Dr Mahalingam College of Engineering & Technology
Department of Computer Science and Engineering
19CSPN6601 - Innovative and Creative Project
Final Review

Title: Feature Selection Of Reviews Using Cross Validation PSO

Team Number: A005

Team Member(s):

Mohammad Faisal J	- 727621BCS009
Monika K	- 727621BCS039
Vikashini M	- 727621BCS067

Faculty Supervisor: B.SUGANYA AP/CSE (SS)

Date: 16/05/2024

CONTENTS

- Problem description
- Literature survey
- Objective
- Existing System Block Diagram
- Disadvantages
- Proposed System Block Diagram
- Module description
- Implementation Screen Shots
- Conclusion
- Result
- Course Certificates
- References

PROBLEM DESCRIPTION

Conventional sentiment analysis techniques may overlook refined opinions about specific aspects of products or topics. Without aspect-based feature selection, sentiment analysis results may lack rudeness and accuracy. There is a need for a more refined approach to sentiment analysis that considers specific aspects to provide deeper insights

LITERATURE SURVEY

S.No	Title of the paper &year	Author	Inference
1	Bio inspired Boolean artificial bee colony based feature selection algorithm-2024	Omar Alqaryouti et.al.,	Managing non-related features and maintaining classification accuracy could be challenging.
2	Sentiment Analysis of Reviews in Natural Language: Roman Urdu as a Case Study-2022	Muhammad Aasim Qureshi et.al.,	Challenges in handling domain-specific language and evolving slang.

LITERATURE SURVEY (CONTD..)

S.No	Title of the paper &year	Author	Inference
3	A review of sentiment analysis for Afaan Oromo: Current trends and future perspectives-2024	Jemal bate ,Faizur Rashid	Potential bias in feature selection that may impact the accuracy and generalization of sentiment analysis models.
4	A review on sentiment analysis from social media platform-2023	Margarita Rodríguez-Ibanez et.al.,	Limited coverage of contextual information and nuances in sentiment.

LITERATURE SURVEY (CONTD..)

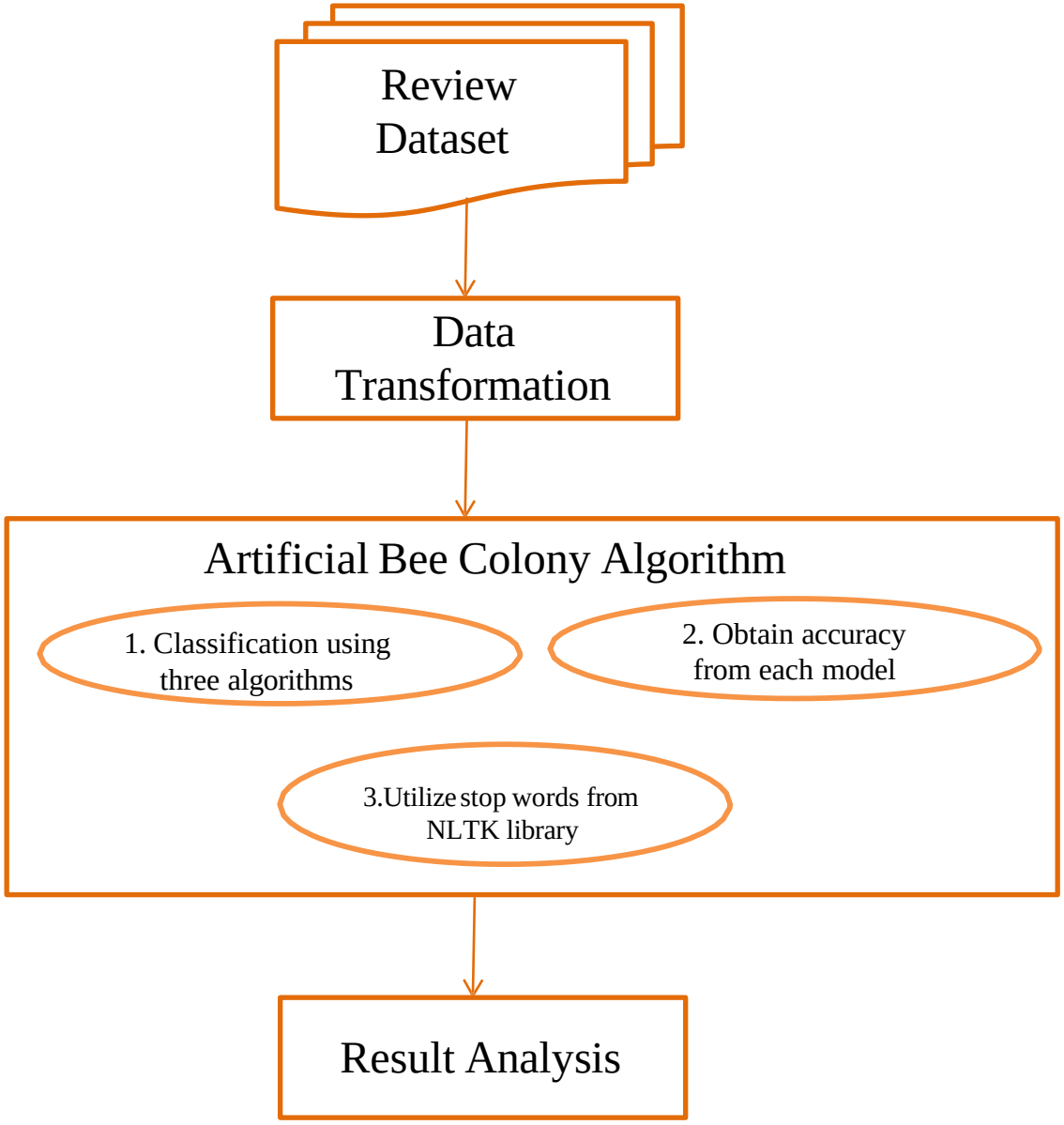
S.No	Title of the paper &year	Author	Inference
5	Sentiment Analysis in Online Product Reviews: Mining Customer Opinions for Sentiment Classification- 2023	Lakshay Bharadwa	Translating analysis results into actionable information
6	Sentiment Analysis of Product Reviews for E- Commerce Recommendation based on Machine Learning-2023	Manal Loukili et.al.,	Lexicon-based methods struggle with nuances, sarcasm, and context-dependent sentiment.

OBJECTIVE

The objective of the project is to

- Enhance sentiment analysis accuracy by focusing on aspect-based feature selection using nature inspired algorithm.
- Streamline sentiment analysis processes by targeting specific aspects within text data and classify the sentiments into positive and negative.

