BLACK COFFER SUMMARY

Presented By

RASHID ALI

Agenda

- ABOUT THE PROJECT
- INTRODUCTION
- WEB SCRAPING
- DATA PREPROCESSING
- EXTRACT FEATURES
- SCRAPING THE DATA
- DATA CRAWLING
- DATA EXTRACTION
- PREPARED DATA

About the Project

- The objective of this assignment is to extract textual data articles from the given URL and perform text analysis to compute variables that are explained below.
- For each of the articles, given in the input.xlsx file, extract the article text and save the extracted article in a text file with URL_ID as its file name.
- While extracting text, please make sure your program extracts only the article title and the article text. It should not extract the website header, footer, or anything other than the article text.
- For each of the extracted texts from the article, perform textual analysis and compute variables, given in the output structure excel file. You need to save the output in the exact order as given in the output structure file, “Output Data Structure.xlsx”

Introduction

What is Data Extraction and Text Analysis?

Data extraction is the process of obtaining data from a database or SaaS platform so that it can be replicated to a **destination** — such as a **data warehouse** — designed to support online analytical processing (OLAP).

Data extraction is the first step in a **data ingestion** process called **ETL — extract, transform, and load**. The goal of ETL is to prepare data for analysis or **business intelligence** (BI).

Suppose an organization wants to monitor its reputation in the marketplace. It may have data from many **sources**, including online reviews, social media mentions, and online transactions. An ETL tool can extract data from these sources and load it into a data warehouse where it can be analyzed and mined for insights into brand perception.

Text analysis (TA) is a machine learning technique used to automatically extract valuable insights from unstructured text data. Companies use text analysis tools to quickly digest online data and documents, and transform them into actionable insights.

You can use text analysis to extract specific information, like keywords, names, or company information from thousands of emails, or categorize survey responses by sentiment and topic.

WEB SCRAPING

- **Web scraping**, **web harvesting**, or **web data extraction** is data scraping used for extracting data from websites. The web scraping software may directly access the World Wide Web using the Hypertext Transfer Protocol or a web browser. While web scraping can be done manually by a software user, the term typically refers to automated processes implemented using a bot or web crawler.
- It is a form of copying in which specific data is gathered and copied from the web, typically into a central local database or spreadsheet, for later retrieval or analysis.
- Web scraping a web page involves fetching it and extracting from it. Fetching is the downloading of a page (which a browser does when a user views a page). Therefore, web crawling is the main component of web scraping, to fetch pages for later processing. Once fetched, then extraction can take place.
- The content of a page may be parsed, searched, reformatted, its data copied into a spreadsheet or loaded into a database.
- Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else.
- An example would be to find and copy names and telephone numbers, companies and their URLs, or e-mail addresses to a list (contact scraping).

DATA PREPROCESSING

Tokenize -To tokenize sentences and words with NLTK, “nltk.word_tokenize()” function will be used. NLTK Tokenization is used for parsing a large amount of textual data into parts to perform an analysis of the character of the text. NLTK for tokenization can be used for training machine learning models, Natural Language Processing text cleaning. The tokenized words and sentences with NLTK can be turned into a data frame and vectorized. Natural Language Tool Kit (NLTK) tokenization involves punctuation cleaning, text cleaning, vectorization of parsed text data for better lemmatization, and stemming along with machine learning algorithm training.

Stop words are words in any language or corpus that occur frequently. For some NLP tasks, they do not provide any additional or valuable information to the text containing them. Words like *a*, *they*, *the*, *is*, *an*, etc. are usually considered stop words.

Sentiment Analysis-Python sentiment analysis is a methodology for analyzing a piece of text to discover the sentiment hidden within it. It accomplishes this by combining machine learning and natural language processing (NLP). Sentiment analysis allows you to examine the feelings expressed in a piece of text.

Positive Score: This score is calculated by assigning the value of +1 for each word if found in the Positive Dictionary and then adding up all the values.**Negative Score**: This score is calculated by assigning the value of -1 for each word if found in the Negative Dictionary and then adding up all the values. We multiply the score with -1 so that the score is a positive number.

