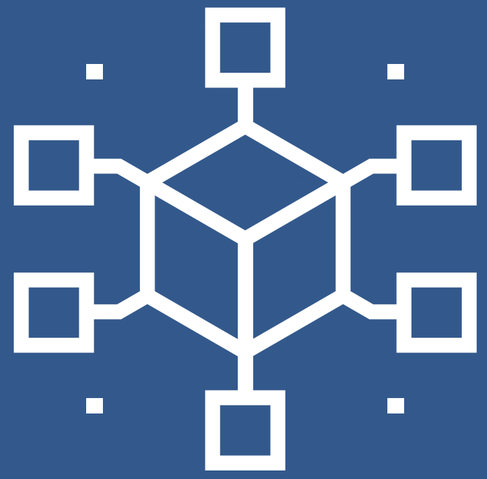
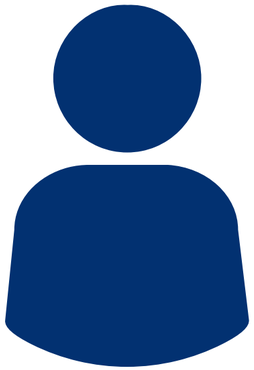
INTERNATIONAL UNIVERSITY - HCMC
School of Computer Science & Engineering



Developing a Personalized Product Recommendation System with PySpark

Course: Big Data Technology
Advisor: Dr. Ho Long Van

Member



Lê Huỳnh Nhã Nguyên

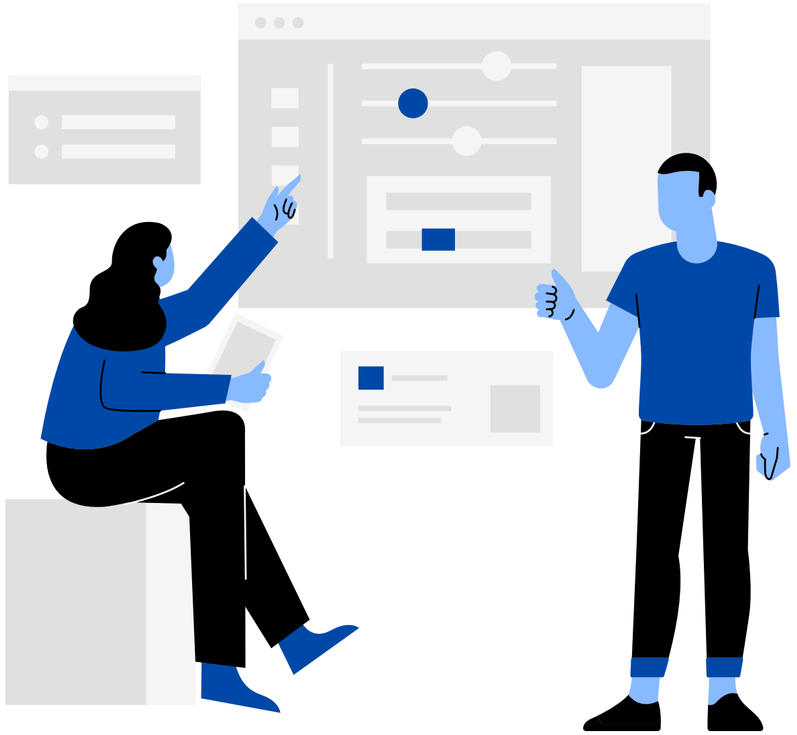
ITDSIU21058

Phan Bảo Trân

ITDSIU21125

Table Of Content

INTRODUCTION



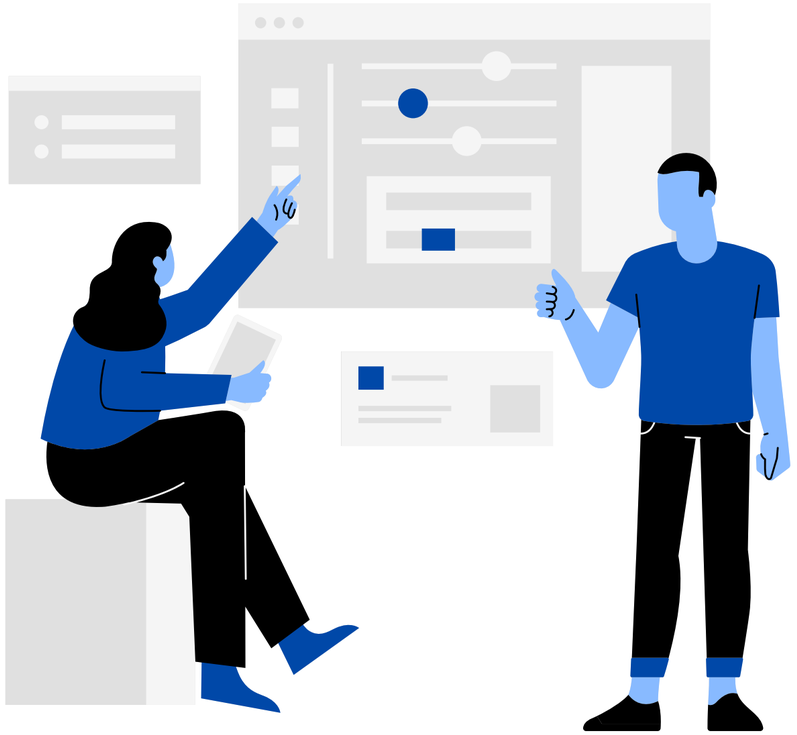
More Details

The reason for choosing the topic

Objectives

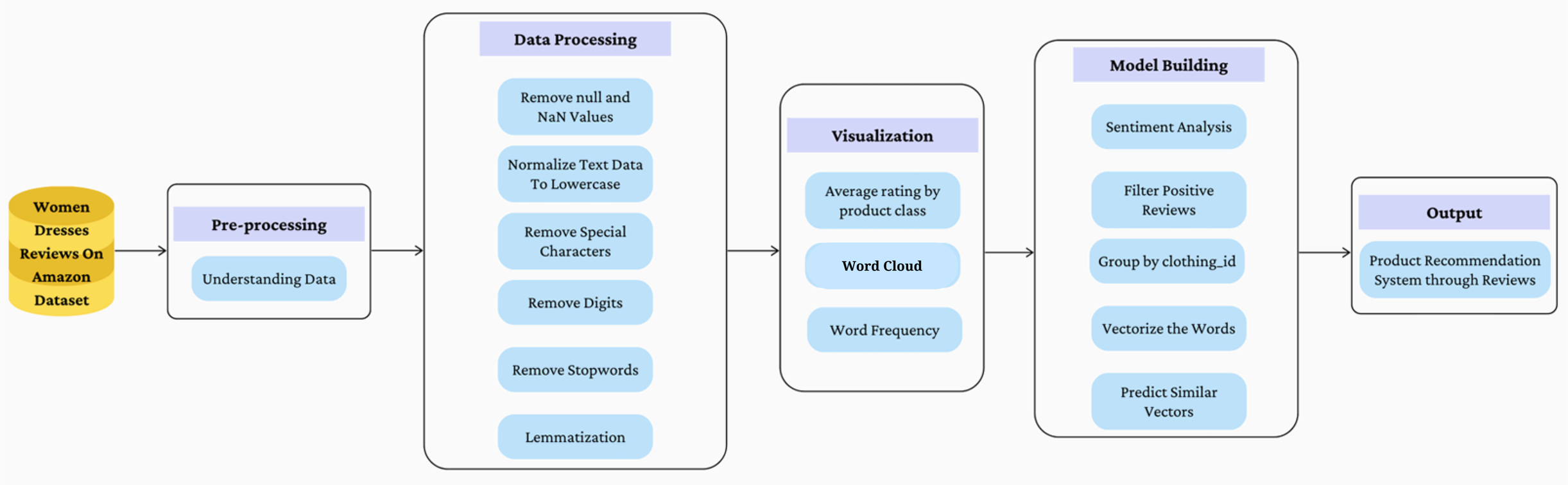
- Analyze Review Data
- Optimize Data Processing
- Build a Recommendation Model
