

Experiment 11: Turney algorithm for unsupervised sentiment analysis

#Objective:

The objective of this experiment is to implement Turney's unsupervised sentiment analysis algorithm, which classifies words or phrases as "Positive," "Negative," or "Neutral" using Pointwise Mutual Information and Information Retrieval (PMI-IR) scores based on predefined word lists.

#Code

```
import math
from collections import Counter
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.util import ngrams
import nltk
# Make sure you have NLTK downloaded
nltk.download('punkt')
nltk.download('stopwords')

positive_words = [
    "good", "great", "excellent", "awesome", "positive", "happy", "love",
    "fantastic", "amazing", "outstanding", "wonderful", "brilliant", "perfect",
    "superb", "satisfied", "incredible", "delightful", "impressive", "terrific",
    "marvelous", "exceptional", "beautiful", "comfortable", "joyful", "remarkable",
    "loved", "enjoyable", "super", "pleasing", "exciting", "affordable", "uplifting",
    "cheerful", "dynamic", "charming", "successful", "inspiring", "motivating",
    "fabulous", "radiant", "rewarding", "admirable", "pleasant", "elevating",
    "thrilling", "outshining", "inviting", "blessed", "refined", "gratifying",
    "heartwarming", "joyous", "amusing", "refreshing"
]

negative_words = [
    "bad", "poor", "terrible", "horrible", "negative", "sad", "hate",
    "awful", "dismal", "disappointing", "frustrating", "unpleasant", "ugly",
    "mediocre", "unreliable", "uncomfortable", "regret", "worst", "undesirable",
    "dreadful", "vile", "unfortunate", "irritating", "depressing", "disastrous",
    "disheartening", "underwhelming", "frustrating", "lousy", "horrendous", "unhappy",
    "miserable", "hopeless", "pathetic", "excruciating", "lousy", "shameful",
    "regrettable", "annoying", "embarrassing", "disturbing", "tragic", "painful",
    "unpleasant", "shoddy", "crummy", "failing", "detestable", "abysmal",
    "displeasing", "unfortunate", "grotesque", "nauseating"
]

# Function to calculate PMI-IR (Pointwise Mutual Information and Information Retrieval)
def pmi_ir(word, seed_word, corpus):
```

```

"""Calculate the PMI-IR score between a target word and a seed word"""
word_count = corpus[word]
seed_word_count = corpus[seed_word]
word_and_seed_count = corpus[(word, seed_word)]
if word_count == 0 or seed_word_count == 0 or word_and_seed_count == 0:
    return 0 # Return 0 if any count is zero, as no association is possible
p_word = word_count / sum(corpus.values())
p_seed = seed_word_count / sum(corpus.values())
p_word_given_seed = word_and_seed_count / seed_word_count

return math.log(p_word_given_seed / (p_word * p_seed))

```

Function to get n-grams from a text

```

def get_ngrams(text, n=1):
    """Generate n-grams from text"""
    tokens = word_tokenize(text.lower())
    tokens = [word for word in tokens if word not in stopwords.words('english')] # Remove
stopwords
    return list(ngrams(tokens, n))

```

sample = "

"The sky is blue today.", "I went to the store.", "This chair is comfortable.",
 "She has a meeting at 3 PM.", "The book is on the table.", "I will call you later.",
 "It is raining outside.", "He is working on a new project.", "I just finished lunch.",
 "The computer is charging.", "I read a book last night.", "They are watching TV.",
 "The car is parked outside.", "She is wearing a red dress.", "The team will meet tomorrow.",
 "I like to take walks in the park.", "We will have a meeting in the afternoon.",
 "The laptop is on the desk.", "I have an appointment at 10 AM.", "It's a good time to relax.",
 "I will send the email by tomorrow.", "The package arrived yesterday.",
 "The door is closed.", "The phone is on the counter.", "She is reading the newspaper.",
 "They are cleaning the house.", "The movie was interesting.", "I am going to sleep early.",
 "We need to buy groceries.", "The train is on time.", "The office is quiet today.",
 "I'm feeling neutral today.", "The streets are empty.", "I have an idea for a new project.",
 "They just finished their presentation.", "I'm attending a webinar later.",
 "The coffee is hot.", "She likes to listen to music in the morning.", "I need to organize my desk.",
 "They are planning a vacation.", "I'm just taking a break.", "I watched a documentary last night.",
 "The road is under construction.", "I took a nap after lunch.", "He is working from home today.",
 "It's been a long day.", "I'm going to the gym tomorrow.", "She enjoys painting.",
 "We should discuss the plan next week.", "I had a sandwich for lunch.",
 "The weather forecast predicts sunny days.", "The bakery is closed today.",
 "This is a fantastic day!", "I am feeling great today.", "That was an amazing experience!",
 "She did an excellent job on her presentation.", "I love spending time with my friends.",
 "This movie was absolutely brilliant.", "The food was delicious!", "The concert was awesome.",
 "I'm so proud of what we accomplished.", "This is the best thing I've ever done.",
 "You look stunning today!", "I had an amazing time at the event.", "We had a wonderful
 evening.",
 "It's great to see everyone so happy.", "This is a beautiful place.", "I'm feeling super excited.",
 "It's such a joyous occasion.", "I absolutely love the way this turned out.",

