



MINI PROJECT PRESENTATION SESSION 2022-23 DEPARTMENT OF CSE (DATA SCIENCE)

Job Application Admitter

Mentored by
Mr. Deepak Aneja

Submitted by
Darshit Agarwal
(2000321540024)
Sanchi Singhal
(2000321540051)

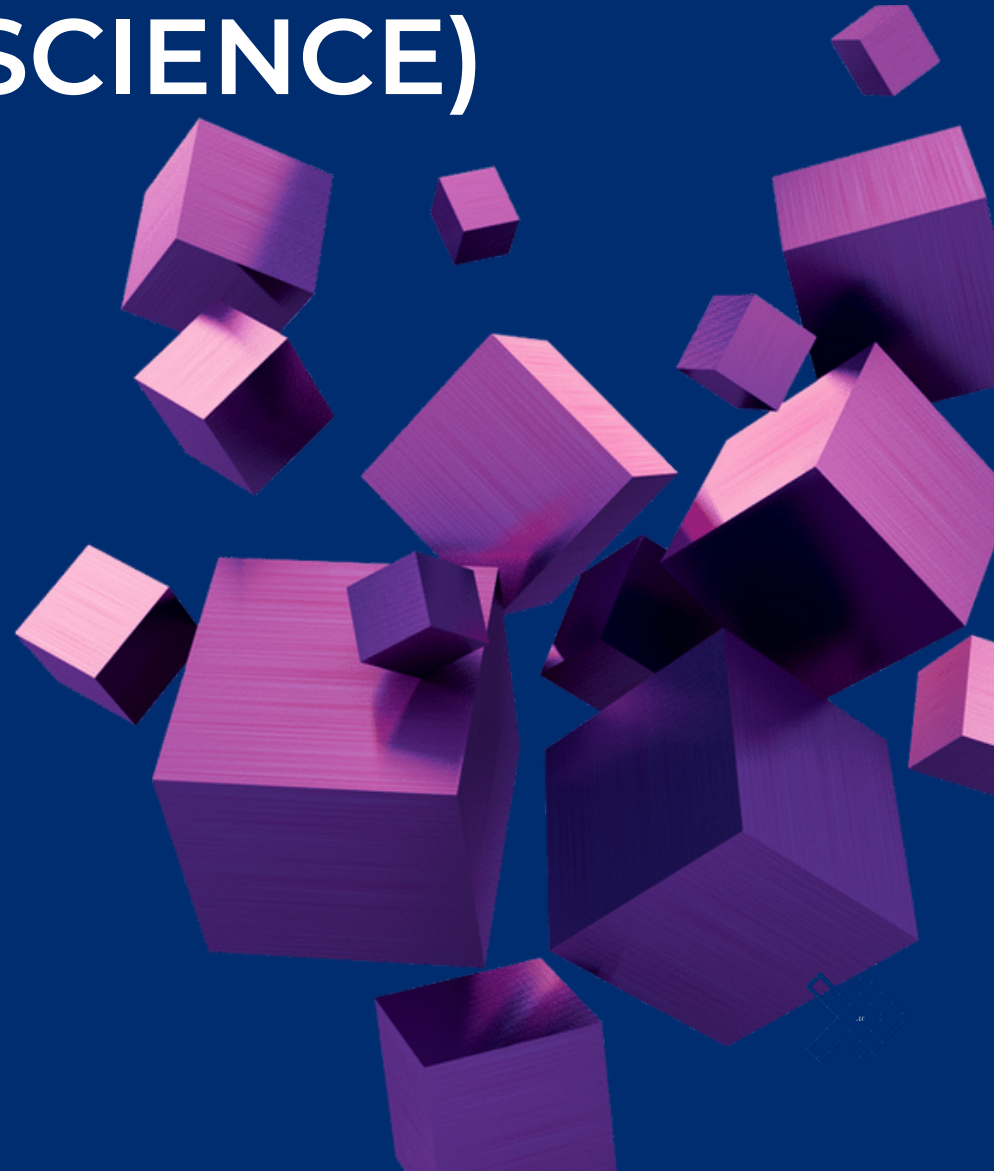
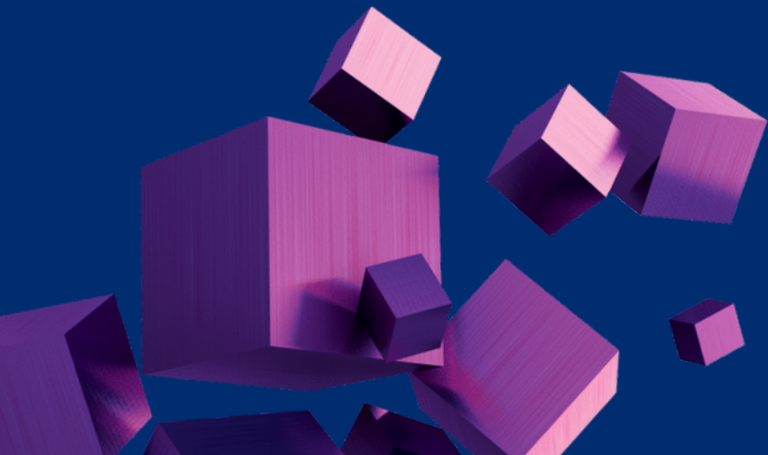




