Subsequence Time Series Clustering for Air-Pollution Pattern Recognition

submitted in partial fulfillment for the degree of master of science

Oriol Aguilar Larruy
14573040

Master Information Studies
data science
Faculty of Science
University of Amsterdam

Submitted on 30.06.2023

	UvA Supervisor
Title, Name	Yen-Chia Hsu
Affiliation	University of Amsterdam
Email	y.c.hsu@uva.nl

Subsequence Time Series Clustering for Air-Pollution Pattern Recognition

Oriol Aguilar Larruy
University of Amsterdam
oriol.aguilar.larruy@student.uva.nl

ABSTRACT

This research explores the effectiveness of subsequence time series clustering to recognize air pollution patterns using data from environmental sensor stations. It is the first paper that uses this technique in environmental data. Besides, while most of the related work in subsequence time series clustering focuses on selecting one approach and doing an extensive domain-specific analysis of the findings, this study compares the results of different approaches to determine which one works better. The research employs various approaches encompassing data preparation, segmentation techniques, and time series clustering models. These approaches are evaluated using internal cluster validation indexes and a subsequence clustering metric. The findings indicate that TimeSeriesKMeans with Bottom-up segmentation is the most effective approach for identifying insightful patterns, outperforming the more complex models, and allowing us to visualize the temporal dependences of the data.

KEYWORDS

Subsequence time series clustering, data mining, unsupervised learning, environmental data, Smell Pittsburgh

GITHUB REPOSITORY

https://github.com/oriolaguilar/STSClustering_SmellPGH

1 INTRODUCTION

In recent years, there has been a significant rise in the number of sensor stations deployed worldwide, collecting vast amounts of environmental data. This is due to the growing concern about climate change and the need for better air quality. As the number of data sources grows, traditional analytical approaches face significant challenges in processing and extracting meaningful insights from such amounts of information. In this context, data mining and machine learning techniques emerge as crucial tools to analyze and make sense of massive quantities of data effectively.

A data mining method that can handle temporal and multivariate air-pollution readings without labeled information is subsequence time series clustering, which identifies similar patterns in time series data. Subsequence time series clustering is considered a subdomain of time series clustering. Nevertheless, instead of grouping multiple time series into different clusters, one time series is segmented into multiple subsequences, which are clustered using time series clustering models. This technique can reveal recurring trends and anomalies that may not be noticeable at a global level and repeat over time. The segmentation and clustering approach allows for extracting meaningful patterns from large and complex datasets and revealing underlying structures that can provide valuable insights into environmental processes.

This research is be done using Smell Pittsburgh (Smell PGH) dataset. Smell PGH is a Carnegie Mellon University CREATE Lab[1] project that engages Pittsburgh residents in tracking pollution odors. Through a mobile app, users can report unpleasant odors in their location, which can then be tracked and linked to episodes of poor air quality by comparing them to the sensor stations' data. The project's final goal is to find patterns that could explain the air-pollution episodes in the region so that actions can be taken and the community can benefit from their valuable odor-reporting efforts. The researchers of this project had already worked with this data to model a decision tree-based machine learning (ML) model that could predict smelly episodes in the short term using the data from sensor stations and wind. However, no study employed an unsupervised learning approach that could distinguish numerous and more complex scenarios and move away from the binary classification to detect other motifs than smelly or not-smelly episodes.

In literature, some studies have used subsequence time series clustering, especially to analyze patterns [23, 29] and to perform anomaly detection [20]. However, to our knowledge there is not any paper that deals with environmental data. Moreover, these studies usually focus on one model and do a domain-specific analysis of findings afterward. Because of this, we present this paper as a proof-of-concept to determine how feasible this approach is for environmental data and which models work best so that in the future, more research can be conducted in this field. To achieve this objective, we frame our research around the following central research question:

To what extent can subsequence time series clustering models be successfully applied to extract patterns from multivariate environmental data with citizen-contributed information?

Some subquestions can be further examined and broken down in the following manner:

- How can we leverage Smell PGH's user-contributed information to enhance pattern extraction from multivariate environmental data while minimizing human bias?
- Which is the optimal segmentation and clustering approach to extracting self-explanatory patterns from clusters in environmental data?
- Which variables exert the most significant influence in defining patterns within the Smell PGH data?

This paper is divided into five sections. Section 2 covers the existing literature on subsequence time series clustering and the previous efforts with Smell PGH. Section 3 describes the methodologies employed and our reasons for choosing them. Experimental results are explained in Section 4 and the findings and future work in Section 5. Finally, we summarize our conclusion in Section 6.

[1]For more information, see CMU CREATE Lab, accessed June 17, 2023, https://www.cmucreatelab.org/projects/Smell_Pittsburgh

2 RELATED WORK

Machine learning and data mining have produced significant research on subsequence time series clustering for pattern recognition. The most relevant studies will be reviewed in the following subsections. Nevertheless, this research presents a unique opportunity to apply different subsequence clustering approaches to the domain of environmental data and air pollution and compare the results.

2.1 Subsequence time series clustering for pattern recognition

Subsequence time series clustering constitutes an influential field of research, finding applications across diverse domains such as finance [22], healthcare [4, 7], and geology [29] among others. Time series data, characterized by sequential observations over time, offer valuable insights into temporal dynamics and patterns.

Until 2003, it was widely regarded as a reliable method for time series analysis. For example, Chung et al. discovered patterns from stock data using the subsequence time series analysis technique [7]. However, Keogh et al. claimed that subsequence time series clustering was meaningless in 2003 because the centroids produced by subsequence time series clustering became sinusoidal pseudo-pattern for almost all kinds of time series data due to the Fourier state [17]. Nevertheless, some researchers continued researching new methods that produce meaningful subsequence clustering results. Chen claimed that the problem in the abovementioned work stems from using the Euclidean distance metric as the distance measure in the subsequence vector space [6]. So that new distances were proposed, especially more efficient and large-scale versions of dynamic time warping (DTW) [26], to make subsequence time series clustering meaningful.

It has to be pointed out that even if Chung et al. work [7] is, to our knowledge, the most referenced paper in subsequence time series clustering, it was written before the most relevant and accurate time series clustering models came out [16, 22, 24]. Hence, it cannot be considered a warning to use this technique as the clustering accuracy has improved, and researchers have kept using this approach for their studies and anaysis. Additionally, segmentation techniques more complex than sliding windows, like change point detection models [28], which have been developed and have made subsequence clustering more robust.

2.2 Time series clustering models

Even if, according to Aghabozorgi et al. in "Time-series clustering a decade review," [1] subsequence time series clustering is considered a different subdomain to whole time series clustering, the latter models play an essential role as they constitute the part of the pipeline that clusters the different subsequences. Considerable time series clustering models have been designed and studied in recent years. As mentioned in the previous section, more complex models enabled achieving more meaningful findings.

