Group 16 - Final Project Report

Suraj Kumar Jha - 2021209(suraj21209@iiitd.ac.in)
Rajat Jaiswal - 2021184(rajat21184@iiitd.ac.in)
Tarun Bansal - 2021210(tarun21209@iiitd.ac.in)

Project Title: Web Scraping tool for extracting the abstracts from Research Papers available on PubMed

Aim: The aim of this project is to scrape and analyze scientific literature from PubMed to identify and document various machine learning and deep learning models used for predicting permeability. The report will provide insights into the dataset size, availability for download, the most commonly mentioned models, top performing model, and extend the analysis to properties beyond permeability, such as Blood barrier Permeability.

Introduction: Permeability prediction is a crucial aspect of drug discovery and pharmaceutical research. Machine learning and deep learning models have gained prominence in predicting permeability, among other properties. This report focuses on the task of scraping relevant scientific literature from PubMed and Google Scholar to gain insights into the models utilized for permeability prediction and other properties like Binding affinity, blood brain barrier permeability.

Methodology: The methodology involves web scraping PubMed and Google Scholar to retrieve articles related to permeability prediction. For each article, the following information is extracted:

1. Title
2. Year of Publication
3. Authors
4. Link to the Article
5. ML and DL Models used
6. Data set size used
7. Download Ability (Availability for download)
8. Top Performing Model

The provided code snippet below performs web scraping on PubMed for articles related to a specified search query, covering the years from 2010 to 2023, covering a broad range of recent research in the field. It iterates through each year and all the available pages of search results, opens the PubMed webpage, waits for results to load, simulates scrolling to fetch additional results, and finally extracts the page source HTML and parses it using BeautifulSoup. This process facilitates the collection of data for subsequent analysis, enabling the identification of machine learning and deep learning models mentioned in the articles related to permeability prediction.

```python
# Loop through each year from 2010 to 2023
for year in range(2010, 2024):
    year_data = []

    # Inside the loop for each page
    for page in range(1, 5):  # Extract data from the first 10 pages
        # Open PubMed
        driver.get(f"https://pubmed.ncbi.nlm.nih.gov/?term={search_query}&filter=years.{year}-{year}&page={pa

        # Wait for the results to load
        time.sleep(5)

        # Scroll down to load more results (you can adjust the number of scrolls)
        for _ in range(3):
            driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
            time.sleep(2)

        # Get the page source and parse it with BeautifulSoup
        page_source = driver.page_source
        soup = BeautifulSoup(page_source, 'html.parser')
```

In the following code segment, for each article in the search results, the title, authors, link, and PubMed ID (PMID) are extracted. The code then navigates to the article's page, waits for it to load, and checks for the presence of a "Save" button to determine if the article is downloadable. It also extracts the article's abstract and the entire text content, converting both to uppercase for consistent comparison. It then iterates through a list of model keywords, checking if any of these models are mentioned in the article's abstract or text. Any mentioned models are recorded. This process allows for the identification of models discussed in the articles retrieved from PubMed.

```python
for result in results:
    title = result.find("a", {"class": "docsum-title"}).text
    authors = result.find("span", {"class": "docsum-authors"}).text
    link = "https://pubmed.ncbi.nlm.nih.gov" + result.find("a", {"class": "docsum-title"})["href"]
    pmid = result.find("span", {"class": "docsum-pmid"}).text

    # Open the link to the article
    driver.get(link)
    time.sleep(5)  # Wait for the article page to load

    # Get the article page source and parse it with BeautifulSoup
    article_page_source = driver.page_source
    article_soup = BeautifulSoup(article_page_source, 'html.parser')

    # Check if the "Save" button element is present inside the article
    save_button = article_soup.find("button", {"id": "save-results-panel-trigger"})
    downloadability = "Yes" if save_button else "No"
    abstract = result.find("div", {"class": "abstract-content"}).text

    # Extract the entire text content of the article page
    article_text = article_soup.get_text()
    article_upper = article_text.upper()

    abstract_upper = abstract.upper()
    models_mentioned = []

    # Iterate through the model keywords and check for mentions in the article text
    for model in model_keywords:
        model_upper = model.upper()
```

Future We Focused on the following aspects:

1. Identifying Top-Performing Models: Analyzing which machine learning or deep learning models are consistently mentioned or associated with higher predictive accuracy in the literature.
2. Dataset Size Investigation: Conducting a systematic search to determine the dataset sizes used in these studies and how they impact model performance.
3. Exploring Other Properties: Extending the analysis to properties beyond permeability, such as Blood Barrier permeability.

```python
data_by_year = {}

def find_data_size(text):
    # Split the text into lines
    lines = text.split('. ')

    max_value = None
    line_with_max_value = None

    for line in lines:
        # Use regular expressions to find all numeric values in the line
        numeric_values = re.findall(r'\d+(?:\.\d+)?', line)

        if numeric_values:
            # Convert the numeric values to float and find the maximum
            line_max_value = max(float(value) for value in numeric_values)

            if max_value is None or line_max_value > max_value:
                max_value = line_max_value
                line_with_max_value = line

    return line_with_max_value

def find_top_performing_model(abstract_text):
    # Split the text into lines
    lines = abstract_text.split('. ')
    line_with_top_model = ""

    for line in lines:
        if re.search(r'\b(?:top|best|leading|outperformed|performance|highest)\b', line, flags=re.IGNORECASE):
            line_with_top_model += line

    return line_with_top_model
```

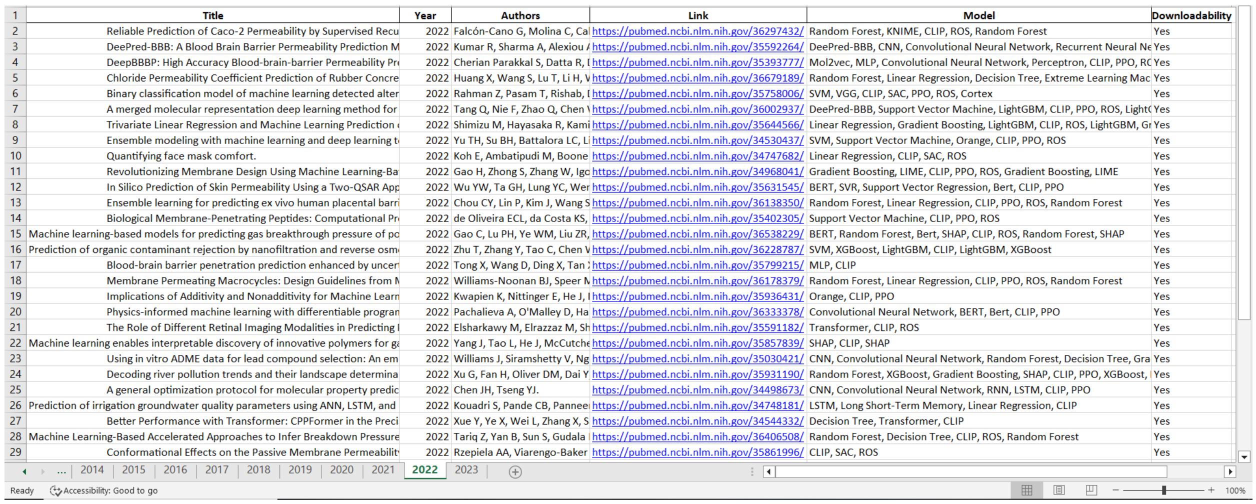
Approach:

1. We use Python with Selenium and BeautifulSoup libraries to automate web scraping.
2. A list of model keywords is provided to identify models mentioned in the articles.
3. We iterate through search results, scroll down to load more articles, and extract relevant information.
4. For each article, we navigate to its page to check if it is available for download and to extract the abstract.
5. The models mentioned in the abstract are recorded for analysis.
6. Data is organized by year, and the results are saved in an Excel file.

