tweet, including the creation time, user ID, and geographic coordinates. We used the geographic coordinates to filter out tweets that were not geotagged or geotagged outside the areas of interest. Moreover, in the first collection scenario, i.e., for the city-level prediction approach, we also collected additional information about Twitter's user posting the tweet, the source, and other extra information.

To derive the fine-grained location labels for our geolocation prediction models, the $FLAIR$ part, we collected points of interest (POIs) from the OpenStreetMap database. We focused on POIs likely relevant to real-life activities, such as restaurants, tourist spots, and shops.

To collect the POIs, we used the Overpass API, which allows us to query the OpenStreetMap database for specific types of POIs in a given area. We queried the database for POIs within a certain radius of each geotagged tweet in our dataset and then reviewed and filtered the results to ensure they were relevant and accurate.

5.4.2 . Data Pre-processing

In order to extract meaningful features that can be used for prediction, a pre-processing step is a must. This subsection describes our pre-processing steps for tweets and points of interest (POIs) data.

The pre-processing of tweets involves several steps, including stop words re-

moval, link removal, and tokenization. Stop words are common words that do not carry much meaning and can be safely removed from the text without affecting its overall meaning. We used a pre-defined list of English, French, and Greek stop words and removed them from the tweet text. Moreover, we removed special characters and performed cleansing to the tweet text.

Links can also be safely removed from the tweet text since they typically do not carry meaningful information about the tweet's content. Those links are the URLs of associated media with the tweets. Usually, they are attached to tweet text when crawling Twitter data. Keeping such information may affect calculating the embedding of words. We used regular expressions to remove links from the text.

Tokenization involves breaking the text into individual words, or tokens, that can be used as features for prediction. We performed tweet text tokenization with the help of NLTK and spaCy python libraries.

In addition to the text pre-processing, we extracted features from the tweet meta-data, such as the creation time and user details, which can provide valuable information for prediction. Moreover, we extracted data from users' profiles for the city-level prediction, such as user home location, account creation time, and other information.

The pre-processing of POIs data involves transforming the data into a format that can be used as input to train our geolocation prediction models. Beforehand, we have identified the relevant POIs to keep and those to remove. To do that, we relied on the type of the POI, so we removed some POIs, such as traffic lights, traffic signs, etc., while keeping the relevant POIs (Restaurants, parks, stores, etc.). We transformed each POI into a sentence by combining the different attributes of the POI, such as its name, address, and category.

There is no standard way for people to mention the POIs in their tweets. It is more common that they use only the name of the POI, or they use the POI name combined with the city or country name. We have generated the most probable combinations the user may write for those reasons.

Overall, the pre-processing of tweets and POIs data has enabled us to extract meaningful features that can be used for accurate geolocation prediction.

5.4.3 . City-Level Model

To accurately predict the location of tweets at a fine-grained level, we proposed the two-fold model training process. The first fold involves training a city-level predictor model using datasets from different countries.

For the city-level predictor model, we used existing models such as DeepGeo2 or Geotagging to landmarks. We trained the model using the pre-processed dataset collected from different countries, which included tweets with their associated metadata, such as user location, timestamp, and text. The model was trained to predict the city label of the tweet based on the available metadata and text information.

For **DeepGeo2**, we have used the code available on GitHub [5]. While for **Geotagging Tweets To Landmarks** following the proposed architecture in [89]. We made a variant of the initially proposed approach and named it **Geotagging Tweets To Cities**, where we map the Tweet text, time, and source to a list of cities. However, we added another piece of information to enhance the accuracy. We made our version of **Geotagging Tweets To Cities***. We use the previously used features and add the user-defined location text as a new feature. We calculate the embedding vectors and then pass them to the convolution neural network. We applied the same process that is applied to Tweet text. Finally, all the features are combined and passed to the fully connected layer to predict its city label.

The city-level predictor model provides an initial prediction of the tweet's location at a coarse-grained level, which helps filter out irrelevant tweets and narrow the search space for the fine-grained prediction. According to the predicted city-level coarse location, we can load the appropriate $FLAIR$ model, as for each city, we will have a different $FLAIR$ model,

5.4.4 . Data Enrichment

Enrichment is adding knowledge to the dataset and transforming data into semantical views. In order to develop accurate geolocation prediction models for tweets, we need to enrich the data by extracting additional information that can be used as features for prediction. This section describes our data enrichment process, which involves applying named entity recognition and text embedding calculation to extract the proper knowledge from the qualitative data (tweet text).

Named entity recognition (NER) involves identifying and classifying entities in text, such as people, locations, organizations, and dates. With the help of NLP techniques specifically, the model developed for *NER* (Named Entity Recognition) that is found on hugging face [6] [7], which has around 88.5% overall precision, we were able to extract additional features related to location, such as the names of cities, countries, and landmarks mentioned in the tweet. Using the organization and location-recognized entities, we will form new sentences. Those sentences will serve as the input to our spatial model.

Conversely, we should transform the raw text into embedding vectors to compute the textual similarity. Text embedding calculation involves representing text as a high-dimensional vector in a semantic space, which can capture the meaning of the text. To perform this, we have used a semantic model [116] found on hugging face [8], this model has an accuracy of around 87.4%. In this phase, we use the pre-processed text (text after cleansing and link removal). Each tweet will be represented as an embedding feature vector.

By calculating text embeddings, we were able to capture the meaning of the

[5] https://github.com/jhlau/twitter-deepgeo
[6] https://huggingface.co/Jean-Baptiste/camembert-ner
[7] https://huggingface.co/xlm-roberta-large-finetuned-conll03-english
[8] https://huggingface.co/sentence-transformers/paraphrase-xlm-r-multilingual-v1

text in a way that is more meaningful than simple bag-of-words approaches. This can provide important contextual information for geolocation prediction, especially for the textual model.

5.4.5 . Model Refinement with FLAIR

This subsection describes the model training phase of $FLAIR$. In order to accurately predict the location of tweets at a fine-grained level, we are using the appropriate $FLAIR$ city model depending on the city-level prediction. This second fold involves training a multi-view learning approach in the FLAIR framework, which combines a spatial model trained on POIs data or location-indicative words and a textual model using tweet embeddings.

$FLAIR$ model is a multi-view learning approach. It is designed to predict the location of a tweet at a fine-grained level based on the available text and timestamp information. $FLAIR$ combines two first-level learners, the spatial and textual models.

The spatial model is trained on the prepared dataset of POIs sentences or extracted location entities from historical tweets in 5.4.2. The model is trained to predict the location of named entities extracted from tweets based on the similarity between the named entity and the POIs/location entities in the dataset. The spatial model uses a pre-trained BERT model to encode the POI/location entity sentences into a high-dimensional vector representation. The pre-trained BERT model is fine-tuned using our dataset to map the entities extracted from a sentence to a cell in the grid map.

The textual model is implemented to find the similarity of each non-geotagged tweet at a specific time with geotagged tweets at that time. It uses the embeddings calculated previously in 5.4.4. The model is trained to predict the location of a tweet based on the similarity between the tweet embedding and the embeddings of other geotagged tweets happening simultaneously. It finds the similarity of each non-geotagged tweet at a specific time with geotagged tweets at that time. Geotagged tweets are grouped into clusters, and we will calculate the cosine similarity of each non-geotagged tweet with all possible geotagged ones.

The spatial and textual models' predictions are combined to generate a new dataset. A meta-learner is trained on top of the new dataset to predict the tweet's location. In our case, we are using the Random Forest classifier as the meta-learner to predict the fine-grained location of the tweet.

5.5 . Experiments

In this section, we describe the experiments we conducted to evaluate the performance of our geolocation prediction models. We divided the experiments into two parts: the city-level prediction model and the FLAIR model. The models are implemented in Python 3.8 using Keras, Tensorflow, and sci-kit-learn. All the experiments are carried on the same environment.

5.5.1 . City-level Experiments

To evaluate the performance of our city-level prediction model, we used a dataset of geotagged tweets collected from different countries worldwide. The dataset was collected from several cities as described in 5.4.1. We have collected approximately the same number of tweets from each city. We had four months of data collection from France(Versailles, Paris, and Marseilles), the USA (Colorado, New Jersey, and New York), the UK (London, New Castle, and Manchester), Belgium (Brussels), and Greece (Santorini). We collected around 70K tweets for each country; just for Santorini, we used the whole dataset, around 186K. Approximately we have a balanced dataset containing tweets from different cities. After pre-processing the tweets and keeping only those containing the user information defined, as most city-level predictors require such features, we had the remaining percentage: Colorado 20K tweets, New Jersey 21K tweets, New York 22K tweets, Paris 28K tweets, Marseilles 25K tweets, Versailles 14K tweets, London 19K tweets, New Castle 26K tweets, Manchester 23K tweets, Brussels 59K tweets, and Santorini 21K tweets. This is considered a balanced dataset and was used to train the city-level models. We evaluated the model's performance using several evaluation metrics, including accuracy, precision, recall, F1-score, and Cohen Kappa score.