TABLE OF CONTENTS

1. Project Title
2. Abstract
3. Project Flow
4. Base Research Paper
5. Methodology
6. Code Implementation
7. Expected Outcome
8. References





Estd. 2000

JOB APPLICATION ADMITTER





Estd. 2000

ABSTRACT

With emerging technologies, the need for people with different skill sets is highly demanding. A candidate is selected based on the skill set mentioned in his resume while applying for a job. Generally, all the resumes are shortlisted manually, making this very time-consuming. Hence, we have proposed a design that ranks all the resumes according to the demands of a company. It will reduce the working time for further recruitment. In this design, we will use Machine Learning and Natural Language Processing. The algorithm is such that it will search for the skills demanded by the company and rank them accordingly. Some top resumes are to be nominated according to the demands of a company to communicate them for the further recruitment procedure. Our model will be highly effective and less time-consuming contrary to the manual ranking.



Estd. 2000

PROJECT FLOW

Phase 1: Resume
Parsing

Phase 2: Resume
Screening

Phase 3: Resume
Ranking

Phase 1: Resume Parsing

A. Importing necessary libraries and datasets.

B. Extracting the text from resume pdfs.

C. Cleaning the resume text

D. Extracting skills from resume text

Phase 2: Resume Screening

A. Visualization of
distribution of categories

B. Training Machine
Learning Model

**Phase 3: Resume
Ranking**

**A. Getting the skills
required by a Company**

**B. Calculating score for
each resume**

**C. Giving ranks to the
resumes**



Estd. 2000

BASE RESEARCH PAPER

Link:-

https://drive.google.com/file/d/1_zlKFwbVffe1qIDnOZE01Q62Ue9ObHq5/view?usp=share_link

International Journal of Research Publication and Reviews, Vol 3, no 4, pp 2567-2569, April 2022



International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Resume Screening Using LSTM

Navale Sakshi¹, Doke Samiksha², Mule Divya³, Prof. Said S. K.⁴

Department of Computer Engineering, Jaihind College of Engineering, Kuran, Pune 410511.