- Enhance Customer Experience

LITERATURE REVIEW



More Details

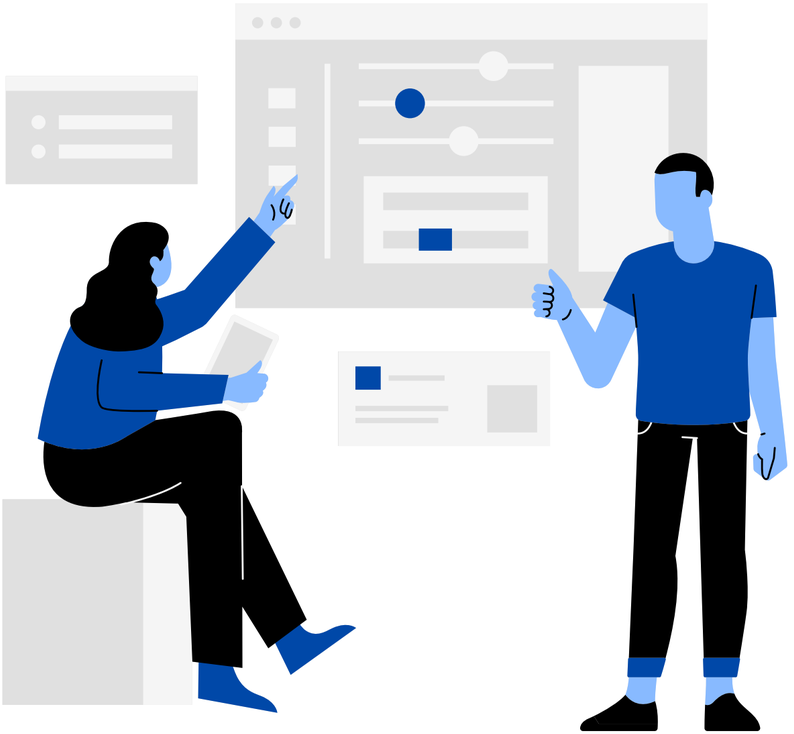
Proposed model



Theoretical Basis

- Sentiment Analysis
- NLTK and Vader
- Word2vec
- Cosine Similarity

DATA COLLECTION & PROCESSING



More Details

Dataset

s_no	age	division_name	department_name	class_name	clothing_id	title	review_text	alike_feedback_count	rating	recommend_index
0	40	General	Bottoms	Jeans	1028	Amazing fit and wash	Like other reviewers i was hesitant to spend this much on	0	5	1
1	62	General Petite	Tops	Blouses	850	Lovely and unique!	As is true of a bunch of the fall clothing photos, the colors	12	5	1