We adapted **DeepGeo2** and **Geotagging To Landmarks** as city-level predictors. Initially, DeepGeo2 is designed to predict the location. However, we are using it to predict the city. During the data preparation phase, we assigned cities as the labels of the tweets. We trained the model on 67% of the dataset and tested it on the remaining 33%. The accuracy reported on the balanced dataset was **92.4**%.

Conversely, we implemented **Geotagging Tweets To Cities**, where we map the Tweet text, time, and source to a list of cities. The approach shows acceptable results with around 67% accuracy. However, for our version **Geotagging Tweets To Cities***. The accuracy improved significantly, reaching 90%. Table 5.5 reports each city's precision, recall, and F1-score.

These results demonstrate that the city-level prediction model can accurately predict the location of tweets at a coarse-grained level, which can be helpful for a wide range of applications, such as social media analysis and urban planning. The high accuracy reported shows we could count on the existing literature to predict a tweet's coarse location. While we still need a complement approach to predict the fine-grained location.

5.5.2 . *FLAIR* Experiments

Experimental Settings

We evaluated the proposed model on real-life data and validated our approach using two datasets. We have collected tweets in the region of Versailles from 2010 until July 2021, and we have found around 370K geotagged tweets. Also, we collected tweets in the region of Santorini between 2015 and 2021, and we found around 186K geotagged tweets. The same experimental settings were used for

City	Precision	Recall	F1-Score
Colorado	0.86	0.86	0.86
New Jersey	0.89	0.9	0.9
New York	0.8	0.84	0.82
Paris	0.74	0.75	0.75
Marseilles	0.87	0.89	0.89
Versailles	0.71	0.66	0.7
London	0.81	0.86	0.83
New Castle	0.92	0.89	0.91
Manchester	0.85	0.86	0.75
Brussels	0.94	0.92	0.93
Santorini	0.78	0.71	0.74

Table 5.5: Precision, Recall, and F1-Score Metrics - Geotagging Tweets To Cities*

both datasets, each dataset is split into a 70% training set, and the rest 30% are used for validation. We have tried different grid splits for the area of interest (*i.e.* Versailles region, Santorini region). We started from 4 columns by four rows to split the map and reached 15 columns by 15 rows, and then we performed a split using 20 columns by 20 rows, 30 by 30, 40 by 40, and reaching 50 by 50.

Table 5.6 reports some results of the performed experiments in the Versailles region. The first two columns in table 5.6 show the area of each cell and the number of classes found concerning different granularities. Only cells containing tweets are considered; in fine granularity, the number of labels increases while the area of the cell decreases. For example, at granularity (4x4), we have only four classes, and the area of each cell is 21 km^2, while for granularity (50x50), we have 227 different labels with an area of 0.07 km^2 for each cell. For the POIs, we have collected all the POIs in the area of interest, and we kept only the relevant ones, such as tourist places, restaurants, street names, etc.

Each POI record has its corresponding geo-coordinates. Using those coordinates, we distribute the POIs over the cells of the grid map. For the geotagged tweets, the same approach is followed; tweets are distributed over the cells based on their geographical coordinates. Hence, in the experiments, POIs and tweets are represented by the cell numbers (labels) and no more their actual geographical coordinates.

Experimental Results

We consider the location prediction problem as a classification problem, where we aim to predict the cell label of the tweet instead of getting the exact geolocation coordinates. We evaluated our model against baselines and proposed approaches

Granularity	Area km^2	Number of labels	MNB	MLP	Geotagging to landmarks	DeepGeo2	Spatial Model	Textual Model	Multi View
(4x4)	21	4	69.8	66.1	71.8	78.4	63.5	83.4	**90.7**
(10x10)	2.3	25	39.3	47.8	45.7	62.6	39.5	68	**79.1**
(15x15)	0.9	46	35.1	47.1	43.8	54.6	33.2	65	**77.1**
(30x30)	0.2	125	29	45.6	40.9	51.9	6.9	62.9	**71.1**
(50x50)	0.07	227	21.3	34.3	42.3	36.4	5.1	59.6	**71.6**

Table 5.6: accuracy of different models Versailles Region

Granularity	Area km^2	Number of labels	MNB	MLP	Geotagging to landmarks	DeepGeo2	Spatial Model	Textual Model	Multi View
(5x5)	19.15	14	57	49	60.8	60	54.9	58.4	**68.9**
(10x10)	3.7	43	42.4	34.1	48.4	45.8	35.9	49.3	**56.1**
(15x15)	1.56	80	36.2	30.1	46.2	42.1	27.2	45.5	**54**
(30x30)	0.36	151	31.5	28	43.1	40.1	20.4	43.2	**52.7**
(50x50)	0.12	219	27.2	16.1	42	32.2	17.3	41.7	**50.5**

Table 5.7: accuracy of different models Santorini Region

in the state-of-art for location prediction methods.

We used the Multinomial Naïve Bayes Classifier (MNB) and Multilayer perceptron network (MLP) for the baselines. Those two classifiers are considered state-of-art methods for text classification and as a baseline in location prediction problems. We validate our approach against other existing approaches in the literature. We considered two approaches, Geotagging Tweets to Landmarks [89] and DeepGeo2 proposed in [81, 127]. We implemented both approaches using deep learning techniques and reported the results on our dataset. For [89], we considered the labels (cell numbers) as the landmarks we wanted to predict.

Table 5.6 reports the experimental results of **Versailles region**. The spatial model trained on the POIs dataset has reported a high accuracy for all granularities when validated on the POIs dataset. It reached around 82%. While when validating the model on locations extracted from tweets, the model accuracy drops, especially at finer granularities. For example, at the (4x4) granularity, the accuracy was around 63.5%; at (10x10), it decreased to 40% and reached 5% at the (50x50) granularity. This decrease in accuracy is due to two factors. The first is that the model has never seen the tweets data (as it is not trained on tweets data). Due to the way of writing on Twitter, most users use shortcuts to mention visited locations, and usually, they do not mention the specific location.

The textual model reported good accuracy among different granularities. At all granularities, it outperforms the baselines and existing approaches.

Combining the results using the multi-view learning approach has enhanced the accuracy. As reported in table 5.6, our proposed approach has outperformed all the other classifiers at different granularities. It reports an accuracy of 71.6% at (50x50) granularity where the area is 0.07 km^2.

This part reports the experimental results of **Santorini region** reported in table 5.7. Although our approach did not show the same accuracy in Santorini as in Versailles, it still outperforms other approaches. Different granularities were tested, starting from (5x5) and reaching (50x50). Our approach shows acceptable accuracy.

The reported results show the superiority of our approach when compared to others. For the two used datasets, the multi-view model outperformed the existing approaches. Although we had good accuracy for $FLAIR$ in the Versailles region and an acceptable one for the Santorini region, we expected better findings.

One of the limitations of our FLAIR model is that it was trained on a dataset of POIs and validated on named entity recognition (NER) extracted from tweets. While this approach allows us to predict the location of tweets at a fine-grained level, it may not generalize well to tweets from different regions or languages.

Santorini is a tourist place, and most tweets in that region correspond to tourists. We expected that the Spatial model (i.e., the model trained on top of POIs) would have better accuracy since most tweets contain location entities.

Two main reasons exist for the low accuracy reported for the spatial model. First, the model is trained using POIs data and validated on tweets data. When validated on POIs data, the fine-tuned model reports a high accuracy of around 82%, but this accuracy drops when validated on tweets data. The way of mentioning location in tweets looks different from the POIs data. This can explain the decrease in accuracy for the spatial model. Moreover, the second reason behind the low accuracy reported is the language of the POIs dataset. In the Santorini region, we collected POIs data using the OSM API, but the retrieved data was in Greek, while the POIs mentioned in the tweets are mainly in English. Those aspects decrease the accuracy of the spatial model. Thus the overall accuracy of the multi-view model drops down.

To address this limitation, We proposed adjusting the spatial model to cope with the low accuracy problem. We learned the location words of each cell from the historical geotagged tweets.

We repeated the experiments using the adjusted spatial model to validate our work. Table 5.8 and table 5.9 reports the results of the spatial model and multi-view model when using the first approach and the second approach for Versailles and Santorini respectively.

Reported results show a significant improvement in the results of the spatial model. For **Versailles region**, it is clear that the new approach improves the accuracy at a finer granularity. Using the spatial model (first approach), we had an accuracy of 6.9% and 5.1% for granularities (30x30) and (50x50), respectively, while with the spatial model (second approach), we had an accuracy of 52.5% and 44.1%. The results improvement of the spatial model is reflected in the results of the multi-view model. Thus we had a better overall accuracy at the different granularities.