Huang et al. proposed in 2015 Time Series k-means (TSkmeans) for clustering time series data [16]. Even if it was not the first approach that adapted k-means to time series, the model outperformed the prior state-of-the-art clustering models. This was achieved by introducing a new objective function to guide the clustering of time series data and the development of novel updating rules for iterative cluster searching. Additionally, the DTW distant measure proposed by Petitjean et al. allows TSkmeans to perform better in non-aligned and varying length time series [25].

Shape-based clustering models, such as k-Shape and Deep Temporal Clustering (DTC), have gained prominence in time series analysis. These models aim to cluster time series data by capturing their inherent shape characteristics. K-Shape, created by Paparrizos and Gravano in 2015, employs a normalized version of the cross-correlation measure to consider the shapes of time series while comparing them [24]. It has already demonstrated its utility in subsequence time series clustering of intracraneal pressure data [23]. DTC, Madiraju et al., combines deep learning and unsupervised learning to extract hierarchical representations, enhancing clustering accuracy [22]. DTC is considered as the state-of-the-art time series clustering model.

When using the algorithms mentioned earlier, it is necessary to pre-specify the number of clusters. Once the hyperparameter k is given, the algorithms will calculate the centroids and assign each data point to the closest centroid. It is worth noting that outliers, noise, and extraneous data can significantly impact clustering accuracy, as all data points are measured to assign clusters.

2.3 Smell Pittsburgh and Machine Learning in Environmental Data

Hsu et al. developed Smell Pittsburgh, a system that enables community members to report odors and track where these odors are frequently concentrated [14, 15]. Additionally, they developed a model to predict upcoming smell events and send push notifications to inform communities. Thus, some work on feature engineering, human-bias mitigation efforts, and pattern extraction has already been made.

The researchers used text analysis from the community's input to do the feature engineering, selecting hydrogen sulfide (H_2S) and wind information combinations as the only model predictors. The model was fitted using a decision tree, and the patterns were extracted using the local explainability that decision trees grant. However, this approach only allows the pattern extraction from the 30% of the smell events that the model predicted accurately. Additionally, the patterns extracted from a decision tree consist of snapshots of combinations of wind with H_2S readings, making it difficult to understand and visualize the actual behavior of these features over time.

Apart from the above-mentioned study, other researchers have worked on ML [19] and deep learning (DL) approaches [3, 13] to predict future air pollution episodes. Han et al. proposed a domain-specific Bayesian DL model to improve the accuracy and interpretability of the existing models [13]. Unsupervised approaches have also been tested for environmental data. Govender and Sivakumar reviewed more than 100 articles over 40 years that dealt with k-means and hierarchical clustering in air pollution data, concluding that k-means was the most used method [11]. Nevertheless, to the best of our knowledge, no study has been conducted on the possibility of performing subsequence clustering in this domain, highlighting the novelty of this paper.

3 METHODOLOGY

Using a subsequence time series clustering approach, we aimed to create a pipeline to detect patterns in a multivariate time series (see Figure 1). In this section, we first describe the dataset used and our data processing approach to convert it into a meaningful time series and reduce bias. Next, we discuss how the time series is segmented and clustered to find common behaviors. Finally, we evaluate how well clustering works.

3.1 Datasets

3.1.1 Reports Dataset (Smell PGH):. It contains 79.853 reports of the users of the Smell PGH app from the 31st of October 2016 to the 11th of December 2022 in the Pittsburgh region in Pennsylvania, USA. Every row in this dataset contains the information of a report: date and time, skewed latitude and longitude, zip code, smell value, and smell description and symptoms linked to odor if provided. The smell value is from 1 to 5, with five being the most severe.

3.1.2 Sensor Stations Datasets (Smell PGH):. It collects air quality datasets from the Pittsburgh region's multiple monitoring stations. All the air quality monitoring station have a unique feed ID. Some stations are operated by the municipality, and some by local citizens. These datasets contain hourly information on wind speed, direction, deviation, and particle measurements like H_2S and sulfur dioxide (SO_2). However, the datasets contain short periods of missing data due to the repair and malfunctioning times of the stations.

3.1.3 External Dataset (Beijing Multi-Site Air-Quality Data Set - Nongzhanguan Station): We introduced this dataset to validate the results of subsequence time series clustering for air-pollution pattern recognition. The dataset comprises 35.064 hourly records from the 1st of March 2013 to the 28th of February 2017. Each record contains air-pollution-related and weather measurements like particles in air, sulfur dioxide, temperature, and atmospheric pressure. It doesn't contain citizen-contributed data and missing data has to be expected as well.

3.2 Data preparation

We needed to transform the valuable information of the datasets into a multivariate time series with equally spaced points in time to segment and cluster the data to find patterns. Therefore, we aggregated the reports' data in 24-hour windows. We chose this window size because the reports' distribution during the day was imbalanced, as most users report a smell between 7:00 a.m. and 9:00 a.m., and the reports drop during the rest of the day. Aggregating the data in a daily time series meant we reduced our dataset to only 2.231 data points. To augment the data and make it more suitable for more complex models, we created 24-hour sliding windows with 5-hour lags, ending with 10.711 data points. We completed the time series by creating features, listed in Table 1, that gathered the most meaningful data from the reports and sensor station datasets.

One of the data preparation challenges was dealing with the reports' smell intensity bias. On average, users report a smell value of 3.66 over 5 when our domain knowledge says that, most days, there is not any smell issue in the region. This bias happens because users are likelier to report a smell when it is malodorous. On the other hand, there is a slight correlation (+0.48) between the average smell value and the number of reports in a day. Thus, the latter can be a better estimator for understanding whether the whole region has an odor issue. As a result, we created a new score by adding the reports' intensity per day that follows a log-normal distribution and matches our domain knowledge, as only the days with many negative reports get placed on the right tail (see Figure 6).

We analyzed the smell description inputs to determine which particles can have an inference to the smell reports. The keywords mentioned the most were industrial, sulfur, rotten eggs and burnt (see Figure 7). With these insights, we selected hydrogen sulfide, which smells like rotten eggs, and sulfur dioxide, which smells like a just-struck match, as the particle features to include in the model. Both features follow a zero-inflated exponential distribution and are calculated by aggregating all the respective sensor stations' readings to simplify the data and minimize missing values.

Wind information was essential in Hsu et al. [15] study to determine the pollution patterns as this information can unveil the origin of the particles. Therefore, we took the sensor stations' wind speed, wind direction, and the standard deviation of wind direction data. All the features extracted from the sensor stations resulted from taking the mean of all station records during the specified window. In the case of wind direction, which deviates a lot during the day, we weighted the wind direction with the percentage of total daily reports at each hour, so the more active periods have a greater influence than the less active ones.

