Using a Deep Autoencoder Model to Characterize Air Pollution

Nicolas Shannon[1], Bishnu Timalsena[2], Raul Chavez[3], Vandana Nunna Lakshmi[2]

[1]Department of Computer Science, Loyola University Chicago, Chicago, IL, US
[2]Department of Computer Science, University North Texas, Denton, TX, US
[3]Department of Electrical and Computer Engineering, University of Texas - Rio Grande Valley, Rio Grande Valley, TX, US ← **fix formatting**

Abstract

Climate change is one of the most complex challenges humanity has ever faced. It is a multi-dimensional problem that poses economic, social, scientific, political, and moral questions that must be addressed. One of the most important ways of combating climate change is to research pollution mitigation strategies so that policy makers and government planners are better equipped to make informed decisions. We focused on the principle cause of global warming: the polluting gases responsible for the rapid shift in global climate. We characterize cities by the amount of pollution present using an undercomplete autoencoder model. By representing the level of eight toxic gases present in numerous cities across the United States, we explore some of the more prominent features that characterize urban pollution. We compare two techniques for reducing the dimensionality of our input - Principle Component Analysis (PCA) and Undercomplete Autoencoders. The first two hidden dimensions represented a large percentage of the explained variance in the model, so we cluster the cities in an effort to extract any useful features. **add more later! Also, maybe make this more specific to air pollution rather than climate change in general...?**

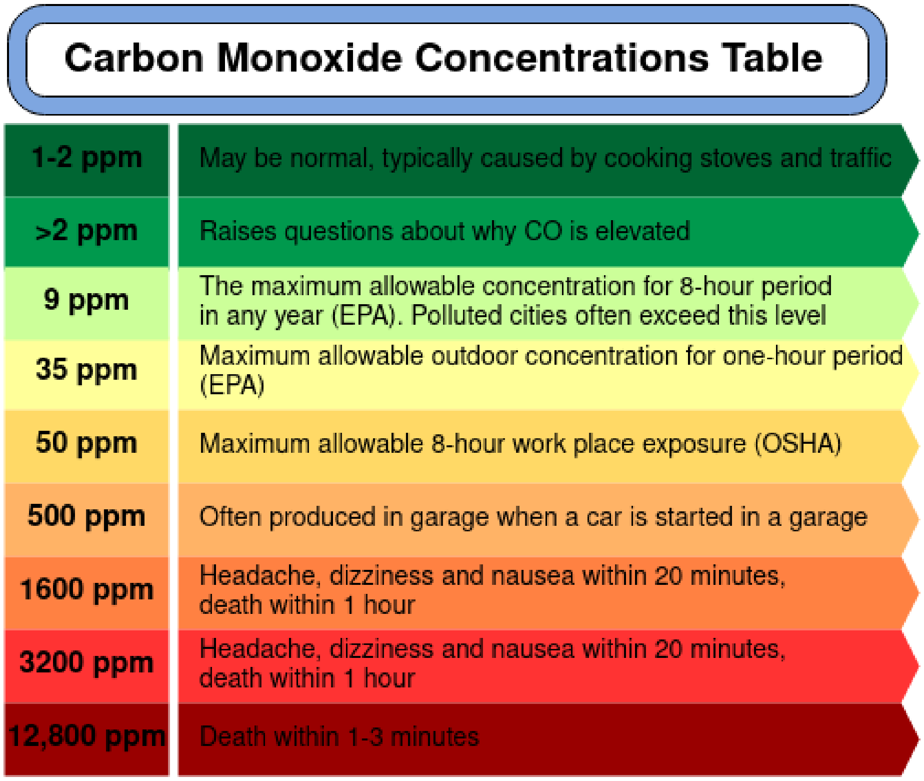
1 Introduction

Over the past several decades, air pollution has been a major concern for human health and the environment. The EPA sets national air quality standards for six major air toxins: ground level Ozone (O^3), Carbon Monoxide (CO), Sulfur Dioxide (SO^2), Lead (Pb), Nitrogen Dioxide (NO^2), and particulate matter (PM). (EPA 2009). The sources of these emissions vary, but the two largest contributors are industrial processes and motorized vehicles.