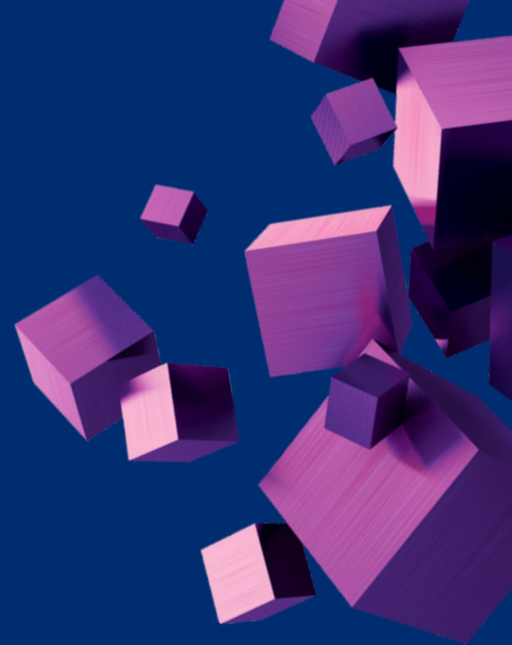
ABSTRACT

Resume Screening is process of determining whether a candidate is qualified for a role based his or her education, experience, and other information captured on their resume. It is a form of pattern matching and classification. Long Short-Term Memory (LSTM) networks are neural networks capable of learning long-term dependencies. Resume Screening using LSTM will take PDF file as input. Data will be extracted from the file. The data extracted will be saved in CSV file. The data from the csv file will then be cleaned and LSTM model will be used to make predictions. The LSTM will be trained and the saved to use in the web application. Applicants GitHub profile will also be retrieved. Web scraping will be used to find details of the GitHub profile. The analysis of the GitHub profile will affect result. User Details will be stored in the database. Data set will be cleaned and then vectorize. The vectorized dataset will then be used for training purpose. The model of bidirectional LSTM will be trained and weights will be saved for further use.



METHODOLOGY

Natural Language Processing(NLP)



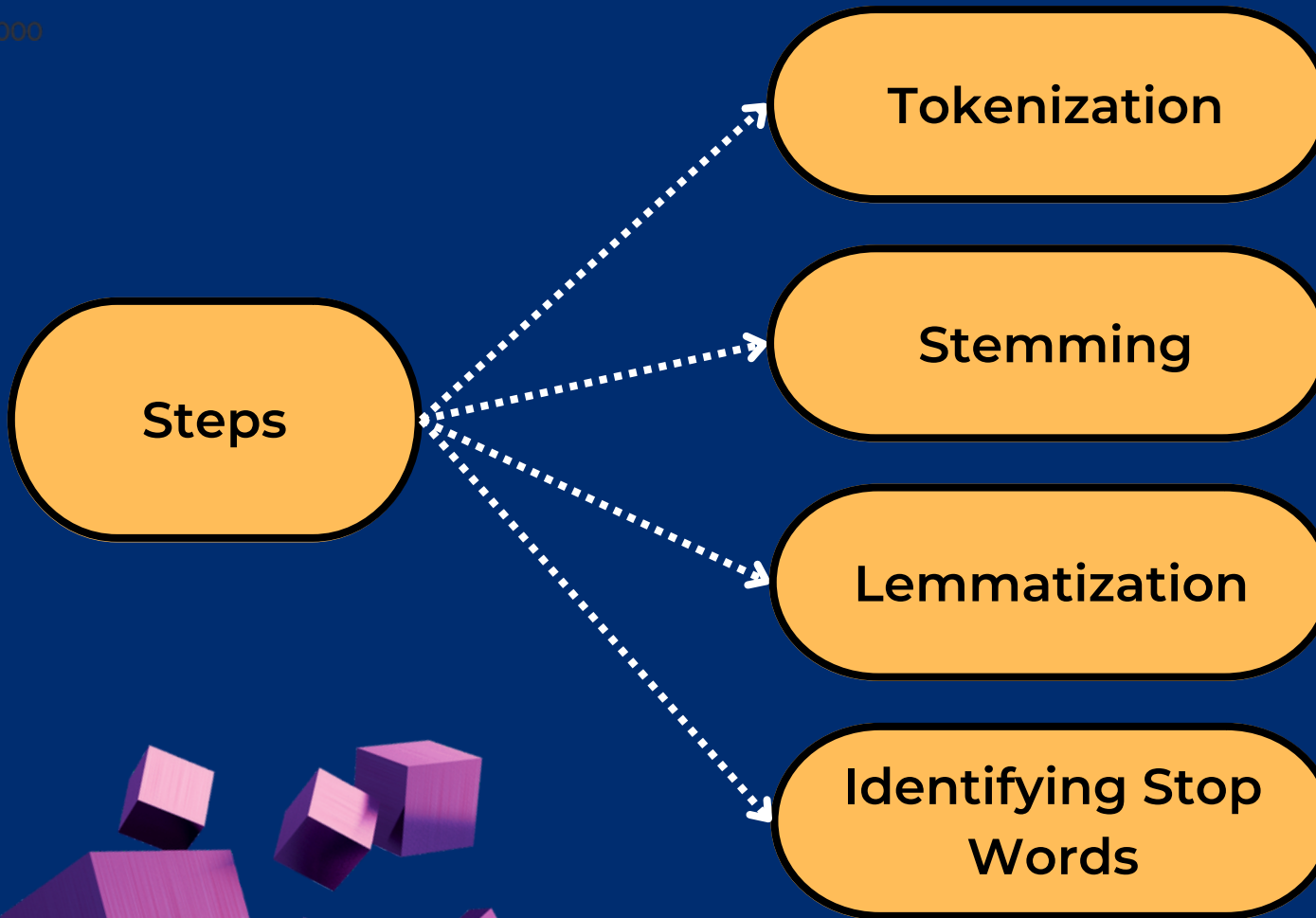
NLP stands for Natural Language Processing, which is a part of Computer Science, Human language, and Artificial Intelligence. It is the technology that is used by machines to understand, analyse, manipulate, and interpret human's languages. It helps developers to organize knowledge for performing tasks such as translation, automatic summarization, Named Entity Recognition (NER), speech recognition, relationship extraction, and topic segmentation.