Additionally, to equalize the effect of features, we normalized each dataset feature but the cosine and sine wind decomposition to zero mean and unit variance. Missing values were replaced with the corresponding linear interpolated values of the adjacent non-missing data points using pandas built-in *interpolate* function.

No data preparation transformations were done to the external dataset other than zero-mean normalization and missing data imputation.

Feature Name	**Type of Distribution**	**Sensor Station Used**
Reports' intensity	Log-Normal	-
Reports' dispersion	Normal	-
Wind speed	Rayleigh/Chi-Squared	1, 27, 29, 3
Wind direction (sin)	Empirical	1, 27, 29, 3
Wind direction (cos)	Empirical	1, 27, 29, 3
Wind direction std.	Empirical	1, 27, 29, 3
SO_2 average	Zero-inflated exp.	1, 29, 3
SO_2 max	Exponential	1, 29, 3
H_2S average	Zero-inflated exp.	1, 29
H_2S max	Exponential	1, 29

(a) Sensor Station Legend. 1: Avalon ACHD, 27: Lawrenceville 2 ACHD, 29: Liberty 2 ACHD, 3: North Braddock ACHD

Table 1: Selected designed features. The types of distributions have been determined with observations from the data

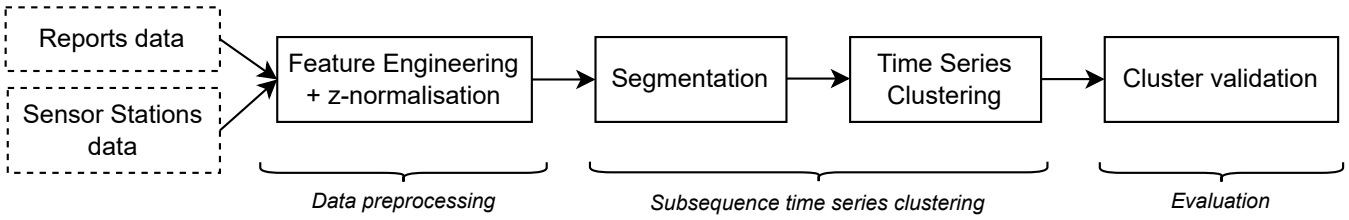

Figure 1: Workflow of the methodology developed in this paper for extracting air pollution patterns. The data is first transformed and normalized into a multivariate time series. Then, it is segmented into multiple subsequences to perform time series clustering. Finally, we evaluate the clustering and the meaningfulness of patterns.

3.3 Time Series Segmentation

Time series segmentation, which refers to the tool for dividing the signal into a discrete number of contiguous sequences, is crucial in subsequence time series clustering. Segmentation aims to identify similar segments in terms of patterns or behaviors.

Several time series segmentation methods exist, including sliding windows and change point detection. Sliding window methods involve dividing the time series into fixed-length windows. Change point detection methods involve identifying points in the multivariate time series where the pattern or behavior changes. In our dataset, we have noticed that the smell events have varying lengths. Thus, we planned to use the bottom-up change point detection method for segmentation. However, some time series clustering models, such as k-Shape and DTC, require fixed-size time series. In these cases, we used the sliding window model instead. Unevenly sized subsequences could be resized to the same length with zero padding or interpolation. Nevertheless, this approach may adversely affect the clustering results.

The process of sliding window segmentation involves dividing a time series into sub-sequences of fixed length with an overlap between them. On the other hand, bottom-up segmentation initially divides the time series into multiple sub-sequences based on a regular grid. Subsequently, contiguous segments are merged based on their level of similarity. The user can set the number of resulting segments as a parameter.

3.4 Time Series Clustering models

Time series clustering models aim to group subsequences obtained from segmentation. This grouping helps identify common patterns that repeat over time. To better evaluate the performance of the subsequence time series models, we run different models and compared their results.

3.4.1 Baseline Models: We chose TimeSeriesKMeans (TSkmeans) and k-Shape from tslearn package [27] for clustering the segmented subsequences. Their ease of use and the fact that they do not require fine-tuning make them a common choice for time series clustering baselines. The main difference between the traditional k-means and the one used with time series is that the latter uses the DTW distance instead of Euclidean. DTW measures the similarity between two time series by finding the optimal alignment. This way, it can be used for clustering time series with different lengths and similar non-aligned shapes. On the other hand, k-Shape uses a shape-based similarity measure to obtain more robust results.

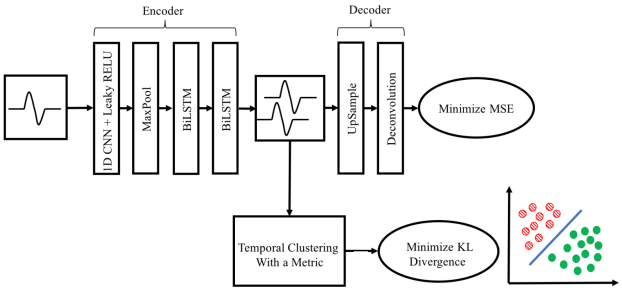

Figure 2: Overview of Madiraju et al. DTC architecture used for our approach.

In addition, we have chosen time point clustering models to have a broader picture of how clustering works in time series. In these models, each data point can be clustered to any class without being restricted by the rest of the segment. Thus getting better cluster validation results and an idea of how effectively clustering can be done per each dataset. However, we cannot use this technique to extract patterns since it does not deal with periods but with punctual events.

We chose k-means and hierarchical clustering as time point clustering models. These methods are raw-data-based clustering techniques and require a prespecified hyperparameter k to determine the number of clusters.

3.4.2 Deep Temporal Clustering (DTC):. Motivated by Madiraju et al. stating the superior performance of DTC compared with k-Shape and other time series clustering approaches [22], we decided to use this temporal autoencoder-based deep neural network architecture to better model this problem.

DTC architecture's (see Figure 2) first level, consisting of a 1D convolution layer followed by a max pooling layer, casts the time series into a more compact representation while retaining the most relevant information, the short-term features (waveforms). Furthermore, this dimensionality reduction is crucial for further processing to avoid very long sequences, which can lead to poor performance. First-level activations are then fed to the second level (Bidirectional Long-Short Term Memory - BiLSTM) to learn temporal changes in both directions and obtain the latent representation. Then, this latent representation is assigned to a cluster using k-means. Finally, reconstruction is provided by an upsampling layer followed by

a deconvolutional layer to obtain autoencoder output. The loss function of the neural network is the sum of the autoencoder loss, the mean square error (MSE) of the input sequence reconstruction, and the clustering loss, Kullback–Leibler divergence loss: $loss_{total} = loss_{reconstruction} + loss_{clustering}$

Even though the original DTC architecture uses a heatmap to localize the main data features contributing to the clustering, it has not been used in this implementation due to the lack of documentation in the author's paper.

