Majority Voting Based Multi-Task Clustering of Air Quality Monitoring Network in Turkey

Goksu Tuysuzoglu 1, Derya Birant 2,* and Aysegul Pala 3

1 Graduate School of Natural and Applied Sciences, Dokuz Eylul University, 35390 Izmir, Turkey; goksu.tuysuzoglu@ceng.deu.edu.tr
2 Department of Computer Engineering, Dokuz Eylul University, 35390 Izmir, Turkey
3 Department of Environmental Engineering, Dokuz Eylul University, 35390 Izmir, Turkey; aysegul.pala@deu.edu.tr
* Correspondence: derya@cs.deu.edu.tr; Tel.: +90-232-301-7401

Received: 1 March 2019; Accepted: 14 April 2019; Published: 18 April 2019

Abstract: Air pollution, which is the result of the urbanization brought by modern life, has a dramatic impact on the global scale as well as local and regional scales. Since air pollution has important effects on human health and other living things, the issue of air quality is of great importance all over the world. Accordingly, many studies based on classification, clustering and association rule mining applications for air pollution have been proposed in the field of data mining and machine learning to extract hidden knowledge from environmental parameters. One approach is to model a region in a way that cities having similar characteristics are determined and placed into the same clusters. Instead of using traditional clustering algorithms, a novel algorithm, named Majority Voting based Multi-Task Clustering (MV-MTC), is proposed and utilized to consider multiple air pollutants jointly. Experimental studies showed that the proposed method is superior to five well-known clustering algorithms: K-Means, Expectation Maximization, Canopy, Farthest First and Hierarchical clustering methods.

Keywords: air pollution; multi-task clustering; air quality; machine learning; data mining

1. Introduction

Air pollution is now recognized as an important problem all over the world. It can be referred as a mixture of multiple pollutants that vary in size and composition. Air pollutants (also referred to as “criteria pollutants”) are commonly grouped as particulate matters such as PM$_{10}$ and PM$_{2.5}$, and ground-level pollutants such as ozone (O$_3$), carbon monoxide (CO), sulfur dioxide (SO$_2$), nitrogen oxides (NO) and nitrogen dioxide (NO$_2$).

It is known that air pollution has negative impacts on human health, range of vision, materials, and plant and animal health. Air pollutants trigger or worsen chronic diseases such as asthma, pneumonia, heart attack, bronchitis and other respiratory problems. Since particulate matters are very small and light, they tend to stay in the air longer than the heavier particles. This increases the likelihood of humans and animals inhaling these particles through respiration. Due to their small size, these particles can easily pass through the nose and throat and penetrate the lungs, and some may even enter the circulatory system. Smoke, a gas mixture of solid and liquid particles resulting from non-burned carbon materials such as solid fuels and fuel oil, is a variety of air pollution and has a reducing effect on range of visibility. Air pollution also has a destructive and disturbing effect on artistic and architectural structures. On plants, they can be lethal and prevent their growth. Thus, high concentrations of air pollutants can harm human health, adversely influence environment, and also cause property damage [1–4].
Due to the seriousness of the issue, air pollution control policies require systematic monitoring and evaluation of air quality. The causes of air pollution should be investigated and necessary precautions should be taken in accordance with the findings. Therefore, it is very important to develop an appropriate tool to understand the air quality in an area. For this purpose, effective methods are continuously developed with new studies.

In this context, this study aimed at examining the air quality monitoring stations in Turkey according to their similarities in terms of five air pollutants, PM$_{10}$, SO$_2$, NO$_2$, NO and O$_3$, and making appropriate inferences based on the analysis of the levels of air pollutants measured at these stations in the time interval from 1 November 2017 to 1 November 2018. In this way, city areas with similar air pollution behavior can be identified so that decision-making authority can canalize emission sources to be located to the regions in need. To perform the experiments, a novel algorithm, named majority-based multi-task clustering (MV-MTC), is proposed, instead of applying traditional clustering algorithms, to benefit from the common decision coming from different pollutant sources. The novelty of this study is the implementation of multi-task clustering (MTC) in the field of environmental science and the examination of air pollution in Turkey with this method for the first time.

The proposed algorithm (MV-MTC) was compared with popular clustering algorithms, namely K-Means, Expectation Maximization, Canopy, Farthest First and Hierarchical clustering methods, in terms of sum of squared error (SSE). The experimental results obtained in this study indicate that the proposed approach produces better clusters than standard clustering algorithms by considering relationships among multiple air pollutants jointly.

The remainder of this study is organized as follows. In Section 2, a detailed literature survey investigating the studies using data mining methods to deal with the air quality control of Turkey is given in addition to the recent studies on the proposed method of multi-task clustering. In Section 3, background information on the applied methodology used in the experiments is explained. The proposed MTC technique and dataset description are mentioned in Sections 4 and 5, respectively. The experimental studies are presented and the obtained results are discussed in Section 6. Lastly, concluding remarks, a brief summary and future directions are given.

2. Related Work

The monitoring stations located in nearby area are characterized by the same specific air pollution characteristics. Many studies have been done in the literature using this information. Data mining and machine learning are intensely applied to environmental subjects to identify interesting structure in large amount of environmental data, where the structure finds patterns, rules, predictive models and relationships among the data. Ignaccolo, Ghigo and Giovenali [5] classified the air quality monitoring network in Piemonte (Northern Italy) using functional cluster analysis based on Partitioning around Medoids algorithm and considering three air pollutants, namely NO$_2$, PM$_{10}$, and O$_3$, to classify sites in homogeneous clusters and identify the representative ones. Barrero, Orza, Cabello and Cantón [6] analyzed and made experiments on the variations of PM$_{10}$ concentrations at 43 stations in the air quality monitoring network of the Basque Country to group them according to their common characteristics. They implemented the autocorrelation function and K-means clustering. Similarly, Lu, He and Dong [7] used principal component analysis and cluster analysis for the management of air quality monitoring network of Hong Kong and for the reduction of associated expenses.

In Turkey, the importance of environmental issues has also gained much attention and studies related to air quality increasingly continue in this direction. Several of environmental data mining studies mentioned thus far on “air quality in Turkey” are compared in Table 1 by displaying the year of the publication, the target pollutants of the study, dataset content used in the experiments, the aim of the study, which data mining task was applied and which algorithms/methods were implemented as well as performance metrics to evaluate the results of the applied methodology. The bold notation in the Algorithms/Methods column shows the algorithm which performs the best among the others. According to the findings, most of the experiments are done using the measurements
of PM$_{10}$ concentrations [8–13] and prediction of air pollutant amount is the main goal. In addition to pollution data, some of the studies also integrate meteorological data such as temperature, wind speed, wind direction, pressure and humidity into the problem domain [8,11–13].

Multi-task learning (MTL) is a learning technique in which useful information contained in a number of relevant tasks is leveraged to help improve the overall performance of all tasks. All of these tasks or at least a subset of them are assumed to be related to each other. In many of the applications, it is found that learning these tasks jointly leads to performance improvement compared with learning them individually. In most fields comprising computer vision, bioinformatics, health informatics, speech and natural language processing, web applications and ubiquitous computing, MTL is used to enhance the overall performance of the applications involved. Learning paradigms including supervised learning (e.g., classification or regression problems) [14–16], unsupervised learning [17–23], semi-supervised learning [24–27], active learning [28–31], reinforcement learning [32–35], multi-view learning [21,36–38], and graphical models [39–41] are generally combined with MTL [42,43].

Multi-task classification and multi-task clustering are two well-known types of multi-task learning recently presented in the literature. Wang, Yan, Lu, Zhang and Li [44] use multi-task classification in the prediction of air pollution particles by implementing a deep multi-task learning framework. On the other hand, multi-task clustering has not been studied until now for the air quality management, neither in the environmental science.

There is an issue to be addressed: “what to share” while learning multiple tasks. The form of sharing type determines which knowledge sharing among all the tasks could occur. Usually, there are three forms of sharing: feature, instance and parameter. Feature-based MTL aims to learn common features among different tasks. Instance-based MTL identifies useful data objects in a task for other tasks and then shares knowledge via the identified instances. Parameter-based MTL uses model parameters in a task to help learn model parameters in other tasks [42]. The proposed method in this study (MV-MTC) is among the popularly applied unsupervised learning schemes of instance-based MTL applications.

