

GOTTFRIED WILHELM LEIBNIZ UNIVERSITÄT HANNOVER  
FAKULTÄT FÜR ELEKTROTECHNIK UND INFORMATIK

# Information extraction from articles on the impacts of COVID-19 lockdowns on air quality

*A thesis submitted in fulfillment of the requirements for the degree of  
Bachelor of Science in Computer Science*

BY

**Quentin Münch**

Matriculation number: 10031323

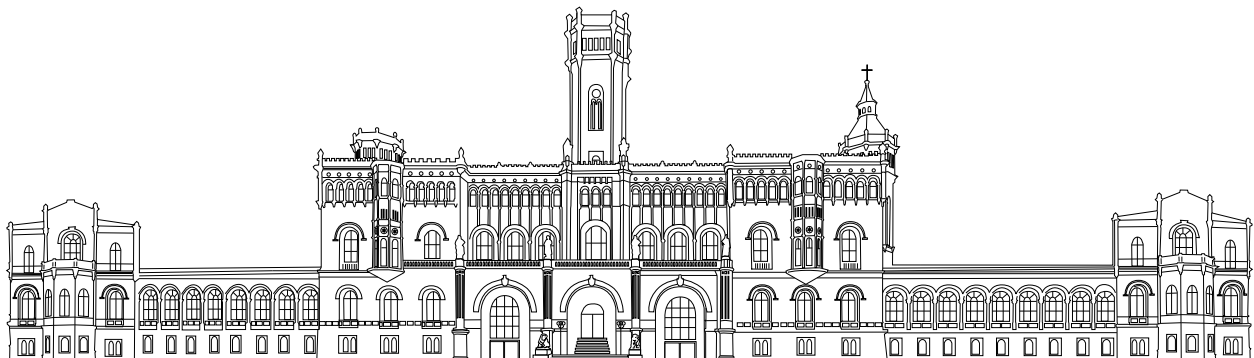
E-mail: [quentin.muench@stud.uni-hannover.de](mailto:quentin.muench@stud.uni-hannover.de)

First evaluator: Prof. Dr. Sören Auer

Second evaluator: Dr. Jennifer D'Souza

Supervisor: Dr. Markus Stocker

31.08.2022





# Declaration of Authorship

I, Quentin Münch, declare that this thesis titled, 'Information extraction from articles on the impact of COVID-19 lockdowns on air quality' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

NAME

Signature: \_\_\_\_\_

Date: \_\_\_\_\_



OPTIONAL



## *Acknowledgements*

XXXXX





## *Abstract*

Your Abstract. Clearly motivate your work (WHY), state what is your problem (WHAT), and describe your solution (HOW). Also, explain how your solution was evaluated (either empirically or formally) and summarized the observed results

*Keywords:*  $KW1$ ,  $KW2$ ,  $KWn$



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Background</b>	<b>3</b>
2.1	Information Extraction . . . . .	3
2.2	Evaluation methods . . . . .	4
2.3	Air pollutants . . . . .	5
<b>3</b>	<b>Related Work</b>	<b>6</b>
<b>4</b>	<b>Approach</b>	<b>7</b>
4.1	Problem Statement . . . . .	7
4.2	Proposed Solution . . . . .	8
4.2.1	Pattern Recognition . . . . .	8
4.2.2	Machine Learning . . . . .	9
<b>5</b>	<b>Implementation</b>	<b>10</b>
5.1	Extraction . . . . .	10
5.1.1	Modules . . . . .	10
5.1.2	Structure . . . . .	11
5.1.3	Patterns . . . . .	11
5.2	Evaluation . . . . .	11
<b>6</b>	<b>Experimental Evaluation</b>	<b>12</b>
<b>7</b>	<b>Conclusions and Future Work</b>	<b>14</b>
	<b>Bibliography</b>	<b>15</b>

