

Lab Manual
For
WEB AND SOCIAL MEDIA
ANALYTICS LAB
MANUAL



Academic Year: 2023-2024

IV Year B. TECH I SEMESTER (R21 AUTONOMOUS)

DEPARTMENT OF CSE (Data Science)

**KG REDDY COLLEGE OF ENGINEERING
AND TECHNOLOGY**

An Autonomous Institution

1.

Department Vision, Mission, POs and PSOs

Institute Vision:

To be a leading Technical Institute to prepare high quality Engineers to cater the needs of the stakeholders in the field of Engineering and Technology with global competence fundamental comprehensive analytical skills, critical reasoning, research aptitude, entrepreneur skills, ethical values and social concern.

Institute Mission:

Imparting Quality Technical Education to young Engineers by providing the state-of-the-art laboratories, quality instructions by qualified and experienced faculty and research facilities to meet the requirements of stakeholders in real time usage and in training them to excel in competitive examinations for higher education and employment to interface globally emerging techno-informative challenges in the growth corridor of techno-excellence.

Department Vision:

To produce excellent standard, quality education of professionals by imparting cognitive learning environment, research and industrial orientation to become innovative Data Science Professional.

Department Mission:

M1: To develop professionals in the areas of math's (probability and statistics, linear algebra and

Calculus), natural language processing, text mining, and problem solving.

M2: To educate the students with latest technologies to update their knowledge in the field of AI and Data science.

M3: To impart quality and value-based education and contribute towards the innovation of computing system, data science to raise satisfaction level of all stakeholders.

M4: Enabling students to get expertise in critical skills with data science education and facilitates socially responsive research and innovation.

M5: Our effort is to apply new advancements in high performance computing hardware and software.

Program Educational Objectives (PEOs)

PEO 1: To prepare the students for successful careers in Computer Science and Engineering and fulfill the need by providing training to excel in competitive examinations for higher education and employment.

PEO 2: To provide students a broad-based curriculum with a firm foundation in Computer Science and Engineering, Applied Mathematics & Sciences. To impart high quality technical skills for designing, modeling, analyzing and critical problem solving with global competence.

PEO 3: To inculcate professional, social, ethical, effective communication skills and entrepreneurial practice among their holistic growth.

PEO 4: To provide Computer Science and Engineering students with an academic environment and members associated with student related to professional bodies for multi-disciplinary approach and for lifelong learning.

PEO 5: To develop research aptitude among the students in order to carry out research in cutting edge technologies, solve real world problems and provide technical consultancy services.

Program Outcomes

Program Outcomes	Statement
PO1	An ability to apply knowledge of mathematics, science, and engineering and knowledge of Fundamental Principles.
PO2	An ability to Identify, formulate and solve engineering problems.
PO3	An ability to design a system, component, or process to meet desired needs in Computer Science and Engineering within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability and sustainability, Design and Modeling.
PO4	An ability to design and conduct experiments, as well as to analyze and interpret data, Experimentation & Interpret/Engineering Analysis.
PO5	An ability to use the techniques, skills and modern Computer Science and Engineering tools necessary for system design with embedded engineering practice.
PO6	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7	The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.
PO8	An understanding of professional and ethical responsibility.
PO9	An ability to function on multidisciplinary teams.
PO10	An ability to communicate effectively.
PO11	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	A recognition of the need for, and an ability to engage in life-long learning.

Program Specific Outcomes

Program Specific Outcomes	Statement
PSO1	To prepare the students ready for industry usage by providing required training in cutting edge technologies.
PSO2	An Ability to use the core concepts of computing and optimization techniques to develop more efficient and effective computing mechanisms.
PSO3	Prepare the graduates to demonstrate a sense of societal and ethical responsibility In their professional endeavors and will remain informed and involved as full participants in the profession and our society.