MTC has been applied in many different areas including bioinformatics, text mining, web mining, image mining, daily activity recognition and so on [18–23,45]. The resulting clustering template of MTC has generally outperformed any single clustering algorithm’s outputs. Table 2 presents a brief list of studies in which different MTC algorithms are proposed and applied in various subject areas. It is experimentally proven that MTC algorithms provide remarkable performance when compared to single task learners.
### Table 1. Summary of the data mining studies with “air pollution in Turkey” as the main subject.

| Ref. | Year | The Target Pollutants | The Dataset Content | Aim | Task | Algorithms/Methods | Performance Metrics |
|------|------|-----------------------|---------------------|-----|-----|-------------------|---------------------|
| [8]  | 2018 | PM$_{10}$             | PD: hourly densities of CO, NO, NO$_x$, NO$_2$, O$_3$, and SO$_2$; M: T, WD, WS, P, RH, min. T, max. T, max. WD and max. WS | Estimation of the density of PM$_{10}$ at Istanbul using datasets with imbalanced class distribution | Prediction | LRC, RFC, ETC, and GBC | Accuracy, AUROC |
| [46] | 2018 | SO$_2$                | The time series of weekly SO$_2$ concentrations | Forecast air pollution in 65 monitoring stations | Clustering | FTS based on FKM, FCMF and GKF | RMSE and PB |
| [9]  | 2017 | PM$_{10}$, SO$_2$     | Four months SO$_2$ and PM$_{10}$ concentrations in Istanbul | Evaluation of the results of regular measurements of PM$_{10}$ and SO$_2$ concentrations in the city of Istanbul | Prediction | ANOVA | SSE and MSE, Kolmogorov-Smirnov Test, t-test |
| [47] | 2017 | AQI (monthly basis)   | SO$_2$, NO$_x$, CO, O$_3$, and PM$_{10}$ (hourly and daily basis) | Determination of the AQI in Ankara | Classification | FLA | |
| [10] | 2016 | PM$_{10}$             | Weekly PM$_{10}$ concentrations in numerous stations in Turkey | Predicting a model to estimate PM$_{10}$ concentrations for 130 monitoring stations | Clustering | FCARM, AR | MAPE, Dickey-Fuller test, p-value |
| [11] | 2014 | PM$_{10}$             | Hourly observations consisted of: M: max. T, avg. T, std. T, max. WS, avg. WS, std. WS, max. WD, avg. WD, std. WD, degree, max. RH, avg. RH, std. RH, PD: max. PM$_{10}$, avg. PM$_{10}$, std. PM$_{10}$ | Forecast maximum Daily PM$_{10}$ concentrations one day ahead in Duzce | Prediction | ANN (MLP), SWR, MLR | IA, FMB, RMSE, R$^2$ |
| [48] | 2013 | SO$_2$                | The amount of SO$_2$ in Ankara | Seasonal fuzzy time series forecasting in Ankara | Clustering | Fuzzy C-means combined with ANN and SARIMA | RMSE, MAPE |
| [12] | 2011 | PM$_{10}$, SO$_2$     | M: T, WS and WD, RH, P, S, C, R | Prediction of the daily and hourly mean concentrations of PM$_{10}$ and SO$_2$ pollutants in the regions of Istanbul | Prediction | CNN, PER | t, d, the Mean Bias Error, MAE and RMSE, t-test (p-value) |
| [13] | 2010 | PM$_{10}$, SO$_2$, CO | M: P, Day T, Night T, H, WS, WD, PD, SO$_2$, CO, PM$_{10}$; GC, Day of Week, Date | Forecasting SO$_2$, CO and PM$_{10}$ levels 3 days in advance for the Besiktas district of Istanbul | Prediction | GFM_NN | Band Error |

LRC, Logistic Regression Classifier; T, temperature; FKM, Fuzzy K-Medoid; MLP, Multi-Layer Perceptron; RFC, Random Forest Classifier; RH, Relative Humidity (H); P, Pressure; R$^2$, coefficient of determination; ETC, Extra Trees Classifier; AQI, Air Quality Index; FTS, Fuzzy Time Series; FMB, Fractional Mean Bias; GBC, Gradient Boosting Classifier; RMSE, Root Mean Squared Error; IA, Index-of-Agreement; MLR, Multiple Linear Regression; PD, Pollution Data; FCMF, FTS Models based on Fuzzy C-means; FCARM, Fuzzy C-Auto Regressive Model; SWR, Stepwise Regression; M, Meteorological Data; GKF, Gustafson–Kessel; AR, Autoregressive model; GKF, Gustafson–Kessel; WS, Wind Speed; PB, Percent Bias; MAPE, Mean Absolute Percentage Error; GC, General Condition; WD, Wind Direction; ANOVA, Analysis of Variance; FLA, Fuzzy Logic Algorithm; ANN, Artificial Neural Network; r, Correlation Coefficient; MAE, Mean Absolute Error; S, Sunshine; R, Rainfall; C, Cloudiness; d, Index of Agreement; PER, Statistical Persistence Method; CNN, Cellular Neural Network; AUROC, The Area under the Receiver Operating Characteristic; SARIMA, Seasonal Autoregressive Integrated Moving Average; GFM_NN, Geographic Forecasting Models using Neural Networks; Percentage Error; GC, General Condition; WD, Wind Direction; ANOVA, Analysis of Variance; FLA, Fuzzy Logic Algorithm; ANN, Artificial Neural Network; r, Correlation Coefficient; MAE, Mean Absolute Error; S, Sunshine; R, Rainfall; C, Cloudiness; d, Index of Agreement; PER, Statistical Persistence Method; CNN, Cellular Neural Network; AUROC, The Area under the Receiver Operating Characteristic; SARIMA, Seasonal Autoregressive Integrated Moving Average; GFM_NN, Geographic Forecasting Models using Neural Networks.
Develop a co-clustering based multi-task multi-view clustering framework which integrates within-view-task clustering, multi-view relationship learning and multi-task relationship learning.

MTCTKI, KM, KKM, Ncut-GK, Ncut-SNN, LSSMTC, DMTFC, SMT–NMF, MTCTKI0

Clustering Accuracy, NMI

Re-weighting the distance between samples in different tasks by learning a shared subspace and selecting the nearest neighbors for each sample from the other tasks.

MTCTKI, KM, KKM, Ncut-GK, Ncut-SNN, LSSMTC, DMTFC, SMT-NMF, MTCTKI0

Clustering Accuracy, NMI

Identify common and context-specific aspects of genome architecture.

Arboretum-Hi-C, KM, HC, SC

DBI, SI, D, Number of enriched clusters, log $p$ value of ANOVA

Automated HiP-2Cells Classification

CMTL and the other 28 methods presented in the HiP-2Cells Classification contest held at the 2012 International Conference on Pattern Recognition

Accuracy

Daily living analysis from visual data gathered from wearable cameras

EMD-MTC with linear and rbf kernel denoted as CEMD-MTC and KEMD-MTC respectively; KM, KKM, CNMF and SemiNMF; SemiEMD-MTC, K5emiEMD-MTC, and the LSMTC method

Clustering Accuracy, NMI

Identifying and avoiding negative effects of the boosting process of MBC and also dealing with nonlinear separable data in the clustering of documents

SMBC and S-MKC, KM and KKM, MBC

Clustering Accuracy, NMI

NMF, Nonnegative Matrix Factorization; Ncut–GK, Normalized Cut with Gaussian Kernel Similarity; BiCo, Bipartite Graph Co-clustering; DMTFC, Convex Discriminative Multi-task Feature Clustering; MTCTKI0, MTCTKI without using Shared Subspace; NSC, Normalized Spectral Clustering; SNMTF, Semi-nonnegativity Matrix Tri-factorization; CoRe, Co-regularized Multi-view Spectral Clustering; CoTr, Co-trained Multi-view Spectral Clustering; SC3, Single-cell Consensus Clustering; SNN-Cliq, Shared Nearest Neighbor Cliq; LSSMTC, Learning the Shared Subspace for Multi-task Clustering; DBI, Davies–Bouldin Index; SI, Silhouette Index; DI, Delta Contact Count; SMT–NMF, Symmetric Multi-task Non-negative Matrix Factorization; ARI, Adjusted Rand Index; HC, Hierarchical Clustering; SC, Spectral Clustering; EMD-MTC, Earth Mover’s Distance Regularized Multi-task Clustering; NMI, Normalized Mutual Information; KM, K-means; S-MKC, Smart Multi-task Kernel Clustering; KKM, Kernel K-means; CMTL, Clustered Multi-task Learning; MBC, Multitask Bregman Clustering; SMBC, Smart Multi-task Bregman Clustering; BMTMVC, Bipartite Graph based Multi-task Multi-view Clustering; Ncut–SNN, Normalized Cut with Shared Nearest Neighbor Similarity; MTCTKI, Multi-task Clustering by Transferring Knowledge of Instances; scVDMC, Variance-Driven Multitask Clustering of Singlecell RNA-seq Data; SMTMVC, Semi-nonnegative Matrix Tri-factorization based Multi-task Multi-view Clustering; Arboretum-Hi-C, Multi-task Spectral Clustering Algorithm for Comparative Analysis of Hi-C Data.

The proposed MV-MTC algorithm has many advantages over existing multi-task clustering methods. First, some methods have a complicated theoretical foundation, which leads to implementation difficulties. For instance, graph-based methods and matrix factorization for nonnegative data are commonly applied (e.g., [21]) by implementing a semi-nonnegative matrix tri-factorization method to co-cluster the data in each view of each task. Likewise, the algorithm introduced in [49] has several sophisticated steps: feature extraction, clustering-based regularization, convex relaxation, and optimization. Spectral clustering that uses the eigenvectors of the Laplacian of a graph for clustering is another way to implement multi-task clustering [20]. In addition to graph-based methods (e.g., [19]), multi-task clustering can be performed by reweighting the distance between data points in different tasks by learning a shared subspace. In this way, clustering operation for each individual task is generated by selecting the nearest neighbors for each sample from the other tasks in the learned shared subspace.
Second, some proposed MTC methods (e.g., scVDMC [18] and Arboretum-Hi-C [20]) were designed as field-specific methods and have a valid use only for bioinformatics data to analyze the genome architecture or to simultaneously capture the differentially expressed genes. These methods are not suitable for the analysis of geographical data (or for the identification of air pollution levels of a region).