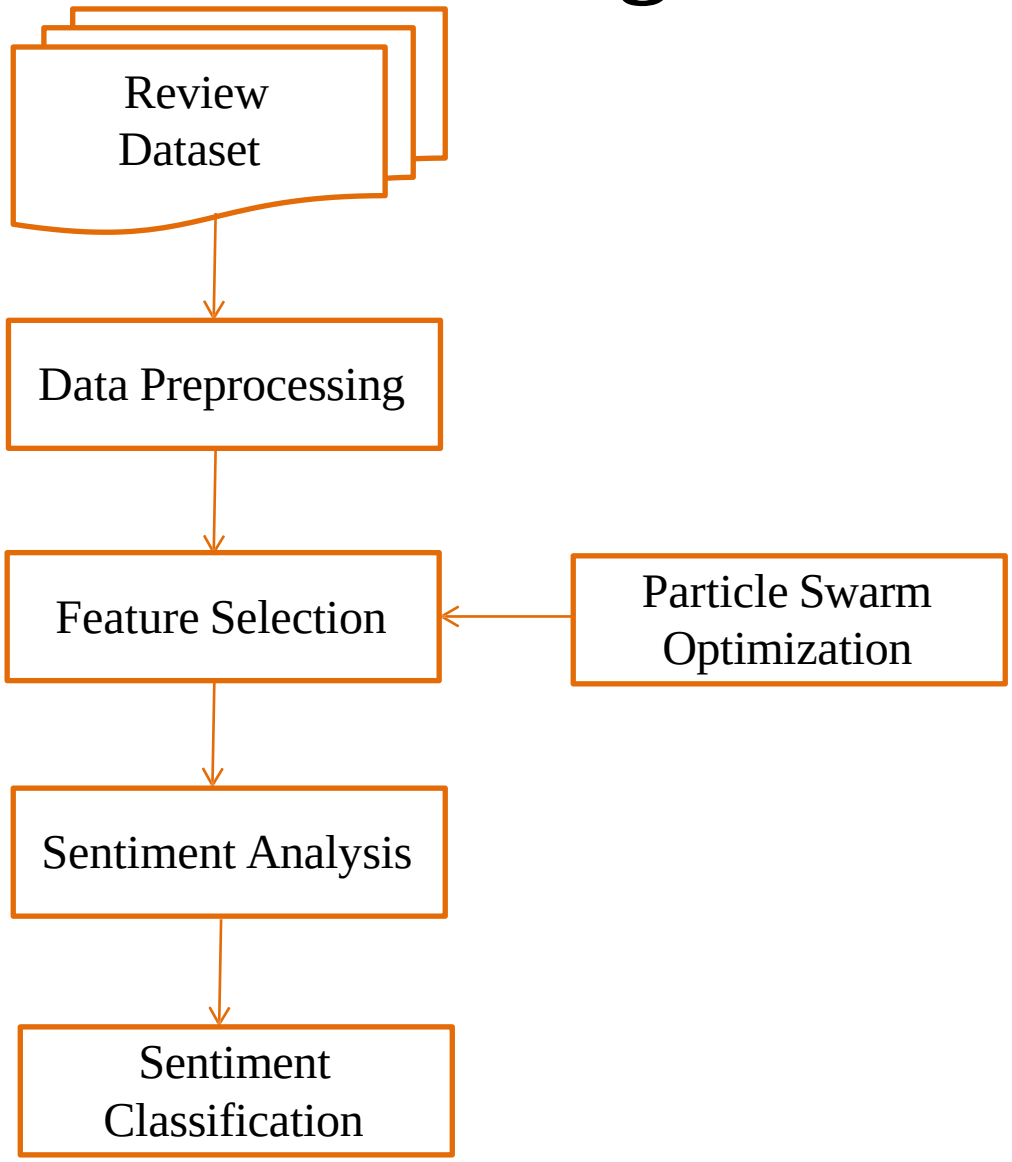
Existing System Block diagram



DISADVANTAGES

- Limited feature selection techniques hinder the identification and prioritization of relevant aspects of sentiment within diverse datasets.
- Lack of robustness in existing methodologies may lead to oversimplified sentiment analysis, failing to capture subtle nuances effectively.
- Prevailing methods may struggle to capture subtle nuances in sentiment expressions, resulting in less precise analysis outcomes.

Proposed System Block diagram



MODULE DESCRIPTION

- **Review dataset**

The dataset includes user reviews for products from the online resource “KAGGLE”. Reviews are collected for various products and categories, saved in CSV format. The dataset contains reviews in multiple languages and noisy data. Pre-processing techniques are applied to clean the data for analysis and improve user experience.

- **Preprocessing**

To get good analytical results using Machine Learning techniques, data is supposed to be very refined and of high quality. The original data had issues like noise, special characters, emojis, and text in different languages. We used techniques like filtering, combining data, making everything lowercase, removing emojis, and making all reviews the same length to get the data ready for analysis.

- **Feature selection**

Feature selection in classification models involves identifying relevant features. For sentiment analysis, reviews are decoded into words and added to a feature vector. filter-based, wrapper-based, or embedded are used. Pragmatic features consider how words are used in context, while emojis, punctuation marks, and slang words convey sentiment.

- **Nature Inspired Algorithm**

Bee Colony Optimization works like bees looking for food. Bees explore options (employed bees), share findings (onlooker bees), and search new areas (scout bees) to find the best solutions. It efficiently solves optimization problems, including feature selection.