Observations:

The web scraping and analysis have yielded valuable insights into the models used for permeability prediction. Key observations include:

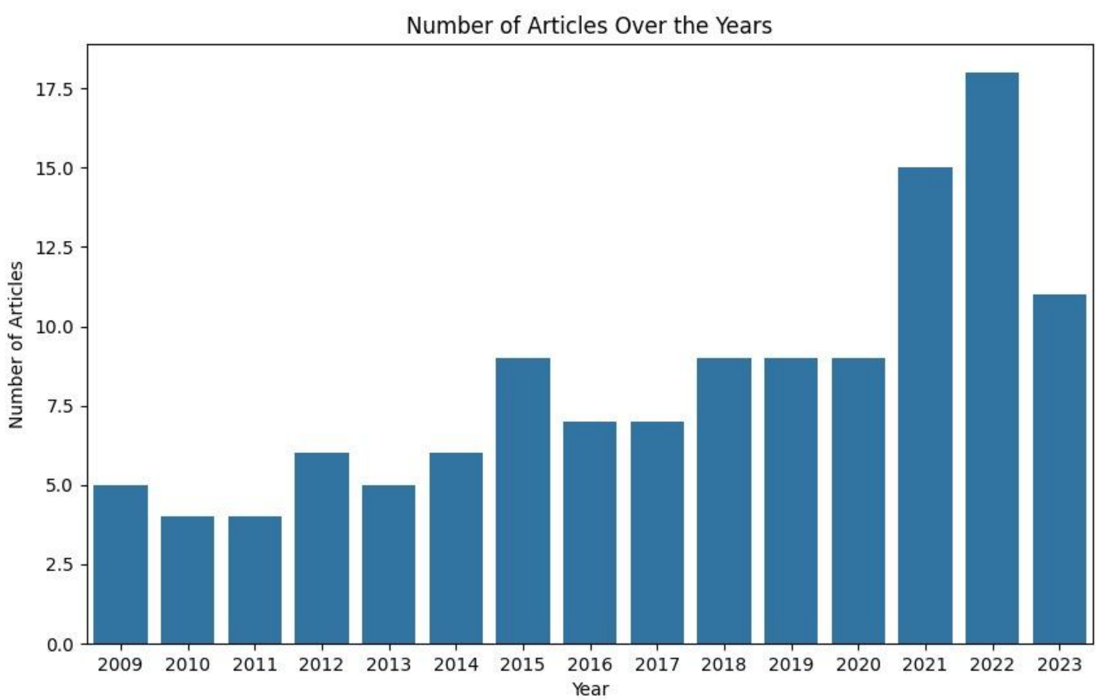
- The dataset size is not directly mentioned in the scraped data.
- Some articles are available for download, while others are not.
- Various machine learning and deep learning models are mentioned, with some being more prevalent than others.
- The report provides a year-wise breakdown of articles, authors, models, and downloadability status.