Estd. 2000

STEPS OF NLP



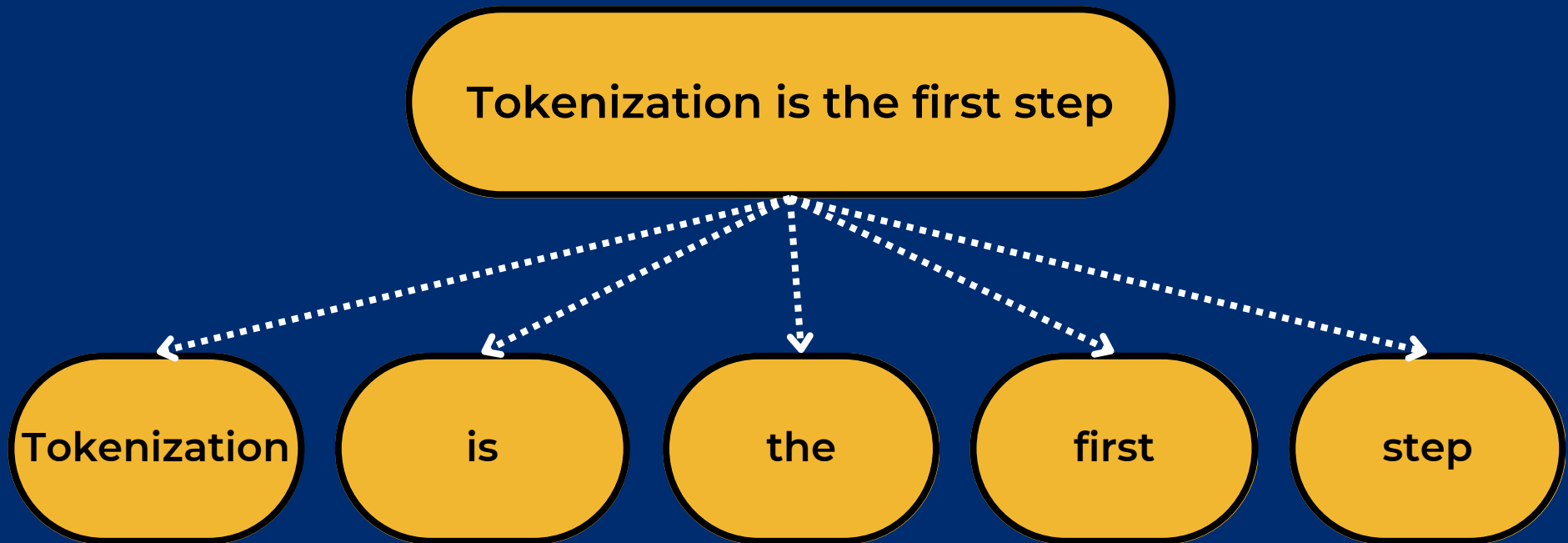


Estd. 2000

Tokenization

To break strings into tokens.