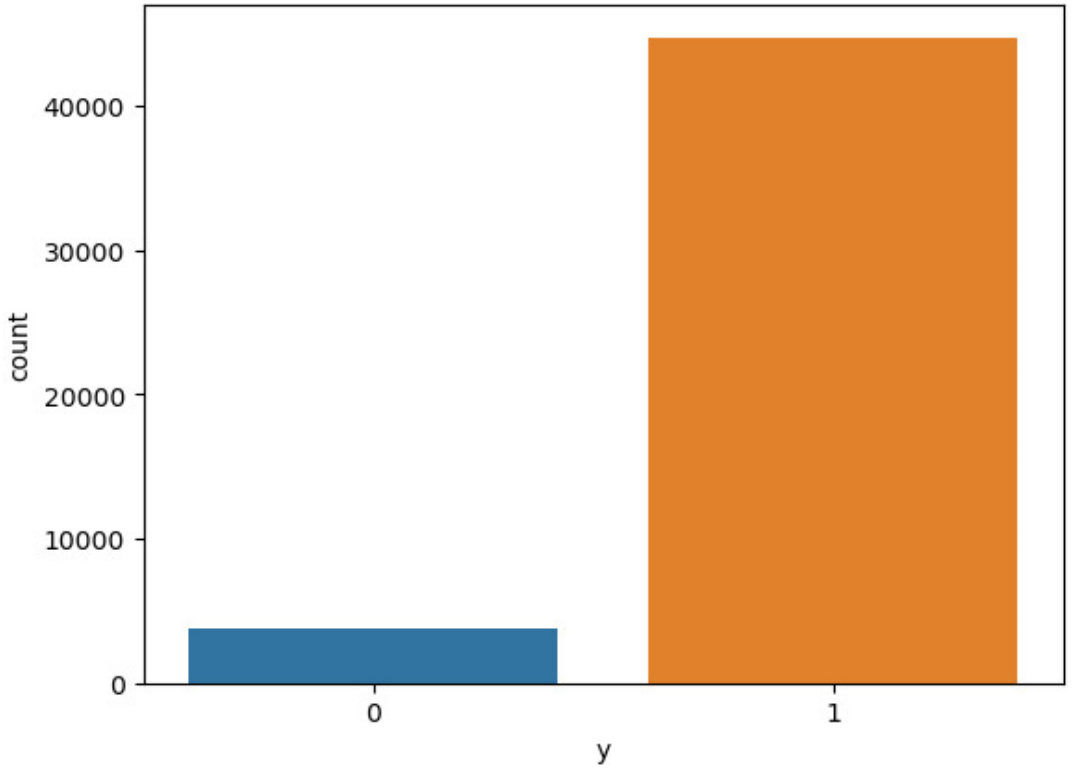
- **Sentiment analysis**

Sentiment analysis involves the process of computationally identifying and categorizing opinions expressed in text to determine whether the attitude is positive, negative, or neutral. It analyzes text data to extract subjective information, such as emotions, opinions, and attitudes, to understand the sentiment behind the text.

Sentiment Classification

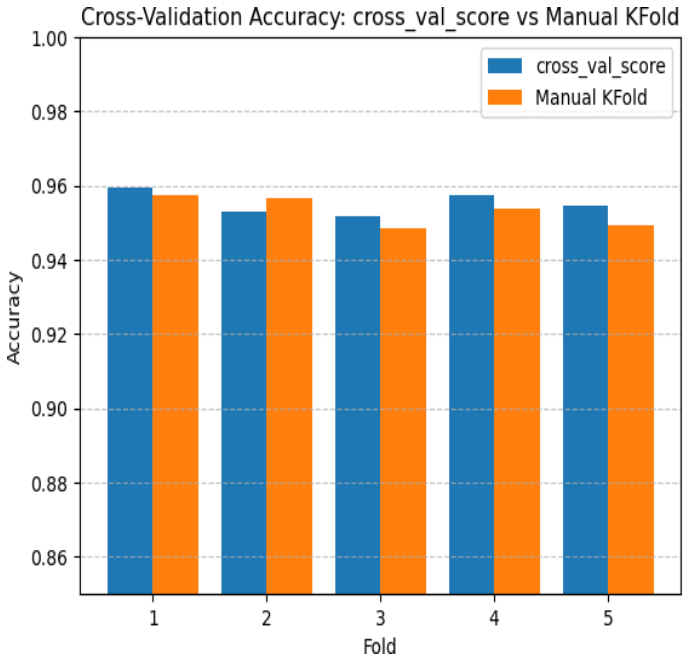
- Sentiment classification is a type of text classification that categorizes text into different sentiment categories, such as positive, negative, or neutral.
- It uses machine learning and natural language processing techniques to analyze text and determine the sentiment expressed in it.
- The goal of sentiment classification is to automatically classify text based on the emotions or opinions conveyed in the text.