"The weather is perfect for a walk.", "I'm so glad I met you.", "This is a lovely surprise!",
 "What an incredible achievement!", "I'm so thankful for everything.", "She gave a superb performance.",
 "It's a fantastic opportunity.", "I'm really enjoying this new project.",
 "The view from here is breathtaking.", "This is exactly what I needed.", "You did a fantastic job!",
 "I'm so happy with the results.", "What a beautiful day it is.", "This new gadget is amazing.",
 "I love spending time outdoors.", "That was a highly satisfying experience.",
 "Everything went perfectly!", "This place feels so welcoming.", "The team did an outstanding job.",
 "This is a perfect way to relax.", "I'm feeling incredibly positive.", "What a wonderful day!",
 "You made my day with that news!", "It's such an inspiring story.", "I'm thrilled with the progress we've made.",
 "It's always a joy to see you.", "The energy in the room was fantastic.", "This is such a rewarding experience.",
 "I can't wait for the next adventure.", "This is so refreshing!", "It's been a truly uplifting day.",
 "I feel so blessed.",
 "I am having a terrible day.", "This situation is awful.", "I feel so frustrated right now.",
 "That was such a disappointing experience.", "I am not happy with the results.", "I hate this place.",
 "This is the worst news I've heard.", "I feel miserable today.", "This is such a bad idea.",
 "I am not feeling great at all.", "This food tastes horrible.", "That movie was dreadful.",
 "I'm feeling very negative today.", "That was a horrible mistake.", "I can't believe this is happening.",
 "This has been a very frustrating process.", "I hate waiting.", "Everything is going wrong today.",
 "I'm so disappointed with the outcome.", "This is the worst experience ever.",
 "I'm feeling down today.", "This is an unpleasant situation.", "I had a terrible time at the event.",
 "I don't want to talk about it.", "This place feels so uncomfortable.", "This has been a really awful week.",
 "I'm so frustrated with the situation.", "That was a complete disaster.", "This is such a regrettable decision.",
 "I am not impressed by the results.", "I feel so miserable.", "This situation is hopeless.",
 "I really don't like this idea.", "This has been an exhausting day.", "I can't stand this anymore.",
 "I am extremely upset about this.", "This place makes me feel uneasy.", "That was an unpleasant experience.",
 "I am so annoyed right now.", "That's such a terrible thought.", "I am just tired of everything.",
 "I don't feel like doing anything today.", "I feel so bad about what happened.", "This is absolutely frustrating.",
 "I'm really upset with how things turned out.", "This is so disappointing.", "I hate being stuck in traffic.",
 "I am done with all of this.", "This has been a truly awful experience.", "I feel so discouraged.",
 "This is a nightmare.", "I feel so irritated.", "That's such a crummy situation."

'''

```
corpus = Counter(get_ngrams(sample, 1)) # Unigrams
```

```
# Function to classify sentiment of a word using Turney's algorithm
```

```
def turney_sentiment_analysis(word, positive_words, negative_words, corpus):  
    pos_score = sum([pmi_ir(word, p_word, corpus) for p_word in positive_words])  
    neg_score = sum([pmi_ir(word, n_word, corpus) for n_word in negative_words])  
  
    if pos_score > neg_score:  
        return "Positive"  
    elif neg_score > pos_score:  
        return "Negative"  
    else:  
        return "Neutral"  
  
target_word = "People study computer science"  
sentiment = turney_sentiment_analysis(target_word, positive_words, negative_words, corpus)  
print(f'Sentiment of the word '{target_word}': {sentiment}')
```

#Output

Sentiment of the word 'People study computer science': Neutral

#Conclusion:

The Turney algorithm effectively classifies sentiment without supervised learning, as demonstrated by the classification of the phrase "People study computer science" as "Neutral." By leveraging PMI-IR scores derived from n-grams and seed word lists, the algorithm can classify text based on its association with positive and negative terms. While the approach works well in the provided context, its accuracy is influenced by the comprehensiveness of the positive and negative word lists and the quality of the corpus. Overall, it offers a scalable method for sentiment analysis in environments lacking labeled datasets.