Ex :-



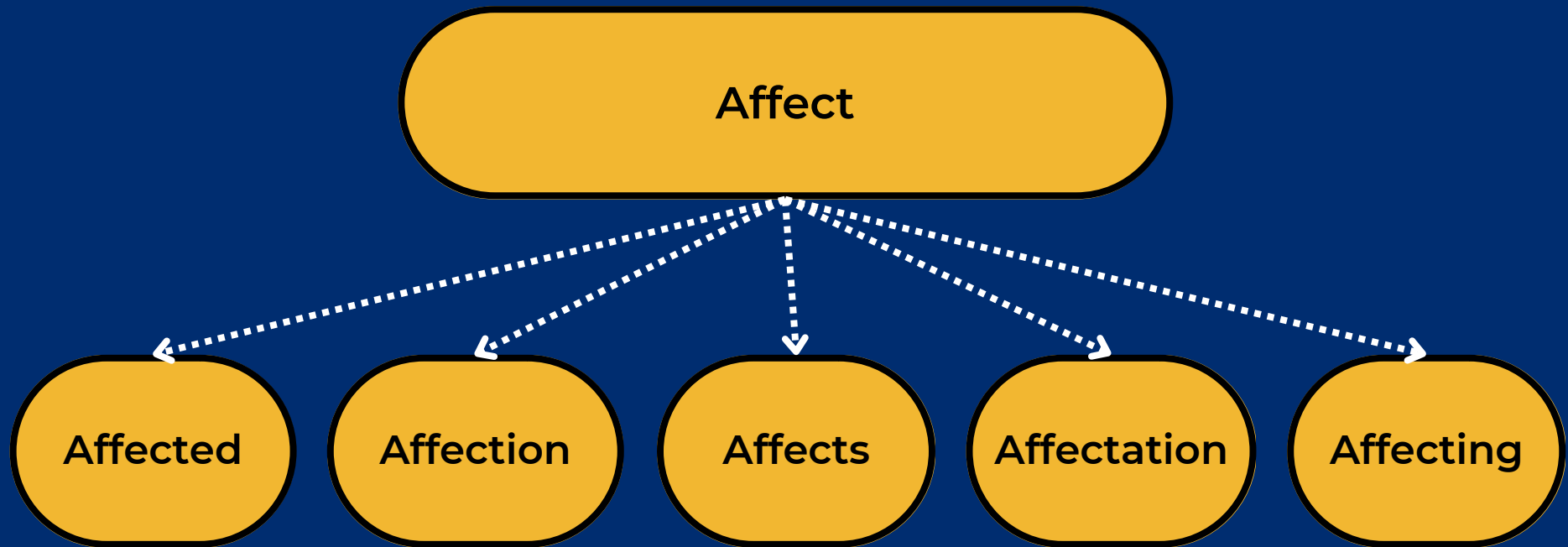


Estd. 2000

Stemming

Normalize words into its base form or root form.