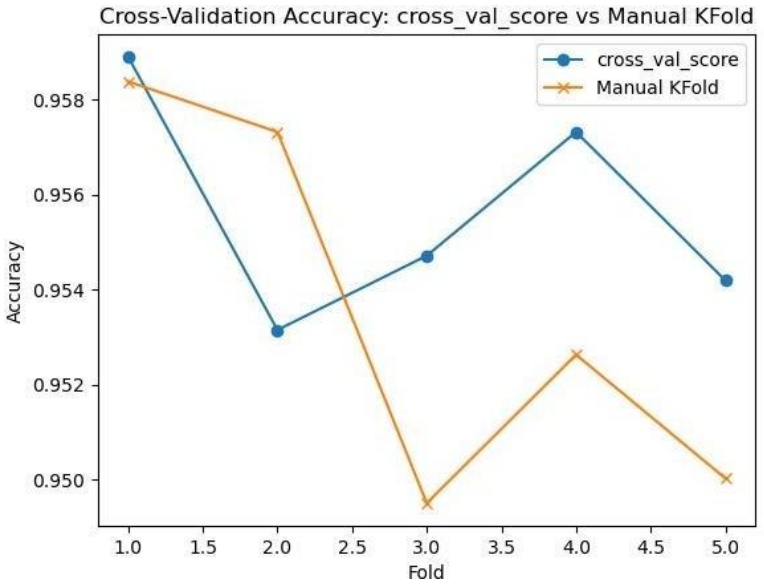
IMPLEMENTATION SCREENSHOT



COUNT OF POSITIVE AND NEGATIVE REVIEWS

IMPLEMENTATION SCREENSHOT

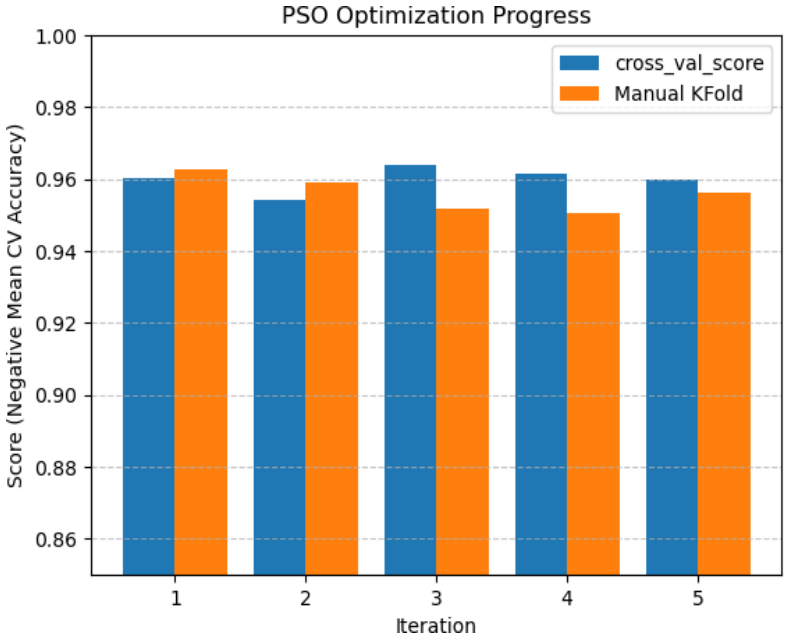


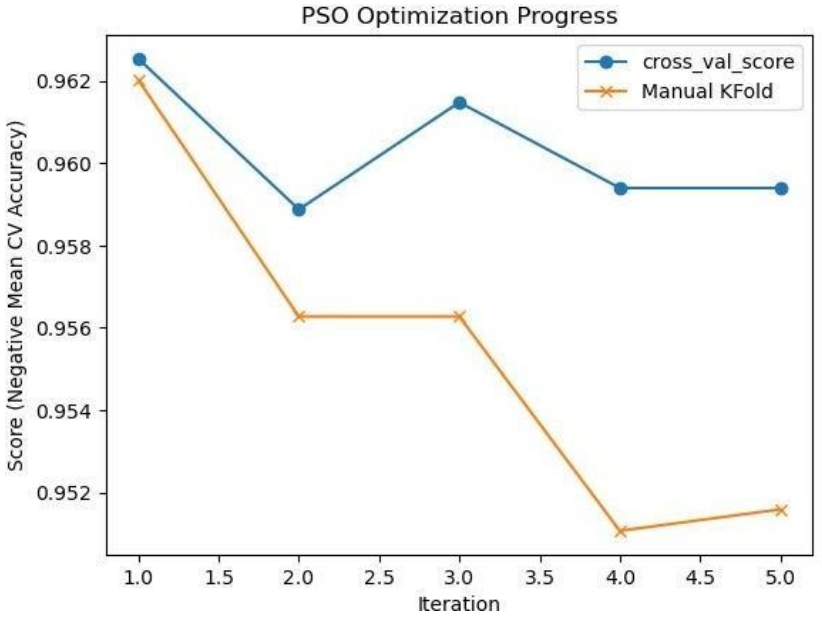


CROSS VALIDATION ACCURACY: CROSS_VAL_SCORE VS MANUAL KFOLD

IMPLEMENTATION SCREENSHOT

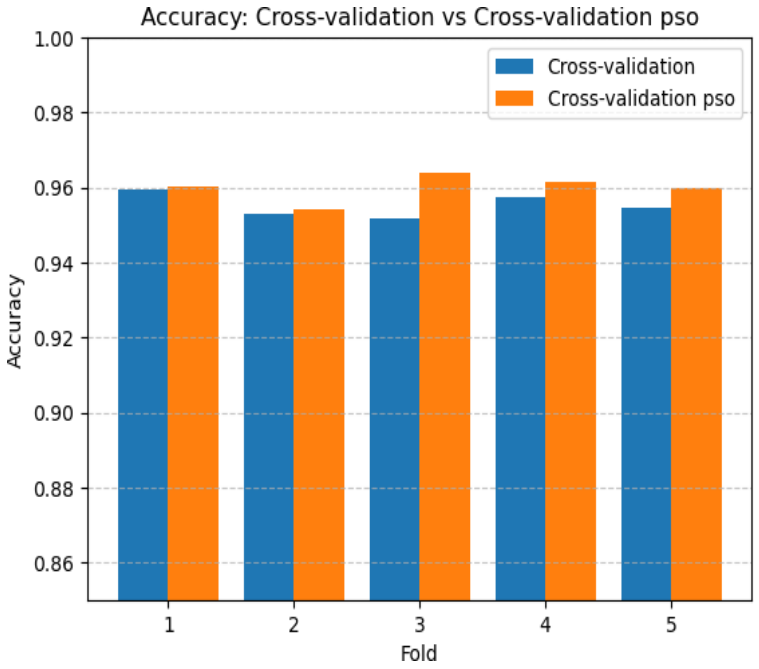
PSO OPTIMIZATION PROGRESS

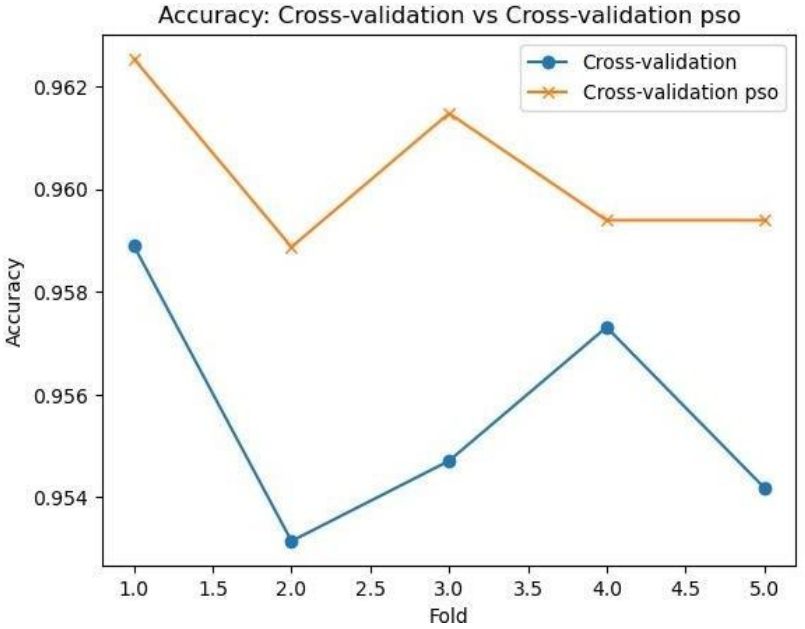




IMPLEMENTATION SCREENSHOT

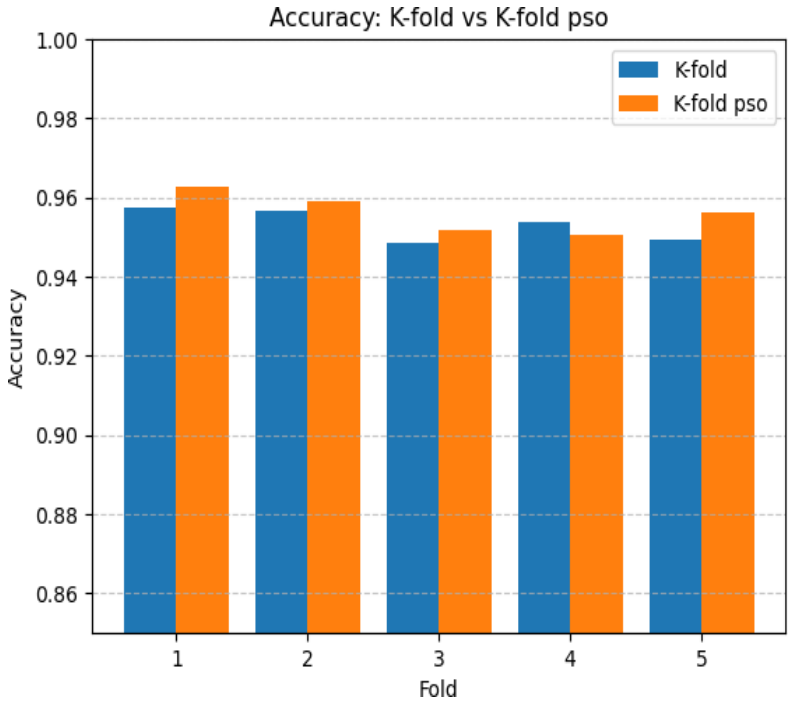
ACCURACY:CROSS-VALIDATION VS CROSS-VALIDATION PSO