# List of Figures

4.1	Example sentence containing a basic pattern . . . . .	8
-----	---	---

# List of Tables

# Acronyms

**GaV** Global-as-View

**GLaV** Global-Local-as-View

**LaV** Local-as-View

**LSLOD** Life Science Linked Open Data

**QEP** Query Execution Plan

**RDF** Resource Description Framework

**RDF-MT** RDF Molecule Template

**RDFS** RDF Schema

**SDL** Semantic Data Lake

**SSQ** star-shaped sub-query

**URI** Universal Resource Identifier



# Chapter 1

## Introduction

The outbreak of the 2019 coronavirus disease (COVID-19) greatly impacted the entire world. Due to the virus being very contagious, hygiene became a lot more important and social interactions had to be reduced to a minimum. While the precise measures to reduce the spread of the virus differed from country to country, lockdowns were introduced in almost every part of the world. People were not allowed to go outside unless it was essential for their living. Companies were told to let the employees work from home, so they didn't need to leave the house. Clubs, bars and restaurants had to close down because of government orders as well as a lack of customers. Considering these and many other points, it is apparent that people were not driving around as much as before the pandemic. Rather they were staying at home, thus considerably reducing their transportation emissions.

Many scientists saw this as a unique opportunity to research how much of an impact such lockdowns have on air quality. While daily life was put on hold because of the restrictions, the streets of major cities were emptier than ever. Many articles regarding air quality have been published, however not all followed scientific methods and reviewing practices. In order to see all the data in one place, the research center Jülich accumulated every article pertaining to the change in air quality during lockdowns. The examined articles are all scientifically verified [3]. These articles contain a great amount of data, which has to be processed manually. The research center Jülich processed each article individually and collected the air quality data. They set up a website which presents the results in various charts.

However, the process of manually having to check each research article proves to be very inefficient. Therefore, models for automated information extraction of such air quality data points could provide a fitting solution. Since scientists are still conducting new research regarding the connection between lockdowns and air quality,



these models would also be useful for later database updates. By automatically extracting the data, the time spent skimming through the papers will be significantly reduced. On the contrary, the technology is errorprone. It is not expected to find every single data point that exists in the text. However, at the very least, it will provide a solid baseline for further examination.

# Chapter 2

## Background

There are huge amounts of text data on the internet. In fact, there is so much of it, that no human can possibly ever read and understand everything. That is why we try to use the computer to help us guide through the data. This is mainly done using information extraction. To understand the development of this thesis, this chapter introduces the main topics.

### 2.1 Information Extraction

There are several strategies for establishing order across texts, the most common being information retrieval (IR), information filtering (IF) and information extraction (IE) [2]. Information retrieval concerns itself with all the activities related to the organization of, processing of, and access to, information of all forms and formats. It can also be seen as a document retrieval system, since it is designed to retrieve information about the existence of documents relevant to a user query [1]. On the other hand, information filtering aims to remove irrelevant data from incoming streams of data items [5]. Information extraction refers to the automatic extraction of structured information such as entities or relations from unstructured sources [7]. In contrast to IR systems, IE systems need to extract facts from the documents itself. The extracted data is often used for filling in databases, which are then available for various applications to further process the data [6]. Since information is often spread across multiple sentences, understanding natural language is fundamental to IE [4]. This can prove to be a great challenge, because computers process information differently compared to humans. While the human perception can easily create relations between entities, the computer has to process each word

bit by bit. Usually it forgets most of it a few words later, hence it has difficulties finding relations between words. However, due to the advancement of technology, these problems have been reduced.

Before the actual extraction occurs, it is often beneficial to employ various preprocessing techniques. These include splitting up sentences into tokens (words, punctuation marks, etc.), recognizing the end of sentences, detecting word types, tracing back words to their original form or even correcting small spelling mistakes [6]. This preprocessing results in an enhanced performance during the extraction process, because most of the hard work has already been completed. The document is now better structured and prepared for further analysis.

## 2.2 Evaluation methods

There are two metrics used for measuring the performance of the retrieval of data from a collection. Let  $N_{correct}$  be the correctly extracted data,  $N_{extracted}$  all extracted data and  $N_{relevant}$  all relevant data. Then [4]

$$\begin{aligned}\text{precision} &= N_{correct}/N_{extracted} \\ \text{recall} &= N_{correct}/N_{relevant}\end{aligned}$$

$N_{correct}$  is also known as true positives (TP).  $N_{extracted}$  consists of TP and wrongly extracted data, called false positives (FP).  $N_{relevant}$  includes TP and relevant data that was overlooked by the program, called false negatives (FN). Precision acts as a general metric to display the ratio of correctly and incorrectly retrieved data, while recall shows the percentage of how much of the relevant data the extraction found. There is also the possibility to combine these two measurements to create an "F score". Traditionally the F score is defined as

$$F = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

However, there are also other possible specialisations of the F score that put further emphasis on either precision or recall [4]. A higher score indicates a better performance of the extraction. Generally, it is expected to reach a precision and recall value of 90%, especially in specialised domains [6].

