Text

Description automatically generated

**Knowledge mining in scientific articles: Converting unstructured data into structured data sources**

Bachelor thesis

University of Groningen

**Author:**

Andreea-Ioana Dan

**Primary supervisor:**

Prof. Dr. Dimka Karastoyanova

**Secondary supervisor:**

Mirela Riveni, PhD

**External supervisor:**

Dennis Mulder

28 June 2022

Abstract

**Objective**: To investigate and create a system that would facilitate the conversion of data from an unstructured form into a structured form.

**Methods**: 10 scientific articles that tackle the subject of vaccines against the recently studied coronavirus disease (Covid-19). We will analyze the following sections: Introduction, Methods, Results, Discussion and Conclusion. The technology used will be based on the cloud technologies offered by the Microsoft Azure platform.

**Results**: We have successfully identified relationships between medical entities within the articles and created graph representations of these. These graphs reflect the overall content of the articles in a structured manner. Further on, these graphs can be queried depending on the specific use case that they are used for.

**Conclusion**: ~~ TBD

Contents

[1. Introduction (needs further work) 3](#_Toc107139004)

[1.1. State of the art (needs further work, mind the refs) 3](#_Toc107139005)

[Current situation 3](#_Toc107139006)

[Current Market 4](#_Toc107139007)

[Conclusion 4](#_Toc107139008)

[1.2. Purpose 5](#_Toc107139009)

[2. Methods 6](#_Toc107139010)

[2.1. Azure Cloud Technologies (needs further work) 6](#_Toc107139011)

[2.2. Implementation 6](#_Toc107139012)

[3. Results 12](#_Toc107139013)

[4. Discussion 14](#_Toc107139014)

[4.1. Limitations 14](#_Toc107139015)

[4.2. Use Cases 14](#_Toc107139016)

[5. Conclusion 15](#_Toc107139017)

[6. References 16](#_Toc107139018)

# Introduction (needs further work)

When looking at the trends of data growth over the past years, we can see that data-driven services have become more in demand. A world development report from the World Bank Group in 2021 stated that in 2018 there were approximately 100,000 petabytes (PB) of digital data flows per month, that represented approximately 6,000 billion US dollars in service. To have a broader overview, let us look at the previous years. In 2015, there have been approximately 80,000 PB digital data flows per month, in 2012 approximately 70,000 PB and in 2009 approximately 60,000 PB (World Bank 2021). Respecting the previous trend, we can expect growth in the next years. In 2020, the amount of data processed and used worldwide has quickly reached 59 zettabytes (ZB). Predictions over the next 5 years project data amounts to at least double its amount, reaching almost 149 zettabytes (Berisha and Meziu 2021).

Even though the aforementioned data refers to overall data present globally, the data produced by scientific research is growing steadily (Fire and Guestrin 2019). For example, in the past years, during the pandemic, there has been a surge of research articles in the medical sphere. Between the 1st of January 2020 and 30th of June 2020, there have been at least 23,634 unique documents published around Covid-19 (Teixeira da Silva, Tsigaris and Erfanmanesh 2021).

Thus, the amount of data held by scientific articles nowadays is already more than humans can individually ingest. Expectations are that these amounts will only continue to grow. Therefore, there is a need of automation of processing and mining the data represented by these articles.