Third, our algorithm is particularly advantageous since it does not need any a priori information about the data. However, Yan et al. [22] proposed a novel algorithm, named Convex Multi-task Clustering (CMTC), which requires some a-priori knowledge about the data relationship.

Fourth, some multi-task clustering algorithms (i.e., [23]) require additional parameters and the results change significantly with different parameter values. It makes it difficult to use the algorithm, since the user should determine the optimal parameter for each problem. Our algorithm does not require any additional parameter tuning.

Fifth, the execution time of some multi-task clustering algorithms (e.g., [45]) increases exponentially when the input data increase. However, our algorithm (MV-MTC) requires computation time that grows linearly with the number of instances, clusters and tasks.

Sixth, our proposed method can effectively avoid the imbalance of cluster distribution by merging multiple models according to majority voting. In addition, the MV-MTC framework can effectively reduce clustering errors by selecting the best clustering algorithm for the problem under consideration.

Our goal is to propose an easily implemented, generally applicable, fast, prior knowledge- and parameter-independent multi-task clustering method. Unlike existing methods, the algorithm in this paper is a new kind of multi-task clustering method that is much easier to understand and implement by taking the jointly obtained common decision from different tasks using cluster labels. It was developed as a new method that can appeal to every area rather than being specific to one area (e.g., [18,20]).

Different types of MTC algorithms have been proposed. For instance, multi-task multi-view clustering [21] is presented to handle the learning problem of multiple related tasks with one or more common views. Each view is associated with one task or multiple related tasks, the inter-task knowledge is transferred to one another, and multi-task and multi-view relationships are exploited to improve clustering performance. In [21], it is applied for webpage and image mining operations under clustering framework.

3. Materials and Methods

In this section, applied methodologies and datasets for experiments in addition to used platforms are presented. The overall goal of the used techniques was to create clusters with a consistent set of similar behavioral points by ensuring the maximum similarities in intra-cluster objects while keeping inter-cluster differences high. The clustering algorithms or techniques used in this study were: K-Means, Expectation Maximization, Canopy, Farthest First and Hierarchical clustering in addition to the proposed technique Multi-task clustering.

3.1. K-Means Clustering

Consider a dataset \( D = \{o_1, o_2, \ldots, o_n\} \) where each \( o_i \) represents an object as a \( p \)-dimensional explanatory variable and \( n \) is the number of objects (instances) in the dataset. Assume that the problem domain is to be divided into \( k \) clusters combination of which is represented as a vector \( C_{KM} = \{C_1, C_2, \ldots, C_k\} \) and the centroids of \( k \) clusters are denoted by \( \mu = \{m_1, m_2, \ldots, m_k\} \).

The first step is to assign \( k \) points as cluster centers at random. The distance between each data point \( o_i \) and each cluster centroids \( m_j \), where \( i = \{1, \ldots, n\} \) and \( j = \{1, \ldots, k\} \), are calculated using one of the distance metrics such as Euclidean, Manhattan, Chebyshev, Minkowski distance, etc. as \( \arg\min_j \text{dist}(o_i, m_j) \) to find the nearest cluster for the respective instance to be assigned. New cluster centroids are calculated by \( m_j = \left(1/n_j\right) \sum_{o_i \in C_j} o_i \) where \( n_j \) denotes the number of objects in cluster \( j \), \( C_j \). This process iteratively continues until no data point changes cluster membership. According to the method used
for the initialization of the process, different techniques instead of random initialization can be used such as K-Means++, Farthest First or Canopy.

3.2. Expectation Maximization Clustering

It extends the K-Means paradigm in a different way. While the K-Means algorithm assigns each data point to a cluster, each object in the Expectation-Maximization (EM) model is assigned to each cluster according to a weight representing the probability of membership. In other words, there is no definite limit between clusters and new centers are calculated in terms of weighted measures [50]. EM clusters data points using a finite mixture density model, i.e., normal distribution, of $k$ probability distributions, where each distribution represents a cluster.

As in the K-Means clustering, the process starts with selecting cluster centroids randomly. The procedure continues with two steps to refine the parameters (i.e., clusters) iteratively based on statistical modeling: Expectation (E) step and Maximization (M) step [51]. In Step E, a function to determine the probability of cluster membership of an instance is generated using the present estimate for the attributes using Equation (1) where $p(o_i|C_j)$ follows the normal distribution and $i = \{1, \ldots, n\}$, $j = \{1, \ldots, k\}$.

$$
P(o_i \in C_j) = p(C_j|o_i) = \frac{p(C_j)p(o_i|C_j)}{p(o_i)},
$$

(1)

Step M is applied as in Equation (2) for re-estimating the model parameters by discovering the attributes which maximizes the expected log-likelihood found in Step E. The iterative process continues until obtaining the optimal value.

$$
m_j = \frac{1}{n} \sum_{i=1}^{n} \frac{o_i P(o_i \in C_j)}{\sum_j P(o_i \in C_j)},
$$

(2)

3.3. Canopy Clustering

The general application of Canopy is on the preprocessing step of other clustering algorithms such as K-Means or Hierarchical clustering to speed up the process in the case of large datasets [52]. The procedure uses two distance metrics $T_1 > T_2$ to be used for later processing and a list of data points to cluster. Initial canopy center is determined randomly from one of the data points and then distances of all other instances to this canopy center are approximated. The instances whose distance value fall within the threshold of $T_1$ is placed into a canopy while the data points whose distance value fall within the threshold of $T_2$ are removed from the list. These removed ones are excluded from being selected as a new canopy center or creating new canopies. The process iteratively continues until the list is empty.

3.4. Farthest First Clustering

It is one of the variants of K-Means clustering where each cluster centroid is selected in turn at the point furthest from the existing cluster centers. This point must lie within the data area. This significantly boosts the speed of clustering in general due to the need of less reassignment and modification [53].

3.5. Hierarchical Clustering

Hierarchical clustering is used to group data objects into a tree of clusters either by bottom-up (agglomerative) or top-down (divisive) fashion [49]. In agglomerative version, each instance of the dataset is put into its own cluster initially and all of these atomic clusters are merged continuously until a single cluster is formed to hold all data points inside or if there is a termination condition. Divisive version is just the opposite of agglomerative clustering because it begins the process with a single cluster where all data points are placed and the later steps are the subdivision of the cluster.
into smaller distinct ones until a termination criterion is satisfied such as a predetermined number of clusters is obtained.

According to the distance calculation method between different clusters, there are many link types used in Hierarchical clustering such as Single (the minimum link that is the closest distance between any items of two different clusters), Complete (the maximum link that is the largest distance between any items of two different clusters), Average (the average distance between the elements of two clusters), Mean (the mean distance of merged cluster) and Centroid (the distance from one centroid to another).

### 4. Multi-task Clustering

A task is generally referred to the construction of a model using a specific dataset for a single target or for a sub-goal. In this sense, “multiple tasks” could mean the modeling of multiple output targets simultaneously by using task-related datasets and by considering task relations. Depending on the definition of “multiple tasks”, we can define multi-task clustering as follows: multi-task clustering (MTC) is a process of generating global clusters that are shared by the multiple related tasks. MTC is desired to merge information among tasks to improve the clustering performance of individual tasks. The most important aspect in MTC is to discover the shared information among tasks. In this paper, a novel algorithm, named Majority Voting based Multi-task Clustering (MV-MTC), is proposed to provide this aspect.

Consider the unlabeled dataset \( D = \{o_1, o_2, \ldots, o_n\} \) where each \( o_i \) represents an object as a \( p \)-dimensional explanatory variable and \( n \) is the number of objects (instances) in the dataset. Assume that the problem domain consists of \( r \) different tasks \( T = \{t_1, t_2, \ldots, t_r\} \), each of which is represented as \( t_i \).

In the first step of the algorithm, the instance set allotted to each task should be properly clustered using one of the traditional clustering algorithms. For \( r \) different tasks, let us denote the resulting clustering assignments as \( C = \{C_{t_1}, C_{t_2}, \ldots, C_{t_r}\} \) where \( C_{t_i} = \{c_{t_i,1}, c_{t_i,2}, \ldots, c_{t_i,k}\} \) for the predetermined number of clusters as \( k \) and each \( c_i \) consists of different \( o_i \)'s from the dataset \( D \). To take the joint decision from all \( C_{t_i} \)'s, a common factor should be determined because the same cluster names do not have to represent the same clustering structure among the task groups. We need to determine common cluster labels meaning the same information through all tasks.