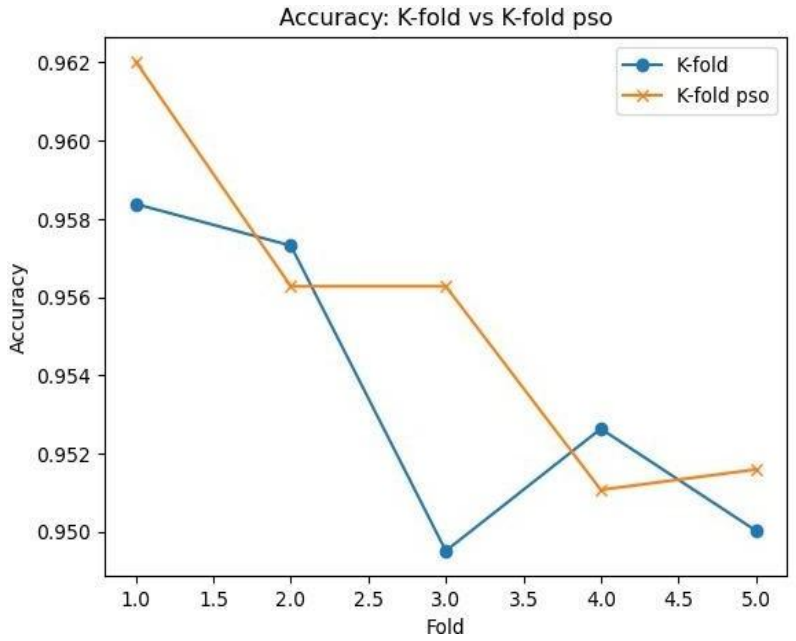




IMPLEMENTATION SCREENSHOT

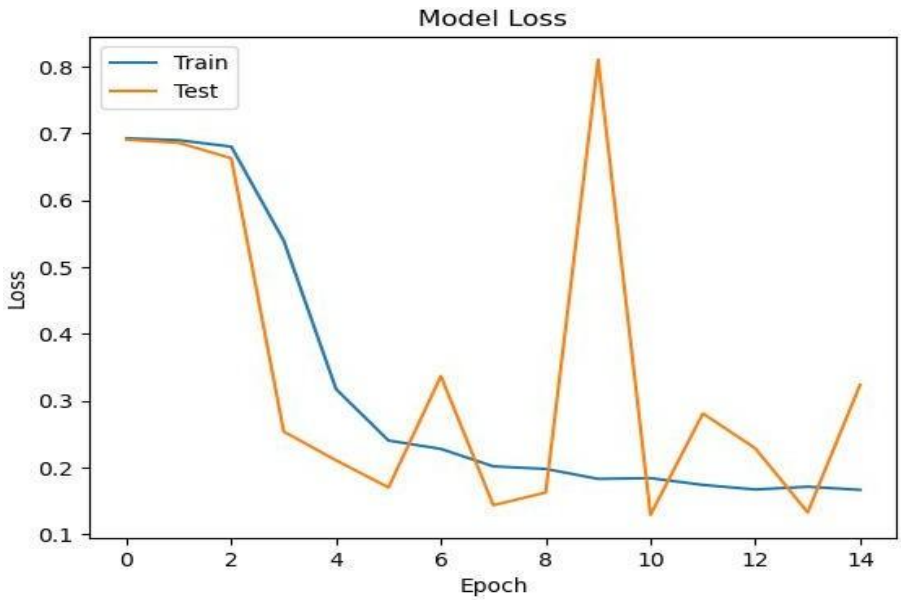
ACCURACY:K-FOLD VS K-FOLD PSO





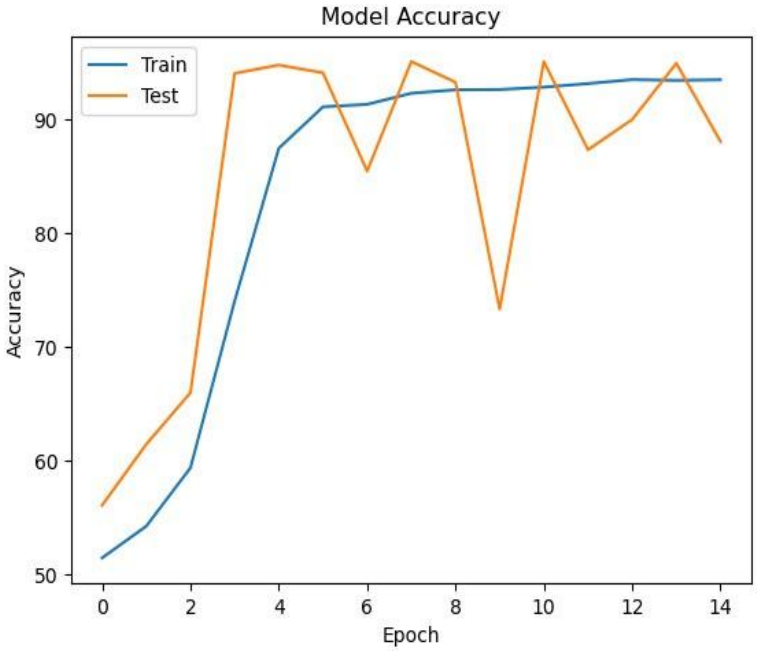
IMPLEMENTATION SCREENSHOT

MODEL LOSS



IMPLEMENTATION SCREENSHOT

MODEL ACCURACY



IMPLEMENTATION SCREENSHOT

[47]:
```
def preprocessing(text):

    text = remove_url(text)
    text = uni.normalize("NFKD", text)
    text = handle_emoji(text)
    text = text.lower()
    text = re.sub(r"[^\w\s]", "", text)
    text = word_tokenizer(text)
    # text = stemming(text)
    text = lemmatization(text)
    text = remove_stopwords(text)
    text = " ".join(text)

    return text
```

[48]:
```
from tqdm import tqdm

tqdm.pandas()

df["clean_review"] = df["review"].progress_map(preprocessing)
```

```
100%|██████████| 9606/9606 [01:03<00:00, 150.92it/s]
```

[49]:
```
df.head()
```

[49]:

	review	y	clean_review
0	great phone in budget. .. pubg performance was...	0	great phone budget pubg performance rough came...
1	Best Smartphone by Mi in this Range. . malfunc...	0	well smartphone mi range malfunction r confuse...
2	Bad smooth phone. . and back camera quality is...	0	bad smooth phone back camera quality bad rear ...
3	Thise is very nice mobile ...I like it very mu...	1	thise nice mobile I like much delivery also fa...
4	I m meagerly dissatisfy 👆 All section r superb...	0	I meagerly dissatisfy thumb section r superb d...