WEB AND SOCIAL MEDIA ANALYTICS LAB

Index

S. No.	List of Experiments
1	Preprocessing text document using NLTK of Python a. Stopword elimination b. Stemming c. Lemmatization d. POS tagging e. Lexical analysis
2	Sentiment analysis on customer review on products
3	Web analytics a. Web usage data (web server log data, clickstream analysis) b. Hyperlink data
4	Search engine optimization- implement spamdexing
5	Use Google analytics tools to implement the following a. Conversion Statistics b. Visitor Profiles
6	Use Google analytics tools to implement the Traffic Sources

1.Pre-processing text document using NLTK of Python

a. Stop word elimination

Stop word elimination is a common text pre-processing technique used in natural language processing to remove common words that do not carry much meaning in a sentence. These words are called stop words.

To perform stop word elimination using NLTK in Python, you can follow these steps:

1.Import the NLTK library and download the stop words corpus:

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

2.Import the stopwords and create a set of them:

```
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
```

3. Tokenize the text into words:

```
from nltk.tokenize import word_tokenize
text = "This is an example sentence for stopword elimination"
words = word_tokenize(text)
```

4. Remove the stop words from the list of words:

```
filtered_words = [word for word in words if word.lower() not in stop_words]
```

complete code

```
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
stop_words = set(stopwords.words('english'))
text = "This is an example sentence for stopword elimination"
words = word_tokenize(text)
filtered_words = [word for word in words if word.lower() not in stop_words]
print(filtered_words)
```

OUTPUT

```
['example', 'sentence', 'stopword', 'elimination']
```

As you can see, the stop words "this", "is", "an", and "for" have been removed from the list of words.

Stopword elimination is a common text pre-processing technique used in natural language processing to remove common words that do not carry much meaning in a sentence. These words are called stop words.

b.Stemming

```
import nltk
nltk.download('punkt')
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
stemmer = PorterStemmer()
text = "I am eating berries"
words = word_tokenize(text)
stemmed_words = [stemmer.stem(word) for word in words]
print(stemmed_words)
```

output

```
['i', 'am', 'eat', 'berri']
```

As you can see, the words "eating" and "berries" have been reduced to their base form "eat" and "berri" using stemming. However, the word "am" remains unchanged as it is not a variant of any other word.

c.Lemmatization

Lemmatization is a text preprocessing technique used in natural language processing to reduce words to their base or root form, similar to stemming. However, unlike stemming, lemmatization takes into account the context and meaning of the words to produce a more accurate base form.

To perform lemmatization using NLTK in Python, you can use the WordNet Lemmatizer algorithm. Here's an example:

```
import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
nltk.download('wordnet')
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize
from nltk import pos_tag
from nltk.corpus import wordnet
lemmatizer = WordNetLemmatizer()
def get_wordnet_pos(tag):
    if tag.startswith('J'):
        return wordnet.ADJ
    elif tag.startswith('V'):
        return wordnet.VERB
    elif tag.startswith('N'):
        return wordnet.NOUN
    elif tag.startswith('R'):
        return wordnet.ADV
    else:
        return wordnet.NOUN
text = "I am eating berries"
words = word_tokenize(text)
tagged_words = pos_tag(words)
lemmatized_words = [lemmatizer.lemmatize(word, pos=get_wordnet_pos(tag)) for word, tag in
tagged_words]
print(lemmatized_words)
```

output

```
['I', 'be', 'eat', 'berry']
```

As you can see, the words "eating" and "berries" have been correctly reduced to "eat" and "berry" respectively using lemmatization. The word "am" has been changed to "be" to reflect its base form as a verb.

d. POS tagging

POS tagging (Part-of-Speech tagging) is a process of marking each word in a given text with its corresponding part of speech, based on both its definition and its context in the sentence.

```
import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
from nltk.tokenize import word_tokenize
from nltk import pos_tag
text = "I am eating berries"
words = word_tokenize(text)
tagged_words = pos_tag(words)
print(tagged_words)
```