3.5 Experimental Set-up

To determine the optimal number of segments, their width, and the width of the overlap for both the bottom-up and sliding window approach, we have used the 'optuna' framework [2]. We tested all possible combinations for each dataset and segmentation technique until finding the one that returned the highest Silhouette score. To avoid the computational load of training the DTC model several times, we used the same segment size (18-time steps) and segment overlap (5-time steps) obtained from the k-Shape simulation.

Before clustering, we used the Silhouette score to determine the optimal number of clusters (see Figure 8). As the final goal is to move away from binary classification and extract multiple patterns represented by the clusters, we constrained the number of clusters to be over three, getting the optimal k value in four.

We used Forest et al. Keras implementation [9] for training the DTC model. We utilized the entire Smell PGH and external datasets for the DTC training sets. A pooling size of 3-time steps, one-sixth of the subsequence length, was used for the Smell PGH model. The similarity metric used for the clustering in our approach is Complexity-Invariant Distance (CID) [5]. It is a widely used distance metric in time series clustering that includes structural features and point-wise differences to capture the time series' inherent complexity. If both input sequences have the same complexity, then the distance is the same as the Euclidean distance. The total training was set to be 250 epochs and 50 pretraining epochs. During each epoch, the model received the training data and adjusted its trainable parameters to reduce the loss. Adam was selected to be the learning rate optimizer during training. Our models were implemented using Python 3.11, ruptures 1.1.17, tslearn 0.5, and Keras 2.12.

3.6 Cluster Validation

Clustering validation indexes (CVI) are commonly used to evaluate clustering results. We can classify cluster validation techniques into internal and external cluster validation. The former uses the clustering process's internal information to evaluate the clusters' goodness without referencing external labels. The latter compares the clustering result to externally provided class labels. Given that the external ground truth labels are not available in our research, we will evaluate the clustering results by the internal index: Silhouette score and Multivariate Time Series Subsequence Clustering Metric (MT3SCM).

Using both metrics, we can overcome the bias that evaluation is not limited to the Silhouette score's perspective of well-separated clusters. Allowing to capture a broader range of cluster characteristics related to subsequence time series clustering. In other words, the effectiveness of clustering should not solely rely on the degree of separation between clusters, but also consider other factors that contribute to meaningful and cohesive grouping of subsequences.

3.6.1 Silhouette Score: It is a widely-used CVI that measures the similarity between each sample and its cluster (intra-cluster distance) compared to the other clusters (nearest-cluster distance). The silhouette ranges from -1 to +1, where a number close to +1 indicates that the sample is well-matched with its cluster and poorly matched with the rest of the clusters. When the value is near 0, it indicates there are overlapped clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster. For data point i in the cluster $C_i (i \in C_i)$, the mean distance between the data point i and the rest of data points in C_i is expressed as $a(i)$, where $d(i, j)$ is the distance between data points i and j.

$$a(i) = \frac{1}{|C_i|-1} \sum_{j \in C_i, i \neq j} d(i, j)$$

The "nearest cluster" or "neighboring cluster" of i, denoted as $b(i)$, is the cluster that is the next best fit for point i, hence has the smallest mean dissimilarity.

$$b(i) = min_{k \neq i} \frac{1}{|C_k|} \sum_{k \in C_k} d(i, j)$$

Thus, the silhouette value for one point $s(i)$ is defined as it follows:

$$s(i) = \frac{b(i) - a(i)}{max(a(i), b(i))}, if |C_i| > 1$$

3.6.2 MT3SCM:. It is a novel internal metric proposed by Kohne et al. in 2023 that tries to treat multivariate time series data as space curves and use the parameterization as to measure similarities between subsequences [18]. This is done in two different ways. First it considers the time-space variations like curvature, acceleration, or torsion in a multidimensional space and its standard deviation for each cluster. Using these curve parameters allows us to measure similarities between subsequences. The second procedure is to apply these newly computed features, which are computed to scalar values per subsequence, onto a well established internal clustering metric, the silhouette score.

$$mt3scm = (cc_w + s_P + s_L)/3$$

The main idea of this approach is to combine three main parts inside one metric. The first incentive is to reward a low standard deviation of the curve parameters between a cluster (accomplished by the weighted curvature consistency - cc_w). Second, to benchmark the clusters' spatial separation based on the new feature space curve parameters (accomplished by the silhouette location based - s_P). Third, to benchmark the clusters' spatial separation based on the median of the subsequence in the original feature space (accomplished by the silhouette curve-parameter based - s_L).

When evaluating with MT3SCM, it is necessary to ensure that the subsequences do not overlap. In cases where the data is divided into segments using a sliding window approach, any overlapping segments should be assigned to the sample with the highest silhouette score.

	Segmentation Technique	Clustering Model	Smell PGH Dataset				External Dataset			
			Silhouette Score		MT3SCM		Silhouette Score		MT3SCM	
			avg	std	avg	std	avg	std	avg	std
Time-point clustering	-	Hierarchical Clustering	0.202	0	0.070	0	0.298	0	0.087	0
		k-means	0.243	0	0.089	0	0.283	0	0.095	0
Subsequence time series clustering	Bottom-Up segmentation	**TSkmeans**	0.142	0.005	0.066	0.005	0.216	0.045	0.104	0.021
	Sliding Window Segmenation	TSkmeans	0.108	0.002	0.061	0.006	0.209	0.024	0.091	0.025
		k-Shape	0.025	0.009	0.028	0.005	0.024	0.007	0.033	0.007
		DTC	0.061	0.001	0.028	0.001	0.056	0.001	0.033	0.001

Table 2: Performance comparison of the different models and segmentation techniques on Smell Pittsburgh dataset and external dataset. The greater the Silhouette and the MT3SCM score the better-clustered.

3.7 Cluster and Pattern Analysis

Determining the effectiveness of the subsequence clustering approach requires more than just a good CVI. Meaningful clusters must appear across multiple periods, be statistically distinct from other clusters, and exhibit a self-interpreting shape or trend that helps to understand air pollution patterns.

To evaluate the meaningfulness of the chosen approach, we use the one-way analysis of variance (ANOVA) to evaluate whether there are any statistically significant differences between the different clusters by comparing the sample's means. Finally, we examine visually the shape of the different barycenters, which are the time series that minimizes the sum of squared distances of each cluster. The barycenter is calculated by first interpolating all the subsequences to a common length and then using the Soft-DTW loss function [8] to get the time series that best illustrates that cluster.

4 RESULTS

4.1 Model Evaluation

We ran different models and compared their outputs to examine the performance of the different clustering algorithms and segmentation techniques on environmental data. Regarding internal validation indexes, we repeated all the clustering methods iteratively 25 times to decrease output variances. We evaluated all the clustering results with the Silhouette value and MT3SCM index at each iteration. The resulting scores were averaged and displayed in Table 2, accompanied by their standard error.