Ex :-



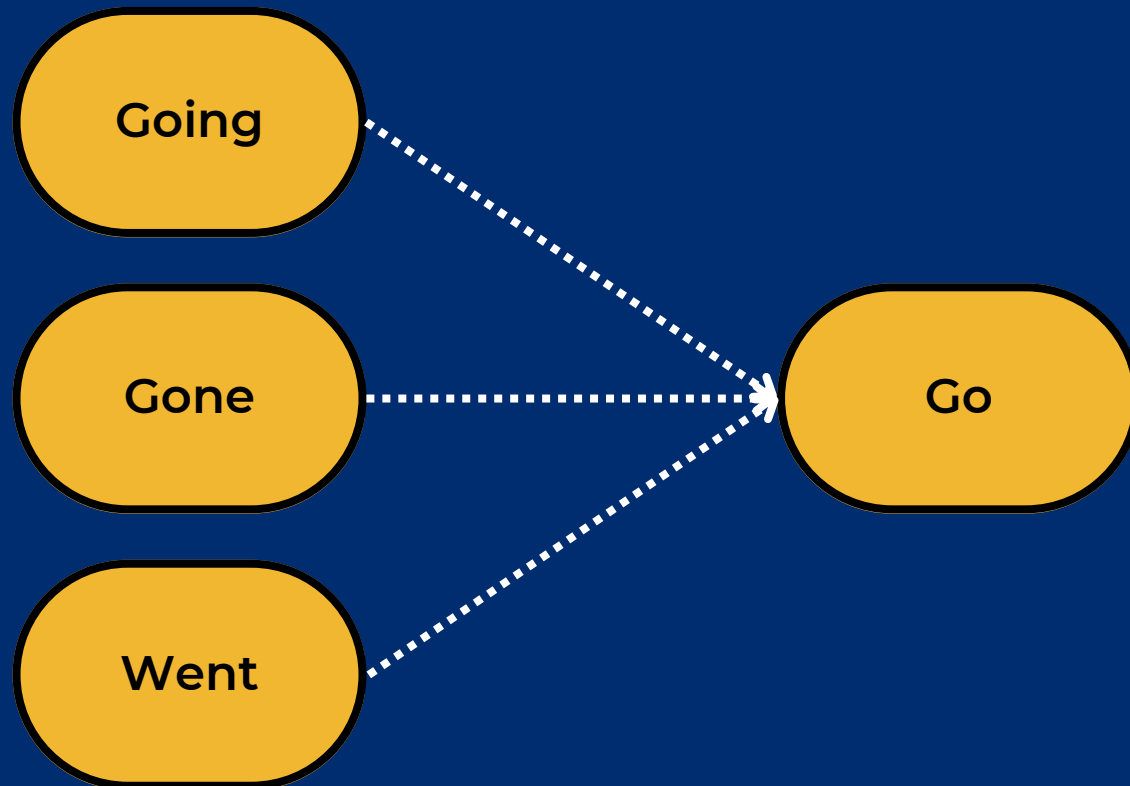


Estd. 2000

Lemmatization

Groups together different inflected forms of a word called lemma.

Ex :-



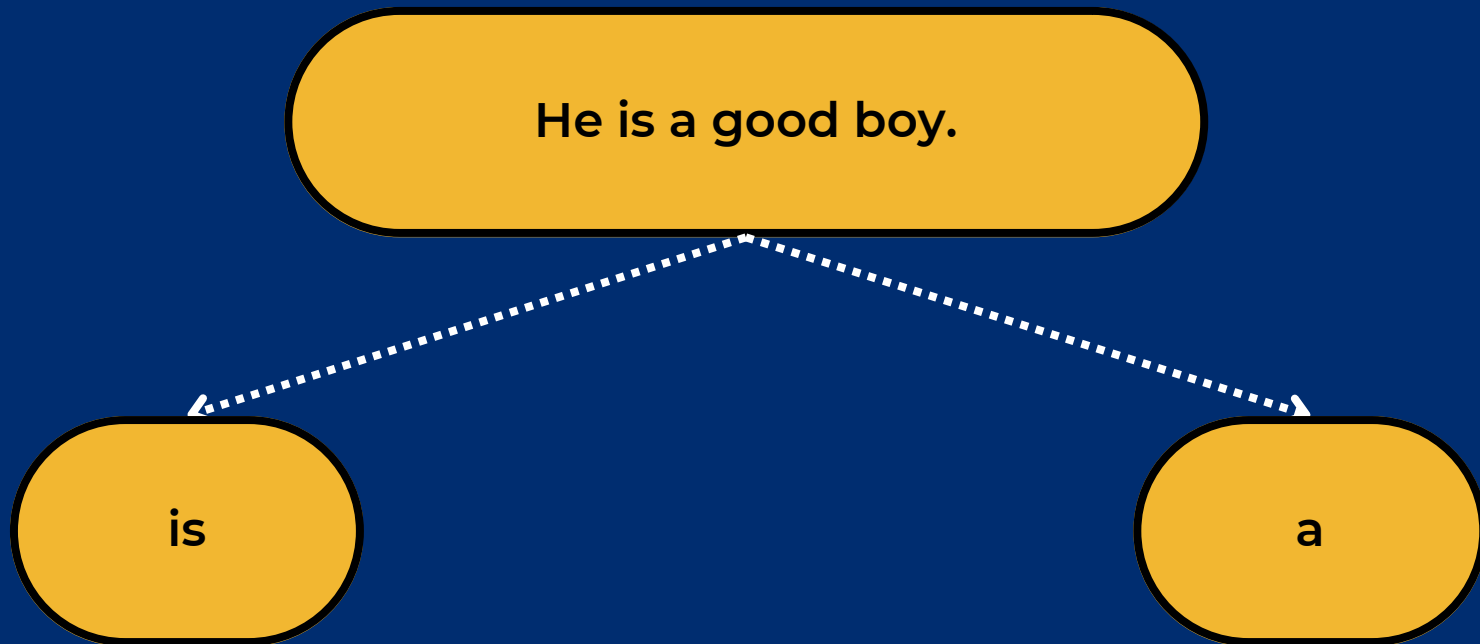


Estd. 2000

Identifying Stop Words

Stop words might be filtered out before doing any statistical analysis.

