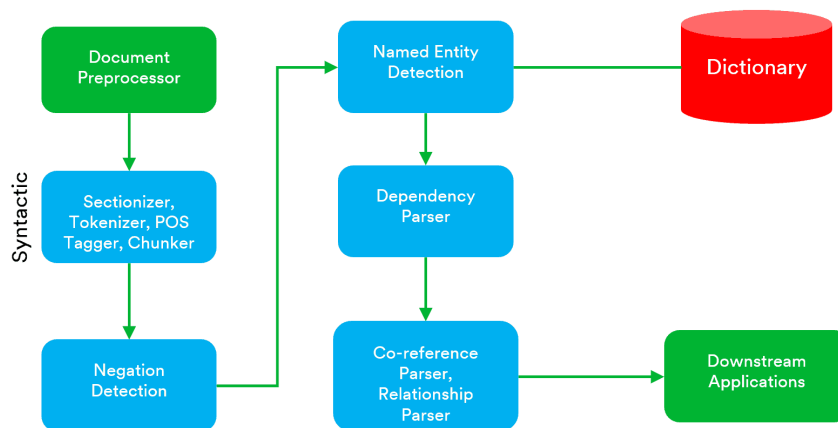


SENTIMENTAL ANALYSIS FOR MARKETING

Employing NLP Techniques :

```
import nltk
import re
from nltk.corpus import stopwords
```

Typical Components of an NLP Application



© 3M 2018. All Rights Reserved. 3M Confidential.