Data Processing

Remove NULL values

Normalize Text to Lowercase

Remove Special Characters

Remove Stopwords

Lemmatization

Data Processing

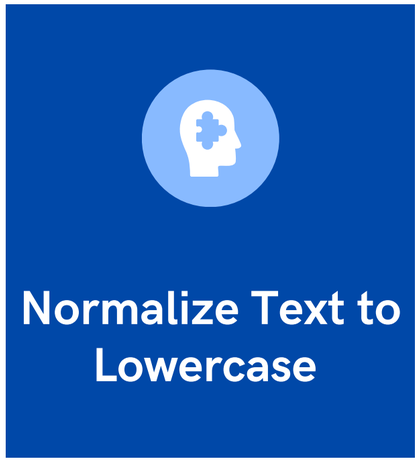
Remove NULL values

```
[ ]  from pyspark.sql.functions import col, isnan
     print("Number of NULL 'age':", df.filter(col("age").isNull() | isnan(col("age"))).count())
     print("Number of NULL 'division_name':", df.filter(col("division_name").isNull() | isnan(col("division_name"))).count())
     print("Number of NULL 'department_name':", df.filter(col("department_name").isNull() | isnan(col("department_name"))).count())
     print("Number of NULL 'class_name':", df.filter(col("class_name").isNull() | isnan(col("class_name"))).count())
     print("Number of NULL 'review_text':", df.filter(col("review_text").isNull() | isnan(col("review_text"))).count())
     print("Number of NULL 'title':", df.filter(col("title").isNull() | isnan(col("title"))).count())
```

```
Number of NULL 'age': 0
Number of NULL 'division_name': 14
Number of NULL 'department_name': 14
Number of NULL 'class_name': 14
Number of NULL 'review_text': 845
```

```
df_cleaned = df.dropna()
```

Data Processing




```
[ ]  from pyspark.sql import Row

     # Convert DataFrame to RDD
     rdd = df_cleaned.rdd

     def process_row(row):
         try:
             return Row(
                 age=row.age,
                 division_name=row.division_name,
                 department_name=row.department_name,
                 class_name=row.class_name,
                 clothing_id=row.clothing_id,
                 title=row.title.lower() if row.title else None,
                 review_text=row.review_text.lower() if row.review_text else None,
                 alike_feedback_count=row.alike_feedback_count,
                 rating=row.rating,
                 recommend_index=row['recommend_index']
             )
         except AttributeError as e:
             print(f"Lỗi xử lý dòng: {row}, lỗi: {e}")
             return None

     rdd = rdd.map(process_row).filter(lambda x: x is not None)
     result = rdd.take(5)

     print("Data after conversion to lowercase in title and review_text:")
     for row in result:
         print(row)
```

```
title='amazing fit and wash', review_text='like other reviewers i was hesitant to spend this much on a pair of jeans.
```

Data Processing


```
import re
def replace_special_characters(text):
    return re.sub(r'[^\w\s]', '', text)

def remove_digits(text):
    return re.sub(r'\d', '', text) if text else None
```

“love the design!”

love the design

Data Processing

Remove Stopwords

```
[ ]  import nltk
     nltk.download('stopwords')
```

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

```
spark = SparkSession.builder.appName("Remove Stopwords").getOrCreate()

sw = set(stopwords.words('english'))

def remove_stopwords(text):
    words = text.split()
    filtered_words = [word for word in words if word.lower() not in sw]
    return ' '.join(filtered_words)
```

Data Processing


```
[ ]  !pip install nltk
     import nltk
     nltk.download('wordnet')
     nltk.download('omw-1.4')
```

```
     from nltk.stem import WordNetLemmatizer
     spark = SparkSession.builder.appName("Lemmatization of Review Text").getOrCreate()

     lemmatizer = WordNetLemmatizer()

     def lemmatize_review(review):
         return " ".join([lemmatizer.lemmatize(word) for word in review.split()])
```