[50]:
```
reviews = df.clean_review.values.tolist()
```

[51]:
```
from tqdm import tqdm

tqdm.pandas()

df["clean_review2"] = df["clean_review"].progress_map(word_tokenizer)
```

```
100%|██████████| 9606/9606 [00:00<00:00, 520192.69it/s]
```

[52]:
```
data_words = df["clean_review2"].values.tolist()
```

```
def handle_emoji(string):
    emojis = demoji.findall(string)

    for emoji in emojis:
        string = string.replace(emoji, " " + emojis[emoji].split(":")[0])

    return string


print(f"Before Handling emoji:- \n {sample}")
print(f"After Handling emoji:- \n {handle_emoji(sample)}")
```

```
Before Handling emoji:-
 Hi Everyone I am Ankit Gupta having the following Kaggle profile
  and I am 😊 to create this notebook
After Handling emoji:-
 Hi Everyone I am Ankit Gupta having the following Kaggle profile
  and I am  smiling face with smiling eyes to create this notebook
```

[17]:
```
def word_tokenizer(text):
    text = text.lower()
    text = text.split()

    return text


sample = "Hi Everyone I am Ankit Gupta."
print(sample)
print(word_tokenizer(sample))
```

```
Hi Everyone I am Ankit Gupta.
['hi', 'everyone', 'i', 'am', 'ankit', 'gupta.']
```

[18]:
```
nltk.download("stopwords")
```

```
[nltk_data] Downloading package stopwords to /home/faiz/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

[18]:
```
True
```

[19]:
```
from nltk.corpus import stopwords

en_stopwords = set(stopwords.words("english"))
print(f"Stop Words in English : \n{ en_stopwords}")
```

```
Stop Words in English :
{'shan', 'we', 'there', 'such', 'y', 'them', 'what', 'most', 'who', 'isn', 'own', 'our', 'off', 'up', 'does', 'any', "mightn't", 'an', 'with'
, "isn't", 'above', 'and', 'this', 'didn', 'weren', 'had', 'has', 'your', "you'd", 'doing', 'few', 'these', 'is', 'you', 'so', 'not', 'yourse
lf', 'me', 'at', 'because', 'during', 'very', 's', "hadn't", 'down', 'should', 'itself', "didn't", "shan't", "shouldn't", 'yours', 'can', 'othe
r', 'm', 'a', 'through', "don't", 'ourselves', 'into', 'his', 'some', "aren't", "doesn't", 'ours', 'll', "you've", "hasn't", 'did', 'about', 'b
e', 'by', 'no', 've', 'been', 'between', 'i', 'hadn', 'am', 'o', 'while', 'it', 'will', 'hasn', "doesn't", 'won', 'have', "you're", 'on', 'ar
```

IMPLEMENTATION SCREENSHOT

```
[58]: fasttext_model.wv.n_similarity("I really like the camera of this phone", "battery")

[58]: 0.94157267

[59]: aspects = ["phone", "camera", "battery", "delivery", "processor"]

def get_similarity(text, aspect):
    try:
        text = " ".join(text)
        return fasttext_model.wv.n_similarity(text, aspect)
    except:
        return 0

[60]: from tqdm import tqdm
tqdm.pandas()
for aspect in aspects:
    df[aspect] = df['clean_review2'].progress_map(lambda text: get_similarity(text, aspect))

100%|██████████| 9606/9606 [00:09<00:00, 1026.78it/s]
100%|██████████| 9606/9606 [00:06<00:00, 1490.01it/s]
100%|██████████| 9606/9606 [00:06<00:00, 1481.46it/s]
100%|██████████| 9606/9606 [00:06<00:00, 1466.14it/s]
100%|██████████| 9606/9606 [00:06<00:00, 1438.70it/s]

[61]: df.head()
```

[61]:

	review	y	clean_review	clean_review2	phone	camera	battery	delivery	processor
0	great phone in budget. .. pubg performance was...	0	great phone budget pubg performance rough came...	[great, phone, budget, pubg, performance, roug...	0.900497	0.877507	0.950030	0.908349	0.903495
1	Best Smartphone by Mi in this Range. . malfunc...	0	well smartphone mi range malfunction r confuse...	[well, smartphone, mi, range, malfunction, r, ...	0.907105	0.840466	0.932822	0.941408	0.864272
2	Bad smooth phone. . and back camera quality is...	0	bad smooth phone back camera quality bad rear ...	[bad, smooth, phone, back, camera, quality, ba...	0.872275	0.923698	0.947014	0.911570	0.931754
3	Thise is very nice mobile ...I like it very mu...	1	thise nice mobile I like much delivery also fa...	[thise, nice, mobile, i, like, much, delivery,...	0.875332	0.844254	0.904018	0.957452	0.844768
4	I m meagerly dissatisfy 👍 All section r superb...	0	I meagerly dissatisfy thumb section r superb d...	[i, meagerly, dissatisfy, thumb, section, r, s...	0.859906	0.860499	0.957805	0.961480	0.889979

```
[62]: spath="dataset/"
df.to_csv(spath+"Clean_Flipkart_Product.csv", index = False)

[63]: import torch
from torch import nn
from torch.utils.data import Dataset
from torch.utils.data import DataLoader
```

IMPLEMENTATION SCREENSHOT

```
    text = preprocessing(text)
    text = numericalize(text)
    text = padding(text)
    return text

def get_similarity(text, aspect):
    try:
        #text = " ".join(text)
        return fasttext_model.wv.n_similarity(text, aspect)
    except:
        return 0

def best_aspect(text, aspects):
    a = []

    for aspect in aspects:
        a.append(get_similarity(text, aspect))
    print(a,np.argmax(a))

    return aspects[np.argmax(a)]
```

[81]:
```
aspects = ["phone", "camera", "battery", "neutral", "processor"]
```

[82]:
```
sample = "I just love the phone , camera , features, bought for my mother and she absolutely love it thanks Flipkart."