Using the spatial model (second approach) in **Santorini region** significantly improves the classification accuracy. The accuracy of the multi-view model (second approach) showed good results compared to the first approach and the state-of-art methods. We used several metrics to evaluate the model, including calculating the precision, recall, and F1-Score per class, as we have an unbalanced classification problem.

Table 5.10 shows the mentioned metrics for the Multi-view (Second approach) model for the granularity (5x5) at the Santorini region. We considered only cells that contain Tweets. For the cells where the metrics are zero, we checked the instances at those cells, and mostly we had only one tweet at that cell. However, we always have good metrics scores for other cells with more than one tweet.

For the rest of the classifiers at finer granularities, we also calculated the same metrics and got acceptable results (>=0.7) per cell with more than one or two tweets. Moreover, we calculated the Cohen kappa metric for the different models. The kappa coefficient is 0.76 at (5x5) granularity, 0.73 at (10x10) granularity, 0.72 at (15x15) granularity, 0.70 at (30x30) granularity, and 0.69 at (50x50) granularity.

Granularity	Spatial Model (First approach)	Spatial Model (Second approach)	Multi View (First approach)	Multi View (Second approach)
(4x4)	63.5	79.8	90.7	**93**
(10x10)	39.5	59	79.1	**84.3**
(15x15)	33.2	59.7	77.1	**83.3**
(30x30)	6.9	52.5	71.1	**78.9**
(50x50)	5.1	44.1	71.6	**78.5**

Table 5.8: accuracy of different models (after spatial model adjustment) Versailles Region

5.5.3 . Coarse to Fine Granularity Prediction Experiments

In this part, we validate the end-to-end hybrid approach proposed in this work. The above experiments independently report the results of the different parts of the proposed framework. The results show that using the existing approaches for the city-level prediction task could be an excellent option to predict the coarse granularity of the tweets. Conversely, the experiments performed using $FLAIR$ on the two datasets from Versailles and Santorini show the strength of $FLAIR$ in geolocating the tweets at fine granularities with acceptable accuracy.

In this subsection, we report the results of the proposed framework when combining the two methods to pass from coarse to fine granularity prediction. This experiment is done on top of datasets collected from Versailles and Santorini. We used all the acquired data for those two cities; as mentioned, we had around 370K

Granularity	Spatial Model (First approach)	Spatial Model (Second approach)	Multi View (First approach)	Multi View (Second approach)
(5x5)	54.9	76.9	68.9	**82.9**
(10x10)	35.9	66	56.1	**77**
(15x15)	27.2	61	54	**75.6**
(30x30)	20.4	53	52.7	**72.8**
(50x50)	17.3	52	50.5	**71.2**

Table 5.9: accuracy of different models (after spatial model adjustment) Santorini Region

tweets for Versailles and 186K for Santorini. For the city-level predictor, we adapted DeepGeo2 for this task. We chose this model as it performed better than other methods reported in the City-level Experiments part 5.5.1. Since the city-level predictors require these features, we pre-processed the dataset and kept only tweets with user location-defined data. The remaining dataset comprises around 228K tweets in Versailles and 25K in Santorini. The city-level predictor performed well on top of this dataset, as the reported accuracy was 98.8%.

As a next step, after finding the coarse granularity of those tweets using the city-level model, we adapted $FLAIR$ to find the fine granularity location of those tweets. Table 5.11 reports the fine granularity location prediction results. The results show good and acceptable accuracy of almost more than 80% for most granularities in the different cities. This experiment validates our hypothesis raised in the beginning that we can go from coarse to fine location prediction using the proposed framework that combines the city level and $FLAIR$ models.

5.6 . Discussions

Based on the results of our experiments, the proposed framework for location prediction is feasible and effective. The experiments demonstrate that adapting existing approaches for city-level prediction achieves higher accuracy than directly predicting at the fine-grained level. The FLAIR model shows that when we know the city label of a text and with minimal features used (text and creation time), we can achieve high accuracy at the fine-grained level.

By combining these findings, we may infer that our suggested framework successfully transitions from coarse to fine-grained location prediction. This is because the framework allows us to utilize the strengths of both techniques, the city-level predictions for improved accuracy and the FLAIR model for fine-grained predictions when the city label is known. However, combining both approaches may require more features, such as user information being unavailable for every tweet.

Cell	Precision	Recall	F1-Score
1	0.59	0.59	0.59
2	0	0	0
3	0.77	0.87	0.82
4	0.89	0.88	0.89
5	0.62	0.51	0.56
6	0.62	0.64	0.63
7	0.84	0.84	0.84
8	0.81	0.75	0.78
9	0.84	0.79	0.81
10	0.84	0.87	0.85
11	0.73	0.7	0.71
12	0.85	0.78	0.82
13	0.72	0.68	0.7
14	0.79	0.58	0.67
15	1	0.25	0.4

Table 5.10: Precision, Recall, and F1-Score Metrics - Multi-view Model (5x5)

In addition to the findings discussed in the previous paragraph, it is worth noting that the FLAIR model has potential applications beyond just predicting location from tweets. It could be used for any form of text data when the city label is known. FLAIR model could also be used to analyze geotagged tweets with imprecise or incomplete location information. Twitter's policy on location data has evolved, so the platform no longer provides precise coordinates for most tweets. Instead, it provides a wide polygon or even the city boundaries as the geotag data for tweets. This presents a challenge for location-based analyses, but the FLAIR model could extract relevant location information even from imprecise geotags.

Moreover, one of the primary features of the suggested framework for location prediction is that it allows us to make reliable predictions at both the coarse and fine-grained levels. This implies that the framework's predictions may be used for various use cases and applications.

For example, the city-level forecasts provided by the framework might be utilized for macro-level assessments of trends and patterns over vast geographic regions. This might benefit urban planning, transit planning, and public health programs, among other applications. However, the framework's fine-grained predictions might be utilized for more particular applications, such as targeted advertising or tailored suggestions based on location. By reliably predicting the location of tweets at a fine-grained level, businesses and organizations may deliver more relevant and tailored services to their consumers.

The suggested framework is a flexible tool with various possible applications

Granularity	$FLAIR$ Versailles	$FLAIR$ Santorini
(5x5)	94.4	86.4
(10x10)	86	81.4
(15x15)	84.5	80.1
(30x30)	80.2	75.9
(50x50)	80	74.5

Table 5.11: Fine grained location prediction for tweets in Versailles and Santorini **after city level prediction**

due to its ability to create reliable predictions at coarse and fine-grained levels. The necessity of precise location prediction will only rise as more data becomes accessible, and the approach described here constitutes an essential addition to the discipline. Overall, the reported results in the previous section demonstrate the potential of our proposed framework for location prediction and suggest that it is a promising avenue for further research.

5.7 . Conclusion

Tweet location prediction has gained the interest of many researchers, especially for applications that use social media data in analysis. Existing approaches succeeded in predicting the location at city or country levels. In this work, we have proposed $FLAIR$, a multi-view learning approach for fine-grain tweet location prediction within a specific area of interest. Our approach is based on top of two models: the spatial model, which learns the location words from a tweet to find its location (either using POIs data or extracted locations from historical tweets), and the textual model assigns labels depending on text similarity.

We trained our city-level prediction model on a dataset collected from different countries and used FLAIR to predict the location of tweets at a fine-grained level. We evaluated our models using different metrics and found that coupling our city-level prediction model with FLAIR improved performance.

Our approach has several applications in various fields, such as disaster response, urban planning, and crime prevention. It can also provide valuable insights into real-life activities by allowing us to predict the location of tweets at different levels of granularity.

$FLAIR$ requires minimal features, as it depends mainly on the tweet text. This approach can be adapted to any text corpus, not just Twitter data. The reported results have shown that our model outperforms the baselines and existing approaches for location prediction problems. Especially when adjusting the spatial model, we obtain a significant improvement in terms of accuracy. The accuracy of the spatial model and that of the textual model drops as the granularity de-

creases. However, the combination of the results using the multi-view model shows acceptable results for all granularities.

We are looking forward to enhancing our approach by adding (at the first-level learners) new views, such as media data. Indeed, using the stack generalization approach allows for adding or removing learners efficiently. Having other views on the data will improve the model's accuracy, yet this needs to be evaluated.

Chapter 6 - Conclusions and Future Work

Contents

In this last chapter of the dissertation, we will summarize the work presented throughout the previous chapters and emphasize our achievements against the research questions presented in Chapter 1 in Section 6.1. Thereby, we will highlight possible future research directions in Section 6.2.

6.1 . Summary of Contributions

Our main goal in this thesis was to enrich and analyze quantitative and qualitative data in the context of environmental monitoring. The aim is to develop multi-dimensional analysis on top of real datasets. Hence we can help decision-makers better plan urbanization policies.