## State of the art (needs further work, mind the refs)

### Current situation

The healthcare sector is one of the biggest generators of data. Once Big Data has come in to play and became popular, the healthcare industry has proved to be a globally impactful source of data [11]. There is a rising need for automation of data extraction, processing but also understanding [15]. In this document we will propose an approach that focuses on data extraction and processing from medical scientific articles, although the healthcare industry offers numerous opportunities for applying automation. Tightly connected with improving the patient experience, precision and personalized medicine is making its way to becoming a standard of treatment. In order to apply such an approach in medicine, decisions have to be taken in a data-driven mode. However, in order to offer support for data-driven recommendations, we require a way in which we can process and summarize data. In the past years, several statistical and analytical methods have been used which have been partially successful. Nonetheless, in order to extract and gain knowledge, we need to teach an algorithm to understand the data, not only to cite it [14] because just citing it, still leans on the healthcare providers and other involved stakeholders to filter the available data. In current times, knowledge mining is usually applied on structured data, such as databases, this process is called Knowledge Discovery in Databases (KDD). The data in such databases is either extracted from sensors, is human generated, or stemming from other sources [7]. On the other hand, knowledge mining in unstructured data sources (e.g. scientific articles) is mostly focused on written text study, creating links between facts and identifying patterns [16].

### Current Market

Let us turn our attention to technology currently available that would allow us to answer questions like “How can we mine the data from scientific articles?” or “Can we design a program that converts unstructured data to structured data?”. There are several tools available that can be used for text mining solutions. These include Vizcontrols created by the company Inxight [6], that make use of visualizations to showcase the links between facts (data points). There are also tools that make use of the cloud environment, such as: IBM’s Watson Natural Language Understanding tool [2]; Google’s Natural Language Understanding tool [1]; 2 Microsoft’s Azure Cognitive Services [5] and many others that make use of Natural Language processing technologies. More focused on Text Mining in Scientific Articles is the CAT (Content Analysis Toolkit) by Indutech Ltd. This tool extracts information, clusters it, creates fact connections and then presents a visualization view that can be seen using Excel [17, 16].

### Conclusion

To conclude from the previous findings about the current state of the art, most available technologies focus on extracting data for immediate visualization or conclusion generation. However, we have found less research results that describe a process or a system that is focused on transforming unstructured data into structured data in order for it to be used in long term applications in comparison with simple data extraction research. Therefore, in our research project we shall continue our research of existing approaches for the aforementioned transformation whilst developing our own implementation. Future views Text mining hides an abundance of opportunities for improvement and additional features. Some of these opportunities can include: adding new features such as research results enhanced by semantic analysis or multilingual data extraction [8] but also improving existing features with a focus on increasing the degree of result consistency and reliability [9].

## Purpose

The purpose of this project is to devise an automated way to ingest unstructured data and to obtain graphs that reflect the overall content of the ingested data. The user is then able to query the graph database as required by their current use case. Use cases will be discussed in a later section. The overall flow is described in Figure 1

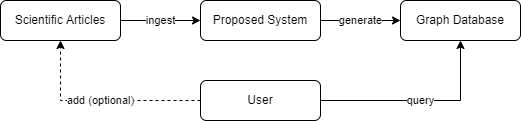


Figure 1: Overall flow

~~ Add more ~~

# Methods

For implementing the proposed project, we used cloud technologies, specifically the Microsoft Azure Cloud Computing Services. The main Azure technologies that have been used are Azure Cognitive Services, Azure Functions, Azure Blob Storage and Azure Cosmos DB.

## Azure Cloud Technologies (needs further work)

~~ Section about Azure overall and then specifically about the technologies that we made use of in general ~~

## Implementation

The idea of this project is to determine an architecture that would allow the ingestion of articles, process the data within and obtain a graph that represents the overall content. However, at further stages of development, we will also test a chosen use case for our system (see later section - TBD).

Visual Studio Code (VS Code) code editor was chosen to develop our system. VS Code is light-weight and intended to be used directly with cloud applications. An advantage of VS Code is that it integrates with Azure via the specifically designed extensions. ([Visual Studio Code Azure Extensions](https://code.visualstudio.com/docs/azure/extensions))

The main chosen language that we used is python. (~~ why)

~~~ (mention VS Code + python)

~~~ (maybe mention the extensions we used?)