For a successful evaluation, a good amount of data is essential. This data splits into two different areas called training and test data. The training data is the most important part and typically gets the largest portion of the available data. Like the name suggests, it is used for training the model. This data includes all previously

collected information that has already been verified and found to be correct. Using this information, the model can now be trained. By feeding past results to the program, we aim to prime the model such that it identifies future results on its own. The larger the training data set is, the better this process becomes.

The remaining amount of available data must not be used for training, but for testing. This data is necessary for the evaluation. To assess the quality of the model's extraction, we run the program on the test data. However, we do not use this data to modify the algorithm in any shape or form. We simply need to compare the results of the program with the already verified test data. For this we use the previously mentioned precision and recall values.

A typical ratio between training and test data would be 80/20 or 70/30. This provides ample data for training as well as testing and ensures a well-rounded model.

## 2.3 Air pollutants

Since the articles for this project focus on the change of air quality during COVID-19 lockdowns, air pollutants play a significant role. It is important to recognise the most important pollutants, so that we know what to look for in the text. The main air pollutants are as follows:

1. nitrogen dioxide (NO<sub>2</sub>)
2. particulate matter (PM<sub>2.5</sub>, PM<sub>10</sub>)
3. ozone (O<sub>3</sub>)
4. carbon monoxide (CO)
5. sulfur dioxide (SO<sub>2</sub>)
6. ammonia (NH<sub>3</sub>)
7. nonmethane volatile organic compound (NMVOCs)

ORKG?

# Chapter 3

## Related Work

Topics related to this thesis have been extensively treated in the literature. This chapter presents an overview of what has been done

With the internet becoming more and more popular over the past decades, available information online increased rapidly. Consequently, the need for efficient algorithms that can find data reliably and quickly increased as well. Thus, extracting information from texts has manifested itself as a very important research field. A lot of research has already been done in this topic, especially in the medical field where a lot of patient data needs to be processed.

maybe one machine learning and pattern recognition?

something regarding air pollution?

information extraction

text information extraction in images and video

<https://ieeexplore.ieee.org/document/7800261>

<https://www.sciencedirect.com/science/article/abs/pii/S0031320303004175>

<https://www-1sciencedirect-1com-19a3k7zno0e6a.shan01.han.tib.eu/journal/pattern-recognition/vol/132/suppl/C>

# Chapter 4

## Approach

This chapter explains the problem at hand and the proposed solution.

### 4.1 Problem Statement

During the COVID-19 pandemic, many countries declared nationwide lockdowns. People were ordered to stay at home, which resulted in traffic and industrial emissions getting significantly reduced. To conduct research on how much the air pollution actually changed during this time, researchers across the globe thoroughly inspected the air quality. They published their results as articles in several scientific journals. To make these results more accessible and visually appealing, the research centre Jülich aims to gather all the information in one place. They manually read each article, searching for information regarding pollution changes. Afterwards, they compress the information and display it on their website. An approach like that has its positives and negatives. On one hand, you can thoroughly search the entire document knowing that you probably did not miss anything. A human can easily understand and recognize relations in text, which enables correct allocation of pollutant, value pairs. On the other hand however, it takes a lot of time and resources to search for information by hand. When the amount of articles increases on a daily basis, it is especially difficult to keep up. On top of that, after reading through the articles for a while, the concentration decreases steadily. This in turn can lead to careless mistakes or not finding all important information in the text.