go

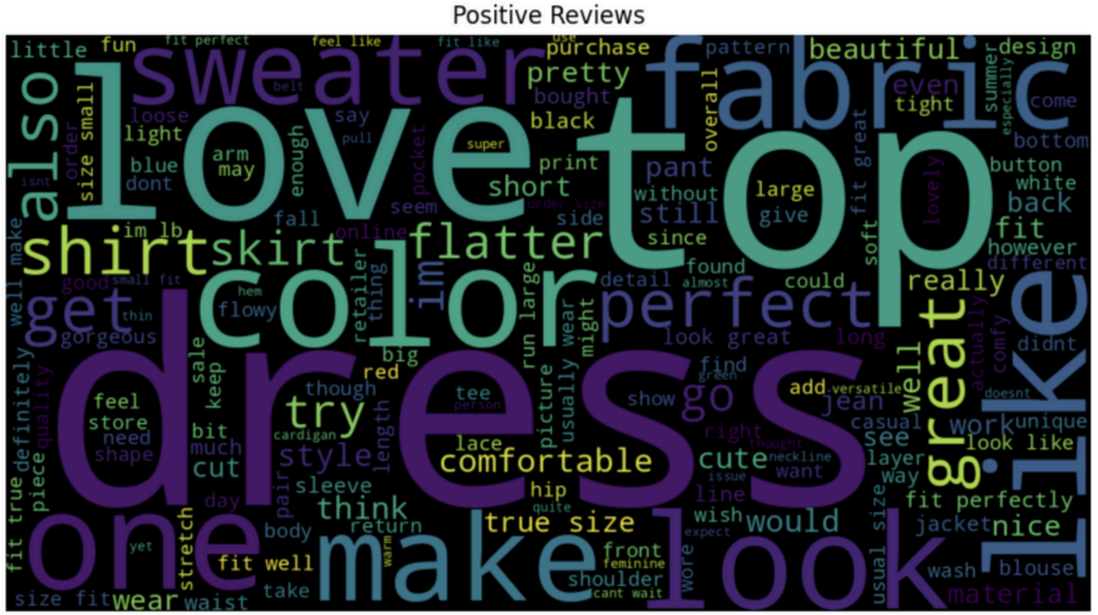
went

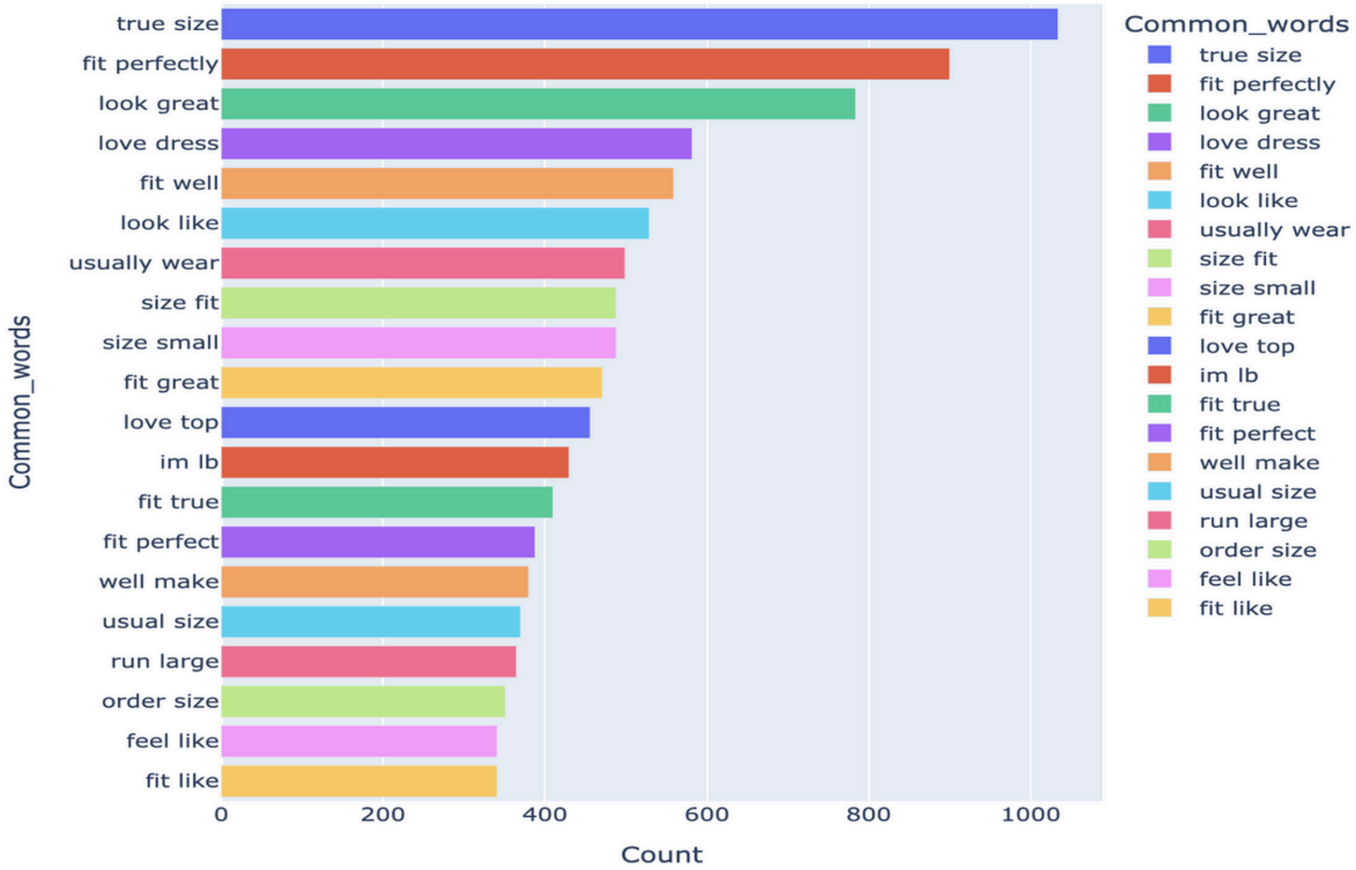
going

go

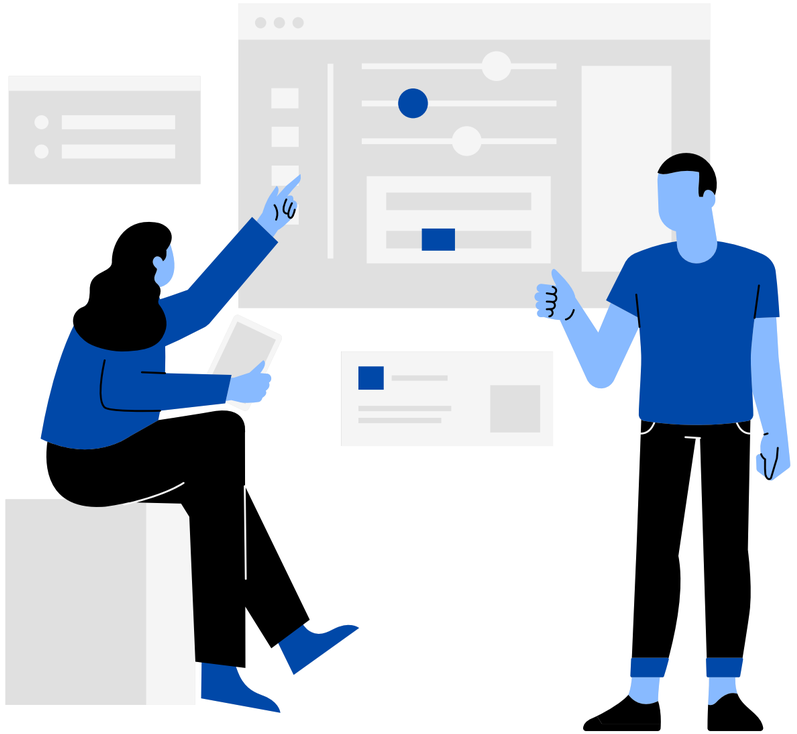
title	review_text
amazing fit wash	like reviewer hesitant spend much pair jean however purchased retailer day andhonest
lovely unique	true bunch fall clothing photo color totally washed model image shame embroidery bri
meh	wanted skirt work love design way way long lb small inch floor step skirt walk
wow	love love hesitant buy first review made seem big wasnt sure kind outfit wanted try
great bigger bust	absolutely love retro look swimsuit first saw blogger amber fillerupclark barefoot b
love pattern color	love sweater im fence keeping havent figured way layer wear really appear best plain
beautiful unique	love sweater soft cozy ruffle overwhelmingjust touch pretty look pretty turquoise ne
unique wonderful	sweater comfortable good weight zipper difficult unzip sweater pull
great look one	love everything sweater well made beautiful material effortless way pull together ru
beauty meet comfort	love top detail neck shoulder lovely front back also comfortable soft allowing look
great fit	searching around new unique clothes wear work top cute comfortable well made quality
adore white sheer	dark blue adore read review later pondering buying white hard find white shirt anymo
red orange	yes much id liked true red kind redorange relieved however orange look monitor sligh
cute flattering	really loved skirt model wasnt flattering received order ordered two size athletic b
curve	c jean size dress fell love pattern particular knowing design might work best body o
itchy	love idea coat foolishly purchased item happy anticipation arrival description say l
great legging	dress short wear dress however great legging boot drape nicely sized like many dress
perfect white pant	love pant purchased white fit great fabric right thickness given white dont want fab
wowser	made dawn suit bomb bottom tiny bit cheeky thong territory enough side leg seriously
pretty bit small	cant decide pant trying store soft beautiful blue color however found bit small usua

Data Visualization





EXPERIMENTATION AND MODEL EVALUATION



More Details

Sentiment analysis process

- Initialize the VADER Model
- Create UDF (User Defined Functions)
- Apply Sentiment Analysis