OUTPUT

```
[('I', 'PRP'), ('am', 'VBP'), ('eating', 'VBG'), ('berries', 'NNS')]
```

As you can see, each word in the text has been tagged with its corresponding part of speech. The POS tags used by NLTK are as follows:

- CC Coordinating conjunction
- CD Cardinal digit
- DT Determiner
- EX Existential there
- FW Foreign word
- IN Preposition or subordinating conjunction
- JJ Adjective
- JJR Adjective, comparative
- JJS Adjective, superlative
- LS List item marker
- MD Modal
- NN Noun, singular or mass
- NNS Noun, plural

NNP Proper noun, singular
NNPS Proper noun, plural
PDT Predeterminer
POS Possessive ending
PRP Personal pronoun
PRP\$ Possessive pronoun
RB Adverb
RBR Adverb, comparative
RBS Adverb, superlative
RP Particle
SYM Symbol
TO to
UH Interjection
VB Verb, base form
VBD Verb, past tense
VBG Verb, gerund or present participle
VBN Verb, past participle
VBP Verb, non-3rd person singular present
VBZ Verb, 3rd person singular present
WDT Wh-determiner
WP Wh-pronoun
WP\$ Possessive wh-pronoun
WRB Wh-adverb

In the example above, "I" is tagged as a personal pronoun (PRP), "am" as a verb, non-3rd person singular present (VBP), "eating" as a verb, gerund or present participle (VBG), and "berries" as a noun, plural (NNS).

e. Lexical analysis

Lexical analysis, also known as tokenization, is the process of breaking a given text into individual words or tokens. In NLTK, you can use the `word_tokenize()` function from the `nltk.tokenize` module to perform lexical analysis.

Here's an example of how to perform lexical analysis in NLTK:

```
import nltk
from nltk.tokenize import word_tokenize
text = "This is an example sentence for lexical analysis."
tokens = word_tokenize(text)
```

```
print(tokens)
```

OUTPUT

```
['This', 'is', 'an', 'example', 'sentence', 'for', 'lexical', 'analysis', '.']
```

As you can see, the text has been broken down into individual tokens, which are individual words in the text. The output of the `word_tokenize()` function is a list of tokens.

Lexical analysis is an important step in natural language processing, as it is often the first step in processing text data. It is also useful for tasks such as counting the frequency of words in a text, finding the most common words, and so on.

2. Sentiment analysis on customer review on products

Sentiment analysis is a technique used to automatically identify the sentiment (positive, negative, or neutral) expressed in a piece of text, such as customer reviews. Sentiment analysis on customer reviews can be helpful for companies to understand how their customers feel about their products, services, or brand, and to identify areas where they can improve.

Install and import the necessary libraries:

```
!pip install nltk
```

```
!pip install vaderSentiment
```

```
import nltk
```

```
from nltk.sentiment.vader import SentimentIntensityAnalyzer
```

Load the customer review data: `import pandas as pd`

```
reviews_df = pd.read_csv('customer_reviews.csv')
```

```
reviews_text = reviews_df['review_text']
```

Initialize the `SentimentIntensityAnalyzer`:

```
sia = SentimentIntensityAnalyzer()
```

Iterate over each customer review, and use the `SentimentIntensityAnalyzer` to obtain a sentiment score for each review:

```
for review in reviews_text:
```

```
    sentiment_score = sia.polarity_scores(review)
```

```
    print(sentiment_score)
```

The `polarity_scores()` function returns a dictionary containing four scores: a positive score, a negative score, a neutral score, and a compound score. The compound score is a normalized score ranging from -1 (most negative) to 1 (most positive).

Here's an example of what the output might look like:

```
{'neg': 0.143, 'neu': 0.523, 'pos': 0.333, 'compound': 0.4404}
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.2732}
{'neg': 0.0, 'neu': 0.517, 'pos': 0.483, 'compound': 0.8316}
{'neg': 0.126, 'neu': 0.469, 'pos': 0.405, 'compound': 0.6486}
```

As you can see, each customer review has been assigned a sentiment score. The compound score can be used as an overall indicator of the sentiment expressed in the review, with higher values indicating more positive sentiment and lower values indicating more negative sentiment.