ba = best_aspect(preprocessing(sample), aspects)
print(ba)

a = infer_processing(sample).to(config.DEVICE)
```

```
[0.8812594, 0.87336266, 0.96708167, 0.9524784, 0.90428954] 2
battery
```

[83]:
```
model.eval()
sentiment = model(a)
sentiment = sentiment.cpu().detach().numpy()[0]
print(sentiment)

if sentiment > 0.5:
    sentiment = 'Positively'
else :
    sentiment = 'Negatively'
```

```
[0.9558885]
```

[84]:
```
print(f"The reviewer is talking {sentiment} about the {ba} of the phone in his/her comment")
```

```
The reviewer is talking Positively about the battery of the phone in his/her comment
```

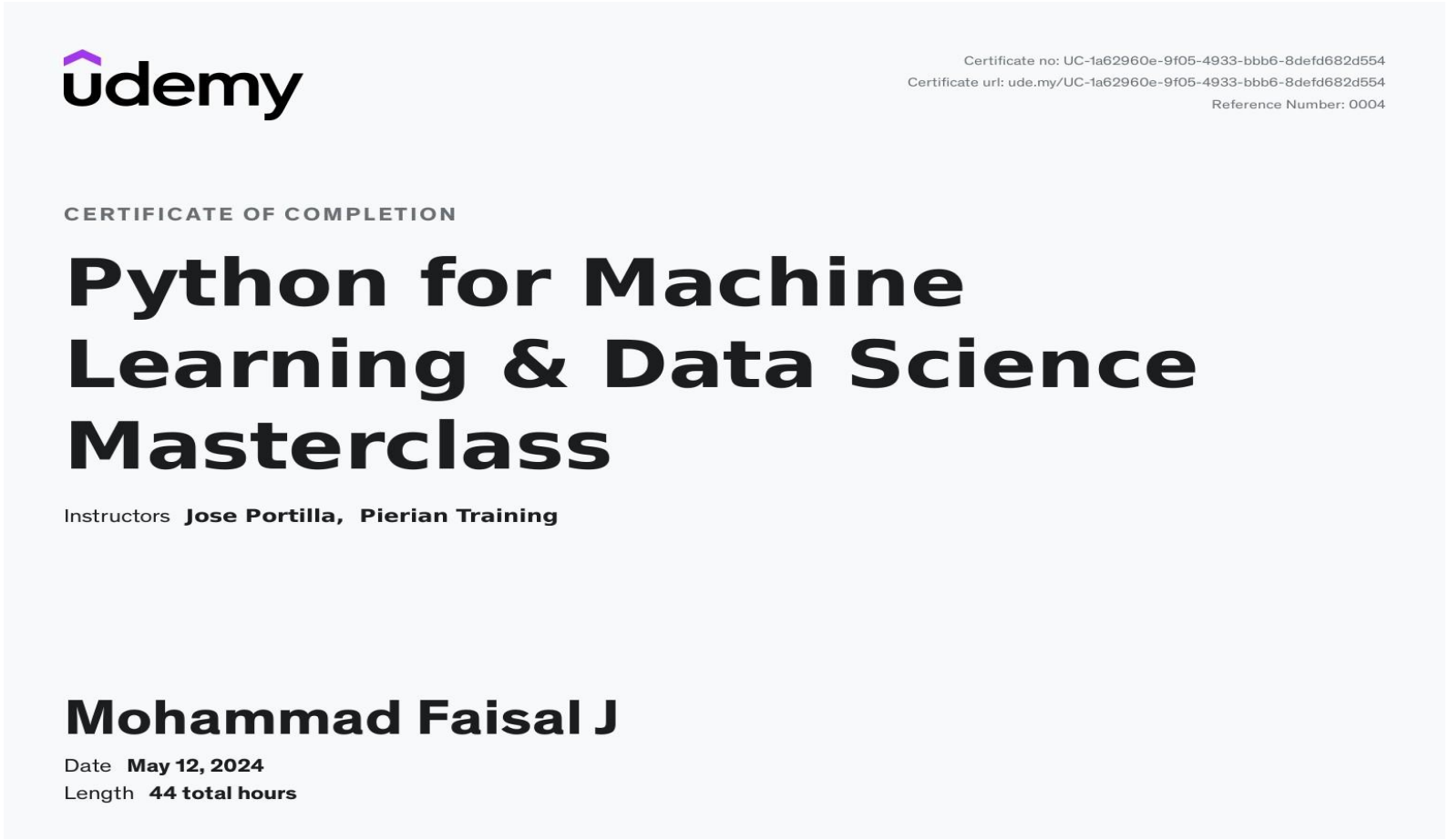
CONCLUSION

The project focuses on enhancing sentiment analysis accuracy by implementing aspect-based feature selection using nature-inspired algorithms. By targeting specific aspects within text data, the aim is to provide deeper insights and improve the rudeness and accuracy of sentiment analysis results. The project also aims to streamline sentiment analysis processes by classifying sentiments into positive and negative categories based on the specific aspects identified within the text data.

RESULT

The culmination of the sentiment analysis projects how cases the successful implementation of aspect-based feature selection using nature-inspired algorithms to enhance sentiment analysis accuracy. The system effectively categorizes sentiments into positive and negative based on specific aspects within the text data, demonstrating improved precision and recall metrics. Through a comprehensive evaluation process, the project highlights the effectiveness of the proposed system in providing deeper insights more refined sentiment analysis results. The streamlined sentiment analysis processes, coupled with the focus on aspect-based feature selection, have proven to be instrumental in achieving superior sentiment classification outcomes, setting a new standard for sentiment analysis accuracy and efficiency.

COURSE CERTIFICATES



udemy

Certificate no: UC-1a62960e-9f05-4933-bbb6-8defd682d554
Certificate url: ude.my/UC-1a62960e-9f05-4933-bbb6-8defd682d554
Reference Number: 0004

CERTIFICATE OF COMPLETION

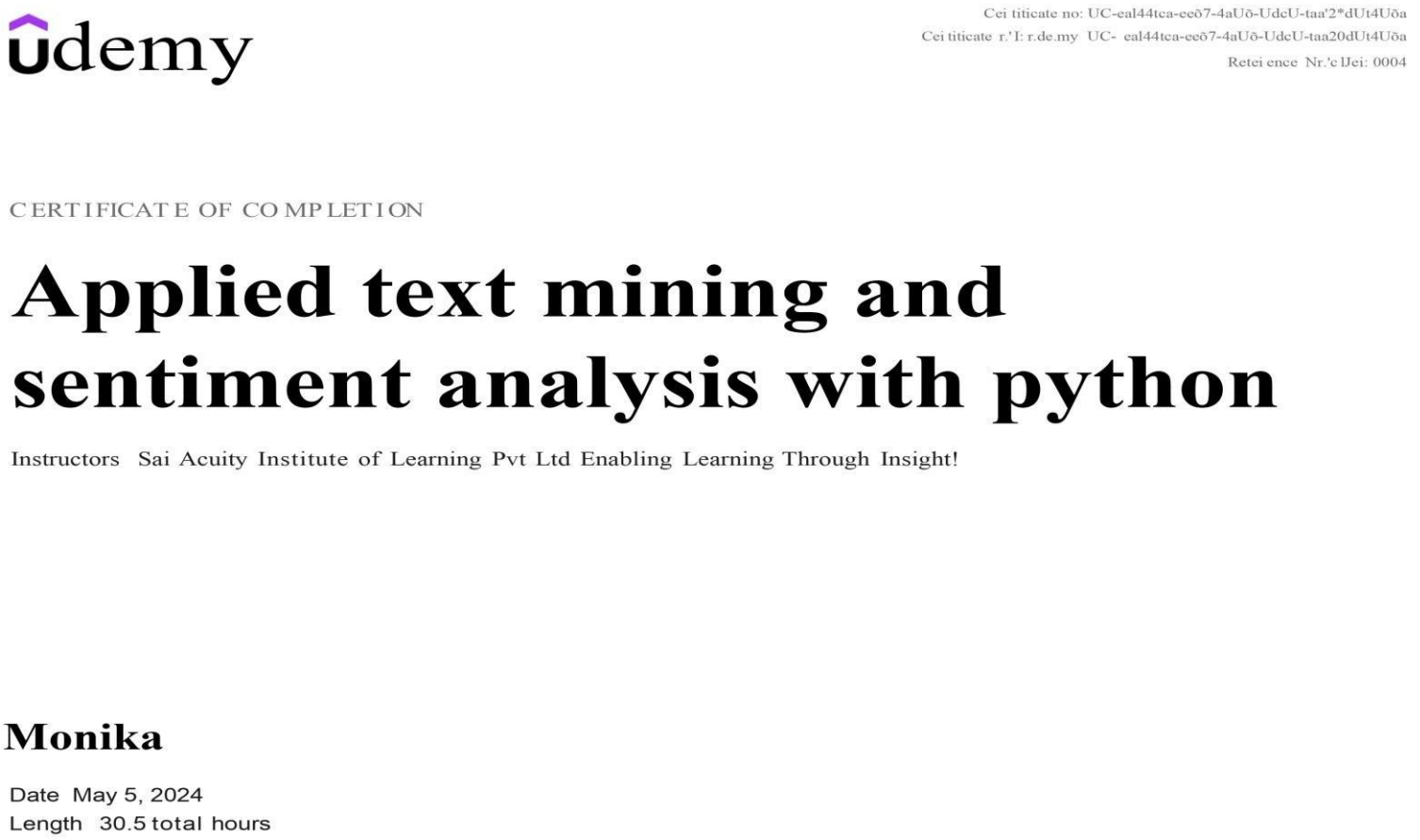
Python for Machine Learning & Data Science Masterclass

Instructors **Jose Portilla, Pierian Training**

Mohammad Faisal J

Date **May 12, 2024**
Length **44 total hours**

COURSE CERTIFICATES(CONTD…)



udemy

Cei titicate no: UC-eal44tca-eeõ7-4aUõ-UdcU-taa'2*dUt4Uõa
Cei titicate r.'I: r.de.my UC- eal44tca-eeõ7-4aUõ-UdcU-taa20dUt4Uõa
Retei ence Nr.'c lJei: 0004

CERTIFICATE OF COMPLETION

Applied text mining and sentiment analysis with python