Time-point clustering models have the highest Silhouette Score as each data point is clustered individually without depending on any sequence. Thus, a higher cluster validation index was expected. However, this solution is unsuitable for approaching this problem as it does not guarantee to find temporal patterns.

Regarding the subsequence time series clustering models, the segmentation technique is slightly positive in the results. Less complex models, like TimeSeriesKMeans, can achieve a better result with bottom-up segmentation as it truncates the time series in critical parts where there is a changing point. Thus, it is easier to match similar time series, making the clustering process easier.

Even though k-Shape and DTC are considered state-of-the-art models, they do not perform as well as the simpler model. K-Shape does not achieve well-separated clusters, and even though DTC performs better, its results do not outperform TimeSeriesKMeans.

About the MT3SCM metric, we got a similar behavior compared with the results with Silhouette Score. Thus, we can observe how simpler models outperform the most complex ones. Nevertheless, the difference between the worst and the best-performing models is lower than regarding the Silhouette Score. This suggests that clusters can still be considered valid even if they are not well-separated in the context of subsequence time series clustering.

TimeSeriesKmeans performs well in both datasets, getting satisfactory results in both metrics no matter the segmentation used. It has a higher MT3SCM (0.104) in the external dataset and a solid Silhouette Score than any other model. Using the Bottom-up segmentation, it has the highest Silhouette Score and MT3SCM average in the Smell PGH dataset (0.142 and 0.066). Consequently, we chose this combination to assign the clusters to perform the cluster analysis.

4.2 Cluster Analysis for Air-Pollution Pattern Recognition

The previously mentioned approach (Bottom-up segmentation and TimeSeriesKmeans) classifies the time series' subsequences into 4 clusters. In order to confirm the successful extraction of air pollution patterns via the subsequence time series clustering approach, we verified the regular appearance of samples from each cluster throughout the time series. All clusters appeared recurrently, indicating that they represent intrinsic behaviors and are likelier to keep appearing in the future.

Besides, the average features of each cluster turned out to have an ANOVA p-value of 0, stating that all features were influential enough to contribute to at least one cluster (see Table 4). Looking at the different cluster averages per each feature, we can confidently say that clusters have been created based on how intense the odor issue was at that moment. Cluster 2 represents the days with many high-intensity reports; meanwhile, Clusters 0 and 3 represent most

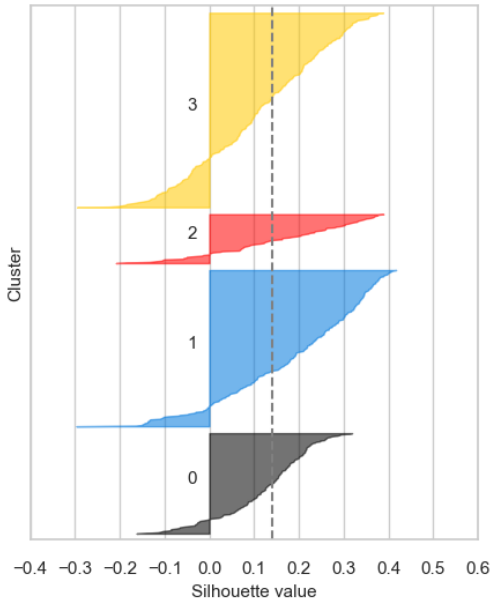

Figure 3: Silhouette score values per each cluster

Features	Cluster 0	Cluster 1	Cluster 2	Cluster 3	*p*-value
Reports Intensity	0.488	-0.341	1.609	-0.123	0
Reports Dispersion	-0.367	0.129	-0.668	0.104	0
Wind Speed	-0.283	0.982	-0.337	-0.412	0
Wind N-S	-0.355	-0.140	-0.374	-0.036	0
Wind E-W	-0.022	-0.478	0.002	0.003	0
Wind std. dir	-0.065	-0.905	-0.195	0.525	0
SO_2 avg.	0.714	-0.299	2.476	-0.299	0
SO_2 max.	0.720	-0.292	2.339	-0.290	0
H_2S avg.	0.605	-0.296	2.643	-0.289	0
H_2S max.	0.638	-0.318	2.458	-0.269	0

Figure 4: Average value per each feature and cluster. The last column contains the *p*-value of the ANOVA test of the different subsamples

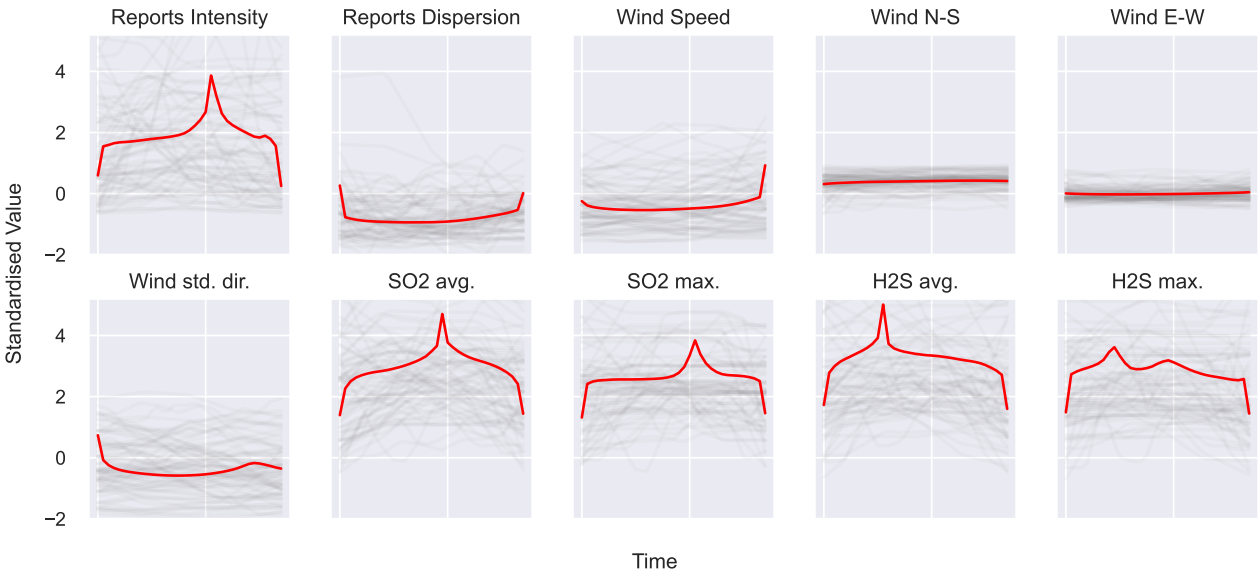

Figure 5: Barycenters of the subsequences of cluster 2. The reports intensity peak matches in time the SO_2 peak but is later in time than the H2S peak.

of the samples, days without any odor issue. The same table gives us more insights into how the odor reports work: There is a positive correlation between high smell intensity and the particles in the air, and wind plays an essential role as the strong wind tends to fade the particles and odor away. In addition, the wind blowing from the south to the north forecasts strong smells unveiling that the cause can be the particles coming from the industrial are in the south of the region.