In this context, after clustering instances of each task by one of the single clustering algorithms, all clusters are labeled from the common label set \( L = \{L_1, L_2, \ldots, L_k\} \) as in Table 3 in terms of the mean weights of intra-cluster objects and \( k \) cluster labels for \( k \) clusters are produced according to the cluster weights. To illustrate if we have three clusters, the heaviest one, the medium one and the lightest one can be labeled as “\( L_3 \)”, “\( L_2 \)” and “\( L_1 \)”, respectively. The same procedure is applied for \( r \) tasks. As shown in the following example, all instances in the dataset are labeled with a suitable cluster label \( L_i \) for each task \( t_i \). As the final stage, as in the majority voting approach, the most common cluster label among all tasks for a given instance \( o_i \) is selected as the final cluster assignment. Therefore, the novel MTC algorithm is called Majority Voting based Multi-task Clustering (MV-MTC).

| Instance | Cluster Label Assignments for \( t_1 \) | Cluster Label Assignments for \( t_2 \) | \( \ldots \) | Cluster Label Assignments for \( t_{r-1} \) | Cluster Label Assignments for \( t_r \) |
|----------|---------------------------------|---------------------------------|---------|---------------------------------|---------------------------------|
| \( o_1 \) | \( L_1 \) | \( L_2 \) | \( \ldots \) | \( L_{k-1} \) | \( L_k \) |
| \( o_2 \) | \( L_2 \) | \( L_2 \) | \( \ldots \) | \( L_{k-1} \) | \( L_3 \) |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| \( o_{n-1} \) | \( L_k \) | \( L_2 \) | \( \ldots \) | \( L_1 \) | \( L_4 \) |
| \( o_n \) | \( L_1 \) | \( L_{k-2} \) | \( \ldots \) | \( L_{k-1} \) | \( L_k \) |
This study proposes two novel concepts: single-task clusters and multi-task clusters. In the first phase, the proposed algorithm discovers local clusters (single-task clusters) from each task data separately, and, in the second phase, these local clusters are combined to produce the global result (multi-task clusters).

**Definition 1.** (Single-Task Clusters) Single-task clusters are groups of instances discovered from the data partition \( D_t \) of a particular task \( t \), i.e., \( D = \bigcup_{t=1}^{r} D_t \), and denoted by \( C_t = \{c_1, c_2, \ldots, c_k\} \), where \( k \) is the number of clusters.

**Definition 2.** (Multi-Task Clusters) Given \( r \) tasks \( T = \{t_i\}_{i=1}^{r} \) where all the tasks are related but not identical, multi-task clusters, which are denoted by \( C = \{C_{t_1}, C_{t_2}, \ldots, C_{t_r}\} \), are groups of instances that mostly appear in the same level of the clusters of the tasks.

Based on these definitions, it is possible to say that there are two elementary factors for multi-task clustering. The first factor is the definition of task. Many real world problems consist of a number of related subtasks. For instance, PM\(_{10}\), SO\(_2\), NO\(_2\), NO and O\(_3\) air pollutants can be considered as the tasks of air quality monitoring problem. The second factor is the definition of ensemble method to combine multiple tasks. In our study, we used majority voting mechanism, which selects the cluster that is the one with the most votes.

To figure out the rationale behind the algorithm, the example scenario in Tables 4 and 5 explain the process step by step. In the first stage, the dataset \( D \), which is full of instances with only one feature, is given. There are three tasks \( (t_1, t_2 \text{ and } t_3) \) and the aim is to group the dataset into three clusters by taking the joint decision from each task. The attribute value of instances can change according to different tasks. The next step is applied for clustering instances by one of the clustering algorithms simultaneously for each task. Instances are properly assigned to one of three clusters \( (C_1, C_2 \text{ or } C_3) \). On the other hand, we need to determine a common decision point on the cluster groups of different tasks to get the final cluster assignments. Therefore, three labels \( (L_1, L_2 \text{ and } L_3) \) are used to generalize the clusters and mean the same groupings under different tasks according to average intra-cluster weights. In the final part, after instances are labeled with the new label set for every task \( (C_{t_1}, C_{t_2} \text{ and } C_{t_3}) \), majority voting scheme is applied to obtain final cluster labels for MV-MTC algorithm.

Figure 1 displays the general framework of multi-task clustering algorithm where each \( t_i \) shows single task of the task space and \( D \) is the unlabeled data. The main purpose is to ensure that the instances in the clusters that are created before the MV-MTC result remain in the same set in the final step. The number of instances remaining in the same cluster is maximized according to Equation (3) where \( C_{ij}(o_r) \) means that the instance \( o_r \) is the member of cluster \( c_j \) of task \( t_i \) and MV – MTC \( j \) indicates the resulting cluster \( c_j \) of MV-MTC algorithm. The pseudo code of the proposed algorithm is given in Algorithm 1.

\[
\max \{ \sum_{i=1}^{r} \sum_{j=1}^{k} \sum_{r=1}^{n} 1 : C_{ij}(o_r) \in \text{MV} - \text{MTC}_j, \quad o_r \in C_{ij} \},
\]  
(3)
Figure 1. The general MV-MTC clustering scheme.

Table 4. An example showing MV-MTC algorithm step by step.

| Ins. ID | Task $t_1$ | Task $t_2$ | Task $t_3$ |
|---------|------------|------------|------------|
| 1       | 3          | 23         | 43         |
| 2       | 10         | 15         | 30         |
| 3       | 21         | 40         | 54         |
| 4       | 9          | 32         | 89         |
| 5       | 18         | 14         | 72         |
| 6       | 12         | 27         | 28         |
| 7       | 6          | 24         | 26         |
| 8       | 4          | 41         | 22         |
| 9       | 17         | 33         | 79         |
| 10      | 19         | 28         | 58         |
| 11      | 10         | 15         | 47         |
| 12      | 3          | 18         | 73         |

(1) Dataset $D$ and its instance values in terms of three tasks.

| Cluster ID | Instance Assignments |
|------------|----------------------|
| $C_1$      | $\{2, 4, 6, 11\}$   |
| $C_2$      | $\{3, 5, 9, 10\}$   |
| $C_3$      | $\{1, 7, 8, 12\}$   |

(2) Instance assignments to clusters under different tasks.

| Cluster ID | Common Label Assignments |
|------------|--------------------------|
| $C_1$      | $L_2$ 10.25 $L_3$ 36.50 $L_2$ 50.50 |
| $C_2$      | $L_3$ 18.75 $L_1$ 15.50 $L_3$ 78.25 |
| $C_3$      | $L_1$ 4.00 $L_2$ 25.50 $L_1$ 26.50 |

(3) Determination of the common cluster labels according to average intra-cluster weights.
As shown in Algorithm 1, the methodology is made up of four steps. In the first step, single-task clusters are generated by taking each individual task into consideration. Step 2 is performed to calculate intra-cluster weights under different tasks. In Step 3, cluster labels are assigned to clusters according to their weight values, assigning $L_1$ to the cluster which has the lowest mean value, and then increasing the label values until giving $L_k$ to the highest one. The last step is the place where joint decision from different tasks is taken by applying a majority voting mechanism. As a result, all data points are placed into the most suitable clusters and final cluster labels are assigned from joint decision.

**Algorithm 1: Majority Voting based Multi-task Clustering (MV-MTC)**

**Inputs:** Dataset $D = \{o_1, o_2, \ldots, o_n\}$  
Task space $T = \{t_1, t_2, \ldots, t_r\}$  
CA: a clustering algorithm  
Cluster label set $L = \{L_1, L_2, \ldots, L_k\}$  
$n$: the number of instances  
k: the number of clusters  
r: the number of tasks

**Process:**

1. **Step 1: Clustering according to task $t_i$**
   1. for $i = 1$ to $r$
   2. $C_i = CA(D, t_i)$
   3. $C.add(C_i)$

2. **Step 2: Determine average intra-cluster weights**
   4. for each $C_i$ in $C$
   5. for $i = 1$ to $k$
   6. for each $o$ in $C_i$
   7. sum = sum + $o$ // value of the instance
   8. $m_i = \text{sum}/|C_i|$

3. **Step 3: Label each cluster $C_i$ for all tasks according to $\mu_i$ values**
   9. for each $C_i$ in $C_i$
   10. for $i = 1$ to $k$
   11. $L_{ci} = L_{\text{index}(m_i)}$ // an appropriate cluster label from $L$ in terms of the index of $m_i$

4. **Step 4: Obtaining joint decision**
   12. for $i = 1$ to $n$
   13. for $j = 1$ to $r$
   14. $L(o_i) = \arg\max_{L_{ci}, L_{cj}} \sum_{j \in \{1, 2, \ldots, r\}} 1$

// final cluster assignment of $o_i$ where each $L_{cj}$ is the cluster label of $o_i$ in task $t_j$ and $j \in \{1, 2, \ldots, r\}$
5. Dataset Description and Used Platforms

There are seven geographical regions, namely as Eastern Anatolia, Central Anatolia, Southeastern Anatolia, Blacksea, Mediterranean, Aegean, and Marmara, in Turkey and numerous air quality monitoring stations (AQMS) at each region. This study was conducted on 49 AQMSs from 32 provinces which are from different regions. The features of each station are listed in Tables 6 and 7 by showing the name of AQMS, in which city it is located, the corresponding county of the city, longitude and latitude information, network type (urban/rural/industrial), and which air pollutants are regularly measured in there.

The National Air Quality Monitoring Network of Turkey includes 330 Air Quality Monitoring Stations. The air quality of all provinces in the country is monitored. To facilitate public access to information on air quality, the monitoring results are published online at the website of http://laboratuvar.cevre.gov.tr [54]. In all of the air pollution measurement stations, SO$_2$ and PM$_{10}$ parameters are measured; in addition, NO, NO$_2$, NO$_x$, CO and O$_3$ are measured automatically in many of them. In this study, all of the AQMSs were investigated and 49 out of 330 stations were selected because the aforementioned air pollutants (PM$_{10}$, SO$_2$, NO$_2$, NO and O$_3$) are regularly measured in these stations together.

