Predicting the Band Gap of Compounds using Machine Learning

Bilal Armiyawo

AiCore

barmiyawo@st.ug.edu.gh

August 3, 2021

Overview

1. Introduction
   - Motivation
2. Methodology
   - Running Project
3. Data Processing and Storage
4. Results
   - Problems
   - Future Directions

Introduction

- Objective
  - Create a data set of elements and compounds with some features.
  - Train a machine model to predict band gap energies of materials based on elemental descriptors
- Motivation
  - Abundance of models for predicting material properties but not band gap energies.
  - Slower to calculate band gap energies from first principles.

Methodology

- Utilized python selenium for extraction of data.
- Mainly using functionalized codes, *OQMD_Database_scraper.py* and *scraper_v3.py*
- Some of the functions within the code include:
  - *get_compounds*()
  - *get_elements*()

Project Resources I

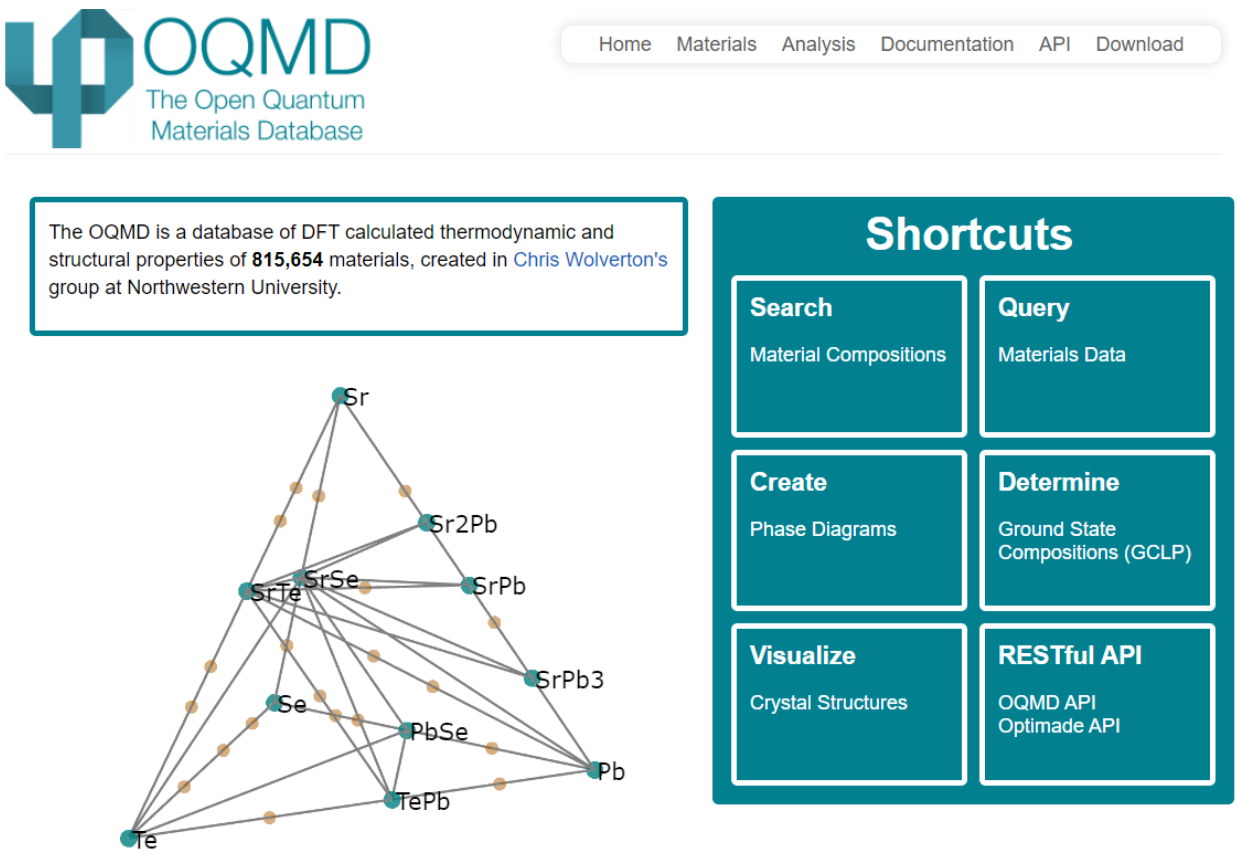
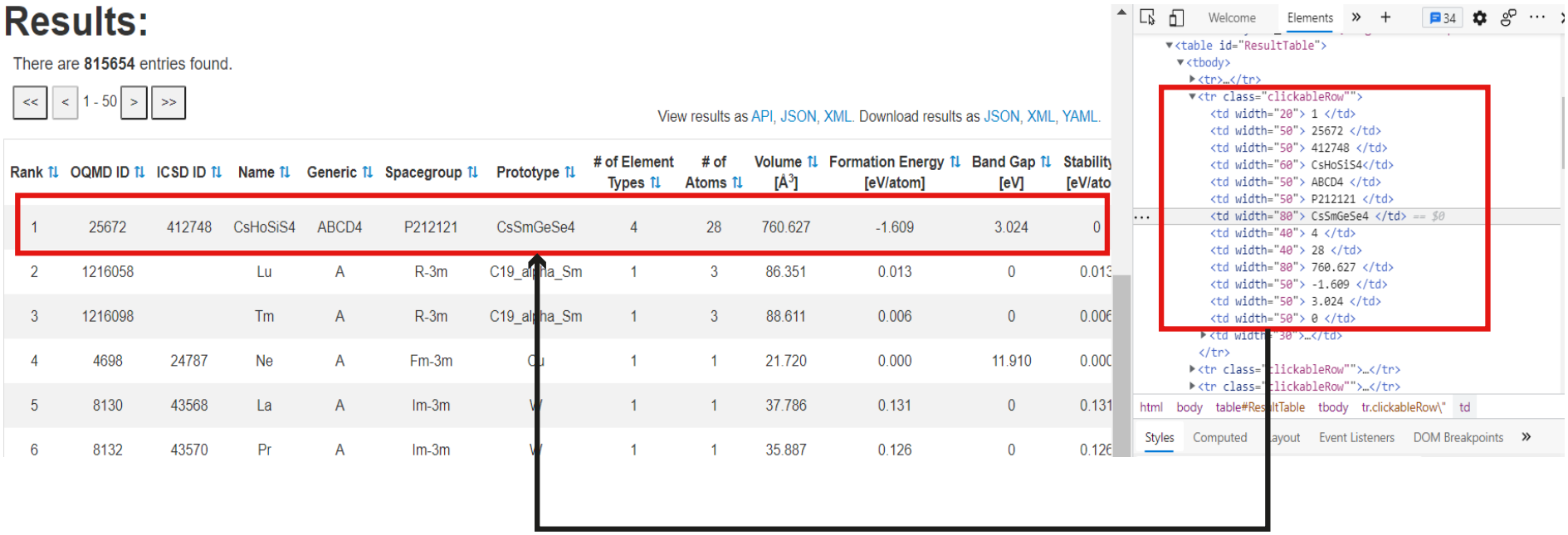