In the first chapter 1, we presented a comprehensive overview of the challenges and objectives addressed in this study. We set the stage by emphasizing the complexity and variability of multi-variate time series data, the necessity for fine-grained location identification, and the limits of fixed station-generated maps in predicting air pollution levels in unmonitored areas. Throughout the dissertation, we highlighted the importance of developing effective methodologies and frameworks to address these challenges. We also emphasized the gaps in existing approaches and the need for innovative solutions when reviewing the related work in Chapter 2.

This conclusion chapter serves as a fitting end to our study, highlighting the consequences and implications of our effort answering key research topics and giving significant insights for future research and practical applications. In this dissertation, we addressed three key research questions aimed at tackling various challenges in different domains:

Research Question #1 focused on dealing with complex, missing, and heterogeneous multi-variate time series data while automating the micro-environment recognition. This question's primary concern is how to automatically identify micro-environments based on abundant trajectories from different sensors (i.e., sources).

To recognize different micro-environment such as *Home*, *Office*, *Walk*, *Car*, etc. We considered our problem as a human activity recognition problem. Thus, we have investigated a wide range of state-of-the-art proposals in activity recognition using GPS data and wearable sensors to answer this question. In addition, other generic methods for Multi-variate time series classification (MTSC) and multi-view learning approaches are independent of the origin of the data.

Therefore, our first contribution ***Contribution #1*** was implementing an end-to-end framework using a machine learning approach based on multi-view learning for micro-environment recognition. We further extended this approach by adding a post-processing layer. Our proposed approach handles the pipeline steps from data collection, pre-processing, learning and prediction, post-processing, and visualization. Throughout our work, we combined different heterogeneous data types (i.e., views) and dealt with missing data by adapting the multi-view learning approach. Our experiments are conducted on top of the real datasets collected within the

scope of the Polluscope project, and the results have shown the efficiency of our framework.

Research Question #2 aimed at estimating air pollution levels in uncovered spots while enhancing fixed station-generated maps with opportunistic Mobile Participatory Monitoring (MPM) data. Within this question, we explored how data collected opportunistically where no targeted places beforehand can help enrich pollution maps and enhance the estimation process. Our aim was to use such data in enriching pollution maps generated from fixed station measures and then estimate the pollution levels in uncovered locations.

Therefore, we went through an extensive review of the existing approaches. We went through the different data used in such approaches, the challenges of each, and the techniques used for estimation. Many existing works combined fixed and mobile sensory data; however, none used opportunistic MPM data.

To answer the raised question, we proposed a methodology that enhances the enrichment of fixed-stations generated maps with opportunistic MPM data and better estimates of air pollution levels, denoted as our second contribution ***Contribution #2***. We showed the usability of enhancing the accuracy and the coverage of air pollution maps by integrating fixed station data with opportunistic MPM data. Within this work, we first expanded pollution estimation's spatial and temporal coverage by identifying clusters of different fixed station data and matching them with corresponding MPM data collected at the same periods. Then by adapting interpolation techniques on top of enriched maps, we obtained better estimate air pollution levels' variability at a higher resolution. We used different interpolation methods, mainly geostatistical and deep learning methods, and we validated our proposed methodology on top of real datasets collected from Versailles and Chicago.

Research Question #3 revolved around identifying the tweets' location precisely at a fine-grained granularity to reduce the distance error between the real and anticipated location. In other words, how to handle the challenge posed by the limited availability of geotagged tweets by integrating methods to enhance location prediction precision.

Throughout the work dedicated to answering this question, we explored different existing approaches in the literature. Existing approaches are classified into two categories based on their predictions: coarse or fine granularity. The coarse granularity approach predicts the country or city of a tweet, while the fine granularity approach predicts the geolocation coordinates of a tweet. Existing approaches are performing very well at city-level prediction. However, we still have a shortage at a finer granularity.

Within our third contribution ***Contribution #3***, we proposed a hybrid coarse-to-fine granularity prediction framework to minimize the distance error of the predictions. Our approach combines the existing approach to predict city-level and investigates $FLAIR$, a fine-grained location prediction algorithm that predicts

the fine granularity of location. Using the multi-view learning approach $FLAIR$ combines predictions of a spatial model and a textual one, which enhances the prediction accuracy. $FLAIR$ uses minimal features compared with other approaches. However, it needs the city label beforehand. Thus, we adapted existing approaches to predict the cit-level, and then we used $FLAIR$ for the fine granularity prediction. Our approach was validated using a real dataset collected from Twitter within the context of the GOGREEN Routes project.

6.2 . Future Work

The future work section of a thesis is highly significant since it discusses prospective topics for future research and development based on the current study's contributions and conclusions. In this section, we look into the possibilities raised by our findings and investigate opportunities to expand our research. We hope to contribute to the ongoing progress and stimulate future research by highlighting unexplored features and possible innovations.

In order to achieve quality and innovation, we foresee a lively and collaborative research community that seizes the opportunity given by our thesis to explore unexplored territory, question prevailing assumptions, and push the limits of knowledge.

6.2.1 . Events Detection

In our work, we automated the detection of micro-environments from MCS data. However, the next important step is recognizing the events during the different activities. Several events can occur within the conducted activities. These events may affect the exposure level. Cooking, Smoking, Opening window, etc., are some events we should consider when calculating exposure to air pollution.

This problem opens the doors to future research. We should investigate such problems and identify how we can consider these events. What is the best way to handle them? Can we build a model that considers recognizing micro-environments and events together, or is it better to build a model handling events independently?

6.2.2 . Exploring Additional Views for Tweet Location Prediction

One of the perspectives of this research is to expand our proposed model $FLAIR$ with additional views. Adding more relevant views can result in enhancing our model's accuracy.

In our work, we considered spatial and textual views; however, much other information is coupled with tweets. As a future direction, we can investigate more in attachments, mainly images linked to those tweets. Such images can encapsulate valuable location information and help our model better identify where a tweet was posted. This direction requires building a robust model that can recognize the location from images and then contribute to our proposed multi-view learning model.

6.2.3 . Exploring Additional Features for Air Pollution Estimation

Air pollution estimation and meteorological, as well as traffic data, have a strong relationship, and their integration has the potential to considerably increase the accuracy and reliability of pollution estimation models. Temperature, wind speed, humidity, and atmospheric stability are all critical elements in regulating the dispersion and transport of pollutants in the atmosphere. Incorporating these meteorological factors into air pollution estimating models can help capture the dynamic nature of pollutant dispersion and provide more accurate estimations. Similarly, because driving significantly contributes to air pollution in urban areas, traffic statistics such as vehicle numbers, traffic flow patterns, and road networks can be helpful indicators of pollutant emissions. Incorporating such traffic-related features can aid estimation models in accounting for localized emissions and traffic-related pollution hotspots.

The strong relationship between these features and air pollution opens the directions for future work. We still need to investigate integrating meteorological and traffic data as additional features in our air pollution estimation framework. This can lead to a more comprehensive air pollution map enrichment, hence better estimation.

6.2.4 . Data Privacy

Sensitive information can be inferred from data collected within the Mobile Crowd Sensing (MCS) paradigm. Nowadays, with the spread of MCS, privacy concerns arise in this context based on the nature of the data collected. Such data includes personal details about individuals. While using mobile devices to gather data, there is a legitimate risk of exposing personal stuff, or locations, resulting in privacy concerns.

Researchers are addressing such problems while enabling effective learning from the collected data. For such problems, a future direction could be to use innovative techniques such as federated learning, a decentralized approach to train machine learning models on a portion of data, ensuring that the sensitive data remains on users' devices. Another future direction is using the power of generative models to generate synthetic data. Such models can generate realistic data that looks like the original data while ensuring the removal of personally identifiable information. These privacy-preserving approaches can provide a promising path toward handling privacy concerns in MCS.

Bibliography

[1] Air pollution, world health organization [online]. available:https://www.who.int/health-topics/air-pollution. 2023.

[2] M. Abboud, H. El Hafyani, J. Zuo, K. Zeitouni, and Y. Taher. Micro-environment recognition in the context of environmental crowdsensing. In *Workshops of the EDBT/ICDT Joint Conference, EDBT/ICDT-WS*, 2021.

[3] M. Abboud, H. E. Hafyani, J. Zuo, K. Zeitouni, and Y. Taher. Micro-environment recognition in the context of environmental crowdsensing. *Proceedings of the Workshops of the EDBT/ICDT 2021 Joint Conference*, 2841, 2021.

[4] M. Abboud, K. Zeitouni, and Y. Taher. Fine-grained location prediction of non geo-tagged tweets: a multi-view learning approach. In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery*, pages 82–91, 2022.