Since the data become roughly periodic after one-year period, only one year of (November 2017 to November 2018) data were used in the experiments. The pollutant concentrations are mean values of daily (24 h) measurements. The application was developed using Weka open source data mining library [55] on Visual Studio.
Table 6. The selected air quality monitoring stations (AQMS) and their features.

| ID | AQMS NAME       | CITY        | COUNTY      | LONGITUDE  | LATITUDE  | TYPE    | THE MEASURED POLLUTANTS                                      |
|----|-----------------|-------------|-------------|------------|-----------|---------|------------------------------------------------------------|
| 1  | Adana-Catalan   | Adana       | Yüreğir     | 35.2619    | 37.1864   | Rural   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$               |
| 2  | Adana-Dogankent | Adana       | Yüreğir     | 35.3491    | 36.8545   | Rural   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$               |
| 3  | Adana-Meteoroloji | Adana       | Karaisalı   | 35.3440    | 37.0041   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO           |
| 4  | Adana-Valilik   | Adana       | Seyhan      | 35.3124    | 36.9991   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO           |
| 5  | Agri-Dogubeyazit | Agri        | Merkez      | 43.0396    | 39.5476   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO           |
| 6  | Agri-Patnos     | Agri        | Patnos      | 42.8530    | 39.2365   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO           |
| 7  | Ankara-Kecioren | Ankara      | Kecioren    | 32.8628    | 39.9672   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO           |
| 8  | Ankara-Can-MTHM | Ankara      | Can         | 27.0498    | 40.2093   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 9  | Artvin          | Artvin      | Merkez      | 41.3182    | 41.1752   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 10 | Bartın          | Bartın      | Merkez      | 32.3564    | 41.6248   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 11 | Bayburt         | Bayburt     | Merkez      | 40.2255    | 40.2558   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 12 | Canakkale-Biga Icdas | Canakkale | Biga       | 27.1072    | 40.4173   | Industrial | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO          |
| 13 | Canakkale-Can-MTHM | Canakkale | Can         | 27.0498    | 40.2093   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 14 | Erzincan        | Erzincan    | Merkez      | 39.4590    | 39.7430   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 15 | Erzurum-Palandoken | Erzurum    | Palandoken  | 41.2728    | 39.8982   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 16 | Erzurum-Pasinler | Erzurum    | Pasinler    | 41.5721    | 40.0335   | Rural   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 17 | Giresun-Gemilercekegi | Giresun | Merkez      | 38.3985    | 40.9144   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 18 | Gumushane       | Gumushane   | Merkez      | 39.4808    | 40.4608   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 19 | Hatay-Iskenderun | Hatay      | Iskenderun  | 36.2239    | 36.7141   | Industrial | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, CO          |
| 20 | Igdir           | Igdir       | Merkez      | 44.0536    | 39.9261   | Urban   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 21 | Iğdır-Aralik    | Iğdır       | Aralik      | 44.6209    | 39.7868   | Rural   | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 22 | Istanbul-Basaksehir-MTHM Istanbul | Basaksehir | 28.7898    | 41.0954   | Industrial | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
| 23 | Istanbul-Basaksehir-MTHM Istanbul | Esenyurt | 28.6688    | 41.0192   | Urban     | PM$_{10}$, SO$_2$, NO$_2$, NO, O$_3$, NO$_X$, PM$_{2.5}$  |
### Table 7. The selected air quality monitoring stations (AQMS) and their features.

| ID  | AQMS NAME                     | CITY          | COUNTY       | LONGITUDE  | LATITUDE  | TYPE       | THE MEASURED POLLUTANTS                                                                 |
|-----|-------------------------------|---------------|--------------|------------|------------|-----------|----------------------------------------------------------------------------------------|
| 28  | Karabuk-Kardemir I            | Karabuk       | Merkez       | 32.6274    | 41.1920    | Industrial | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                 |
| 29  | Kars-Istasyon Mah.             | Kars          | Merkez       | 43.1044    | 40.6050    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                 |
| 30  | Kirklaireli-Limankoy-MTHM      | Kirklaireli   | Limankoy     | 28.0559    | 41.8852    | Rural      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                   |
| 31  | Kirsehir                      | Kirsehir      | Merkez       | 34.1686    | 39.1381    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO, PM\(_{2.5}\)                   |
| 32  | Kocaeli-Gebe-MTHM             | Kocaeli       | Gebe         | 29.4365    | 40.8108    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO, PM\(_{2.5}\)                   |
| 33  | Kocaeli-Kirkef-MTHM           | Kocaeli       | Kirkef       | 29.7888    | 40.7461    | Industrial | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), PM\(_{2.5}\), PM\(_{2.5}\) Flow Rate|
| 34  | Kocaeli-Yenikoy-MTHM          | Kocaeli       | Yenikoy      | 29.8844    | 40.7042    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\)                                     |
| 35  | Manisa-Soma                   | Manisa        | Soma         | 27.6129    | 39.1814    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                  |
| 36  | Ordu-Unye                     | Ordu          | Unye         | 37.2802    | 41.1214    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), PM\(_{2.5}\) Flow Rate             |
| 37  | Rize                          | Rize          | Merkez       | 40.5328    | 41.0217    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\)                                     |
| 38  | Rize-Ardesen                  | Rize          | Ardesen      | 41.0475    | 41.1273    | Rural      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), PM\(_{2.5}\)                     |
| 39  | Samsun-Atakum                 | Samsun        | Atakum       | 36.2965    | 41.3253    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), PM\(_{2.5}\), PM\(_{10}\) Flow Rate, PM\(_{2.5}\) Flow Rate |
| 40  | Seyyar-1(06 THL 77)-Malatya Arapgir | Malatya     | Arapgir      | 38.4878    | 39.0457    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\)                                                |
| 41  | Seyyar-2 (06 THL 79)-Sincan OSB | Ankara       | Sincan       | 33.0364    | 39.0908    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\)                                                |
| 42  | Seyyar-4 (06 DV 9975)-Iskarya Kizildag | Iskarya    | Kizildag     | 31.3549    | 38.0442    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\)                                                |
| 43  | Tekirdag-Corlu-MTHM           | Tekirdag      | Corlu        | 27.8154    | 41.1806    | Industrial | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\)                                                |
| 44  | Trabzon-Akaabat               | Trabzon       | Akaabat      | 39.5923    | 41.0143    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                |
| 45  | Trabzon-Uzungol               | Trabzon       | Uzungol      | 40.2890    | 40.6173    | Rural      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\)                                                |
| 46  | Trabzon-Valilik               | Trabzon       | Merkez       | 39.7123    | 41.0059    | Urban      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                |
| 47  | Yalova-Armutlu-MTHM           | Yalova        | Armutlu      | 28.7845    | 40.5292    | Rural      | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), PM\(_{2.5}\)                     |
| 48  | Zonguldak-Enerji Lise         | Zonguldak     | Catalagzi    | 31.8801    | 41.4964    | Industrial | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                |
| 49  | Zonguldak-Enerji Terpekooy    | Zonguldak     | Catalagzi    | 31.9374    | 41.5269    | Industrial | PM\(_{10}\), SO\(_{2}\), NO, O\(_{3}\), NO\(_{x}\), CO                                |
6. Experimental Results

In this study, the proposed MTC method, MV-MTC, was compared with traditional clustering algorithms K-Means (KM), Expectation Maximization (EM), Hierarchical Clustering (HIER), Canopy and Farthest First (FFFIRST). Each task was clustered by the selected algorithm and then their decision from consensus was obtained in MV-MTC framework. Performance evaluation was done via sum of squared error (SSE) calculation. Before constructing the model, data were normalized and missing data imputation was performed using the mean values.

The number of clusters, \( k \), was selected as 10% of the number of instances in the dataset, therefore it was 5. Distance metric was chosen as Euclidean distance. To take the joint decision from each single clustering algorithm, each cluster was labeled according to the weights calculated as the average value of the instances of intra-cluster. According to this scheme, five cluster labels were determined as “\( L_1 \)”, “\( L_2 \)”, “\( L_3 \)”, “\( L_4 \)” and “\( L_5 \)”. Table 8 displays the average normalized weight of each cluster in terms of different air pollutants and their corresponding cluster labels. As a result of the joint decision of different tasks, where evaluation of PM\(_{10}\), SO\(_2\), NO\(_2\), NO and O\(_3\) pollutants were assumed as a new task, final cluster assignments were done.