Figure: An image showing the OQMD website

Project Resources II



Results:

There are **815654** entries found.

<< < 1 - 50 > >>

View results as API, JSON, XML. Download results as JSON, XML, YAML.

Rank	OQMD ID	ICSD ID	Name	Generic	Spacegroup	Prototype	# of Element Types	# of Atoms	Volume [Å^3]	Formation Energy [eV/atom]	Band Gap [eV]	Stability [eV/ato
1	25672	412748	CsHoSiS4	ABCD4	P212121	CsSmGeSe4	4	28	760.627	-1.609	3.024	0
2	1216058		Lu	A	R-3m	C19_alpha_Sm	1	3	86.351	0.013	0	0.013
3	1216098		Tm	A	R-3m	C19_alpha_Sm	1	3	88.611	0.006	0	0.006
4	4698	24787	Ne	A	Fm-3m	Cu	1	1	21.720	0.000	11.910	0.000
5	8130	43568	La	A	Im-3m	W	1	1	37.786	0.131	0	0.131
6	8132	43570	Pr	A	Im-3m	W	1	1	35.887	0.126	0	0.126

Figure: An image showing example results of a query for materials data

Project Resources III

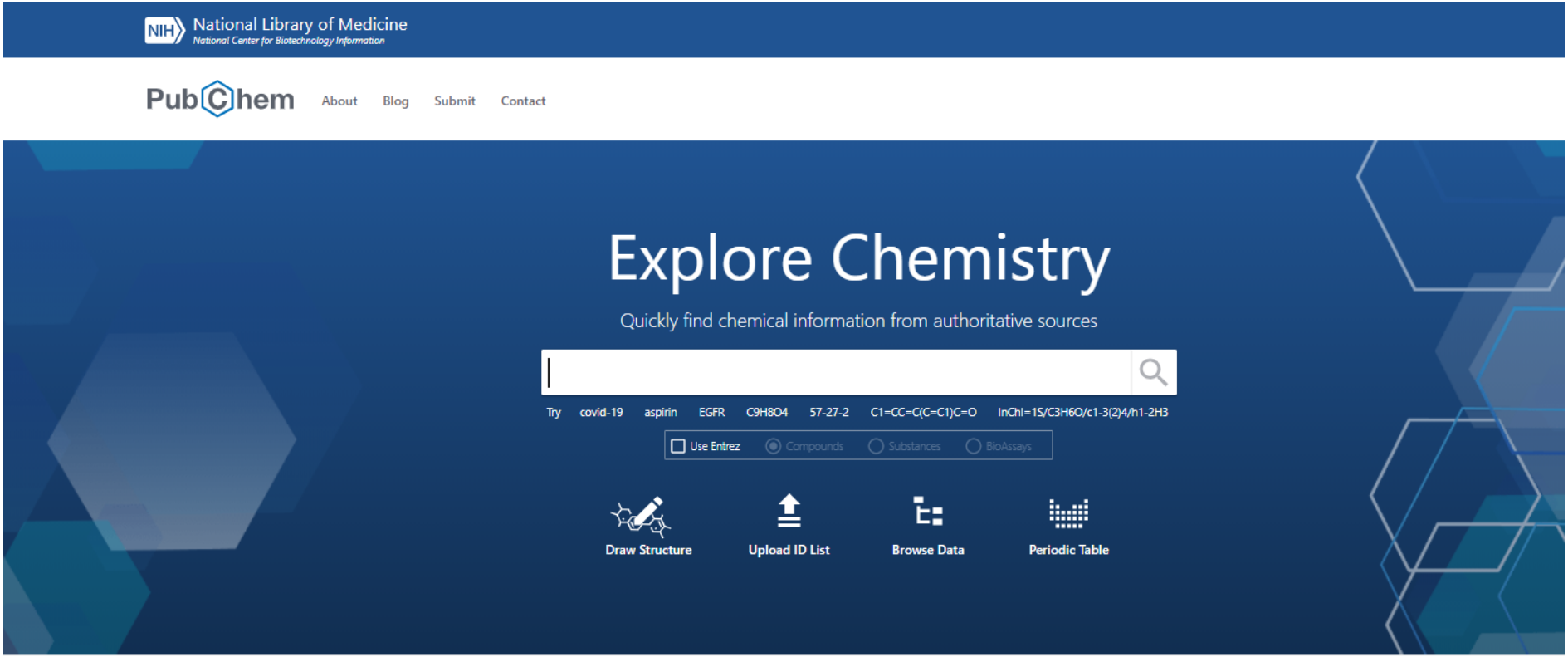


Figure: An image showing the periodic table webpage

Project Resources IV

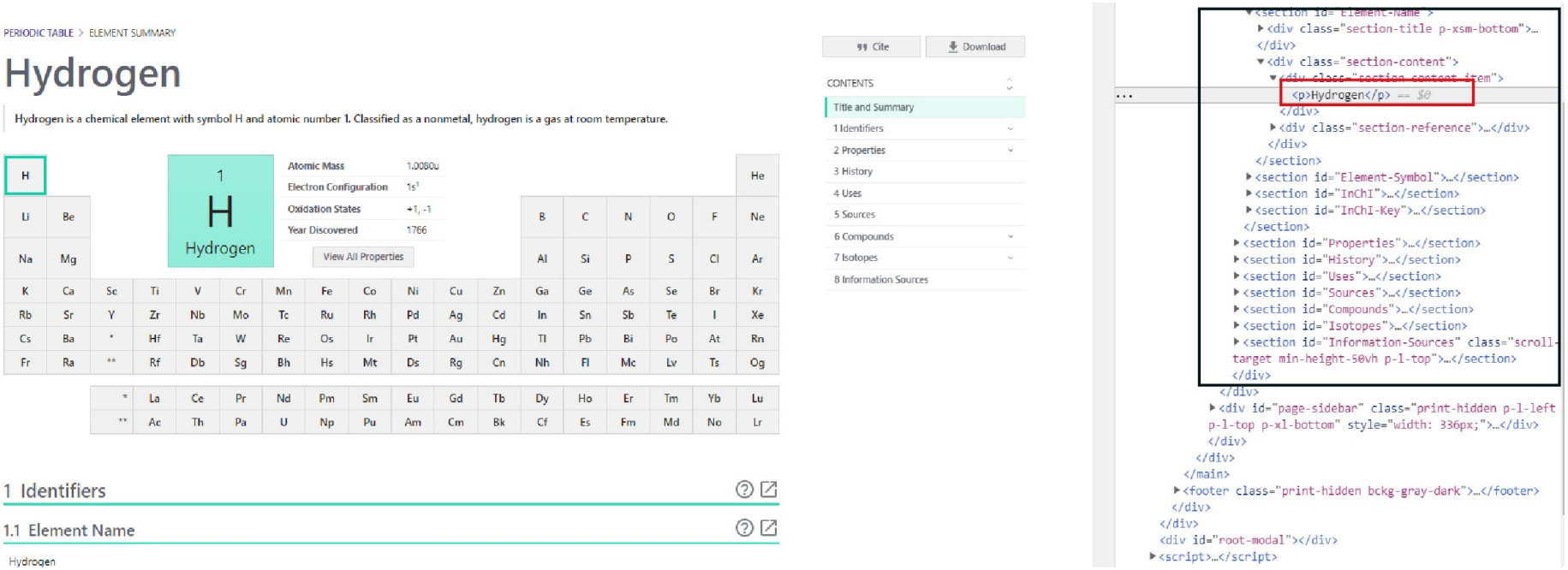


Figure: An image showing Hydrogen page with attributes found in web elements

Running Project

Scraping from OQMD website

To extract data from the OQMD website, run the code:

```
python OQMD_Database_scraper.py
```

Scraping from Periodic Table website

Similarly, extract element data by running the code,

```
python scraper_v3.py
```

Executing with a bash script

Can be executed in a bash script,

```
sh execute.sh
```

Data Processing and Storage

Pandas

1. Determining storage format
2. Cleaning the data
3. Importing to SQL

For the desired structured output, the python library 'Pandas' was utilized.

Table I

Name	Space Group	Volume	Band Gap
CsHoSiS4	P212121	760.627	3.024
CsAu	Pm-3m	77.418	1.040
RbAu	Pm-3m	69.171	0.392

Table: Compounds with some distinct features

Table II

Element Name	Atomic Number	Electronegativity	Boiling Point
Hydrogen	1	2.2	20.28 K
Li	3	0.98	1615 K
RbAu	4	1.57	2744 K

Table: Elements with some distinct features

Results

- Scraper ran successfully locally.
- Included READme file delineating code usage.
- Data was stored locally in csv format.

Problems

- Connection refused when connecting to Postgre due to working from a Windows Subsystem Linux (WSL).
- TimeOutException error when running code.

Future Directions

Steps

- Ran the code remotely on the AWS EC2 server
- Store the data in S3 bucket
- Train a model to predict the band gap energies of simulated combinations of elements using elemental descriptors.