[5] K. Aberer, S. Sathe, D. Chakraborty, A. Martinoli, G. Barrenetxea, B. Faltings, and L. Thiele. Opensense: open community driven sensing of environment. In *Proceedings of the ACM SIGSPATIAL International Workshop on GeoStreaming*, pages 39–42, 2010.

[6] A. Agarwal, O. Chapelle, M. Dudík, and J. Langford. A reliable effective terascale linear learning system. *The Journal of Machine Learning Research*, 15(1):1111–1133, 2014.

[7] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, 33(8):831–838, 2015.

[8] S. Aminikhanghahi and D. J. Cook. Enhancing activity recognition using cpd-based activity segmentation. *Pervasive and Mobile Computing*, 53:75 – 89, 2019.

[9] A. Anjomshoaa, F. Duarte, D. Rennings, T. J. Matarazzo, P. deSouza, and C. Ratti. City scanner: Building and scheduling a mobile sensing platform for smart city services. *IEEE Internet of things Journal*, 5(6):4567–4579, 2018.

[10] A. Antoniou, A. Storkey, and H. Edwards. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017.

[11] J. Bao, Y. Zheng, and M. Mokbel. Location-based and preference-aware recommendation using sparse geo-social networking data. In *Proceedings of the 20th international conference on advances in geographic information systems*, pages 199–208, 11 2012. doi: 10.1145/2424321.2424348.

[12] A. Bardoutsos, G. Filios, I. Katsidimas, T. Krousarlis, S. Nikoletseas, and P. Tzamalis. A multidimensional human-centric framework for environmental intelligence: Air pollution and noise in smart cities. In *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pages 155–164. IEEE, 2020.

[13] A. Bekkar, B. Hssina, S. Douzi, and K. Douzi. Air-pollution prediction in smart city, deep learning approach. *Journal of big Data*, 8(1):1–21, 2021.

[14] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, USA:, 1994.

[15] J. Cai, Y. Ge, H. Li, C. Yang, C. Liu, X. Meng, W. Wang, C. Niu, L. Kan, T. Schikowski, et al. Application of land use regression to assess exposure and identify potential sources in pm2. 5, bc, no2 concentrations. *Atmospheric Environment*, 223:117267, 2020.

[16] B. Cao, F. Chen, D. Joshi, and S. Y. Philip. Inferring crowd-sourced venues for tweets. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 639–648. IEEE, 2015.

[17] T. Cassard, G. Jauvion, and D. Lissmyr. High-resolution air quality prediction using low-cost sensors. *arXiv preprint arXiv:2006.12092*, 2020.

[18] B. Chaix, Y. Kestens, K. Bean, C. Leal, N. Karusisi, K. Meghiref, J. Burban, M. Fon Sing, C. Perchoux, F. Thomas, et al. Cohort profile: residential and non-residential environments, individual activity spaces and cardiovascular risk factors and diseases—the record cohort study. *International journal of epidemiology*, 41(5):1283–1292, 2012.

[19] B. Chaix, Y. Kestens, C. Perchoux, N. Karusisi, J. Merlo, and K. Labadi. An interactive mapping tool to assess individual mobility patterns in neighborhood studies. *American journal of preventive medicine*, 43(4):440–450, 2012.

[20] L. Chatzidiakou, A. Krause, M. Kellaway, Y. Han, Y. Li, E. Martin, F. J. Kelly, T. Zhu, B. Barratt, and R. L. Jones. Automated classification of time-activity-location patterns for improved estimation of personal exposure to air pollution. 2022.

[21] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Intell. Res. (JAIR)*, 16:321–357, 06 2002. doi: 10.1613/jair.953.

[22] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, and Y. Liu. Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges and Opportunities. *arXiv:2001.07416 [cs]*, Jan. 2020. URL http://arxiv.org/abs/2001.07416. arXiv: 2001.07416.

[23] W. Cheng, Y. Shen, Y. Zhu, and L. Huang. A neural attention model for urban air quality inference: Learning the weights of monitoring stations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[24] L. Chi, K. H. Lim, N. Alam, and C. Butler. Geolocation prediction in twitter using location indicative words and textual features. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, Osaka, Japan, 12 2016. The COLING 2016 Organizing Committee.

[25] H. Cho and S. M. Yoon. Divide and conquer-based 1d cnn human activity recognition using test data sharpening. *Sensors*, 18(4):1055, 2018.

[26] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[27] W.-H. Chong and E.-P. Lim. Exploiting contextual information for fine-grained tweet geolocation. In *Eleventh International AAAI Conference on Web and Social Media*, 2017.

[28] W.-H. Chong and E.-P. Lim. Tweet geolocation: Leveraging location, user and peer signals. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1279–1288, 2017.

[29] D. Choujaa and N. Dulay. Tracme: Temporal activity recognition using mobile phone data. In *2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, volume 1, pages 119–126. IEEE, 2008.

[30] C. T. Cowie, F. Garden, E. Jegasothy, L. D. Knibbs, I. Hanigan, D. Morley, A. Hansell, G. Hoek, and G. B. Marks. Comparison of model estimates from an intra-city land use regression model with a national satellite-lur and a regional bayesian maximum entropy model, in estimating no2 for a birth cohort in sydney, australia. *Environmental research*, 174:24–34, 2019.

[31] S. Dabiri and K. Heaslip. Inferring transportation modes from gps trajectories using a convolutional neural network. *Transportation research part C: emerging technologies*, 86:360–371, 2018.

[32] H. Deng, G. Runger, E. Tuv, and M. Vladimir. A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153, 2013.

[33] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[34] T. M. T. Do and D. Gatica-Perez. The Places of Our Lives: Visiting Patterns and Automatic Labeling from Longitudinal Smartphone Data. *IEEE Transactions on Mobile Computing*, 13(3):638–648, Mar. 2014. ISSN 1558-0660. doi: 10.1109/TMC.2013.19.

[35] M. Dredze, M. Osborne, and P. Kambadur. Geolocation for twitter: Timing matters. In *Proceedings of the 2016 conference of the north American chapter of the association for computational linguistics: human language technologies*, pages 1064–1069, 2016.

[36] G. Dünnebeil, M. Marjanović, and I. P. Žarko. Approaches to fuse fixed and mobile air quality sensors. In *International symposium on environmental software systems*, pages 71–84. Springer, 2017.

[37] F. Dutt and S. Das. Fine-grained geolocation prediction of tweets with human machine collaboration. *arXiv preprint arXiv:2106.13411*, 2021.

[38] M. Eeftens, R. Meier, C. Schindler, I. Aguilera, H. Phuleria, A. Ineichen, M. Davey, R. Ducret-Stich, D. Keidel, N. Probst-Hensch, et al. Development of land use regression models for nitrogen dioxide, ultrafine particles, lung deposited surface area, and four other markers of particulate matter pollution in the swiss sapaldia regions. *Environmental Health*, 15:1–14, 2016.

[39] H. El Hafyani. *Spatio-temporal data analytics in the context of environmental crowdsensing*. PhD thesis, Université Paris-Saclay, 2022.

[40] H. El Hafyani, M. Abboud, J. Zuo, K. Zeitouni, and Y. Taher. Tell me what air you breath, i tell you where you are. In *17th International Symposium on Spatial and Temporal Databases*, SSTD '21, page 161–165, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384254. doi: 10.1145/3469830.3470914. URL https://doi.org/10.1145/3469830.3470914.

[41] H. El Hafyani, M. Abboud, J. Zuo, K. Zeitouni, Y. Taher, B. Chaix, and L. Wang. Learning the micro-environment from rich trajectories in the context of mobile crowd sensing. *GeoInformatica*, pages 1–44, 2022.

[42] M. Etemad, A. Soares Júnior, and S. Matwin. Predicting transportation modes of gps trajectories using feature engineering and noise removal. In *Advances in Artificial Intelligence: 31st Canadian Conference on Artificial Intelligence, Canadian AI 2018, Toronto, ON, Canada, May 8–11, 2018, Proceedings 31*, pages 259–264. Springer, 2018.

[43] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963, 2019.

[44] H. I. Fawaz, B. Lucas, G. Forestier, C. Pelletier, D. Schmidt, J. Weber, G. I. Webb, L. Idoumghar, P.-A. Muller, and F. Petitjean. Inceptiontime: Finding alexnet for time series classification. *ArXiv*, abs/1909.04939, 2020.

[45] A. Galal and A. Elkorany. Enabling semantic user context to enhance twitter location prediction. In *ICAART (1)*, pages 223–230, 2016.

[46] E. Garcia-Ceja, C. E. Galván-Tejada, and R. Brena. Multi-view stacking for activity recognition with sound and accelerometer data. *Information Fusion*, 40:45–56, Mar. 2018. ISSN 1566-2535. doi: 10.1016/j.inffus.2017.06.004. URL http://www.sciencedirect.com/science/article/pii/S1566253516301932.