To evaluate the performance of the applied methodology, values of sum of squared error were calculated. SSE is the sum of the squared differences between each observation and its group’s mean. In Equation (4), \( o_i \) represents an instance of dataset \( D \), \( C_j \) represents the \( j \)th cluster, \( m_j \) is the centroid value of the specified cluster \( j \) where \( o_i \) is assigned and \( k \) is the number of cluster. Total SSE of a method is the sum of all separate SSE calculations coming from distinct clusters.

\[
SSE = \sum_{j=1}^{k} \sum_{o \in C_j} (o_i - m_j)^2, \tag{4}
\]

| Cluster 1 | Cluster 2 | Cluster 3 | Cluster 4 | Cluster 5 |
|-----------|-----------|-----------|-----------|-----------|
| Weight    | Label     | Weight    | Label     | Weight    | Label     | Weight    | Label     | Weight    | Label     |
| PM\(_{10}\) | 0.2002    | \( L_2 \) | 0.3142    | \( L_3 \) | 0.1309    | \( L_4 \) | 0.3982    | \( L_4 \) | 0.5977    | \( L_5 \) |
| SO\(_2\)  | 0.0713    | \( L_1 \) | 0.2328    | \( L_3 \) | 0.1497    | \( L_2 \) | 0.5799    | \( L_5 \) | 0.4996    | \( L_4 \) |
| NO\(_2\)  | 0.0902    | \( L_1 \) | 0.2177    | \( L_2 \) | 0.3322    | \( L_3 \) | 0.4396    | \( L_4 \) | 0.6593    | \( L_5 \) |
| NO        | 0.0726    | \( L_1 \) | 0.1382    | \( L_2 \) | 0.5186    | \( L_3 \) | 0.1781    | \( L_3 \) | 0.1783    | \( L_4 \) |
| O\(_3\)   | 0.4208    | \( L_3 \) | 0.3237    | \( L_2 \) | 0.7482    | \( L_5 \) | 0.4873    | \( L_4 \) | 0.2082    | \( L_1 \) |

Final assignments are obtained both by MV-MTC and single clustering algorithms. Clustering algorithms is applied on each single task, i.e., the model is formed just by taking one pollutant into consideration. The SSE results of different pollutants under different algorithms are shown as \( C_{\text{pollutantName}} \) where “pollutantName” is one of the pollutants (PM\(_{10}\), SO\(_2\), NO\(_2\), NO or O\(_3\)) in Table 8. \( C_{\text{ALL}} \) is the average SSE value coming from all pollutants. KM is applied with two different versions in terms of initialization method used. KM with the random initialization is denoted as KM and KM initialized with K-Means++ is displayed as KM++. Hierarchical clustering is implemented with different link types among clusters. HIER\(_{\text{Sing}}\), HIER\(_{\text{Comp}}\), HIER\(_{\text{Avg}}\), HIER\(_{\text{Mean}}\) and HIER\(_{\text{Centro}}\) represent the hierarchical clustering types with single link, complete link, average link, mean link and centroid link, respectively. The bold notations in Table 9 show the best results in the respective rows.

We can conclude that the proposed MV-MTC method outperforms all single clustering algorithms that similar AQMSs are assigned to the same cluster group more accurately when multi-task clustering is applied. Besides, the most promising output of MV-MTC is obtained by KM++. In the case of single clustering algorithms, EM performs the best among the other applied techniques.

Final cluster assignments after performing MV-MTC with KM++ are shown in Figure 2. It points out the geographical locations of AQMSs in the map of Turkey with different colored markers where each color represents a cluster.
After single-task clustering step, the merging process takes the runtime cost of $O(n^2 \times n \times k \times I)$. This time complexity indicates that the proposed MV-MTC algorithm requires computation time that grows linearly with the number of instances, clusters and tasks. Thus, the execution time of the algorithm will still be reasonable even if we process a large volume of data.

Table 9 shows the execution time (in seconds) to perform MV-MTC algorithm in terms of different clustering methods. Single task clustering results are also shown as $C_{\text{PollutantName}}$, and $C_{\text{ALL}}$ represents the sum of the running time of all single task clustering results. Experiments were performed on a desktop computer with Intel Core i7-6700 3.40 GHz processor and 8 GB memory. In each experiment, the algorithms were executed 10 times and then the average values were reported. The empirical results show that the running time of the proposed K-means++ algorithm under MV-MTC framework is better than EM and hierarchical clustering algorithms. Besides, the MV-MTC algorithm has comparable speed with the traditional clustering algorithms when we compare $C_{\text{ALL}}$ and MV-MTC results on the datasets.

In MV-MTC approach, a clustering algorithm is performed for $t$ tasks and a merging operation is done in the final step (“majority voting”). In this study, the best results were obtained using the KMeans++ algorithm. The time complexity of K-Means++ is $O(n \times k + n \times k \times I)$, where $n$ is the number of instances, $k$ is the number of clusters, and $I$ is the number of iterations needed for convergence [56]. After single-task clustering step, the merging process takes the runtime cost of $O(t \times n)$, where $t$ is the number of tasks. Considering this, the total time complexity of MV-MTC algorithm is $O(n \times k + n \times k \times I + t \times n)$. This time complexity indicates that the proposed MV-MTC algorithm requires computation time that grows linearly with the number of instances, clusters and tasks. Thus, the execution time of the algorithm will still be reasonable even if we process a large volume of data.

Table 10 shows the execution time (in seconds) to perform MV-MTC algorithm in terms of different clustering methods. Single task clustering results are also shown as $C_{\text{PollutantName}}$, and $C_{\text{ALL}}$ represents the sum of the running time of all single task clustering results. Experiments were performed on a desktop computer with Intel Core i7-6700 3.40 GHz processor and 8 GB memory. In each experiment, the algorithms were executed 10 times and then the average values were reported. The empirical results show that the running time of the proposed K-means++ algorithm under MV-MTC framework is better than EM and hierarchical clustering algorithms. Besides, The MV-MTC algorithm has comparable speed with the traditional clustering algorithms when we compare $C_{\text{ALL}}$ and MV-MTC results on the datasets.

Table 9. The results of the performance evaluation in terms of SSE values.

| PollutantName | KM | KM++ | EM | Canopy | FFirst | HIERMean | HIERComp | HIERAvg | HIERMean | HIERCentro |
|---------------|----|------|----|--------|--------|----------|----------|----------|----------|------------|
| PM10          | 396.40 | 406.53 | 392.42 | 476.39 | 415.03 | 532.51 | 411.91 | 516.16 | 456.02 | 526.03 |
| NO2           | 260.31 | 194.96 | 226.03 | 415.04 | 214.27 | 216.13 | 204.56 | 214.27 | 214.27 | 214.27 |
| NO            | 367.95 | 315.79 | 319.25 | 406.52 | 356.39 | 437.90 | 317.32 | 437.90 | 373.54 | 437.90 |
| CO            | 335.17 | 351.29 | 294.79 | 426.31 | 305.73 | 319.82 | 303.40 | 303.40 | 319.82 | 319.82 |
| ALL           | 393.83 | 394.42 | 371.31 | 478.08 | 418.25 | 645.29 | 402.39 | 410.27 | 514.38 | 446.52 |
| MV-MTC        | 350.73 | 322.60 | 320.76 | 440.47 | 341.93 | 430.33 | 327.92 | 376.40 | 375.61 | 388.91 |
|               | 113.80 | 108.12 | 116.16 | 122.84 | 134.41 | 161.96 | 124.92 | 161.96 | 144.16 | 161.96 |

Figure 2. Results from the implementation of multi-task clustering obtained by KM++.
**Table 10.** Comparisons of different clustering methods in terms of execution time (in seconds).

|          | KM | KM+  | EM  | Canopy | FFirst | HIERSing | HIERComp | HIERAvg | HIERMean | HIERCentro |
|----------|----|------|-----|--------|--------|-----------|----------|---------|----------|------------|
| C\textsubscript{PM10} | 0.04 | 0.05 | 0.32 | 0.03  | 0.03   | 0.11      | 0.11     | 0.11    | 0.11     | 0.16       |
| C\textsubscript{SO2}   | 0.04 | 0.03 | 0.36 | 0.04  | 0.03   | 0.11      | 0.11     | 0.11    | 0.11     | 0.17       |
| C\textsubscript{NO2}  | 0.03 | 0.04 | 0.29 | 0.03  | 0.03   | 0.11      | 0.11     | 0.11    | 0.11     | 0.16       |
| C\textsubscript{NO}   | 0.04 | 0.04 | 0.26 | 0.03  | 0.03   | 0.11      | 0.11     | 0.11    | 0.11     | 0.17       |
| C\textsubscript{O3}   | 0.05 | 0.04 | 0.30 | 0.03  | 0.03   | 0.11      | 0.11     | 0.11    | 0.11     | 0.16       |
| C\textsubscript{ALL}  | 0.20 | 0.20 | 1.53 | 0.16  | 0.15   | 0.55      | 0.55     | 0.55    | 0.56     | 0.82       |
| MV-MTC     | 0.31 | 0.60 | 1.55 | 0.33  | 0.29   | 0.73      | 0.73     | 0.73    | 0.76     | 1.05       |

MV-MTC algorithm was compared with one of the recently proposed MTC methods, MTCMRL [45], in terms of time complexity. In [45], multi-task clustering is combined with model relation learning (MTCMRL) method to automatically learn the model parameter relatedness between each pair of tasks by providing a solution to a non-convex optimization problem. Even though the proposed algorithm has a better clustering performance compared to other multi-task clustering methods, it still does not offer the expected performance in terms of time complexity, which is $O(n^2 \times m)$, where $m$ is the number of features and $n$ is the number of instances per task, thus it increases exponentially when $n$ is increased to larger volumes. On the other hand, MV-MTC is still reasonable to be preferred because of its linearly changing time complexity.