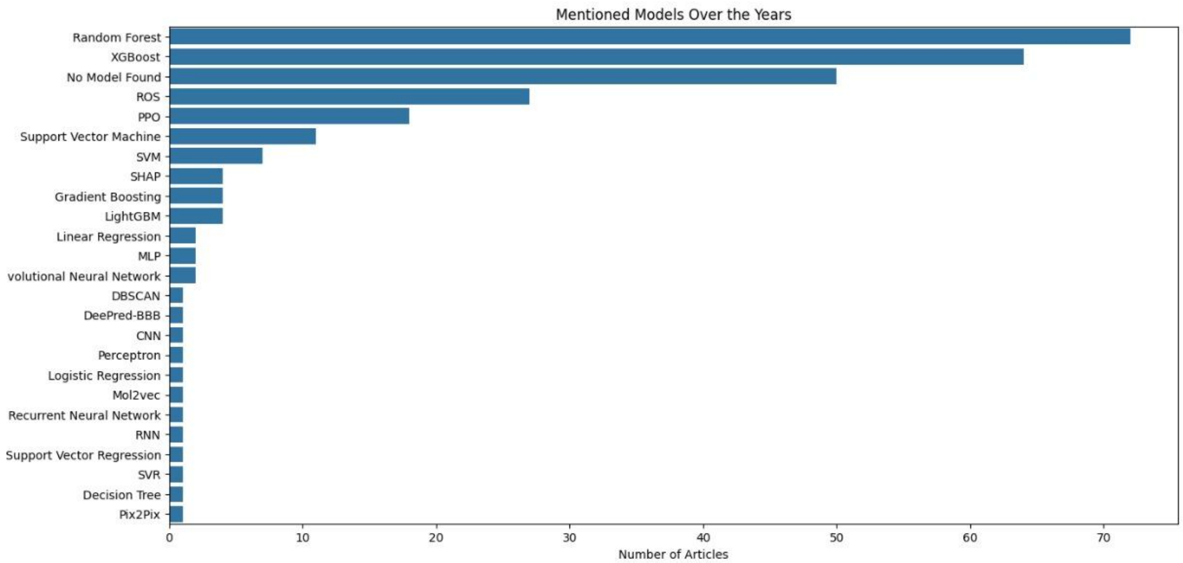


1	Title	Year	Authors	Link	Model	Downloadability
2	Reliable Prediction of Caco-2 Permeability by Supervised Recu	2022	Falcón-Cano G, Molina C, Cal	https://pubmed.ncbi.nlm.nih.gov/36297432/	Random Forest, KNIME, CLIP, ROS, Random Forest	Yes
3	DeePred-BBB: A Blood Brain Barrier Permeability Prediction M	2022	Kumar R, Sharma A, Alexiou A	https://pubmed.ncbi.nlm.nih.gov/35592264/	DeePred-BBB, CNN, Convolutional Neural Network, Recurrent Neural Ne	Yes
4	DeepBBBP: High Accuracy Blood-brain-barrier Permeability Pr	2022	Cherian Parakkal S, Datta R, [	https://pubmed.ncbi.nlm.nih.gov/35393777/	Mol2vec, MLP, Convolutional Neural Network, Perceptron, CLIP, PPO, RC	Yes
5	Chloride Permeability Coefficient Prediction of Rubber Concre	2022	Huang X, Wang S, Lu T, Li H, V	https://pubmed.ncbi.nlm.nih.gov/36679189/	Random Forest, Linear Regression, Decision Tree, Extreme Learning Mac	Yes
6	Binary classification model of machine learning detected alter	2022	Rahman Z, Pasam T, Rishab, [	https://pubmed.ncbi.nlm.nih.gov/35758006/	SVM, VGG, CLIP, SAC, PPO, ROS, Cortex	Yes
7	A merged molecular representation deep learning method for	2022	Tang Q, Nie F, Zhao Q, Chen \	https://pubmed.ncbi.nlm.nih.gov/36002937/	DeePred-BBB, Support Vector Machine, LightGBM, CLIP, PPO, ROS, Light(	Yes
8	Trivariate Linear Regression and Machine Learning Prediction (	2022	Shimizu M, Hayasaka R, Kami	https://pubmed.ncbi.nlm.nih.gov/35644566/	Linear Regression, Gradient Boosting, LightGBM, CLIP, ROS, LightGBM, Gr	Yes
9	Ensemble modeling with machine learning and deep learning t	2022	Yu TH, Su BH, Battalora LC, Li	https://pubmed.ncbi.nlm.nih.gov/34530437/	SVM, Support Vector Machine, Orange, CLIP, PPO, ROS	Yes
10	Quantifying face mask comfort.	2022	Koh E, Ambatipudi M, Boone	https://pubmed.ncbi.nlm.nih.gov/34747682/	Linear Regression, CLIP, SAC, ROS	Yes
11	Revolutionizing Membrane Design Using Machine Learning-Ba	2022	Gao H, Zhong S, Zhang W, Igo	https://pubmed.ncbi.nlm.nih.gov/34968041/	Gradient Boosting, LIME, CLIP, PPO, ROS, Gradient Boosting, LIME	Yes
12	In Silico Prediction of Skin Permeability Using a Two-QSAR App	2022	Wu YW, Ta GH, Lung YC, Wer	https://pubmed.ncbi.nlm.nih.gov/35631545/	BERT, SVR, Support Vector Regression, Bert, CLIP, PPO	Yes
13	Ensemble learning for predicting ex vivo human placental barri	2022	Chou CY, Lin P, Kim J, Wang S	https://pubmed.ncbi.nlm.nih.gov/36138350/	Random Forest, Linear Regression, CLIP, PPO, ROS, Random Forest	Yes
14	Biological Membrane-Penetrating Peptides: Computational Pr	2022	de Oliveira ECL, da Costa KS,	https://pubmed.ncbi.nlm.nih.gov/35402305/	Support Vector Machine, CLIP, PPO, ROS	Yes
15	Machine learning-based models for predicting gas breakthrough pressure of po	2022	Gao C, Lu PH, Ye WM, Liu ZR,	https://pubmed.ncbi.nlm.nih.gov/36538229/	BERT, Random Forest, Bert, SHAP, CLIP, ROS, Random Forest, SHAP	Yes
16	Prediction of organic contaminant rejection by nanofiltration and reverse osm	2022	Zhu T, Zhang Y, Tao C, Chen V	https://pubmed.ncbi.nlm.nih.gov/36228787/	SVM, XGBoost, LightGBM, CLIP, LightGBM, XGBoost	Yes