[47] J. D. Gonzalez Paule, Y. Moshfeghi, J. M. Jose, and P. Thakuriah. On fine-grained geolocalisation of tweets. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pages 313–316, 2017.

[48] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[49] A. Graves and J. Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18 (5-6):602–610, 2005.

[50] B. Guo, Z. Wang, Z. Yu, Y. Wang, N. Y. Yen, R. Huang, and X. Zhou. Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm. *ACM computing surveys (CSUR)*, 48(1):1–31, 2015.

[51] R. Guo, Y. Qi, B. Zhao, Z. Pei, F. Wen, S. Wu, and Q. Zhang. High-resolution urban air quality mapping for multiple pollutants based on dense monitoring data and machine learning. *International journal of environmental research and public health*, 19(13):8005, 2022.

[52] M. Habermann, M. Billger, and M. Haeger-Eugensson. Land use regression as method to model air pollution. previous results for gothenburg/sweden. *Procedia Engineering*, 115:21–28, 2015.

[53] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[54] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

[55] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[56] K. Hu, A. Rahman, H. Bhrugubanda, and V. Sivaraman. Hazeest: Machine learning based metropolitan air pollution estimation from fixed and mobile sensors. *IEEE Sensors Journal*, 17(11):3517–3525, 2017.

[57] Z. Hu. Spatial analysis of modis aerosol optical depth, pm2. 5, and chronic coronary heart disease. *International journal of health geographics*, 8(1): 1–10, 2009.

[58] B. Hui, H. Chen, D. Yan, and W.-S. Ku. Edge: Entity-diffusion gaussian ensemble for interpretable tweet geolocation prediction. *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 1092–1103, 2021.

[59] Y. M. Idir, O. Orfila, V. Judalet, B. Sagot, and P. Chatellier. Mapping urban air quality from mobile sensors using spatio-temporal geostatistics. *Sensors*, 21(14):4717, 2021.

[60] J. Iglesias, J. Cano, A. M. Bernardos, and J. R. Casar. A ubiquitous activity-monitor to prevent sedentariness. In *2011 ieee international conference on pervasive computing and communications workshops (percom workshops)*, pages 319–321. IEEE, 2011.

[61] H. Iso, S. Wakamiya, and E. Aramaki. Density estimation for geolocation via convolutional mixture density network. *arXiv preprint arXiv:1705.02750*, 2017.

[62] M. Izbicki, V. Papalexakis, and V. Tsotras. Geolocating tweets in any language at any location. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 89–98, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450369763. doi: 10.1145/3357384.3357926. URL https://doi.org/10.1145/3357384.3357926.

[63] L. C. Jatoba, U. Grossmann, C. Kunze, J. Ottenbacher, and W. Stork. Context-aware mobile health monitoring: Evaluation of different pattern recognition methods for classification of physical activity. In *2008 30th annual international conference of the ieee engineering in medicine and biology society*, pages 5250–5253. IEEE, 2008.

[64] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2013.

[65] W. Jiang and Z. Yin. Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1307–1310, 2015.

[66] W. Jiang and Z. Yin. Human Activity Recognition Using Wearable Sensors by Deep Convolutional Neural Networks. In *Proceedings of the 23rd ACM international conference on Multimedia*, MM '15, pages 1307–1310, New York, NY, USA, Oct. 2015. Association for Computing Machinery. ISBN 978-1-4503-3459-4. doi: 10.1145/2733373.2806333. URL https://doi.org/10.1145/2733373.2806333.

[67] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

[68] X. Jurado. *Atmospheric pollutant dispersion estimation at the scale of the neighborhood using sensors, numerical and deep learning models*. PhD thesis, Université de Strasbourg, 2021.

[69] X. Jurado, N. Reiminger, M. Benmoussa, J. Vazquez, and C. Wemmert. Deep learning methods evaluation to predict air quality based on computational fluid dynamics. *Expert Systems with Applications*, 203:117294, 2022.

[70] F. Karim, S. Majumdar, H. Darabi, and S. Harford. Multivariate lstm-fcns for time series classification. *Neural Networks*, 116:237–245, 2019.

[71] Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, pages 1746–1751, 2014.

[72] S. Kinsella, V. Murdock, and N. O'Hare. " i'm eating a sandwich in glasgow" modeling locations with tweets. In *Proceedings of the 3rd international workshop on Search and mining user-generated contents*, pages 61–68, 2011.

[73] I. Kontopoulos, K. Chatzikokolakis, K. Tserpes, and D. Zissis. Classification of vessel activity in streaming data. In *Proceedings of the 14th ACM International Conference on Distributed and Event-based Systems*, pages 153–164, 2020.

[74] G. Kordopatis-Zilos, S. Papadopoulos, and I. Kompatsiaris. Geotagging text content with language models and feature mining. *Proceedings of the IEEE*, PP:1–16, 08 2017. doi: 10.1109/JPROC.2017.2688799.

[75] M. Kranz, A. Möller, N. Hammerla, S. Diewald, T. Plötz, P. Olivier, and L. Roalter. The mobile fitness coach: Towards individualized skill assessment using personalized mobile devices. *Pervasive and Mobile Computing*, 9(2): 203–215, 2013.

[76] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84–90, 2017.

[77] P. Kumar, L. Morawska, C. Martani, G. Biskos, M. Neophytou, S. Di Sabatino, M. Bell, L. Norford, and R. Britter. The rise of low-cost sensing for managing air pollution in cities. *Environment international*, 75: 199–205, 2015.

[78] S. Lal, L. Tiwari, R. Ranjan, A. Verma, N. Sardana, and R. Mourya. Analysis and classification of crime tweets. *Procedia Computer Science*, 167:1911–1919, 2020. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2020.03.211. URL https://www.sciencedirect.com/science/article/pii/S1877050920306761. International Conference on Computational Intelligence and Data Science.

[79] B. Languille, V. Gros, N. Bonnaire, C. Pommier, C. Honoré, C. Debert, L. Gauvin, S. Srairi, I. Annesi-Maesano, B. Chaix, et al. A methodology for the characterization of portable sensors for air quality measure with the goal of deployment in citizen science. *Science of the Total Environment*, 708: 134698, 2020.

[80] O. D. Lara and M. A. Labrador. A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials*, 15(3): 1192–1209, 2012.

[81] J. H. Lau, L. Chi, K.-N. Tran, and T. Cohn. End-to-end network for Twitter geolocation prediction and hashing. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 744–753, Taipei, Taiwan, Nov. 2017. Asian Federation of Natural Language Processing. URL https://aclanthology.org/I17-1075.

[82] V.-D. Le, T.-C. Bui, and S.-K. Cha. Spatiotemporal deep learning model for citywide air pollution interpolation and prediction. In *2020 IEEE international conference on big data and smart computing (BigComp)*, pages 55–62. IEEE, 2020.

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

[84] K. Lee, R. Ganti, M. Srivatsa, and L. Liu. When twitter meets foursquare: tweet location prediction using foursquare. In *11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2014.

[85] S. Li, Y. Li, and Y. Fu. Multi-view time series classification: A discriminative bilinear projection approach. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 989–998, 2016.

[86] W. Li, P. Serdyukov, A. P. de Vries, C. Eickhoff, and M. Larson. The where in the tweet. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2473–2476, 2011.

[87] Z. Li, K.-F. Ho, H.-C. Chuang, and S. H. L. Yim. Development and intercity transferability of land-use regression models for predicting ambient pm 10, pm 2.5, no 2 and o 3 concentrations in northern taiwan. *Atmospheric Chemistry and Physics*, 21(6):5063–5078, 2021.

[88] C. C. Lim, H. Kim, M. R. Vilcassim, G. D. Thurston, T. Gordon, L.-C. Chen, K. Lee, M. Heimbinder, and S.-Y. Kim. Mapping urban air quality using mobile sampling with low-cost sensors and machine learning in seoul, south korea. *Environment international*, 131:105022, 2019.

[89] K. H. Lim, S. Karunasekera, A. Harwood, and Y. George. Geotagging tweets to landmarks using convolutional neural networks with text and posting time. In *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion*, New York, NY, United States, 03 2019. Association for Computing Machinery. doi: 10.1145/3308557.3308691.

[90] L. Lin, Y. Liang, L. Liu, Y. Zhang, D. Xie, F. Yin, and T. Ashraf. Estimating pm2. 5 concentrations using the machine learning rf-xgboost model in guanzhong urban agglomeration, china. *Remote Sensing*, 14(20):5239, 2022.

[91] J. Lines, S. Taylor, and A. Bagnall. HIVE-COTE: The Hierarchical Vote Collective of Transformation-based Ensembles for Time Series Classification. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 1041–1046, 2016.