Since time is a valuable resource and errors are always undesired, getting the assistance of a computer may improve the situation. Computers are fast, efficient workers, albeit not particularly intelligent in regards to understanding human concepts. They need precise instructions in order to do exactly what we need them to

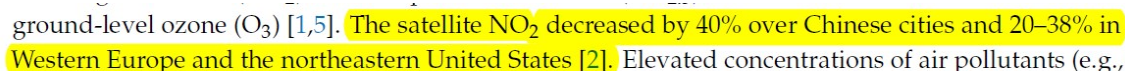
do. This simple knitted system that computers are based on also has its upsides. They are predictable. If we write a program to execute a specific task, the computer will do exactly that. There is no conscience that could influence its actions nor will there be any lack of concentration. Thus a combination of the automatic extraction by a computer and the manual extraction by a human would provide a great trade-off between efficiency and effectiveness.

## 4.2 Proposed Solution

There are generally two viable solutions that can be explored. These are pattern recognition and machine learning. In information extraction there is an important concept to keep in mind. The more data you have for training your model, the better it will be at the extraction process. For our project we have a training data set of 154 articles containing over 1000 points of data. These are the articles that have already been manually searched through by the research centre Jülich. While 1000 data points might seem like a lot at first, in reality it is actually not that much. Consequently the quality of our model will be lower than anticipated.

### 4.2.1 Pattern Recognition

First we will have a look at pattern recognition, which is the most basic form of information extraction. For starters, we will have to investigate the articles themselves. Whenever the text contains any sign of indicating a change in air quality, we need to analyse the sentence in question. By analysing the kind of words that precede or follow the pollutant and the associated value, we can create patterns. The created patterns now consist of a set of words and numbers that represent a sentence containing valuable information regarding air pollutants. An example sentence is shown in 4.1. Examining this sentence, we can deduce the first basic pattern. A pollutant (in this case NO<sub>2</sub>) decreased by a certain amount (40%). Now we just need to let our program know that it should look for this pattern in the text. Following that, each time a part of a sentence matches this pattern, the computer finds it and presents the contained information.



ground-level ozone (O<sub>3</sub>) [1,5]. The satellite NO<sub>2</sub> decreased by 40% over Chinese cities and 20–38% in Western Europe and the northeastern United States [2]. Elevated concentrations of air pollutants (e.g.,

Figure 4.1: Example sentence containing a basic pattern

Unfortunately, having only this one pattern is not enough. There are various different possibilities of describing a change in air pollution. Therefore it is necessary to look through more articles and thus define new patterns, while also further improving existing ones. In the end there will be a wide range of patterns, so that almost every possible expression is covered.

Although the concept of more data equals better results still applies here, it is not essential for the success of the program. As long as you know what you are looking for, you can think of different ways of communicating the information. That way you can create patterns that you think might appear in a document, despite not actually having seen them. This enables training beyond actual training data. However, these patterns have to be carefully evaluated to avoid many false positives. Since our total training data was limited to 153 articles, we decided to pursue this pattern recognition approach. To cover as many different sentences as possible, 59 distinct patterns have been created in total. These are the arrangements of words which the program is looking for in the text.

### 4.2.2 Machine Learning

The second possible approach to an IE problem is machine learning. Like the name suggests, the computer should learn on its own how to accomplish a task. On top of that, it should improve its execution each time it tries anew. For this purpose, the program needs a fitness function which assesses the quality of the result. The aim of the program is to maximise said function. Regarding information extraction, the fitness function could just be the f score, i.e. a compound between precision and recall of the training data set.

There are two types of machine learning approaches, called supervised and unsupervised learning. For supervised learning, the program gets to use labelled data. These labels help guide the computer to improve accuracy and speed up the training process. Usually a human has been processing the data beforehand and labelled them accordingly. Unsupervised learning on the other hand does not use labelled data. It works on its own to determine the structure of the text. Hence, not having any sense of direction, there is a lot more training data needed. Otherwise, the results can end up considerably inaccurate. This is also the main drawback of machine learning. You need an enormous amount of training data for your model to produce satisfactory results. Having only about 150 articles to train from is certainly not sufficient. For this reason we can not apply machine learning to our project and rather concentrate on the pattern recognition approach.



# Chapter 5

## Implementation

The implementation consists of the extraction and evaluation. They are both written using python. Extraction.py represents the core of the project. It contains the code for the extraction of the values, as well as the highlighting of the important text passages. Evaluation.py however, does not influence the extraction process at all. We need it to evaluate the quality of the extraction. It compares the original training data with the output of the extraction and measures the performance.

This chapter lays out the structure of these two scripts and describes the way they operate.