Ex :-





Estd. 2000

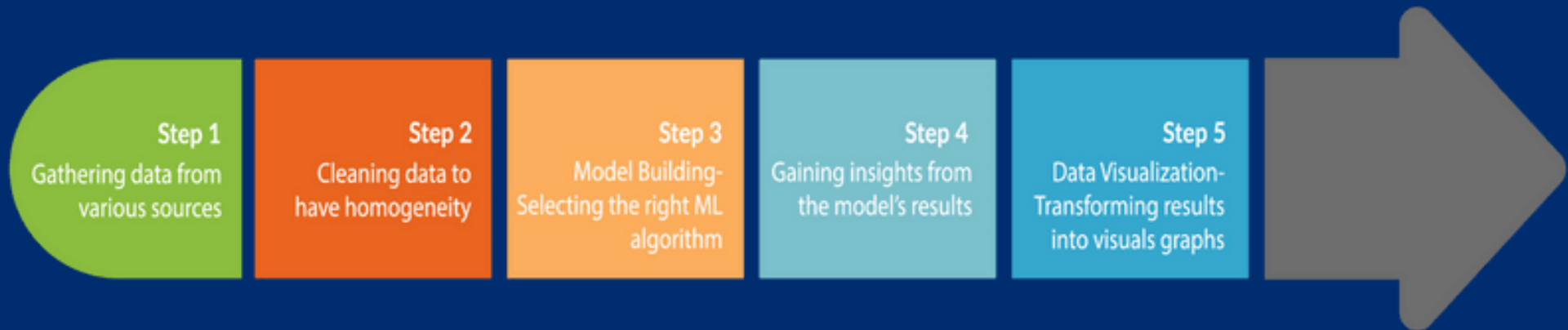
Machine Learning

Machine Learning is the field of study that gives computers the capability to learn without being explicitly programmed. ML is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn. Machine learning is actively being used today, perhaps in many more places than one would expect.



Estd. 2000

Machine Learning Process





CODE IMPLEMENTATION

Importing modules

```
#Data Loading/Data Manipulation
import pandas as pd
import numpy as np

#nltk
import nltk
import re
from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords

#data visualization
import matplotlib.pyplot as plt
import seaborn as sns

# preprocessing and model training
from sklearn.preprocessing import LabelEncoder
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
from sklearn.metrics import confusion_matrix
```




CODE IMPLEMENTATION

Data Visualization

```
plt.figure(figsize=(20,5))
plt.xticks(rotation=90)
ax=sns.countplot(x='Category', data=data)
for p in ax.patches:
    ax.annotate(str(p.get_height()),(p.get_x(), p.get_height()))
plt.show()
```



CODE IMPLEMENTATION

Model Training

```
x_train, x_test, y_train, y_test=train_test_split(Word_feature, target, random_state=0, test_size=0.4)
print(x_train.shape)
print(x_test.shape)
```

```
(625, 2000)
(418, 2000)
```

```
from sklearn.svm import SVC
model=SVC()
model.fit(x_train, y_train)
```

```
SVC()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