[92] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[93] L. Liu, Y. Peng, S. Wang, M. Liu, and Z. Huang. Complex activity recognition using time series pattern dictionary learned from ubiquitous sensors. *Inf. Sci.*, 340-341:41–57, 2016.

[94] Z. Liu and Y. Huang. Where are you tweeting? a context and user movement based approach. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1949–1952, 2016.

[95] T. Lu, J. D. Marshall, W. Zhang, P. Hystad, S.-Y. Kim, M. J. Bechle, M. Demuzere, and S. Hankey. National empirical models of air pollution using microscale measures of the urban environment. *Environmental Science & Technology*, 55(22):15519–15530, 2021.

[96] R. Ma, X. Xu, H. Y. Noh, P. Zhang, and L. Zhang. Generative model based fine-grained air pollution inference for mobile sensing systems. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, pages 426–427, 2018.

[97] R. Ma, N. Liu, X. Xu, Y. Wang, H. Y. Noh, P. Zhang, and L. Zhang. A deep autoencoder model for pollution map recovery with mobile sensing networks. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, pages 577–583, 2019.

[98] R. Mahajan and V. Mansotra. Predicting geolocation of tweets: using combination of cnn and bilstm. *Data Science and Engineering*, 6:402–410, 2021.

[99] S. Mailler, L. Menut, D. Khvorostyanov, M. Valari, F. Couvidat, G. Siour, S. Turquety, R. Briant, P. Tuccella, B. Bessagnet, et al. Chimere-2017: From urban to hemispheric chemistry-transport modeling. *Geoscientific Model Development*, 10(6):2397–2423, 2017.

[100] J. D. Mazimpaka and S. Timpf. Trajectory data mining: A review of methods and applications. *Journal of Spatial Information Science*, 2016(13):61–99, 2016.

[101] X. Meng, L. Chen, J. Cai, B. Zou, C.-F. Wu, Q. Fu, Y. Zhang, Y. Liu, and H. Kan. A land use regression model for estimating the no2 concentration in shanghai, china. *Environmental research*, 137:308–315, 2015.

[102] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[103] P. Mishra. Geolocation of tweets with a BiLSTM regression model. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 283–289, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Linguistics (ICCL). URL https://aclanthology.org/2020.vardial-1.27.

[104] A. Murga, Y. Sano, Y. Kawamoto, and K. Ito. Integrated analysis of numerical weather prediction and computational fluid dynamics for estimating cross-ventilation effects on inhaled air quality inside a factory. *Atmospheric Environment*, 167:11–22, 2017.

[105] G. Nayak, V. Mithal, X. Jia, and V. Kumar. Classifying multivariate time series by learning sequence-level discriminative patterns. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 252–260. SIAM, 2018.

[106] F. J. Ordóñez and D. Roggen. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1):115, 2016.

[107] O. Ozdikis, H. Ramampiaro, and K. Nørvåg. Locality-adapted kernel densities for tweet localization. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1149–1152, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. doi: 10.1145/3209978.3210109. URL https://doi.org/10.1145/3209978.3210109.

[108] L. Pappalardo, F. Simini, G. Barlacchi, and R. Pellungrini. scikit-mobility: A python library for the analysis, generation, and risk assessment of mobility data. *Journal of Statistical Software*, 103(1):1–38, 2022. doi: 10.18637/jss.v103.i04. URL https://www.jstatsoft.org/index.php/jss/article/view/v103i04.

[109] J. D. G. Paule, Y. Sun, and Y. Moshfeghi. On fine-grained geolocalisation of tweets and real-time traffic incident detection. *Information Processing & Management*, 56(3):1119–1132, 2019.

[110] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[111] J. Pärkkä, M. Ermes, P. Korpipää, J. Mäntyjärvi, J. Peltola, and I. Korhonen. Activity classification using realistic data from wearable sensors. *IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society*, 10(1):119–128, Jan. 2006. ISSN 1089-7771. doi: 10.1109/titb.2005.856863.

[112] X. Qin, L. Platisa, T. H. Do, E. Tsiligianni, J. Hofman, V. P. L. Manna, N. Deligiannis, and W. Philips. Context-based analysis of urban air quality using an opportunistic mobile sensor network. In *Science and Technologies for Smart Cities: 5th EAI International Summit, SmartCity360, Braga, Portugal, December 4-6, 2019, Proceedings*, pages 285–300. Springer, 2020.

[113] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[114] A. Rebeiro-Hargrave, N. H. Motlagh, S. Varjonen, E. Lagerspetz, P. Nurmi, and S. Tarkoma. Megasense: Cyber-physical system for real-time urban air quality monitoring. In *2020 15th IEEE conference on industrial electronics and applications (ICIEA)*, pages 1–6. IEEE, 2020.

[115] K. Rehrl, S. Gröchenig, and S. Kranzinger. Why did a vehicle stop? a methodology for detection and classification of stops in vehicle trajectories. *International Journal of Geographical Information Science*, 34(10): 1953–1979, 2020.

[116] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL http://arxiv.org/abs/1908.10084.

[117] A. P. Ruiz, M. Flynn, and A. Bagnall. Benchmarking Multivariate Time Series Classification Algorithms. *arXiv:2007.13156 [cs, stat]*, July 2020. URL http://arxiv.org/abs/2007.13156. arXiv: 2007.13156.

[118] A. P. Ruiz, M. Flynn, J. Large, M. Middlehurst, and A. Bagnall. The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 35(2):401–449, 2021.

[119] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak. Convolutional, long short-term memory, fully connected deep neural networks. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4580–4584. Ieee, 2015.

[120] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[121] J. L. Santiago, F. Martín, and A. Martilli. A computational fluid dynamic modelling approach to assess the representativeness of urban monitoring stations. *Science of the total environment*, 454:61–72, 2013.

[122] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015.

[123] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014.

[124] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[125] D. Simpson, A. Benedictow, H. Berge, R. Bergström, L. D. Emberson, H. Fagerli, C. R. Flechard, G. D. Hayman, M. Gauss, J. E. Jonson, et al. The emep msc-w chemical transport model–technical description. *Atmospheric Chemistry and Physics*, 12(16):7825–7865, 2012.

[126] L. S. Sloan, J. Morgan, W. Housley, M. L. Williams, A. Edwards, P. Burnap, and O. F. Rana. Knowing the tweeters: Deriving sociologically relevant demographics from twitter. *Sociological Research Online*, 18:74 – 84, 2013.

[127] L. Snyder, M. Karimzadeh, R. Chen, and D. Ebert. City-level geolocation of tweets for real-time visual analytics. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery*, pages 85–88, 11 2019. ISBN 9781450369572. doi: 10.1145/3356471.3365243.

[128] J. Song and M. E. Stettler. A novel multi-pollutant space-time learning network for air pollution inference. *Science of The Total Environment*, 811: 152254, 2022.

[129] J. Song, K. Han, and M. E. Stettler. Deep-maps: Machine-learning-based mobile air pollution sensing. *IEEE Internet of Things Journal*, 8(9):7649–7660, 2020.

[130] R. Tavenard, J. Faouzi, G. Vandewiele, F. Divo, G. Androz, C. Holtz, M. Payne, R. Yurchak, M. Rußwurm, K. Kolar, and E. Woods. Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, 21(118):1–6, 2020. URL http://jmlr.org/papers/v21/20-091.html.

[131] A. Tella and A.-L. Balogun. Prediction of ambient pm10 concentration in malaysian cities using geostatistical analyses. *Journal of Advanced Geospatial Science & Technology*, 1(1):115–127, 2021.

[132] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pages 4489–4497. IEEE, 2015.

[133] C. M. Vera-Burgos and D. R. Griffin Padgett. Using twitter for crisis communications in a natural disaster: Hurricane harvey. *Heliyon*, 6 (9):e04804, 2020. ISSN 2405-8440. doi: https://doi.org/10.1016/j.heliyon.2020.e04804. URL https://www.sciencedirect.com/science/article/pii/S2405844020316479.

[134] B. Wang, T. Jiang, X. Zhou, B. Ma, F. Zhao, and Y. Wang. Time-Series Classification Based on Fusion Features of Sequence and Visualization. *Applied Sciences*, 10(12):4124, Jan. 2020. doi: 10.3390/app10124124. URL https://www.mdpi.com/2076-3417/10/12/4124.

[135] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu. Deep Learning for Sensor-based Activity Recognition: A Survey. *Pattern Recognition Letters*, 119: 3–11, Mar. 2019. ISSN 01678655. doi: 10.1016/j.patrec.2018.02.010. URL http://arxiv.org/abs/1707.03502. arXiv: 1707.03502.

[136] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision (ECCV)*, pages 20–36. Springer, 2016.

