SATHYABAMA INSTITUTE OF SCIENCE & TECHNOLOGY

SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SCSA 2604 NATURAL LANGUAGE PROCESSING LAB

LAB – 1 : CASE STUDY

AIM: To Enhance Customer Feedback Analysis through NLP-based Text Processing

PROBLEM STATEMENT:

A company receives a large volume of customer feedback across various channels such as emails, social media, and surveys. Understanding and categorizing this feedback manually is time-consuming and inefficient. The goal is to develop an NLP-based program to automatically process and analyze customer feedback to extract valuable insights.

OBJECTIVE:

Utilize spaCy and NLP techniques to process customer feedback text, extract tokens, perform lemmatization, and conduct dependency parsing to uncover underlying relationships between words.

APPROACH:

Data Collection:

Gather a dataset containing customer feedback from different sources, including emails, social media comments, and survey responses.

Text Processing with spaCy:

Utilize spaCy to process the customer feedback text. Extract tokens to identify individual words and perform lemmatization to obtain their base forms.

Dependency Parsing Analysis:

Use spaCy's dependency parsing feature to identify the syntactic relationships between words. Analyze the dependency tree to understand how different parts of the feedback sentences are connected.

Insight Generation:

Categorize the feedback based on sentiment, identify frequently occurring topics, or extract key phrases related to specific issues or praises mentioned by customers.

Implementation:

Use Python and spaCy to develop the program for text processing and analysis.

Incorporate visualization techniques (e.g., graphs, word clouds) to represent the findings and insights derived from the processed feedback.

Evaluation:

Evaluate the accuracy and efficiency of tokenization, lemmatization, and dependency parsing in handling different types of customer feedback.

Measure the program's ability to extract meaningful insights and categorize feedback accurately.

PROGRAM:

```python
import spacy


# Load English tokenizer, tagger, parser, NER, and word vectors

nlp = spacy.load("en_core_web_sm")


# Sample customer feedback data

customer_feedback = [

    "The product is amazing! I love the quality.",

    "The customer service was terrible, very disappointed.",

    "Great experience overall, highly recommended.",

    "The delivery was late, very frustrating."
```

```
]

def analyze_feedback(feedback):
    for idx, text in enumerate(feedback, start=1):
        print(f"\nAnalyzing Feedback {idx}: '{text}'")
        doc = nlp(text)

        # Extract tokens and lemmatization
        tokens = [token.text for token in doc]
        lemmas = [token.lemma_ for token in doc]
        print("Tokens:", tokens)
        print("Lemmas:", lemmas)

        # Dependency parsing
        print("\nDependency Parsing:")
        for token in doc:
            print(token.text, token.dep_, token.head.text, token.head.pos_,
                [child for child in token.children])

if __name__ == "__main__":
    analyze_feedback(customer_feedback)
```

OUTPUT:

Analyzing Feedback 1: 'The product is amazing! I love the quality.'

Tokens: ['The', 'product', 'is', 'amazing', '!', 'I', 'love', 'the', 'quality', '.']

Lemmas: ['the', 'product', 'be', 'amazing', '!', 'I', 'love', 'the', 'quality', '.']

Dependency Parsing:

The det product NOUN []

product nsubj is AUX [The]

is ROOT is AUX [product, amazing, !]

amazing acomp is AUX []

! punct is AUX []

I nsubj love VERB []

love ROOT love VERB [I, quality, .]

the det quality NOUN []

quality dobj love VERB [the]

. punct love VERB []

Analyzing Feedback 2: 'The customer service was terrible, very disappointed.'

Tokens: ['The', 'customer', 'service', 'was', 'terrible', ',', 'very', 'disappointed', '.']

Lemmas: ['the', 'customer', 'service', 'be', 'terrible', ',', 'very', 'disappointed', '.']

Dependency Parsing:

The det service NOUN []

customer compound service NOUN []

service nsubj was AUX [The, customer]

was ROOT was AUX [service, disappointed, .]

terrible amod disappointed ADJ []

, punct disappointed ADJ []

very advmod disappointed ADJ []

disappointed acomp was AUX [terrible, ,, very]

. punct was AUX []

Analyzing Feedback 3: 'Great experience overall, highly recommended.'

Tokens: ['Great', 'experience', 'overall', ',', 'highly', 'recommended', '.']

Lemmas: ['great', 'experience', 'overall', ',', 'highly', 'recommend', '.']

Dependency Parsing:

Great amod experience NOUN []

experience nsubj recommended VERB [Great]

overall advmod recommended VERB []

, punct recommended VERB []

highly advmod recommended VERB []

recommended ROOT recommended VERB [experience, overall, ,, highly, .]

. punct recommended VERB []

Analyzing Feedback 4: 'The delivery was late, very frustrating.'

Tokens: ['The', 'delivery', 'was', 'late', ',', 'very', 'frustrating', '.']

Lemmas: ['the', 'delivery', 'be', 'late', ',', 'very', 'frustrating', '.']

Dependency Parsing:

The det delivery NOUN [ ]

delivery nsubj was AUX [The]

was ROOT was AUX [delivery, frustrating, .]

late advmod frustrating ADJ [ ]

, punct frustrating ADJ [ ]

very advmod frustrating ADJ [ ]

frustrating acomp was AUX [late, ,, very]

. punct was AUX [ ]

CONCLUSION:

The developed NLP-based program utilizing spaCy proves to be an efficient solution for processing and analyzing customer feedback. Its capability to extract tokens, perform lemmatization, and conduct dependency parsing aids in understanding the sentiment, identifying key topics, and establishing relationships within the feedback data. This enables companies to derive actionable insights, prioritize issues, and enhance customer satisfaction based on the analysis of their feedback.

RESULT:

This case study demonstrates the practical application of the provided code snippet using spaCy in a business context, specifically for customer feedback analysis, showcasing how NLP techniques can be employed to extract valuable insights from unstructured text data.

Mrs. Parveen . A , Assistant Professor, Dept. of CSE, SIST