Before describing the implementation details, we would like to present the system architecture diagram (Figure 2) that has been created, in order to ease the understanding of this section:

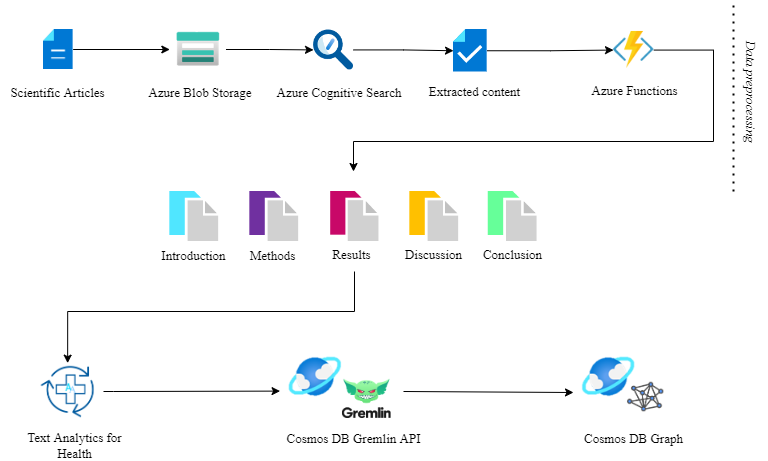


Figure 2: System architecture

(~~ describe in short the diagram)