Note that the accuracy of sentiment analysis can be affected by the quality of the text data, as well as the complexity of the language used in the reviews. Therefore, it is important to carefully evaluate the results of sentiment analysis and consider them in context.

3.Web analytics

a. Web usage data (web server log data)

Web Server Log Data

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, classification_report

# Read log data from CSV file
log_data = pd.read_csv('web_server_logs.csv')

# Preprocessing: Extract features from log data
X = log_data['request']
y = log_data['status']

# Feature extraction: Convert request text into numerical features
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(X)

# Split data into training and testing sets
X_train, X_test = X[:8000], X[8000:]
y_train, y_test = y[:8000], y[8000:]

# Train a Naive Bayes classifier
clf = MultinomialNB()
clf.fit(X_train, y_train)
```

```

# Predict on test data
y_pred = clf.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
classification_report = classification_report(y_test, y_pred)

# Print results
print("Accuracy: ", accuracy)
print("Classification Report:\n", classification_report)

```

output:

```

Accuracy: 0.82
Classification Report:

```

	precision	recall	f1-score	support
200	0.83	0.96	0.89	1576
404	0.81	0.58	0.68	424
accuracy			0.82	2000
macro avg	0.82	0.77	0.78	2000
weighted avg	0.82	0.82	0.81	2000

Hyperlink data

```

import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, classification_report
from sklearn.model_selection import train_test_split

# Read hyperlink data from CSV file
hyperlink_data = pd.read_csv('hyperlink_dataset.csv')

# Preprocessing: Extract features from hyperlink data
X = hyperlink_data['source_url']
y = hyperlink_data['target_url']

# Feature extraction: Convert source_url text into numerical features
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(X)

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train a Naive Bayes classifier
clf = MultinomialNB()
clf.fit(X_train, y_train)

# Predict on test data

```

```
y_pred = clf.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
classification_report = classification_report(y_test, y_pred)

# Print results
print("Accuracy: ", accuracy)
print("Classification Report:\n", classification_report)
```

CLICKSTREAM ANALYSIS

Clickstream Analysis using Apache Spark and Apache Kafka

Clickstream analysis is the process of collecting, analyzing, and reporting about which web pages a user visits, and can offer useful information about the usage characteristics of a website. Some popular use cases for clickstream analysis include:

A/B Testing: Statistically study how users of a web site are affected by changes from version A to B. [Read more](#)

Recommendation generation on shopping portals: Click patterns of users of a shopping portal website, indicate how a user was influenced into buying something. This information can be used as a recommendation generation for future such patterns of clicks.

Targeted advertisement: Similar to recommendation generation, but tracking user clicks "across websites" and using that information to target advertisement in real-time.

Trending topics: Clickstream can be used to study or report trending topics in real time. For a particular time quantum, display top items that gets the highest number of user clicks.

In this Code Pattern, we will demonstrate how to detect real-time trending topics on the Wikipedia web site. To perform this task, Apache Kafka will be used as a message queue, and the Apache Spark structured streaming engine will be used to perform the analytics. This combination is well known for its usability, high throughput and low-latency characteristics.

When you complete this Code Pattern, you will understand how to:

- Use Jupyter Notebooks to load, visualize, and analyze data
- Run streaming analytics interactively using Notebooks in IBM Watson Studio

- Interactively develop clickstream analysis using Apache Spark Structured Streaming on a Spark Shell
 - Build a low-latency processing stream utilizing Apache Kafka.
1. User connects with Apache Kafka service and sets up a running instance of a clickstream.
 2. Run a Jupyter Notebook in IBM's Watson Studio that interacts with the underlying Apache Spark service. Alternatively, this can be done locally by running the Spark Shell.
 3. The Spark service reads and processes data from the Kafka service.
 4. Processed Kafka data is relayed back to the user via the Jupyter Notebook (or console sink if running locally).

Install Spark and Kafka

Install by downloading and extracting a binary distribution from Apache Kafka (0.10.2.1 is the recommended version) and Apache Spark 2.2.0 on your system.

2. Setup and run a simulated clickstream

NOTE: These steps can be skipped if you already have a clickstream available for processing. If so, create and stream data to the topic named 'clicks' before proceeding to the next step.

Use the following steps to setup a simulation clickstream that uses data from an external publisher:

Download and extract the Wikipedia Clickstream. Select any data set, the set 2017_01_en_clickstream.tsv.gz was used for this Code Pattern.

Create and run a local Kafka service instance by following the instructions listed in the Kafka Quickstart Documentation. Be sure to create a topic named clicks.

The Kafka distribution comes with a handy command line utility for uploading data to the Kafka service. To process the simulated Wikipedia data, run the following commands:

NOTE: Replace ip:port with the correct values of the running Kafka service, which is defaulted to localhost:9092 when running locally.