We analyzed cluster 2 as most of the samples have a Silhouette score greater than the average (see Figure 3) and represents smelly episodes. The visual representation of the subsequences' barycenters of this cluster (see Figure 5) reveals meaningful motifs. We can observe that the increase in smell intensity coincides with the spike in SO_2 levels, but it occurs after the H_2S spike. This finding implies that the H_2S spike may help predict the impeding citizen's reaction.

5 DISCUSSION AND FUTURE WORK

This paper addresses the research to study the effectiveness of subsequence clustering for environmental data with citizen-contributed information. It is the first work that evaluates and compares different approaches, including segmentation techniques and time series clustering models, for environmental or equivalent data, unlike other works that focus on one model and do an extensive domain-specific study afterward.

The data preprocessing efforts allowed us to transform the longitudinal reports into a time series where subsequence time series clustering could be performed, and human bias could be mitigated. The later cluster analysis proved the efficiency of aggregating the reports' and sensor stations' features into sliding windows and measuring the smell intensity feature by considering the number of reports rather than just the intensity average.

The model evaluation stated that TimeSeriesKMeans is the better-suited model for finding well-clustered patterns using subsequence time series clustering in environmental data. The results of the Smell PGH and the external dataset proved its superiority over the more complex models, explained later in Section 5.1.

H_2S and SO_2 particles are the most influential features in detecting smelly patterns. Besides, the wind should always be considered as it can vanish them. In Table 4, Cluster 1 represents windy days with minimal smell intensity and particle measurements. Additionally, wind directions can determine the origin of the odor. All the clusters with high smell reports have a predominant and constant south wind. This finding aligns with Hsu et al. [15] finding when they mainly used H_2S readings from Liberty Monitor Station, south of the region, to predict the upcoming smell event.

5.1 Subsequence Time Series Clustering in Environmental Data

The cluster and pattern analysis prove the efficiency of this approach. However, TimeSeriesKMeans' superiority over k-Shape and DTC raises some questions. Why have the models frequently used for time series clustering in a wide range of domains not only been able to surpass the results of TimeSeriesKMeans but have yet to perform well? Why do even the time point clustering models have a better MT3SCM than these models, even if the formers do not consider the temporal dimension?

K-Shape and DTC are shape-based clustering models, as they cluster sequences based on shape similarity. K-Shape, for example, compares the shapes of time series by aligning them with DTW and then measures their difference using a shape-based distance metric. TimeSeriesKMeans, on the other hand, applies the k-means algorithm directly to time series data using DTW as the distance measure to work with non-aligned sequences. This method clusters sequences based on the similarity of their data points. Therefore, if two subsequences contain similar high values above the mean, they will likely be clustered in the same cluster.

Environmental data, especially from air-polluting particle measurements, often follows an exponential or log-normal distribution. The data values usually lay on the distribution's left side (lower values), and occasionally, there are unusual situations when higher values are measured. The higher the value, the less frequent, and the likelier to be a smelly episode. As a result, TimeSeriesKMeans does particularly well in clustering these unusual days, as it can group them based on the intensity of the smell reports. Moreover, as we deal with short-term patterns, subsequences are stationary, irregular, and short. Thus, no common shapes or complex patterns can be extracted other than the natural shape, a peak, of a smelly episode. That is also the reason behind the performance of time point clustering, as it merely focuses on the amplitude the data points and those share information with the adjacent samples.

If, instead of focusing on short-term patterns, we had decided to extract long-term patterns, where the weather seasonality was an essential factor to consider, shape-based models could be more decisive. Another possibility for using shape-based models would be to only cluster the anomalies, in our case, the smelly episodes, so that high-intensity episodes could be clustered based on shape features instead of amplitude. Similar work has been done for identifying different types of intracranial pressure anomaly signals in medical data [23].

5.2 Feature Engineering and Data Preprocessing

All the decisions regarding the details of the approach, like the number of clusters, the size of the segments, and the model hyperparameters, have been chosen with the Silhouette score as the reference to compare the different performances. However, we have yet to evaluate whether a more complex model is better for the proposed cluster and pattern analysis.

Setting the number of clusters to four was a rather conservative decision that allowed us to understand multiple patterns without the help of a domain expert and simultaneously manage to have better cluster validation indexes. Nevertheless, more experiments with higher numbers of clusters should be performed and evaluated by domain experts to find more complex patterns that could give less trivial but still accurate insights.

Another way to extract more complex patterns from the data would have been to use more features. In the data preprocessing steps, the information from all the sensor stations was aggregated to make the model less complex and to deal with the missing data. By using the information of the different sensor stations independently, we could get more insights into the location of the particles and their origin, as each sensor station is encountered in a different place in the region. However, the missing values of the sensor stations make this approach less likely to be effective. In this work, the missing data imputation has been done with linear interpolation, even though the development of dimensions is likely non-linear, so it may be beneficial to explore other alternatives.

A Generative Adversarial Network (GAN) can be considered to improve the quality of imputed missing data [10]. Given a training set, a GAN learns to generate new data with the same statistics as the training set, being able to replicate the patterns of data in the missing values. Luo et al. proposed a GAN for multivariate time series imputation using a modified Gate Recurrent Unit (GRU) [21]. Gupta et al. proposed a Bi-Directional GAN for time series imputation and prediction [12]. By obtaining a more accurate representation of missing values of the sensor stations, more features can be added to the cluster analysis.

6 CONCLUSION

This is the first study that proposes and compares different subsequence time series clustering approaches for pattern extraction in environmental data, as most existing work only tests one approach. We have used the Smell PGH dataset as a design choice to analyze the model's outcomes.

The research aims to answer the best subsequence time series clustering approach for environmental data. TimeSeriesKMeans outperformed the other clustering models, creating well-separated clusters and meaningful and cohesive subsequences based on the anomalies' amplitude rather than their shape, as the other tested models: k-Shape and DTC. The further cluster analysis demonstrates the validity of the approach as suggestive insights and patterns related to the temporal dependence of the Smell PGH dataset's particle readings could be extracted.

Additionally, our cluster analysis verifies the efficiency of aggregating features from the longitudinal reports and sensor stations into sliding windows while measuring smell intensity based on the number of reports rather than the average intensity. These findings highlight the potential of incorporating user-contributed data to enhance pattern extraction in environmental analysis and ensure a more objective and self-interpreting approach.

Further work would include using a different segmentation technique to be able to solely cluster the anomalies based on the shape and not only the amplitude, along with the use of GANs to deal with the missing data of the Smell PGH's sensor stations and be able to extract more complex yet accurate patterns.