Polarity Score: This is the score that determines if a given text is positive or negative in nature. It is calculated by using the formula: polarity Score = (Positive Score – Negative Score)/ ((Positive Score + Negative Score) + 0.000001),Range is from -1 to +1

Subjectivity Score: This is the score that determines if a given text is objective or subjective. It is calculated by using the formula: Subjectivity Score = (Positive Score + Negative Score)/ ((Total Words after cleaning) + 0.000001),Range is from 0 to +1

Average Sentence Length = the number of words / the number of sentences

Percentage of Complex words = the number of complex words / the number of words

Fog Index = 0.4 * (Average Sentence Length + Percentage of Complex words)

We count the number of Syllables in each word of the text by counting the vowels present in each word. We also handle some exceptions like words ending with "es","ed" by not counting them as a syllable.

Personal Pronouns

To calculate Personal Pronouns mentioned in the text, we use regex to find the counts of the words - “I,” “we,” “my,” “ours,” and “us”. Special care is taken so that the country name US is not included in the list.

Average Word Length

Average Word Length is calculated by the formula:Sum of the total number of characters in each word/Total number of words

EXTRACT FEATURES

1. POSITIVE SCORE
2. NEGATIVE SCORE
3. POLARITY SCORE
4. SUBJECTIVITY SCORE
5. AVG SENTENCE LENGTH
6. PERCENTAGE OF COMPLEX WORDS
7. FOG INDEX
8. AVG NUMBER OF WORDS PER SENTENCE
9. COMPLEX WORD COUNT
10. WORD COUNT
11. SYLLABLE PER WORD
12. PERSONAL PRONOUNS
13. AVG WORD LENGTH

SCRAPING THE DATA

```
r = requests.get(url.URL[1], headers={"User-Agent": "XY"})
data = r.text
soup = BeautifulSoup(data)
print(soup)
```

```
<!DOCTYPE html>
<!--[if IE 8]>    <html class="ie8" lang="en"> <![endif]--><!--[if IE 9]>    <html class="ie9" lang="en"> <![en
dif]--><!--[if gt IE 8]><!--><html lang="en-US"> <!--<![endif]-->
<head>
<meta charset="utf-8"/>
<meta content="width=device-width, initial-scale=1.0" name="viewport"/>
<link href="https://insights.blackcoffer.com/xmlrpc.php" rel="pingback"/>
<meta content="index, follow, max-image-preview:large, max-snippet:-1, max-video-preview:-1" name="robots"/>
<meta content="https://insights.blackcoffer.com/wp-content/uploads/2022/04/kolpp.jpg" property="og:image"/><lin
k href="https://insights.blackcoffer.com/wp-content/uploads/2018/12/Black76.png" rel="apple-touch-icon-precompo
sed" sizes="76x76"/><link href="https://insights.blackcoffer.com/wp-content/uploads/2018/12/Black120.png" rel
="apple-touch-icon-precomposed" sizes="120x120"/><link href="https://insights.blackcoffer.com/wp-content/upload
s/2018/12/Black152.png" rel="apple-touch-icon-precomposed" sizes="152x152"/><link href="https://insights.blackc
offer.com/wp-content/uploads/2018/12/Black114.png" rel="apple-touch-icon-precomposed" sizes="114x114"/><link hr
ef="https://insights.blackcoffer.com/wp-content/uploads/2018/12/Black144.png" rel="apple-touch-icon-precompose
d" sizes="144x144"/>
<!-- This site is optimized with the Yoast SEO plugin v19.2 - https://yoast.com/wordpress/plugins/seo/ -->
<title>Is telehealth the future of healthcare? - Blackcoffer Insights</title>
<meta content="The growing advancements in technology has enabled the creation of several wearable technologies
which would help in better self-tracking of health." name="description"/>
<link href="https://insights.blackcoffer.com/is-telehealth-the-future-of-healthcare-3/" rel="canonical"/>
<meta content="en_US" property="og:locale"/>
<meta content="article" property="og:type"/>
<meta content="Is telehealth the future of healthcare? - Blackcoffer Insights" property="og:title"/>
<meta content="The growing advancements in technology has enabled the creation of several wearable technologies
which would help in better self-tracking of health." property="og:description"/>
<meta content="https://insights.blackcoffer.com/is-telehealth-the-future-of-healthcare-3/" property="og:url"/>
<meta content="Blackcoffer Insights" property="og:site_name"/>
<meta content="https://www.facebook.com/blackcoffer.consulting" property="article:publisher"/>
<meta content="2022-04-28T15:45:11+00:00" property="article:published_time"/>
<meta content="2022-04-28T15:45:13+00:00" property="article:modified_time"/>
<meta content="https://insights.blackcoffer.com/wp-content/uploads/2022/04/kolpp.jpg" property="og:image"/>
<meta content="1200" property="og:image:width"/>
<meta content="630" property="og:image:height"/>
```