```
cd kafka_2.10-0.10.2.1
```

```
tail -200 data/2017_01_en_clickstream.tsv | bin/kafka-console-producer.sh --broker-list ip:port --topic clicks --producer.config=config/producer.properties
```

TIP: Unix head or tail utilities can be used for conveniently specifying the range of rows to be sent for simulating the clickstream.

3. Run the script

Go to the Spark install directory and bootstrap the Spark shell specifying the correct versions of Spark and Kafka:

```
cd $SPARK_DIR
```

```
bin/spark-shell --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.2.0
```

In the spark shell prompt, specify the schema of the incoming wikipedia clickstream and parse method:

TIP: For conveniently copying and pasting commands into the spark shell, spark-shell supports a `:paste mode*`

```
scala> import scala.util.Try
```

```
scala> case class Click(prev: String, curr: String, link: String, n: Long)
```

```
scala> def parseVal(x: Array[Byte]): Option[Click] = {  
    val split: Array[String] = new Predef.String(x).split("\\t")  
    if (split.length == 4) {  
        Try(Click(split(0), split(1), split(2), split(3).toLong)).toOption  
    } else  
        None  
}
```

Setup structured streaming to read from Kafka:

NOTE: Replace ip:port with the correct values of ip and port of the running Kafka service, which is defaulted to localhost:9092 when running locally.

```
scala> val records = spark.readStream.format("kafka")  
    .option("subscribe", "clicks")  
    .option("failOnDataLoss", "false")  
    .option("kafka.bootstrap.servers", "ip:port").load()
```

Process the records:

```
scala>  
    val messages = records.select("value").as[Array[Byte]]  
    .flatMap(x => parseVal(x))  
    .groupBy("curr")  
    .agg(Map("n" -> "sum"))  
    .sort($"sum(n)".desc)
```

Output to the console and start streaming data (using the tail clickstream command described above):