### 5.1 Extraction

#### 5.1.1 Modules

For this project, several external modules need to be imported.

**pandas** Pandas is a valuable tool that provides data structures and data analysis tools. It is especially useful when working with tables. It can automatically convert a csv file to a table in a pandas data frame. Conversely, it can also write a pandas data frame to a csv file. Since the training data is in csv format, using pandas for this project is an obvious choice.

**spacy**

**pymupdf**

**re**

tabula

os

BytesIO

### 5.1.2 Structure

The Program is divided into different sections. Starting with the definitions of different important variables. Noteworthy are the pollutants, the trend words, and the regular expression that forms a number. These three When executed, it...

### 5.1.3 Patterns

## 5.2 Evaluation

yeah

spacy - benefit of kind of words

pymupdf - needed for highlighting

citation for the modules?

# Chapter 6

## Experimental Evaluation

The experimental evaluation is reported in this section. Please, include your research questions.

The research questions addressed by this thesis are: **RQ1)** YYY **RQ2)** XXX **RQ3)** TT **RQ4)** OPP

The remainder of this chapter is structured as follows: First, the used benchmark is described. Second, the data preparation is presented. Afterwards, the setup of the experiment is depicted. Finally, the results are shown and analyzed.

**Benchmark:**

**Metrics:**

**Implementations:**

multicolumn text is challenging to extract from, especially if coming from pdf  
tables not correctly transforming, contain lots of information  
graphics -> very difficult to extract from  
matching numbers to the actual pollutants  
not always accurate, sometimes average of the actual values that got extracted  
problem when sentence spans over two pages -> need the pages for highlighting, cant  
just put it in one line  
many different ways of writing -  
human error - NOx = NO2 ??? sometimes + and - confused, sometimes wrong cell  
(PM2.5 at seattle)  
no or wrong DOI for evaluation  
testdata einzeln evaluieren bitte  
patterns can't be too basic, otherwise precision plummets  
Question: Does the script actually help or is it just a burden  
sometimes rounded numbers in the training data

---

fi from significantly  
difficulty of distinguishing between multipatterns (e.g. 3 vs 5 - northern-china.pdf)  
too many different layouts for tables, none are the same, hard to create a general  
extraction rule  
not a single absolute value found in the articles, taken from graphics, general averages

## Chapter 7

# Conclusions and Future Work

This chapter presents the lessons learned and future work

# Bibliography

- [1] Gobinda G Chowdhury. *Introduction to modern information retrieval*. Facet publishing, 2010.
- [2] Jim Cowie and Wendy Lehnert. “Information Extraction”. In: *Commun. ACM* 39.1 (Jan. 1996), pp. 80–91. ISSN: 0001-0782. DOI: 10.1145/234173.234209. URL: <https://doi.org/10.1145/234173.234209>.
- [3] Georgios I. Gkatzelis et al. “The global impacts of COVID-19 lockdowns on urban air pollution: A critical review and recommendations”. In: *Elementa: Science of the Anthropocene* 9.1 (Apr. 2021). 00176. ISSN: 2325-1026. DOI: 10.1525/elementa.2021.00176. eprint: <https://online.ucpress.edu/elementa/article-pdf/9/1/00176/458795/elementa.2021.00176.pdf>. URL: <https://doi.org/10.1525/elementa.2021.00176>.
- [4] Ralph Grishman. “Information extraction: Techniques and challenges”. In: *Information Extraction A Multidisciplinary Approach to an Emerging Information Technology*. Ed. by Maria Teresa Pazienza. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 10–27. ISBN: 978-3-540-69548-6.
- [5] Uri Hanani, Bracha Shapira, and Peretz Shoval. “Information filtering: Overview of issues, research and systems”. In: *User modeling and user-adapted interaction* 11.3 (2001), pp. 203–259.
- [6] Peter Klügl and Martin Toepfer. “Informationsextraktion”. In: (2014). ISSN: 0170-6012, 1432-122X. DOI: 10.1007/s00287-014-0776-6. URL: <https://www.tib.eu/de/suchen/id/springer%3Adoi%7E10.1007%252Fs00287-014-0776-6>.
- [7] Sunita Sarawagi. *Information extraction*. Now Publishers Inc, 2008.