Sentiment analysis process

age	division_name	department_name	class_name	clothing_id	title	review_text	alike_feedback_count	rating	recommend_index
40	General	Bottoms	Jeans	1028	amazing fit and wash	like reviewer hes...	0	5	1
62	General Petite	Tops	Blouses	850	lovely and unique	true bunch fall c...	12	5	1
47	General Petite	Bottoms	Skirts	993	meh	wanted skirt work...	3	1	0
45	General Petite	Bottoms	Pants	1068	wow	love love hesitan...	0	5	1
37	Initmates	Intimate	Swim	24	great for bigger ...	absolutely love r...	0	5	1
43	General	Tops	Sweaters	933	love the pattern ...	love sweater im f...	0	4	1
83	General	Tops	Sweaters	937	beautiful and unique	love sweater soft...	4	5	1
34	General	Tops	Knits	868	unique and wonderful	sweater comfortab...	2	5	1
49	General Petite	Tops	Fine gauge	900	great look all in...	love everything s...	4	5	1
49	General	Tops	Knits	873	beauty meets comfort	love top detail n...	0	5	1
32	General	Tops	Knits	872	great fit	searching around ...	0	5	1
41	General Petite	Tops	Knits	873	adore but the whi...	dark blue adore r...	0	4	1
53	General	Bottoms	Skirts	1008	not red not orange	yes much id liked...	0	5	1
23	General	Bottoms	Skirts	1020	cute but not flat...	really loved skir...	0	2	0
31	General	Dresses	Dresses	1086	not for my curves	c jean size dress...	5	3	0

- Remove rows where recommend_index = 0 or rating = 1 or 2

```
df = df_loaded.filter((df_loaded["recommend_index"] != 0) & (df_loaded["rating"] != 1) & (df_loaded["rating"] != 2))
df.show()
```

age	division_name	department_name	class_name	clothing_id	title	review_text	alike_feedback_count	rating	recommend_index
40	General	Bottoms	Jeans	1028	amazing fit and wash	like reviewer hes...	0	5	1
62	General Petite	Tops	Blouses	850	lovely and unique	true bunch fall c...	12	5	1
45	General Petite	Bottoms	Pants	1068	wow	love love hesitan...	0	5	1
37	Initmates	Intimate	Swim	24	great for bigger ...	absolutely love r...	0	5	1
43	General	Tops	Sweaters	933	love the pattern ...	love sweater im f...	0	4	1
83	General	Tops	Sweaters	937	beautiful and unique	love sweater soft...	4	5	1
34	General	Tops	Knits	868	unique and wonderful	sweater comfortab...	2	5	1
49	General Petite	Tops	Fine gauge	900	great look all in...	love everything s...	4	5	1
49	General	Tops	Knits	873	beauty meets comfort	love top detail n...	0	5	1
32	General	Tops	Knits	872	great fit	searching around ...	0	5	1
41	General Petite	Tops	Knits	873	adore but the whi...	dark blue adore r...	0	4	1
53	General	Bottoms	Skirts	1008	not red not orange	yes much id liked...	0	5	1
50	General Petite	Dresses	Dresses	1078	great with leggings	dress short wear ...	13	4	1
50	General Petite	Bottoms	Pants	1060	perfect white pants	love pant purchas...	0	5	1
40	Initmates	Intimate	Swim	286	wowser	made dawn suit bo...	1	5	1
38	General	Bottoms	Pants	1053	pretty but a bit ...	cant decide pant ...	12	4	1

Initialize the VADER Model

- To calculate the sentiment polarity scores for the reviews

```
!pip install nltk
import nltk

nltk.download('vader_lexicon')
```

```
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.9.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.4.2)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2024.9.11)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.6)
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
[nltk_data]   Package vader_lexicon is already up-to-date!
True
```

```
from pyspark.sql.functions import udf, col
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from pyspark.sql.types import StringType, FloatType
```

• Create UDF

```
def get_polarity_score(review):
    return float(sia.polarity_scores(review)['compound'])

def get_sentiment_label(review):
    score = sia.polarity_scores(review)['compound']
    if score > 0.05:
        return "positive"
    elif score < -0.05:
        return "negative"
    else:
        return "neutral"

polarity_score_udf = udf(get_polarity_score, FloatType())
sentiment_label_udf = udf(get_sentiment_label, StringType())
```

- get_polarity_score(), calculates the overall sentiment score for each review
- get_sentiment_label(), assigns a sentiment label to the reviews

• Apply Sentiment Analysis

```
polarity_score_udf = udf(get_polarity_score, FloatType())
sentiment_label_udf = udf(get_sentiment_label, StringType())

new_data = new_data.withColumn("polarity_score", polarity_score_udf(col("review_text")))
new_data = new_data.withColumn("Sentiment_Label", sentiment_label_udf(col("review_text")))
```

Result:

department_name	class_name	clothing_id	title	review_text	rating	recommend_index	polarity_score	Sentiment_Label
Bottoms	Jeans	1028	amazing fit and wash	like reviewer hes...	5	1	0.6908	positive
Tops	Blouses	850	lovely and unique	true bunch fall c...	5	1	0.8718	positive
Bottoms	Pants	1068	wow	love love hesitan...	5	1	0.9767	positive
Intimate	Swim	24	great for bigger ...	absolutely love r...	5	1	0.5563	positive
Tops	Sweaters	933	love the pattern ...	love sweater im f...	4	1	0.9227	positive
Tops	Sweaters	937	beautiful and unique	love sweater soft...	5	1	0.9719	positive
Tops	Knits	868	unique and wonderful	sweater comfortab...	5	1	0.5719	positive
Tops	Fine gauge	900	great look all in...	love everything s...	5	1	0.9632	positive
Tops	Knits	873	beauty meets comfort	love top detail n...	5	1	0.9661	positive
Tops	Knits	872	great fit	searching around ...	5	1	0.9186	positive
Tops	Knits	873	adore but the whi...	dark blue adore r...	4	1	-0.3612	negative
Bottoms	Skirts	1008	not red not orange	yes much id liked...	5	1	0.9517	positive
Dresses	Dresses	1078	great with leggings	dress short wear ...	4	1	0.9442	positive
Bottoms	Pants	1060	perfect white pants	love pant purchas...	5	1	0.9358	positive
Intimate	Swim	286	wowser	made dawn suit bo...	5	1	0.979	positive
Bottoms	Pants	1053	pretty but a bit ...	cant decide pant ...	4	1	0.875	positive
Bottoms	Shorts	177	great summer staple	fit quite well ra...	5	1	0.7901	positive
Jackets	Jackets	967	great jacket	one cutest jacket...	5	1	0.9544	positive
Dresses	Dresses	1094	perfect for irish...	bought dress gree...	5	1	0.9442	positive
Tops	Blouses	829	even more vibrant...	color blouse beau...	5	1	0.9578	positive

Algorithm Experimentation

Data for model building

clothing_id	department_name	class_name	title	combined_review_text
100	Intimate	Intimates	sizing is off	good design comfortable fit however sizing way c small fit minim
1000	Bottoms	Skirts	awesome skirt	fit like model even though im really happy casual summer find lo
1004	Bottoms	Skirts	great skirt and yetno pockets	bought faded pink rose color inexplicably called red motif comfo
1006	Bottoms	Skirts	beautiful but not exactly as pictured	beautiful well made skirt flow curve beautifully complaint pictu
1008	Bottoms	Skirts	love this skirt	skirt fit amazing run bottom went little give fabric always plus
101	Intimate	Intimates	so cute	expensive panty worth money unique flattering
1010	Bottoms	Skirts	petites please	absolutely adore skirt sadly lot skirt im foot nothing would muc
1012	Bottoms	Skirts	adorable	pretty pocket also nice quality price interesting texture love s
1013	Bottoms	Skirts	great work skirt	skirt nice medium weight lined normal size fit perfect wish avai
1016	Bottoms	Skirts	beautiful skirt	love skirt lovely soft gauze layer lining underneath live summer
1017	Bottoms	Skirts	nice basic	nice basic pencil skirt closet think fit right problem billowing
1021	Bottoms	Skirts	love this skirtget it	beautiful skirt great piece collect erin fetherston design cut f
1022	Bottoms	Jeans	a long last perfect leggings	cant say enough pant often find legging revealing wearing long l
1027	Bottoms	Jeans	love these	th pair currentelliot jean ive bought im length perfect flat sof
103	Intimate	Layering	shimmer in silvergrey size down	ribbed camis stretch size prettydelicate eg fabric thin silvergr
1030	Bottoms	Jeans	bought 2 pairs	cord went sale bought nd pair great work week pant teach comfort
1038	Bottoms	Jeans	floral fun	absolutely love jean rise mid perfect low high embroidered flowe
1039	Bottoms	Jeans	great summer jeans	jean fit true size look amazing wear athletic build look great e
1040	Bottoms	Jeans	great fall jeans	fit incredible print subtle make little bit edgy easy pair many
1041	Bottoms	Pants	sophisticated feminine overalls	yeah know sound ridiculous really nice piece fabric thin soft li

Algorithm Experimentation

Word2Vec

- To transform both old and new reviews into vectors.

```
from pyspark.sql import SparkSession
from pyspark.ml.feature import Word2Vec
from pyspark.sql import functions as F

spark = SparkSession.builder.appName("ReviewSimilarity").getOrCreate()

new_review_text = "This is so comfortable and beautiful"
new_review_df = spark.createDataFrame([(new_review_text,)], ["combined_review_text"])
tokenized_df = aggregated_comments.select("class_name","combined_review_text",F.split(F.col("combined_review_text"), " ").alias("words"))

tokenized_new_review = new_review_df.select(F.split(F.col("combined_review_text"), " ").alias("words"))

word2Vec = Word2Vec(vectorSize=100, minCount=0, inputCol="words", outputCol="result")
model = word2Vec.fit(tokenized_df)

result_df = model.transform(tokenized_df)

new_result_df = model.transform(tokenized_new_review)
new_vector = new_result_df.collect()[0].result
```

Algorithm Experimentation

- Calculate the similarity between the new review and the old reviews to determine similarity

```
def cosine_similarity(v1, v2):
    norm_v1 = float(v1.norm(2))
    norm_v2 = float(v2.norm(2))
    if norm_v1 == 0 or norm_v2 == 0:
        return 0.0
    return float(v1.dot(v2)) / (norm_v1 * norm_v2)

similarities = []
for row in result_df.collect():
    similarity = cosine_similarity(new_vector, row.result)
    similarities.append((row.words, similarity, row.class_name, row.combined_review_text))
```