Instructors Sai Acuity Institute of Learning Pvt Ltd Enabling Learning Through Insight!

Monika

Date May 5, 2024
Length 30.5 total hours

COURSE CERTIFICATES(CONTD…)

REFERENCES

- Miriam Amendola , Danilo Cavaliere , Carmen De Maio , Giuseppe Fenza, Vincenzo Loia “Towards echo chamber assessment by employing aspect-based sentiment analysis and GDM consensus metrics” , Elsevier , pp.1-11 (2024)
- Muhammad Aasim Qureshi,Muhammad Asif,“Sentiment Analysis of Reviews in Natural Language: Roman Urdu as a Case Study” , *IEEE Access* ,pp. 24945-24954(2022).
- Lakshay Bharadwa,“Sentiment Analysis in Online Product Reviews: Mining Customer Opinions for Sentiment Classification”, *ReasearchGate* , pp.1-30(2023).
- Manal Loukili, Fayçal Messaoudi, and Mohammed El Ghazi, “Sentiment Analysis of Product Reviews for E- Commerce Recommendation based on Machine Learning” , pp1-14(2023).
- Mayur Wankhade, Annavarapu Chandra Sekhara Rao, Chaitanya Kulkarni1, “A survey on sentiment analysis methods, applications, and challenges” , *Springer* , pp.5731-5780(2022)

REFERENCES(CONTD.,)

- Ziedhan Alifio Dieksona , Muhammad Rivyan Bagas Prakosoa , Muhammad Savio Qalby Putraa ,"Sentimental Analysis for Customer Reviews" , *ScienceDirect* , pp.1-9(2023)
- Men Li, Yucheng Shi "Sentimental analysis and prediction model based on Chinese government affairs microblogs" , *Elsevier* , pp.1-16 (2023)
- Yan Zhou, Xiaodong Li ,"Sentimental Contrastive Learning for event representation" , *Elsevier* , pp.1-11 (2023)
- Mark Swillus , Andy Zaidman "Sentiment overflow in the testing stack: Analyzing software testing posts on Stack Overflow" , *Elsevier* , pp.1-21 (2023)
- Jamin Rahman Jim, Md Apon Riaz Talukder, Partha Malakar, Md Mohsin Kabir, Kamruddin Nur, M.F. Mridha "Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review" , *Elsevier*, pp.1-21 (2023)

REFERENCE(CONTD.,)

- Jemal Abate ,Faizur Rashid “A review of sentiment analysis for Afaan Oromo: Current trends and future perspectives” , *Elsevier* , pp.1-9 (2024)
- Omar Alqaryouti and Nur Siyam, Azza Abdel Monem, Khaled Shaalan “Aspect-based sentiment analysis using smart government review data”, *Emerald Insights*, pp 142-161. (2019)
- Margarita Rodríguez-Ibanez, Antonio Casanez-Ventura “A review on sentiment analysis from social media platform” , *Elsevier* , pp.1-14(2023).
- Houssem Lahiani, Mondher Frikha “A Systematic Review of Social Media Data Mining on Android” , *Elsevier* , pp.2018-2027 (2023)
- Wei Liu, Shenchao Cao, Sun Zhang “Multimodal consistency-specificity fusion based on information bottleneck for sentiment analysis” , *Elsevier* , pp.1-10 (2024)

THANK YOU