[137] S. Wang, J. Cao, and P. S. Yu. Deep learning for spatio-temporal data mining: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3681–3700, 2022. doi: 10.1109/TKDE.2020.3025580.

[138] H. Wei, H. Zhou, J. Sankaranarayanan, S. Sengupta, and H. Samet. Delle: Detecting latest local events from geotagged tweets. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Analytics for Local Events and News*, pages 1–10, 2019.

[139] L. Wei and E. Keogh. Semi-supervised time series classification. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 748–753, New York, NY, USA, Aug. 2006. Association for Computing Machinery. ISBN 978-1-59593-339-3. doi: 10.1145/1150402.1150498. URL https://doi.org/10.1145/1150402.1150498.

[140] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241 – 259, 1992. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(05)80023-1. URL http://www.sciencedirect.com/science/article/pii/S0893608005800231.

[141] T. Yao, Y. Pan, Y. Li, and T. Mei. Spatio-temporal attention based on long short-term memory networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4849–4857, 2015.

[142] L. Ye and E. Keogh. Time series shapelets: A New Primitive for Data Mining. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*, page 947–956, 2009.

[143] J. Yoon, D. Jarrett, and M. van der Schaar. Time-series generative adversarial networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/c9efe5f26cd17ba6216bbe2a7d26d490-Paper.pdf.

[144] C. Zhang, G. Zhou, Q. Yuan, H. Zhuang, Y. Zheng, L. Kaplan, S. Wang, and J. Han. Geoburst: Real-time local event detection in geo-tagged tweet streams. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 513–522, New York, NY, United States, 07 2016. Association for Computing Machinery. doi: 10.1145/2911451.2911519.

[145] C. Zhang, L. Liu, D. Lei, Q. Yuan, H. Zhuang, T. Hanratty, and J. Han. Triovecevent: Embedding-based online local event detection in geo-tagged tweet streams. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 595–604, New York, NY, United States, 08 2017. Association for Computing Machinery. doi: 10.1145/3097983.3098027.

[146] D. Zhang and S. S. Woo. Real time localized air quality monitoring and prediction through mobile and fixed iot sensing network. *IEEE Access*, 8: 89584–89594, 2020.

[147] M. Zhang and A. A. Sawchuk. Motion primitive-based human activity recognition using a bag-of-features approach. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, pages 631–640, 2012.

[148] M. Zhang and A. A. Sawchuk. Motion primitive-based human activity recognition using a bag-of-features approach. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, IHI '12, pages 631–640, New York, NY, USA, Jan. 2012. Association for Computing Machinery. ISBN 978-1-4503-0781-9. doi: 10.1145/2110363.2110433. URL https://doi.org/10.1145/2110363.2110433.

[149] Q. Zhang, Y. Han, V. O. Li, and J. C. Lam. Deep-air: A hybrid cnn-lstm framework for fine-grained air pollution estimation and forecast in metropolitan cities. *IEEE Access*, 2022.

[150] X. Zhang, Y. Gao, J. Lin, and C.-T. Lu. TapNet: Multivariate Time Series Classification with Attentional Prototypical Network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6845–6852, 2020.

[151] Y. Zhang, X. Zhang, L. Wang, Q. Zhang, F. Duan, and K. He. Application of wrf/chem over east asia: Part i. model evaluation and intercomparison with mm5/cmaq. *Atmospheric Environment*, 124:285–300, 2016.

[152] X. Zheng, J. Han, and A. Sun. A survey of location prediction on twitter. *IEEE Transactions on Knowledge and Data Engineering*, 30(9):1652–1671, Sep. 2018. ISSN 1558-2191. doi: 10.1109/TKDE.2018.2807840.

[153] Y. Zheng. Trajectory data mining: An overview. *ACM Trans. Intell. Syst. Technol.*, 6(3), May 2015. ISSN 2157-6904. doi: 10.1145/2743025. URL https://doi.org/10.1145/2743025.

[154] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma. Understanding mobility based on GPS data. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 312–321. Association for Computing Machinery, New York, NY, USA, Sept. 2008. ISBN 978-1-60558-136-1. URL https://doi.org/10.1145/1409635.1409677.

[155] Y. Zheng, L. Liu, L. Wang, and X. Xie. Learning transportation mode from raw gps data for geographic applications on the web. In *Proceedings of the 17th international conference on World Wide Web*, pages 247–256, 2008.

[156] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma. Recommending friends and locations based on individual location history. *ACM Transactions on the Web (TWEB)*, 5(1):1–44, 2011.

[157] Y. Zheng, F. Liu, and H.-P. Hsieh. U-air: When urban air quality inference meets big data. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1436–1444, 2013.

[158] L. Zhou, D. Zhang, C. Yang, and Y. Wang. Harnessing social media for health information management. *Electronic Commerce Research and Applications*, 27, 12 2017. doi: 10.1016/j.elerap.2017.12.003.

[159] Z.-H. Zhou. *Ensemble Methods: Foundations and Algorithms*. CRC press, 2012.

[160] J. Zuo, K. Zeitouni, and Y. Taher. Incremental and adaptive feature exploration over time series stream. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 593–602, 2019.

[161] J. Zuo, K. Zeitouni, and Y. Taher. Exploring interpretable features for large time series with se4tec. In *Proc. EDBT*, pages 606–609, 2019.

Appendix A
Résumé en Français

Avec la diffusion rapide des appareils de l'Internet des objets (IoT), l'enrichissement des données est crucial. Cependant, ces données sont souvent massives, complexes et non structurées, ce qui rend difficile l'extraction d'information pertinente. Les algorithmes d'apprentissage automatique fournissent une solution robuste pour l'extraction automatique de motifs, de tendances et de corrélations dans les données de l'IoT, augmentant ainsi leur valeur. Cet enrichissement facilite l'analyse prédictive et améliore le processus décisionnel. Cette thèse se place dans le contexte de la collecte participative (Mobile Crowd Sensing ou MCS) de la qualité de l'air et utilise l'apprentissage automatique pour enrichir les données. Nous nous intéressons à trois thématiques : l'enrichissement par le micro-environnement des données de MCS vu que la qualité de l'air dépend fortement du contexte ; l'enrichissement par des sources de données externes, ici Twitter, pour aider à comprendre l'exposition à la pollution ; et enfin, la combinaison des données de stations fixes et de MCS pour améliorer les cartes de pollution. Nous décrivons les motivations et les challenges de chacune de ces thématiques avant de résumer nos contributions.

En effet, le micro-environnement influe considérablement la qualité de l'air à laquelle les personnes sont exposées. Il est de plus préférable de réduire la charge des participants en automatisant la détection des micro-environnements, évitant des annotations manuelles. Par ailleurs, l'enrichissement des cartes de pollution générées par les stations fixes avec des données de MCS est un autre sujet étudié dans cette thèse. La combinaison des deux sources pourrait améliorer l'estimation de la pollution dans les endroits non couverts. Cependant, la collecte opportuniste des données rend la tâche complexe.

Le troisième sujet est l'enrichissement des données par des données externes. L'usage des tweets est très utile notamment dans la détection d'événements localisés. Cependant, peu de tweets sont géolocalisés et peuvent être directement exploités, d'où la nécessité de prédire la localisation par apprentissage automatique.

Notre objectif est de développer des méthodes adéquates de fouille de données et concernent :

- La reconnaissance automatique du micro-environnement dans le contexte du MCS.

- La prédiction de la localisation des tweets en vue de la détection d'événements.

- L'amélioration des cartes de la qualité de l'air par l'intégration des données de MCS collectées en mode opportuniste.

Les solutions ont été mises en œuvre et évaluées à partir de données réelles complexes à grande échelle. Elles traitent les problèmes inhérents à ces données incomplétude, imprécision, rareté de l'annotation et différencient le traitement selon le type de données. Les questions de recherche dans le cadre de ce travail sont les suivantes :

- Comment traiter des données de séries temporelles multivariées complexes, avec des valeurs manquantes et hétérogènes tout en automatisant la reconnaissance du micro-environnement ?
- Comment prédire plus précisément l'emplacement d'un tweet à une granularité fine ?
- Comment estimer la qualité de l'air dans les zones non couvertes tout en enrichissant les cartes générées par les stations fixes avec des données de MCS opportunistes ?

Afin d'établir des modèles d'enrichissement de données efficaces dans le contexte de l'IoT, notre travail apporte les contributions majeures suivantes :

- Il propose un pipeline de bout en bout pour la reconnaissance du micro-environnement en utilisant l'approche d'apprentissage multi-vue.
- Il implémente un framework hybride et un algorithme de prédiction pour la géolocalisation des tweets à granularité fine.
- Il offre une méthodologie pour améliorer l'enrichissement des stations fixes et des données de MCS collectées de manière opportuniste afin d'améliorer la précision de l'estimation de la qualité de l'air.