REFERENCES

[1] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. 2015. Time-series clustering – A decade review. *Information Systems* 53 (2015), 16–38. https://doi.org/10.1016/j.is.2015.04.007

[2] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. (7 2019). http://arxiv.org/abs/1907.10902

[3] Federico Amato, Fabian Guignard, Sylvain Robert, and Mikhail F. Kanevski. 2020. A novel framework for spatio-temporal prediction of environmental data using deep learning. *Scientific Reports* 10 (2020).

[4] Wenjun Bai, Okito Yamashita, and Junichiro Yoshimoto. 2023. Learning task-agnostic and interpretable subsequence-based representation of time series and its applications in fMRI analysis. *Neural Networks* 163 (2023), 327–340. https://doi.org/10.1016/j.neunet.2023.03.038

[5] Gustavo E. A. P. A. Batista, Eamonn J. Keogh, Oben M. Tataw, and Vinícius M. A. de Souza. 2013. CID: an efficient complexity-invariant distance for time series. *Data Mining and Knowledge Discovery* 28 (2013), 634 – 669.

[6] Jason R Chen. 2005. Making Subsequence Time Series Clustering Meaningful.

[7] Fu-Lai Chung, Vincent T Y Ng, Robert Wing, Pong Luk, Tak-Chung Fu, Vincent Ng, and Robert Luk. 2001. Pattern discovery from stock time series using self-organizing maps Pattern Discovery from Stock Time Series Using Self-Organizing Maps †. https://www.researchgate.net/publication/228771755

[8] Marco Cuturi and Mathieu Blondel. 2018. Soft-DTW: a Differentiable Loss Function for Time-Series. arXiv:1703.01541 [stat.ML]

[9] Florent Forest and Arthur Guilherme. 2020. DeepTemporalClustering. https://github.com/FlorentF9/DeepTemporalClustering.

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Networks. (6 2014). http://arxiv.org/abs/1406.2661

[11] Paulene Govender and Venkataraman Sivakumar. 2020. Application of k-means and hierarchical clustering techniques for analysis of air pollution: A review (1980–2019). *Atmospheric Pollution Research* 11 (2020), 40–56.

[12] Mehak Gupta and Rahmatollah Beheshti. 2020. Time-series Imputation and Prediction with Bi-Directional Generative Adversarial Networks. https://github.com/mehak25/BiGAN

[13] Yang Han, Jacqueline C. K. Lam, Victor O. K. Li, and Qi Zhang. 2022. A Domain-Specific Bayesian Deep-Learning Approach for Air Pollution Forecast. *IEEE Transactions on Big Data* 8 (2022), 1034–1046.

[14] Yen-Chia Hsu, Jennifer Cross, Paul Dille, Michael Tasota, Beatrice Dias, Randy Sargent, Ting-Hao 'Kenneth' Huang, and Illah Nourbakhsh. 2019. Smell Pittsburgh: Engaging Community Citizen Science for Air Quality. (12 2019). http://arxiv.org/abs/1912.11936

[15] Yen-Chia Hsu, Jennifer Cross, Paul Dille, Michael Tasota, Beatrice Dias, Randy Sargent, Ting-Hao 'Kenneth' Huang, and Illah Nourbakhsh. 2020. Smell Pittsburgh: Community-Empowered Mobile Smell Reporting System. arXiv:1810.11143 [cs.HC]

[16] Xiaohui Huang, Yunming Ye, Liyan Xiong, Raymond Y.K. Lau, Nan Jiang, and Shaokai Wang. 2016. Time series k-means: A new k-means type smooth subspace clustering for time series data. *Information Sciences* 367-368 (2016), 1–13. https://doi.org/10.1016/j.ins.2016.05.040

[17] Eamonn Keogh and Jessica Lin. 2005. Clustering of time-series subsequences is meaningless: Implications for previous and future research. *Knowledge and Information Systems* 8 (2005), 154–177. Issue 2. https://doi.org/10.1007/s10115-004-0172-7

[18] Jonas Kohne, Lars Henning, and Clemens Guhmann. 2023. Autoencoder-Based Iterative Modeling and Multivariate Time-Series Subsequence Clustering Algorithm. *IEEE Access* 11 (2023), 18868–18886. https://doi.org/10.1109/ACCESS.2023.3247564

[19] K. Kumar and B. P. Pande. 2022. Air pollution prediction with machine learning: a case study of Indian cities. *International Journal of Environmental Science and Technology* 20 (2022), 5333 – 5348.

[20] Jinbo Li, Hesam Izakian, Witold Pedrycz, and Iqbal Jamal. 2021. Clustering-based anomaly detection in multivariate time series data. *Applied Soft Computing* 100 (2021), 106919. https://doi.org/10.1016/j.asoc.2020.106919

[21] Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, and Xiaojie Yuan. 2018. Multivariate Time Series Imputation with Generative Adversarial Networks.

[22] Naveen Sai Madiraju, Seid M. Sadat, Dimitry Fisher, and Homa Karimabadi. 2018. Deep Temporal Clustering : Fully Unsupervised Learning of Time-Domain Features. (2 2018). http://arxiv.org/abs/1802.01059

[23] Isabel Martinez-Tejada, Casper Schwartz Riedel, Marianne Juhler, Morten Andresen, and Jens E. Wilhjelm. 2022. k-Shape clustering for extracting macro-patterns in intracranial pressure signals. *Fluids and Barriers of the CNS* 19 (12 2022). Issue 1. https://doi.org/10.1186/s12987-022-00311-5

[24] John Paparrizos and Luis Gravano. 2015. k-Shape: Efficient and Accurate Clustering of Time Series.

[25] François Petitjean, Alain Ketterlin, and Pierre Gançarski. 2011. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition* 44, 3 (2011), 678–693. https://doi.org/10.1016/j.patcog.2010.09.013

[26] Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. 2012. Searching and mining trillions of time series subsequences under dynamic time warping. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 262–270.

[27] Romain Tavenard, Johann Faouzi, Gilles Vandewiele, Felix Divo, Guillaume Androz, Chester Holtz, Marie Payne, Roman Yurchak, Marc Rußwurm, Kushal Kolar, and Eli Woods. 2020. Tslearn, A Machine Learning Toolkit for Time Series Data. *Journal of Machine Learning Research* 21, 118 (2020), 1–6. http://jmlr.org/papers/v21/20-091.html

[28] Charles Truong, Laurent Oudre, Nicolas Vayatis, and Paris Saclay. 2020. Selective review of offline change point detection methods.