With this study, it was aimed to identify similar regions in terms of air quality. It enables flexible decision-making at the cluster level. Thus, decision makers on the control of air quality can take actions similarly for the members of the same cluster. Since many air quality monitoring station data are summarized in several clusters, it provides richer but compacted information for control and modeling. It finds structure in air quality data and is therefore exploratory in nature. Representing the whole environmental data by few clusters may offer the great advantage of simplification in analyzing the data. Identification of the monitoring station groups can be used to understand why these stations in a same cluster are similar. Clustering monitoring stations minimizes the overload of information. Grouping similar information and summarizing common characteristics help the environmental scientists understand the current situation more clearly. In addition, it is also possible to classify a new station by assigning it to the cluster with the closest center.

The potential contributions of this study to the prediction of air quality can be listed as follows:

- The multi-task clustering can also be used to label all the observed elements before air quality prediction, by calculating the distance between each centroid and each element in the data, and then selecting the cluster label (or level) with minimum distance.
- Multi-task clustering can also be used as a preprocessing step to improve the speed and performance of the classification algorithm that is used to predict air quality index.
- In the application to predict air quality index, temporal data clustering results can give information about air quality variations, such that a set of forecasting systems, which are dedicated to reflect temporal changes, can be formed.
- The identification of the air pollution levels of the different regions by clustering can be useful to design air quality monitoring network structure. Such networks must consider the monitoring location, sampling frequencies and the pollutants concern. For instance, clustering results lead to design an optimal network, i.e., a network providing maximum data with minimum measurement devices. The spatial relationship analysis is used to compare the information given by the potential sites that may form the network.
- On forecasting the level of air pollution, it is possible to find the closest cluster of a new instance to be predicted, and then use the values in this cluster for prediction.
- Multi-task clustering can also be useful for detecting the extreme air pollution events and can help predict future exceedances. In this sense, an air pollutant value of a region may be considered as an outlier if it exceeds the minimum or maximum value of the cluster it belongs to.
7. Conclusions

The main purpose of this study was to present a new multi-task clustering algorithm to determine which provinces of Turkey have the same air pollution characteristics so that similar precautions for the reduction of pollution can be taken by the decision-making authority for the cities in the same group. The main air pollutants for the experiments were selected as PM$_{10}$, SO$_2$, NO$_2$, NO and O$_3$ and their mean daily concentrations were taken into consideration. All of the data were taken from 49 air quality monitoring stations from different regions of Turkey. Two phases were performed under MV-MTC scheme: single-task clustering and multi-task clustering. In single-task clustering, each air pollutant was handled individually and air quality monitoring stations were assigned to respective clusters (local clustering). In multi-task clustering phase, clusters were labeled according to the intra-cluster weights so that taking common decision from different tasks becomes easier by applying majority voting on these cluster labels per each instance. Final cluster labels were obtained in this phase by combining the results of single-task clusters (global clustering). According to the results of the sum of squared error, the proposed multi-task clustering method MV-MTC performed well compared to classical single clustering algorithms K-Means, Expectation Maximization, Canopy, Farthest First and Hierarchical clustering. MV-MTC with K-Means, which was initialized with K-Means++, provides promising results in the detection of similar AQMSs.

With this study, the following benefits can be obtained:

- Similar regions can be detected easily so that similar air quality management strategies can be applied for them by the decision-making authority.
- Collecting similar information together and summarizing common features help environmental scientists figure out the present situation more clearly.
- Data analysis becomes easier due to dealing with only few cluster instances instead of whole environmental data.
- Data summarization is performed resulting in compact and useful information, thus one does not need to handle huge amounts of redundant data.
- It can be used as a pre-processing step before performing the essential environmental study.
- Inherent hidden patterns of air quality data can be discovered.
- In the case of a new station to be classified, the process can be achieved by placing the station into the cluster that has the nearest cluster center.

In the future, other unsupervised learning methods such as association rule mining or outlier detection or time series analysis can be applied on Turkey’s air pollution data. Instead of using only pollutant levels, meteorological factors such as temperature, humidity, wind speed and direction, pressure, etc. are going to be added into the problem domain because they can significantly influence the air quality level of a region. Seasonal changes can also be observed instead of using yearly data. The severity of air quality may be clustered based on the impact on the health issue or the potential damage to the environment. Furthermore, a new study could be conducted to investigate the main causes of pollution by utilizing data such as fuel, exhaust and industrial waste.

PM$_{2.5}$ is one of the most dangerous particulate matters. However, in Turkey, there is a missing data problem considering the measurements of PM$_{2.5}$ particulate matter. The same case is also valid for CO pollution, thus it is not dealt with in this study. If the study is extended to be applied on other countries, new air pollutants can also be handled.

Author Contributions: G.T., D.B. and A.P. were the main investigators; G.T. and D.B. contributed to writing paper and critically reviewed the paper; D.B. contributed to the design of the paper; G.T. performed the review of the literature; G.T. implemented the methodology; D.B. supervised the work and provided experimental insights; and A.P. critically reviewed the paper and contributed to its final edition.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.
References

1. Rich, D.Q. Accountability Studies of Air Pollution and Health Effects: Lessons Learned and Recommendations for Future Natural Experiment Opportunities. Environ. Int. 2017, 100, 62–78. [CrossRef] [PubMed]

2. Xing, Y.F.; Xu, Y.H.; Shi, M.H.; Lian, Y.X. The Impact of PM$_{2.5}$ on the Human Respiratory System. J. Thorac. Dis. 2016, 8, E69–E74. [CrossRef]

3. Mannucci, P.M.; Franchini, M. Health Effects of Ambient Air Pollution in Developing Countries. Int. J. Environ. Res. Public Health 2017, 14, 1048. [CrossRef] [PubMed]

4. Hava Kirliliğin Çevre ve İnsan Sağlığına Etkileri. Available online: http://cevrevonline.com/hava-kirliliginin-cevre-ve-insan-sagligina-etkileri/ (accessed on 20 January 2019).

5. Ignaccolo, R.; Ghigo, S.; Giovenali, E. Analysis of Air Quality Monitoring Networks by Functional Clustering. Environmetrics 2008, 19, 672–686. [CrossRef]

6. Barrero, M.A.; Orza, J.A.G.; Cabello, M.; Cantón, L. Categorisation of Air Quality Monitoring Stations by Evaluation of PM$_{10}$ Variability. Sci. Total Environ. 2015, 524, 225–236. [CrossRef] [PubMed]

7. Lu, W.Z.; He, H.D.; Dong, L.Y. Performance Assessment of Qir Quality Monitoring Networks Using Principal Component Analysis and Cluster Analysis. Build. Environ. 2011, 46, 577–583. [CrossRef]

8. Kaya, K.; Öğüdücü, Ş.G. A binary classification model for PM$_{10}$ levels. In Proceedings of the 3rd International Conference on Computer Science and Engineering (UBMK 2018), Sarajevo, Bosnia-Herzegovina, 20–23 September 2018; pp. 361–366.

9. Oral, A.E.; Bayramlar, O.F.; Ezirmik, E.; Gulle, B.T.; Canatar, F.; Calik, D.; Nacar, D.D.; Aydin, L.E.; Baran, A.; Harbawi, Z.K. Evaluation of Air Quality in the City of Istanbul during the Years 2013 and 2015. J. Environ. Sci. Eng. 2017, 6, 465–470. [CrossRef]

10. Güler, N.; İsci, O.G. The Regional Prediction Model of PM$_{10}$ Concentrations for Turkey. Atmos. Res. 2016, 180, 64–77. [CrossRef]

11. Başpinar, F.; Bozkurt, Z. Application of artificial neural networks and regression models in the prediction of daily maximum PM$_{10}$ concentration in Düzce, Turkey. Fresenius Environ. Bull. 2014, 23, 2450–2459.

12. Şahin, Ü.A.; Ucan, O.N.; Bayat, C.; Tolluoglu, O. A New Approach to Prediction of SO$_2$ and PM$_{10}$ Concentrations in Istanbul, Turkey: Cellular Neural Network (CNN). Environ. Forensics 2011, 12, 253–269. [CrossRef]

13. Kurt, A.; Oktay, A.B. Forecasting Air Pollutant Indicator Levels with Geographic Models 3 Days in Advance Using Neural Networks. Expert Syst. Appl. 2010, 37, 7986–7992. [CrossRef]

14. Xue, Y.; Liao, X.; Carin, L.; Krishnapuram, B. Multi-task Learning for Classification with Dirichlet Process Priors. J. Mach. Learn. Res. 2007, 8, 35–63.

15. Liu, P.; Qu, X.; Huang, X. Recurrent neural network for text classification with multi-task learning. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI 2016), New York, NY, USA, 9–15 July 2016; pp. 2873–2879.

16. Liu, X.; Gao, J.; He, X.; Deng, L.; Duh, K.; Wang, Y.Y. Representation Learning Using Multi-task Deep Neural Networks for Semantic Classification and Information Retrieval. Available online: https://www.microsoft.com/en-us/research/publication (accessed on 2 January 2019).