3M

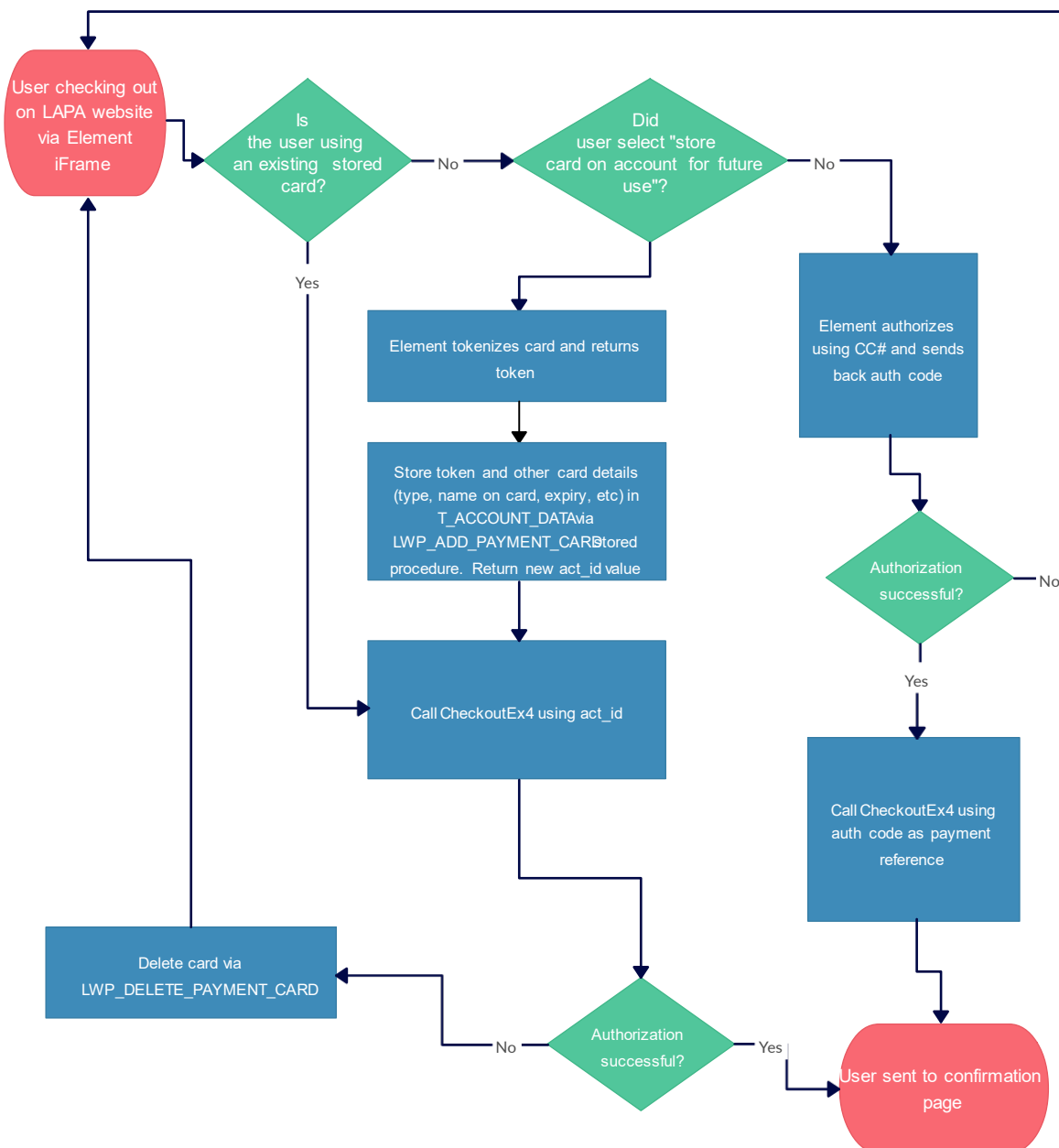
2

- **Importing the NLTK Library:** `import nltk`: This statement imports the Natural Language Toolkit (NLTK) library, which is a powerful library for working with human language data.
- **Importing Regular Expression Module:** `import re`: This statement imports the regular expression module (re) in Python. Regular expressions are used for pattern matching and text manipulation.
- **Importing NLTK Stopwords:** `from nltk.corpus import stopwords`: This line imports the stopwords corpus from NLTK. Stopwords are common words (e.g., "and," "the," "is") that are often removed from text data because they don't contribute much to the meaning.

Now, let's discuss the typical use of these imports in NLP preprocessing:

- **NLTK (Natural Language Toolkit):** NLTK provides tools for tasks such as tokenization, stemming, tagging, parsing, and more. It's widely used for natural language processing and text analysis in Python.
- **Regular Expression (re) Module:** Regular expressions are powerful tools for pattern matching and text manipulation. In NLP, they are often used for tasks like text cleaning, removing special characters, or extracting specific patterns from text.
- **Stopwords:** Stopwords are words that are commonly used in a language but are often filtered out in text processing because they don't carry much information about the content of the text. NLTK's stopwords corpus provides a list of common stopwords for different languages.

Tokenization :



```
df['sentiment'] = df['airline_sentiment'].apply(lambda x:0 if x=='negative' else 1)
df['clean_tweet'] = df['text'].apply(lambda x:tweet_to_words(x))
df['tweet_length'] = df['text'].apply(lambda x:tweet_len(x))
train, test = train_test_split(df,test_size=0.2,random_state=42)
```

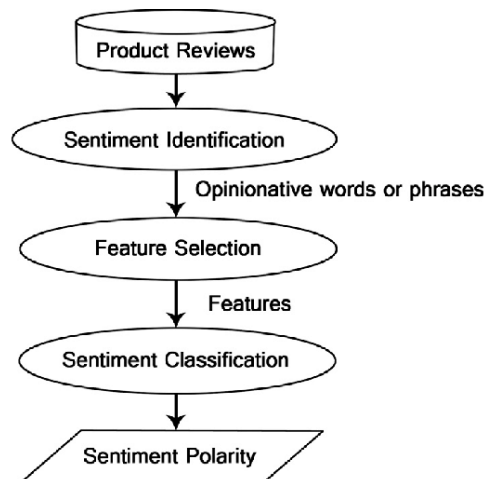
- **Sentiment Labeling:** This line creates a new column called 'sentiment' in the DataFrame df. It assigns a value of 0 if the corresponding value in the 'airline_sentiment' column is 'negative' and 1 otherwise. This step converts the sentiment labels into numerical values for binary classification (0 for negative, 1 for positive).
- **Text Cleaning:** A new column 'clean_tweet' is added, and it is created by applying a function tweet_to_words(x) to each element in the 'text' column of the DataFrame. This suggests that tweet_to_words is a function designed to clean and preprocess text data, likely for preparing it for natural language processing (NLP) tasks.
- **Tweet Length Calculation:** Another new column 'tweet_length' is added, and it is created by applying a function tweet_len(x) to each element in the 'text' column. This implies that tweet_len is a function designed to calculate the length of each tweet.
- **Train-Test Split:** The DataFrame df is split into training and testing sets using the train_test_split function. The test_size parameter is set to 0.2, meaning that 20% of the data will be used for testing, and the random_state parameter is set to 42 for reproducibility.

```
from sklearn.feature_extraction.text import CountVectorizer
v = CountVectorizer(analyzer='word')
train_features = v.fit_transform(train_clean_tweet)
test_features = v.transform(test_clean_tweet)
```