[29] Rahul Kumar Vijay and Satyasai Jagannath Nanda. 2023. Earthquake pattern analysis using subsequence time series clustering. *Pattern Analysis and Applications* 26 (2 2023), 19–37. Issue 1. https://doi.org/10.1007/s10044-022-01092-1

Appendix A EXPLORATORY DATA ANALYSIS

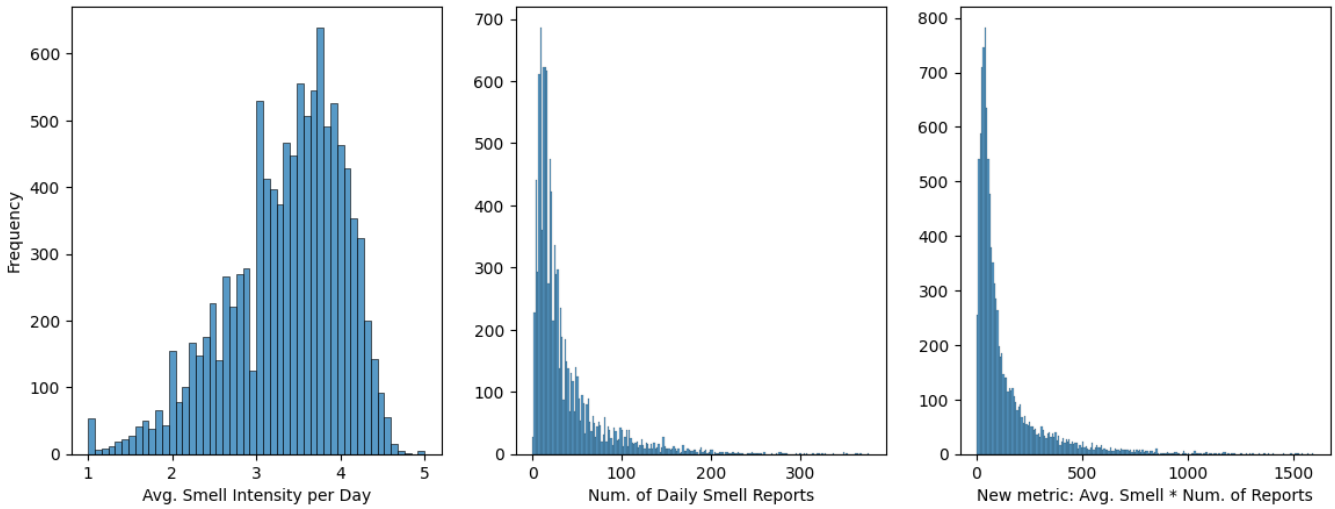

Figure 6: From left to right. Distribution of average smell intesity per day. Distribution of the amount of smell reports per day. Distribution of the metric used in the thesis.

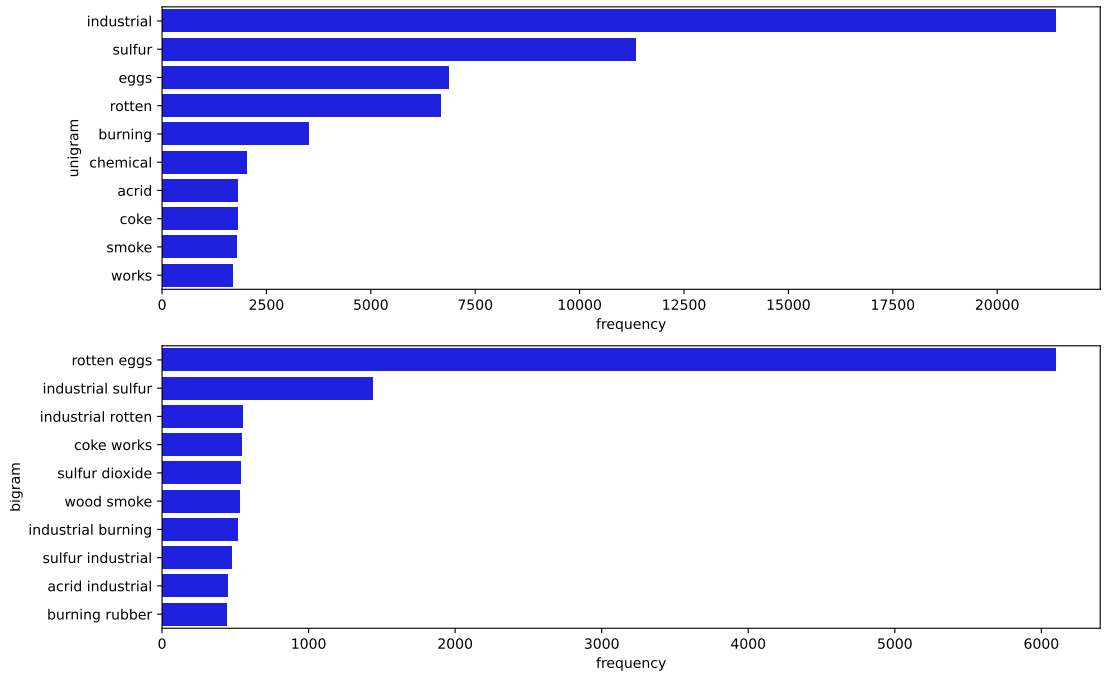

Figure 7: Frequency count of unigrams (top) and bigrams (bottom) of the "smell description" input

Appendix B EXPERIMENTAL SET-UP

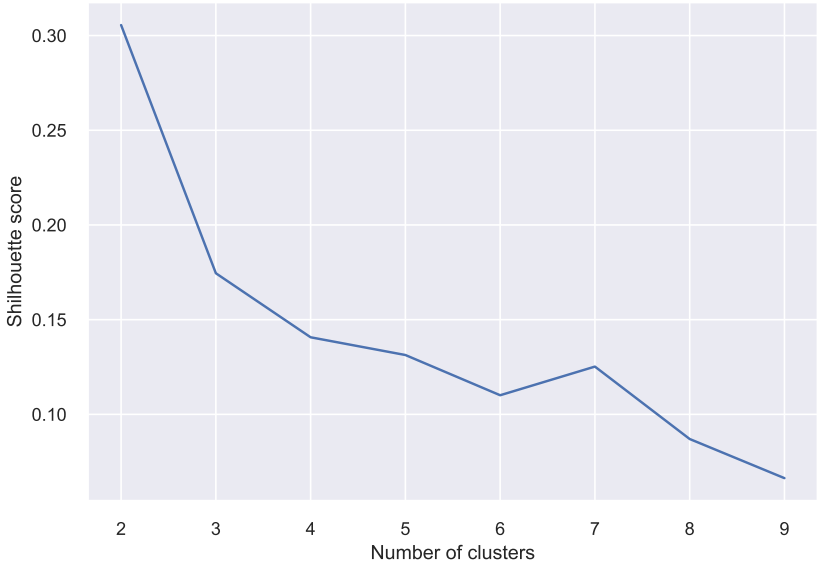

Figure 8: Silhouette score per number of clusters using Bottom-Up + TSKMeans approach