Short and long term exposure to air toxins pose significant risk to human health. A study by Wang et al. (2019) found a correlation between increased ambient CO levels and outpatient visits for respiratory, cardiovascular, genitourinary, and gastrointenstinal diseases. Several studies performed by the EPA also suggest that people with pre-existing health conditions are at higher risk when exposed to CO over long periods.

1.1 Air Toxins

We wanted to explore how CO and other gases can be characterized with respect to location in an effort to provide researchers and policy makers with the context and tools necessary to make informed decisions. There are a litany of technologies and tools for pollution prevention and control, but most of these options require contextual information and large amounts of data.
By creating an embedding of time series pollution data for numerous cities throughout the U.S., we can explore the more interesting features of the model as well as apply it to tangential areas such as the pollution generated by industrial farming and animal husbandry activities.



Carbon Monoxide Concentrations Table

1-2 ppm	May be normal, typically caused by cooking stoves and traffic
>2 ppm	Raises questions about why CO is elevated
9 ppm	The maximum allowable concentration for 8-hour period in any year (EPA). Polluted cities often exceed this level
35 ppm	Maximum allowable outdoor concentration for one-hour period (EPA)
50 ppm	Maximum allowable 8-hour work place exposure (OSHA)
500 ppm	Often produced in garage when a car is started in a garage
1600 ppm	Headache, dizziness and nausea within 20 minutes, death within 1 hour
3200 ppm	Headache, dizziness and nausea within 20 minutes, death within 1 hour
12,800 ppm	Death within 1-3 minutes

Figure 1: Carbon Monoxide Exposure Table

1.2 A Deep Learning Approach

In this paper we compare the autoencoder model with principle component analysis (PCA), a much older technique for dimensional reduction. Using a deep autoencoder has several advantages. First, deep autoencoders are better at compressing highly complex data into few dimensions. Autoencoders with a nonlinear encoder and decoder function can learn more powerful nonlinear generalizations than PCA. (Goodfellow, Bengio, and Courville 2016). **Autoencoder grapic from poster here**

2 Related Work

The idea of generating embeddings for a location is a highly potential expanse for research. The two major forces behind its competency are, the fact that large volumes of data is being effectively compressed and that these embeddings, when plugged into any model could effectively represent any given location. A lot of research is being done on location embeddings in recent times. For instance, location embeddings are being generated based on human-mobility data (Crivellari, A.; 2019) to understand the behavioral proximity between different locations irrespective of their spatial distances. The sequence of locations that are frequently travelled were fed to a Skip-gram Word2vec model to generate the location embeddings. Another approach that was implemented to leverage location embeddings is, GPS2Vec (Yifang Yin, 2019). This application consists of a neural network that would generate embeddings for a location, based on the semantic contexts that are sourced from the user content published on multiple digital platforms. With the integration of these embeddings, a successful geo-tagged image classification task is demonstrated. In another effort (Carl Yang, 2019), the location embeddings derived from user content such as addresses, phone numbers, images, place names and all other details from multiple individual online posts, are compressed to form low dimensional embeddings to solve the place duplication issue. Deduplication of locations, is a paramount task for any digital platform since they provide a place graph to its users, which is the result of merging location data from multiple sources. This merge leads to duplicity since different sources could hold different names for the same place. By applying K-NN search, pair-wise duplication prediction on these generated location embeddings, the application can identify the duplicates and eliminate them to result in an efficient place graph.

3 Methodology

We created three models to test our proposed autoencoder approach. We performed a simple linear regression on the encoded data that was generated by both the PCA and AE models, as well as on the unencoded values as a control.

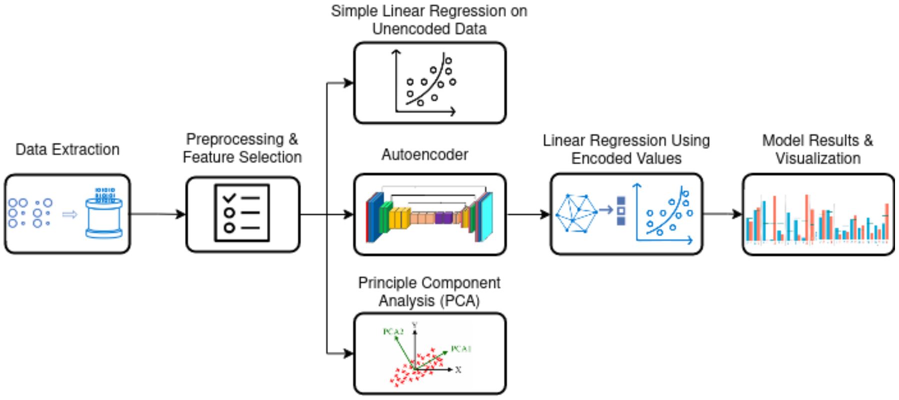

Simple Linear Regression on Unencoded Data
Data Extraction
Preprocessing & Feature Selection
Autoencoder
Linear Regression Using Encoded Values
Model Results & Visualization
Principle Component Analysis (PCA)
PCA2
y
PCA1
x