**Article selection**

To create our dataset that will be ingested by the system, we researched a number of articles that can be used as an initial dataset. For this, we made use of free searchable databases, such as PubMed® but also ScienceDirect® that provided access to peer-reviewed medical journals.

~~~ maybe talk a bit about the articles that we used?

**Article storage**

[Introduction to Blob (object) storage - Azure Storage | Microsoft Docs](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blobs-introduction)

After article selection, they are stored in a Blob Storage container. Blob Storage has been chosen because it can be accessed easily via HTTP/HTTPS from anywhere in the world. Also, there are specific client libraries for python that can be used to manage the stored data easier.

Furthermore, Azure Blob Storage is one of the main data sources that can be used by the Azure Cognitive Search service.

**Extracting Content**

After storing the dataset in the cloud, the Cognitive Search extracts the data and metadata from the files.

~~ what kind of metadata per PDF’s

Cognitive Search has 4 basic elements that handle the data extraction: [Search components - Learn | Microsoft Docs](https://docs.microsoft.com/en-us/learn/modules/create-azure-cognitive-search-solution/3-search-components?ns-enrollment-type=learningpath&ns-enrollment-id=learn.wwl.implement-knowledge-mining-azure-cognitive-search)

1. The index: stores the data extracted from the files
2. The datasource: manages the connection to the Blob Storage
3. The skillset: defines what kind of data will be extracted from the files
4. The indexer: leads the extraction pipeline and connects the previous 3 elements

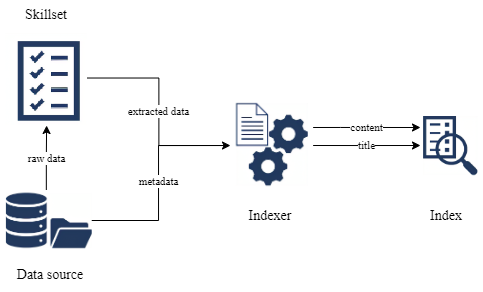
An overview of how the extraction is done:

Figure 3: Overview of the indexing process

After everything was initialized and run, we can query the results via REST API calls as shown in Figure 4 in order to get the data from the built index:



Figure 4: REST API calls to get contents and titles of Blob files

~~ way more to be written here

**Section identification**

The content stored in the index represents the text from the scientific articles as resulted from the *optical character recognition* processes, as it can be observed in Figure 5:



Figure 5: Content stored in search index. *Disclaimer: Actual section is longer, but it was shortened to facilitate reading*

In order to extract specific sections from the content, custom Azure Functions were used. Thus, we made use of regex functions to extract the sections, as seen in the code from Figure 6:



Figure 6: Regex function to extract sections

For this, we assumed that every section is bordered by newlines. The section names have been set to have several values, so that a recursive function verifies several possibilities of sections. For this aspect we encountered several deficiencies that we will discuss in a later section. (~~ discuss the need for standardization).

(~~ discuss the REST API call to the Az Function? Use directly from the cloud;)

**Medical entity and relationship recognition**

In order to store the data in a graph database, we require to obtain relationships between entities. For this, we are making use of a Cognitive Service, namely Text Analytics for Health. This service is identifying medical entities from the given text and the relationships between them.

(~~ discuss the types of relationships that it can find

~~ discuss the types of entities that it can find

~~ discuss the medical dictionary to which it is connected)

In Figure 7 we can see an example of conclusion extracted from an article (~~ give article name) and then few relationships recognized:



Figure 7: Text Analytics for Health applied on a conclusion section

**Creating and populating graphs**

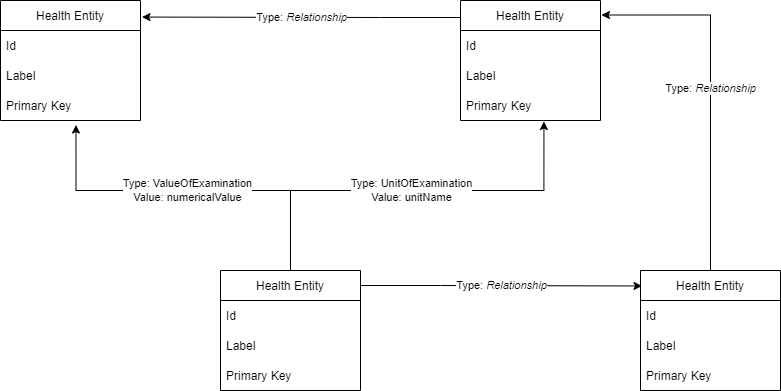
After we have obtained the entity relationships, we are going to use them to design a graph. The following graph schema (Figure 8) has been adopted:

Figure 8: Graph schema; *Relationship* can have different names but no value

(~~ graphs are always unidirectional)

Graphs have been created using the Gremlin Query Language designed by Apache TinkerPop™. Gremlin is a graph computing framework that can be used for both graph analytic systems (OLAP) and graph databases (OLTP). Currently, our system can be classified as an OLTP system.

(~~ maybe show some snippets of code? Not sure)

# Results

After implementing the proposed system, several graphs have been obtained. An example of graph can be observed in Figure 9:



Figure 9: Graph examples; *Disclaimer: Names and values of edges are not seen via the Cosmos DB visualization tool*

The obtained graphs can be queried depending on specific use cases. A section for use cases will be discussed later in this paper.

# Discussion

## Limitations

* Difference in spelling:
  + E.g. Covid-19 or Covid- 19 or Covid – 19
* Abbreviation:
  + E.g. Polymerase chain reaction or PCR
* Word separation:
  + E.g. Current situation or cur-rent situation
* Different article structure
  + E.g. Material and Methods / Methods / Methods and Materials

## Use Cases

* **Twitter messages**

(~~ add more details)

* More use cases…

~~ options for graph visualization

# Conclusion

# References

Berisha, Blend, and Endrit Meziu. 2021. "Big Data Analytics in Cloud Computing: An overview." *[Preprint].* doi:10.13140/RG.2.2.26606.95048.

Fire, Michael, and Carlos Guestrin. 2019. "Over-optimization of academic publishing metrics: Observing Goodhart's Law in action." *GigaScience* 8. doi:10.1093/gigascience/giz053.

Teixeira da Silva, Jaime, Panagiotis Tsigaris, and Mohammadamin Erfanmanesh. 2021. "Publishing volumes in major databases related to Covid-19." *Scientometrics.* doi:10.1007/s11192-020-03675-3.

World Bank. 2021. "World Development Report 2021: Data for Better Lives." Washington, DC. Accessed 03 14, 2022. https://www.worldbank.org/en/publication/wdr2021.