```
prediction=model.predict(x_test)
```

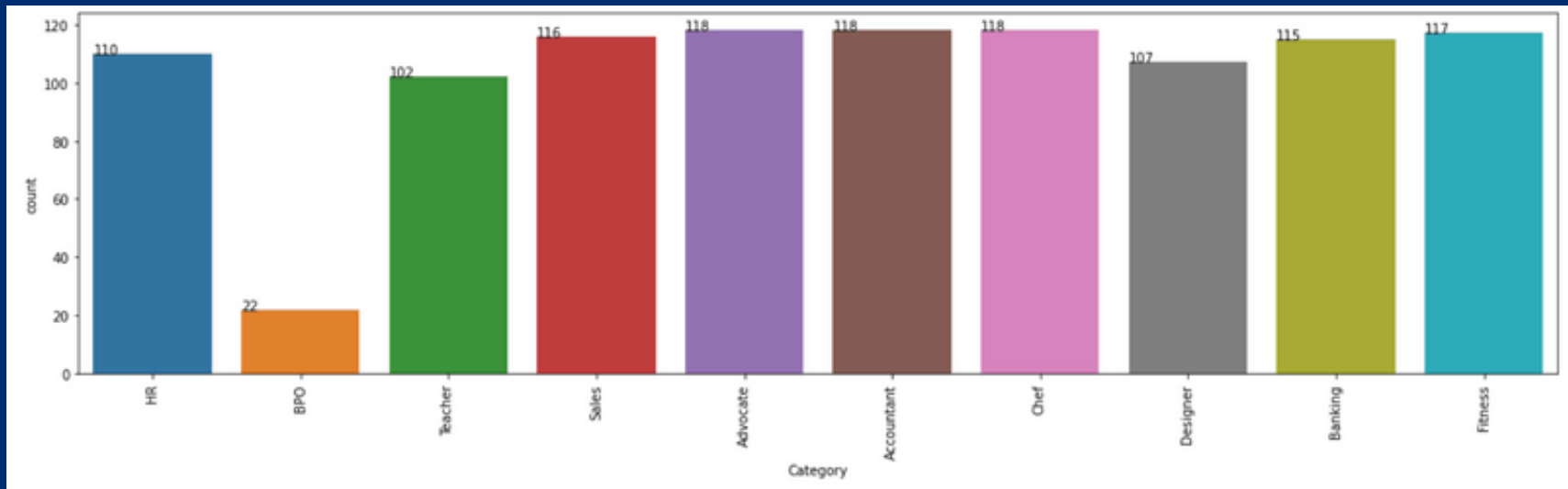
```
print("training Score: {:.2f}".format(model.score(x_train, y_train)))
print("test Score: {:.2f}".format(model.score(x_test, y_test)))
```

```
training Score: 1.00
test Score: 0.78
```



EXPECTED OUTCOME

Top resumes in accordance with recruiter's demand of skills.





Estd. 2000

REFERENCES

Resume Dataset:-

<https://www.kaggle.com/datasets/snehanbawal/resume-dataset>

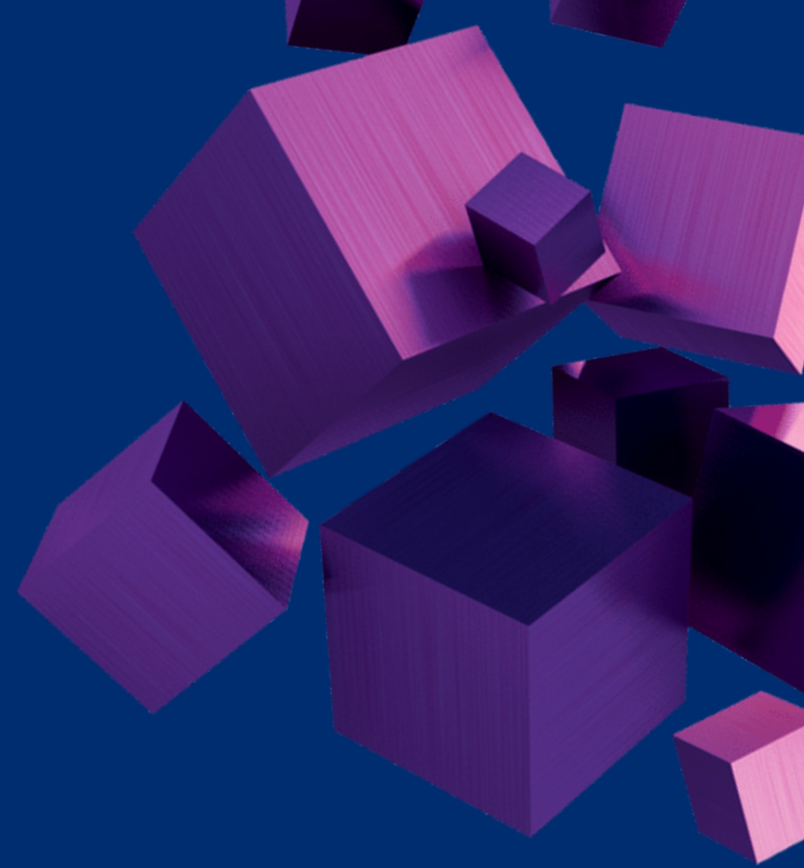
Methodology:-

<https://www.javatpoint.com/nlp>

<https://www.geeksforgeeks.org/machine-learning/>

Research Paper:-

https://drive.google.com/file/d/1_zlKFwbVffe1qlDnOZE01Q62Ue9ObHq5/view?usp=share_link





Estd. 2000

Thank You