17. Zhang, X.L. Convex Discriminative Multitask Clustering. IEEE Trans. Pattern Anal. Mach. Intell. 2015, 37, 28–40. [CrossRef] [PubMed]

18. Zhang, H.; Lee, C.A.A.; Li, Z.; Garbe, J.R.; Eide, C.R.; Petegrosso, R.; Kuang, R.; Tolar, J. A multitask clustering approach for single-cell RNA-seq analysis in Recessive Dystrophic Epidermolysis Bullosa. PLoS Comput. Biol. 2018, 14, e1006053. [CrossRef] [PubMed]

19. Zhang, X.; Zhang, X.; Liu, H.; Liu, X. Multi-task Clustering through Instances Transfer. Neurocomputing 2017, 251, 145–155. [CrossRef]

20. Siahpirani, A.F.; Ay, F.; Roy, S. A Multi-task Graph-Clustering Approach for Chromosome Conformation Capture Data Sets Identifies Conserved Modules of Chromosomal Interactions. Genome Biol. 2016, 17, 114. [CrossRef] [PubMed]

21. Zhang, X.; Zhang, X.; Liu, H.; Liu, X. Multi-Task Multi-View Clustering. IEEE Trans. Knowl. Data Eng. 2016, 28, 3324–3338. [CrossRef]

22. Yan, Y.; Ricci, E.; Liu, G.; Sebe, N. Egocentric Daily Activity Recognition via Multitask Clustering. IEEE Trans. Image Process. 2015, 24, 2984–2995. [CrossRef]
23. Zhang, X.; Zhang, X.; Liu, H. Smart Multitask Bregman Clustering and Multitask Kernel Clustering. *ACM Trans. Knowl. Discov. Data* 2015, 10, 8. [CrossRef]

24. Liu, Q.; Liao, X.; Carin, L. Semi-supervised multitask learning. In *Proceedings of the 20th International Conference on Neural Information Processing Systems* (NIPS 2007), Vancouver, BC, Canada, 3–6 December 2007; pp. 937–944.

25. Qi, Y.; Tastan, O.; Carbonell, J.G.; Klein-Seetharaman, J.; Weston, J. Semi-supervised Multi-task Learning for Predicting Interactions between HIV-1 and Human Proteins. *Bioinformatics* 2010, 26, i645–i652. [CrossRef]

26. Lu, X.; Li, X.; Mou, L. Semi-supervised Multitask Learning for Scene Recognition. *IEEE Trans. Cybern.* 2015, 45, 1967–1976. [CrossRef] [PubMed]

27. Zhang, Y.; Yeung, D.Y. Semi-supervised multi-task regression. In *Machine Learning and Knowledge Discovery in Databases, Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Bled, Slovenia, 6–10 September 2009; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5782, pp. 617–631.

28. Reichart, R.; Tomanek, K.; Hahn, U.; Rappoport, A. Multi-task active learning for linguistic annotations. In Proceedings of the Association for Computational Linguistics: Human Language Technology Conference (ACL: HLT 2008), Columbus, OH, USA, 15–20 June 2008; pp. 861–869.

29. Zhang, Y. Multi-task active learning with output constraints. In Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI 2010), Atlanta, Georgia, USA, 11–15 July 2010; pp. 667–672.

30. Acharya, A.; Mooney, R.J.; Ghosh, J. Active Multitask Learning Using Supervised and Shared Latent Topics. In *Pattern Recognition and Big Data*; World Scientific: Singapore, 2017; pp. 75–112.

31. Fang, M.; Tao, D. Active multi-task learning via bandits. In Proceedings of the SIAM International Conference on Data Mining (SDM 2015), Vancouver, BC, Canada, 30 April–2 May 2015; pp. 505–513.

32. Wilson, A.; Fern, A.; Ray, S.; Tadepalli, P. Multi-task reinforcement learning: A hierarchical Bayesian approach. In Proceedings of the 24th International Conference on Machine Learning (ICML 2007), Corvalis, OR, USA, 20–24 June 2007; pp. 1015–1022.

33. Parisotto, E.; Ba, J.L.; Salakhutdinov, R. Actor-Mimic: Deep Multitask and Transfer Reinforcement Learning. Available online: https://arxiv.org/abs/1511.06342 (accessed on 12 January 2019).

34. Li, H.; Liao, X.; Carin, L. Multi-task Reinforcement Learning in Partially Observable Stochastic. *J. Mach. Learn. Res.* 2009, 10, 1131–1186.

35. Lazaric, A.; Ghavamzadeh, M. Bayesian multi-task reinforcement learning. In Proceedings of the 27th International Conference on Machine Learning (ICML 2010), Haifa, Israel, 21–24 June 2010; pp. 599–606.

36. Zhao, J.; Xie, X.; Xu, X.; Sun, S. Multi-view Learning Overview: Recent Progress and New Challenges. *J. Adv. Inf. Fusion* 2017, 38, 43–54. [CrossRef]

37. He, J.; Lawrence, R. A graph-based framework for multi-task multi-view learning. In Proceedings of the 28th International Conference on Machine Learning (ICML 2011), Bellevue, WA, USA, 28 June–2 July 2011; pp. 25–32.

38. Gao, Z.; Li, S.H.; Zhang, G.T.; Zhu, Y.J.; Wang, C.; Zhang, H. Evaluation of Regularized Multi-task Learning Algorithms for Single/Multi-view Human Action Recognition. *Multimed. Tools Appl.* 2017, 76, 20125–20148. [CrossRef]

39. Honorio, J.; Samaras, D. Multi-task learning of gaussian graphical models. In Proceedings of the 27th International Conference on Machine Learning (ICML 2010), Haifa, Israel, 21–24 June 2010; pp. 447–454.

40. Oyen, D.; Lane, T. Leveraging domain knowledge in multitask Bayesian network structure learning. In Proceedings of the 26 AAAI Conference on Artificial Intelligence (AAAI 2012), Toronto, ON, Canada, 22–26 July 2012; pp. 1091–1097.

41. Yan, Y.; Ricci, E.; Subramanian, R.; Lanz, O.; Sebe, N. No matter where you are: Flexible graph-guided multi-task learning for multi-view head pose classification under target motion. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2013), Sydney, Australia, 3–6 December 2013; pp. 1177–1184.

42. Zhang, Y.; Yang, Q. A Survey on Multi-Task Learning. Available online: https://arxiv.org/abs/1707.08114 (accessed on 10 January 2019).

43. Zhang, Y.; Yang, Q. An Overview of Multi-task Learning. *Natl. Sci. Rev.* 2017, 5, 30–43. [CrossRef]

44. Wang, B.; Yan, Z.; Lu, J.; Zhang, G.; Li, T. Deep Multi-task Learning for Air Quality Prediction. In *Lecture Notes in Computer Science*; Cheng, L., Leung, A., Ozawa, S., Eds.; Springer: Cham, Switzerland, 2018; Volume 11305, ISBN 978-3-030-04221-9.
45. Zhang, X.; Zhang, X.; Liu, H.; Luo, J. Multi-task clustering with model relation learning. In Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI 2018), Stockholm, Sweden, 13–19 July 2018; pp. 3132–3140.
46. Dincer, N.G.; Akku¸s, Ö. A New Fuzzy Time Series Model based on Robust Clustering for Forecasting of Air Pollution. Ecol. Inform. 2018, 43, 157–164. [CrossRef]
47. Atacak, I.; Arici, N.; Guner, D. Modelling and Evaluating Air Quality with Fuzzy Logic Algorithm-Ankara-Cebeci Sample. Int. J. Intell. Syst. Appl. Eng. 2017, 5, 263–268. [CrossRef]
48. Cagcag, O.; Yolcu, U.; Egrioglu, E.; Aladag, C.H. A Novel Seasonal Fuzzy Time Series Method to the Forecasting of Air Pollution Data in Ankara. Am. J. Intell. Syst. 2013, 3, 13–19. [CrossRef]
49. Liu, A.; Lu, Y.; Nie, W.; Su, Y.; Yang, Z. HEp-2 Cells Classification via Clustered Multi-task Learning. Neurocomputing 2016, 195, 195–201. [CrossRef]
50. Han, J.; Pei, J.; Kamber, M. Data Mining: Concepts and Techniques, 3rd ed.; Morgan Kaufmann Publishers: Waltham, MA, USA, 2011; ISBN 9780123814807.
51. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum Likelihood from Incomplete Data via the EM Algorithm. J. R. Stat. Soc. Ser. B 1977, 39, 1–22. [CrossRef]
52. McCallum, A.; Nigam, K.; Ungar, L.H. Efficient clustering of high-dimensional data sets with application to reference matching. In Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM:KDD 2000), Boston, MA, USA, 20–23 August 2000; pp. 169–178.
53. Sharma, N.; Bajpai, A.; Litoriya, M.R. Comparison the Various Clustering Algorithms of Weka Tools. Int. J. Emerg. Technol. Adv. Eng. 2012, 4, 78–80.
54. Hava Kalitesi İzleme İstasyonları Web Sitesi. Available online: http://laboratuvar.cevre.gov.tr (accessed on 20 November 2018).
55. Frank, E.; Hall, M.A.; Witten, I.H. The WEKA Workbench. Online Appendix for “Data Mining: Practical Machine Learning Tools and Techniques”, 4th ed.; Morgan Kaufmann: Cambridge, MA, USA, 2016; ISBN 9780128043578.
56. Wu, H.; Li, H.; Jiang, M.; Chen, C.; Lv, Q.; Wu, C. Identify High-quality Protein Structural Models by Enhanced-means. BioMed Res. Int. 2017, 2017, 7294519. [CrossRef] [PubMed]

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).