DATA CRAWLING

```
mydivs=soup.find_all("div", class_="td-post-content")
a=str(mydivs[0].text)
print(a)
```

```
Covid-19 has paved the way for advancements in Telehealth services, which have currently become a significant c
omponent of the healthcare industry to cope with the social distancing and lockdown measures, people adapted to
telehealth services to avail doctor expertise.
Telehealth's global market size is projected to grow to $55.6 billion by 2025 from $25.4 billion in 2020 at a C
AGR of 16.9% during the period.  According to the statistics provided by the Medical Council of India, the doct
or-patient ratio currently is 1:2000. Such a low doctor-patient ratio results in most of the population without
any access to proper healthcare facilities. The budding digital infrastructure covers the gaps created by the l
ack of quality hospital infrastructure and medical staff, allowing doctors to converse with patients in the far
-flung rural areas, through telemedicine.
Telemedicine will be a major contributor to filling the current gaps in the healthcare industry, a some of the
factors contributing to its growth are:
 Internet Connectivity:
As internet connectivity in India is currently growing at a rapid pace of 45% as of 2021, this growth will help
in further penetrating the market, thus reaching people in areas lacking healthcare facilities. This would avai
l quality health care services to the people, allowing for efficient and effective treatment of diseases preval
ent in the area.
Growing healthcare startups in India
The rapid growth in Digital infrastructure has led to a booming increase in the number of healthcare startups p
resent in India. These start-ups are dedicated to the development of disease tracking and prevention tools. Sin
ce most remote areas do not have proper medical facilities or are unable to afford quality healthcare services,
these start-ups have developed tools that enable the in-need patients to connect with far-flung doctors to rece
ive affordable medical advice without incurring huge transportation costs. Start-ups such as Practo and 1 MG pr
ovide delivery of medicines that are at times not easily available.
Cloud-based and AI telehealth solutions provide tools for employees and companies. These tools help in understa
nding the onset of diseases, current analysis of the causal factors of prominent diseases, and the most prevale
nt diseases in the population. Such tools better analyze the causal factors and develop prominent solutions for
prevention, thus helping in better disease control for the organization and country.
Companies such as Tata Digital Health provide cloud-based platforms for storing and understanding patient medic
al records and prescriptions. These help in understanding the track record of the patient, helping doctors to p
rescribe suitable medicines for the disease.
Self-monitoring and Wearable technology
The growing advancements in technology have enabled the creation of several wearable technologies which would h
elp in better self-tracking of health. The onset of smart-watches and easily available medical devices has resu
lted in a better collection of medical data. Such devices would help understand the warning signs of certain me
dical diseases, analyze the trends in their progress and develop better solutions to prevent them.
The increasing cost of medical facilities
With the improvement of medical facilities, the cost of acquiring the facilities is also growing which the majo
rity of the Indian population cannot afford. This issue can be resolved with the advancements of telemedicine w
hich can reduce various expenses like travel for the people which people can utilize in availing better medical
practitioners.
Government initiatives
The resolution of the Indian government and various state governments to improve the healthcare facilities and
make them accessible to the majority of the people has led them to launch various schemes like Ayushman Bharat
which provides free health insurance to the bottom 50% of the population, and Mohalla Clinic, the initiative of
Delhi government to provide free medical services for everyone.
```