17	Blood-brain barrier penetration prediction enhanced by uncer	2022	Tong X, Wang D, Ding X, Tan X	https://pubmed.ncbi.nlm.nih.gov/35799215/	MLP, CLIP	Yes
18	Membrane Permeating Macrocycles: Design Guidelines from M	2022	Williams-Noonan BJ, Speer M	https://pubmed.ncbi.nlm.nih.gov/36178379/	Random Forest, Linear Regression, CLIP, PPO, ROS, Random Forest	Yes
19	Implications of Additivity and Nonadditivity for Machine Learn	2022	Kwapien K, Nittinger E, He J, I	https://pubmed.ncbi.nlm.nih.gov/35936431/	Orange, CLIP, PPO	Yes
20	Physics-informed machine learning with differentiable progran	2022	Pachalieva A, O'Malley D, Ha	https://pubmed.ncbi.nlm.nih.gov/36333378/	Convolutional Neural Network, BERT, Bert, CLIP, PPO	Yes
21	The Role of Different Retinal Imaging Modalities in Predicting I	2022	Elsharkawy M, Elrazzaz M, Sh	https://pubmed.ncbi.nlm.nih.gov/35591182/	Transformer, CLIP, ROS	Yes
22	Machine learning enables interpretable discovery of innovative polymers for ga	2022	Yang J, Tao L, He J, McCutche	https://pubmed.ncbi.nlm.nih.gov/35857839/	SHAP, CLIP, SHAP	Yes
23	Using in vitro ADME data for lead compound selection: An em	2022	Williams J, Siramshetty V, Ng	https://pubmed.ncbi.nlm.nih.gov/35030421/	CNN, Convolutional Neural Network, Random Forest, Decision Tree, Gra	Yes
24	Decoding river pollution trends and their landscape determina	2022	Xu G, Fan H, Oliver DM, Dai Y	https://pubmed.ncbi.nlm.nih.gov/35931190/	Random Forest, XGBoost, Gradient Boosting, SHAP, CLIP, PPO, XGBoost, I	Yes
25	A general optimization protocol for molecular property predic	2022	Chen JH, Tseng YJ.	https://pubmed.ncbi.nlm.nih.gov/34498673/	CNN, Convolutional Neural Network, RNN, LSTM, CLIP, PPO	Yes
26	Prediction of irrigation groundwater quality parameters using ANN, LSTM, and	2022	Kouadri S, Pande CB, Pannee	https://pubmed.ncbi.nlm.nih.gov/34748181/	LSTM, Long Short-Term Memory, Linear Regression, CLIP	Yes
27	Better Performance with Transformer: CPPFormer in the Preci	2022	Xue Y, Ye X, Wei L, Zhang X, S	https://pubmed.ncbi.nlm.nih.gov/34544332/	Decision Tree, Transformer, CLIP	Yes
28	Machine Learning-Based Accelerated Approaches to Infer Breakdown Pressure	2022	Tariq Z, Yan B, Sun S, Gudala	https://pubmed.ncbi.nlm.nih.gov/36406508/	Random Forest, Decision Tree, CLIP, ROS, Random Forest	Yes
29	Conformational Effects on the Passive Membrane Permeabilit	2022	Rzepiela AA, Viarengo-Baker	https://pubmed.ncbi.nlm.nih.gov/35861996/	CLIP, SAC, ROS	Yes

... 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023

Ready Accessibility: Good to go 100%





Mentioned Models Over the Years
Random Forest
XGBoost
No Model Found
ROS
PPO
Support Vector Machine
SVM
SHAP
Gradient Boosting
LightGBM
Linear Regression
MLP
volutional Neural Network
DBSCAN
DeePred-BBB
CNN
Perceptron
Logistic Regression
Mol2vec
Recurrent Neural Network
RNN
Support Vector Regression
SVR
Decision Tree
Pix2Pix
0
10
20
30
40
50
60
70
Number of Articles