- **Importing CountVectorizer:** from sklearn.feature_extraction.text import CountVectorizer: This line imports the CountVectorizer class from Scikit-learn's feature extraction module. The CountVectorizer is used to convert a collection of text documents to a matrix of token counts.
- **Initializing CountVectorizer:** v = CountVectorizer(analyzer='word'): This line creates an instance of the CountVectorizer class. The analyzer parameter is set to 'word', indicating that the analyzer will be based on words.
- **Transforming Training Data:** train_features = v.fit_transform(train_clean_tweet): The fit_transform method is applied to the training data (train_clean_tweet). This method both fits the vectorizer to the data (learns the vocabulary) and transforms the training data into a document-term matrix. The result (train_features) is a sparse matrix where each row corresponds to a document (tweet), and each column corresponds to a unique word in the entire training dataset. The matrix contains counts of how many times each word appears in each document.

- **Transforming Test Data:** `test_features = v.transform(test_clean_tweet)`: The transform method is applied to the test data (`test_clean_tweet`). This uses the vocabulary learned from the training data to transform the test data into a similar document-term matrix. It's important to use the same vectorizer instance (with the same vocabulary) for both training and test data to ensure consistency.

Insights For Sentimental Analysis For Marketing:



Sentiment analysis in marketing can provide valuable insights into customer perceptions, preferences, and satisfaction. Here are some key insights that can be generated through sentiment analysis in the context of marketing:

- **Overall Sentiment Distribution:** Analyze the distribution of sentiments across customer feedback to understand the overall sentiment landscape. This provides an overview of how positive, negative, and neutral sentiments are distributed.
- **Trend Analysis Over Time:** Track sentiment trends over time to identify patterns or changes in customer sentiment. This insight can help marketers understand the impact of marketing campaigns, product launches, or other events on customer perception.
- **Identifying Product Strengths and Weaknesses:** Analyzing sentiment in product-related discussions can reveal specific features or aspects that customers appreciate (positive sentiment) and areas that need improvement (negative sentiment).
- **Campaign Effectiveness:** Assess the effectiveness of marketing campaigns by analyzing sentiment around campaign-related content. Positive sentiment may indicate successful campaigns, while negative sentiment can highlight areas for improvement.
- **Customer Feedback for Product Development:** Utilize sentiment analysis to extract valuable feedback from customers regarding products or services. Positive sentiments can indicate features that resonate well, while negative sentiments can point to areas for innovation or enhancement.

- **Competitor Benchmarking:** Compare sentiment across different brands or competitors to understand how your brand is positioned in the market relative to others. Identify competitive strengths and weaknesses.
- **Identifying Brand Advocates and Influencers:** Positive sentiment can help identify brand advocates and influencers who are positively impacting the brand. Engage with these individuals to amplify positive sentiment and increase brand reach.
- **Customer Satisfaction Metrics:** Use sentiment analysis as a metric for customer satisfaction. Track changes in sentiment to assess the effectiveness of customer service, product quality, and overall customer experience.
- **Social Media Engagement Analysis:** Analyze sentiment on social media platforms to understand how customers are engaging with the brand. Identify popular topics, monitor brand mentions, and respond to customer concerns promptly.
- **Crisis Management:** Detect negative sentiment spikes early to address potential crises promptly. Sentiment analysis can serve as an early warning system for emerging issues that may impact the brand negatively.
- **Segmentation of Customer Sentiment:** Segment customer sentiment based on demographics, geography, or other factors to tailor marketing strategies for different customer groups. Understand how sentiment varies among different audience segments.
- **Adjusting Marketing Strategies:** Use sentiment insights to adapt marketing strategies. If a particular campaign receives positive feedback, consider replicating its elements in future campaigns. Conversely, adjust strategies that generate negative sentiment.
- **Content Strategy Optimization:** Optimize content strategies based on the types of content that generate positive sentiment. Identify content themes, formats, or channels that resonate well with the audience.
- **Measuring Brand Loyalty:** Positive sentiment can be an indicator of brand loyalty. Identify and nurture loyal customers who consistently express positive sentiments about the brand.

By leveraging sentiment analysis in marketing, businesses can gain a deeper understanding of customer sentiments, make data-driven decisions, and continuously improve their marketing strategies to build positive brand perception and enhance customer satisfaction.