DATA EXTRACTION

In [87]:

```
for i in range(len(report)):
    content=a
    tokenized_words = tokenize(content)
    #print(f'Total tokenized words are {len(tokenized_words)}')
    words = remove_stopwords(tokenized_words, Stopwords)
    num_words = len(words)
    #print(f'Total words after removing stop words are {len(words)}')
    avg_word_length=average_word(words)/ num_words
    positive_score = countfunc(positive_dictionary, words)
    negative_score = countfunc(negative_dictionary, words)
    #print(f'Total positive score is {positive_score}')
    #print(f'Total negative score is {negative_score}')
    polarity_score = polarity(positive_score, negative_score)
    #print(polarity_score)
    subjectivity_score = subjectivity(positive_score, negative_score, num_words)
    #print(subjectivity_score)
    #print(sentiment(polarity_score))
    sentences = sent_tokenize(content)
    num_sentences = len(sentences)
    average_sentence_length = num_words/num_sentences
    #print(average_sentence_length)
    average_word_per_sentence=num_words/num_sentences


    num_complexword=0
    personal_pronouns= 0
    syllable_per_word=0

    for word in words:
        w,c=syllable_morethan2(word)
        if(w):
            num_complexword = num_complexword+1
        if(pronouns(word)):
            personal_pronouns+=1

        avg_word_length=average_word(word)/num_words
        syllable_per_word = syllable_per_word+c


    percentage_complexwords = num_complexword/num_words
    #print(percentage_complexwords)
    fog_index = fog_index_cal(average_sentence_length, percentage_complexwords)
    #print(fog_index)



    url_out['POSITIVE SCORE'][i] = positive_score
    url_out['NEGATIVE SCORE'][i]= negative_score
    url_out['POLARITY SCORE'][i] = polarity_score
    url_out['SUBJECTIVITY SCORE'][i] = subjectivity_score
    url_out['AVG SENTENCE LENGTH'][i] = average_sentence_length
    url_out['PERCENTAGE OF COMPLEX WORDS'][i] = percentage_complexwords
    url_out['FOG INDEX'][i] = fog_index
    url_out['COMPLEX WORD COUNT'][i] = num_complexword
    url_out['AVG NUMBER OF WORDS PER SENTENCE'][i] = average_word_per_sentence
    url_out['WORD COUNT'][i] = num_words
    url_out['SYLLABLE PER WORD'][i] = syllable_per_word
    url_out['PERSONAL PRONOUNS'][i] = personal_pronouns
    url_out['AVG WORD LENGTH'][i] = avg_word_length
```

->Processes of the data extraction

[86]:

```
def tokenize(text):
    text = re.sub(r'[^A-Za-z]',' ',text.upper())
    tokenized_words = word_tokenize(text)
    return tokenized_words

def remove_stopwords(words, stop_words):
    return [x for x in words if x not in stop_words]

def countfunc(store, words):
    score = 0
    for x in words:
        if(x in store):
            score = score+1
    return score

def sentiment(score):
    if(score < -0.5):
        return 'Most Negative'
    elif(score >= -0.5 and score < 0):
        return 'Negative'
    elif(score == 0):
        return 'Neutral'
    elif(score > 0 and score < 0.5):
        return 'Positive'
    else:
        return 'Very Positive'


def polarity(positive_score, negative_score):
    return (positive_score - negative_score)/((positive_score + negative_score)+ 0.000001)


def subjectivity(positive_score, negative_score, num_words):
    return (positive_score+negative_score)/(num_words+ 0.000001)

def syllable_morethan2(word):
    count =0
    if(len(word) > 2 and (word[-2:] == 'es' or word[-2:] == 'ed')):
        return False,count

    vowels = ['a','e','i','o','u']
    for i in word:
        if(i.lower() in vowels):
            count = count +1

    if(count > 2):
        return True,count
    else:
        return False,count

def fog_index_cal(average_sentence_length, percentage_complexwords):
    return 0.4*(average_sentence_length + percentage_complexwords)


def pronouns(word):
    p=[ 'I,','we,','my,','ours,','us']
    for i in p:
        if word==i:
            return True
    return False

def average_word(word):
    add=0
    for i in word:
        add+=len(word)
    return add
```

PREPARED DATA

```
In [88]: url_out=url_out.set_index("URL_ID")
         url_out
```

Out[88]:

URL_ID	URL	POSITIVE SCORE	NEGATIVE SCORE	POLARITY SCORE	SUBJECTIVITY SCORE	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER O WORDS PER SENTENC
1	https://insights.blackcoffer.com/is-telehealth...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
2	https://insights.blackcoffer.com/how-telehealt...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
3	https://insights.blackcoffer.com/is-telemedici...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
4	https://insights.blackcoffer.com/is-telehealth...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
5	https://insights.blackcoffer.com/how-people-di...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
...	...	...	...	...	...	...	...	...	.
146	https://insights.blackcoffer.com/blockchain-fo...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
147	https://insights.blackcoffer.com/the-future-of...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
148	https://insights.blackcoffer.com/big-data-anal...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
149	https://insights.blackcoffer.com/business-anal...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363
150	https://insights.blackcoffer.com/challenges-an...	0.0	0.0	0.0	0.0	15.863636	0.262655	6.450517	15.86363

150 rows × 14 columns

Thank you