Algorithm Experimentation

```
similarities_df = spark.createDataFrame(similarities, ["words", "similarity", "class_name", "combined_review_text"])
similarities_df = similarities_df.select("class_name", "similarity") \
    .orderBy("similarity", ascending=False)
```

```
similarities_df.show()
```

```
+----------+------------------+
|class_name|        similarity|
+----------+------------------+
|    Lounge|0.8240463035813141|
|      Swim| 0.818630007178348|
|    Lounge|0.8118630790012138|
|    Lounge|0.8013311285498402|
| Intimates|0.7771936148549765|
|    Shorts|0.7768788095691238|
|    Shorts|0.7732644691660442|
|  Layering|0.7681678930943401|
|      Swim|0.7664632296083066|
|    Lounge|0.7630835723484342|
|    Lounge|0.7588442006808428|
```

- Convert the list of results into a DataFrame
- Filter products, similarity and sorting them in descending order

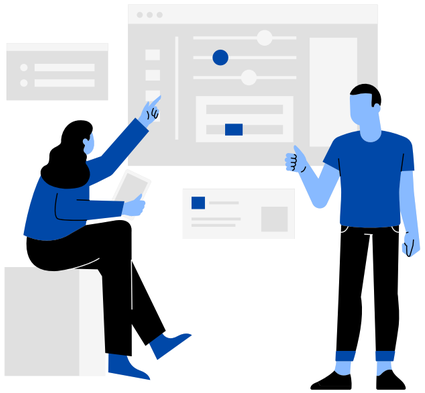
```
top_5_similarities = similarities_df.limit(5)

for row in top_5_similarities.collect():
    print(f"Recommended Products: '{row.class_name}'")
```

Filter out the top 5 reviews with the highest similarity to the new review

Results

- Successfully recommended five products based on the similarity between the new review and the old review.

```
Recommended Products: 'Lounge'
Recommended Products: 'Swim'
Recommended Products: 'Lounge'
Recommended Products: 'Lounge'
Recommended Products: 'Intimates'
```



Model Evaluation

01

Accuracy:

- Effectively recommends relevant products.
- Repetition of 'Lounge' indicates its ability to match user needs and preferences with precision.

02

Consistency:

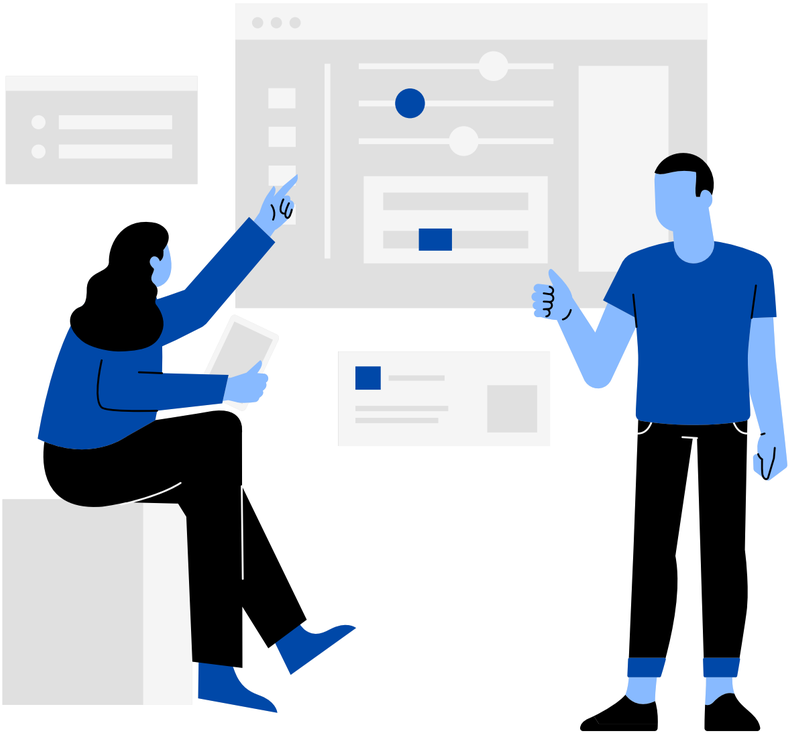
- Providing relevant product recommendations.
- Ensures reliable and diverse product suggestions.

03

Potential for Improvement:

- Broaden product recommendations for more options.
- Upgrade model for detailed and specific reviews to boost accuracy and flexibility.

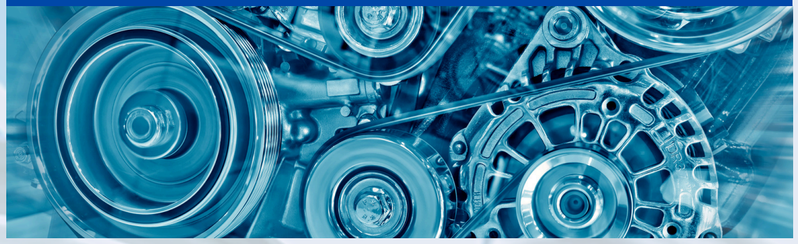
CONCLUSION



More Details

Small and non-diverse dataset

Lack of evaluation metrics



Limitations & Work Future

- Explore advanced ML & deep learning for improved accuracy.
- Develop web/mobile apps for online execution and real-time recommendations.

Thank you for your attention