Figure 2: Data Pipeline

3.1 Data Collection and Preprocessing

Our data was from OpenWeather's Air Pollution API. We initially looked data for eight polluting gases across a six-and-a-half month time period: January 27th, 2020 through July 6th, 2021. However, we found that we achieved better results with just four months of of data for our input features. Our final dataset included just over eighteen-thousand cities in the United States and other U.S. territories.

Table 1: Dataset Characteristics

OpenWeather Pollution API	
Time period	6.5 months
Time frequency	Daily
# of cities	18,526
# of features	120 (originally 190)
# of component gases	8
Dependent variable	single day of time series (3/27/21)

We removed the dependent y from the input data and normalized the x values for each of the eight component gases.

3.2 Autoencoder and PCA Models

The PCA model was trained with k-fold cross- validation

The main use of an undercomplete autoencoder is not in its ability to perfectly reconstruct the input, but in the useful features it is capable of distilling. In our case, we wanted to see what properties the autoencoder model could extract when given time series air pollution data for various cities throughout the United States. Using data from OpenWeather's Air Pollution API, we made 28,338 requests from U.S. state city's coordinates or U.S. territories city's coordinates for all air pollutant information from November 27, 2020 0600 GMT-4 to June 8, 2021 0500 GMT-4 in a hourly basis. This gave us a total of 26100 successful request. We divided the request in 6 different parts creating 6 different files, to minimize overall run time and could request only the files needed incase of error. We then created 8 new data frames for each pollutant, with the daily average of each, and combined form the 6 API call files. We then removed any row that had any missing values and removed any cities with the same name and pollutant daily averages since we did know which was the original city. While this made the data quickly useable, a more ideal solution would be to add a state/country code to distinguish between cities and to impute any missing values using k nearest values. For more on this see the future improvements section. Our final 8 data frames left us with 18526 rows and 195 columns. In the columns there was city, lat, lon, and 192 days. We compared the autoencoder model to Principle Component Analysis, a much older technique used for dimensionality reduction that has been a part of statistical literature since the early twentieth century.[2] The main distinction between autoencoders and PCA is that autoencoders can span a nonlinear subspace, whereas PCA learns the principle subspace. Autoencoders that employ nonlinear encoder and decoder functions can create a more powerful generalization than PCA. However, an autoencoder model that is allowed too much capacity is at risk of reconstructing the input "too perfectly" without extracting any useful information. For PCA we set the y values as 2021/06/06. We also normalized the data frame values and did a 5 split K-fold cross validation. Then we ran a linear regression PCA for each pollutant. For auto encoder we set the y values as 2021/03/28. We also normalized the data frame values. We ran a grid search first looking only for the best activation model. We ran 15 different dimensions, which 8 were between 2 and 25 dimensions, with different activations and the rest of the parameters the same. We then choose the activation that appeared more often as the best activation mode for each dimension threw the grid search. This was Leaky ReLU. We did a new grid search with Leaky ReLU and we changed threw the parameters of learning rate, batch and epoch. We once again choose the parameters that appeared more often as the best. The final parameters were learning rate at 0.001, batch size at 64, and epochs at 150. We then ran the auto encoder for each pollutant.[3]

4 Results

5 Discussion

References

EPA (2009). “Carbon Monoxide NAAQS: Scope and Methods Plan for Health Risk and Exposure Assessment”. In: `https://www.epa.gov/sites/production/files/2020-07/documents/2009_04_coscopeandmethodsplan.pdf`.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press.

Wang, Yu et al. (2019). “Carbon Monoxide and risk of outpatient visits due to cause-specific diseases”. In: `https://ehjournal.biomedcentral.com/articles/10.1186/s12940-019-0477-3`.