```
val query = messages.writeStream
```

```
    .outputMode("complete")
```

```
    .option("truncate", "false")
```

```
    .format("console")
```

```
    .start()
```

```
scala> .....
```

```
Batch: 0
```

```
+.....+
|curr                |sum(n) |
+.....+
|Gavin_Rossdale      |1205584|
|Unbreakable_(film)  |1100870|
|Ben_Affleck         |939473 |
|Jacqueline_Kennedy_Onassis |926204 |
|Tom_Cruise         |743553 |
|Jackie_Chan         |625123 |
|George_Washington  |622800 |
|Bill_Belichick      |557286 |
|Mary,_Queen_of_Scots |547621 |
|The_Man_in_the_High_Castle |529446 |
|Clint_Eastwood      |526275 |
|Beyoncé            |513177 |
|United_States_presidential_line_of_succession|490999 |
|Sherlock_Holmes     |477874 |
|Winona_Ryder        |449984 |
```

Titanic_(1997_film)	400197
Watergate_scandal	381000
Jessica_Biel	379224
Patrick_Swayze	373626
+.....+.....+	

only showing top 20 rows

The resultant table shows the Wikipedia pages that had the most hits. This table updates automatically whenever more data arrives from Kafka. Unless specified otherwise, structured streaming performs processing as soon as it sees any data.

Here we assume the higher number of clicks indicates a "Hot topic" or "Trending topic". Please feel free to contribute any ideas on how to improve this, or thoughts on any other types of clickstream analytics that can be done.

4.Search engine optimization- implement spamdexing

SEO as it relates to Google because Google will likely account for the vast majority of your inbound search traffic. Additionally, if you rank highly on Google, you will probably do well on other search engines anyway. Just like in football if you could play in the Major league you would most likely kill it in the minor league I would begin by explaining why SEO is important talk a little bit about what SEO is about and talk about how those concepts relate to the world wide web. Then talk about some of the things you could do to optimize your site from top to bottom of a typical webpage.

Why it's important

When you want to hide something on google put it on the second page. #SEO @searchdecoder

Winner takes almost everything

More than 80 percent of shoppers research big purchases online first

Opportunity for business

88% Of Consumers Trust Online Reviews As Much As Personal Recommendations

72% Of Consumers Say That Positive Reviews Make Them Trust A Local Business More

93% of online experiences begin with a search engine.

70% of the links search users click on are organic

75% of users never scroll past the first page of search results

Search is the #1 driver of traffic to content sites, beating social media by more than 300%

Understanding SEO

Search Engine Optimization SEO is the ongoing strategy of generating content and making pages more accessible for people to find on the web. SEO as a librarian

Page structure

It imports for you to write well-structured code, addresses with in an

tag articles with in tags so that search engine spiders could have a better understanding of what content on the page represent.

What should be in your header

Title

Page titles have a huge impact on SEO ranking and should be different for every page on your site. Title tags—technically called title elements—define the title of a document. Title tags are often used on search engine results pages (SERPs) to display preview snippets for a given page and are important both for SEO and social sharing.

The title element of a web page is meant to be an accurate and concise description of a page's content. This element is critical to both user experience and search engine optimization. It creates value in three specific areas: relevancy, browsing, and in the search engine results pages.

```
<title>Example Title</title>
```

Optimal Format

Primary Keyword - Secondary Keyword | Brand Name

Optimal Length for Search Engines Google typically displays the first 50-60 characters

Meta Description

The meta description is a HTML attributes that provide concise explanations of the contents of web pages. Meta descriptions are commonly used on search engine result pages (SERPs) to display preview snippets for a given page.

```
<meta name="description" content="This is an example of a meta description. This will often show up in search results.">
```

Meta description tags, while not important to search engine rankings, are extremely important in gaining user click-through from SERPs. These short paragraphs are a webmaster's opportunity to advertise content to searchers and to let them know exactly whether the given page contains the information they're looking for.

The meta description should employ the keywords intelligently, but also create a compelling description that a searcher will want to click. Direct relevance to the page and uniqueness between each page's meta description is key.

Optimal Length for Search Engines

Roughly 155 Characters

```
<meta name="keywords" content="HTML,CSS,XML,JavaScript">
```

Author

Google also allows you to place your name next to specific website you have created on the web. By adding the small snippet of code below to the header of your page and adding the link to the site to the “Contributors” section on your Google+ profile.

```
<link rel="author" href="https://plus.google.com/u/0/109859280204979591787/posts"/>
```

In a similar manner as a search algorithm may recommend a book by an established author so too Google’s search algorithm would recommend web pages/site/applications higher if they were witter by an establish author. Check out this website for more detailed infomation on how to implement the rel author tag to your page.

PRO TIP: Referencing the author of the web page as a Google Plus user is a good way to have establish a firm authority.

Improve speed

Okay this is a big one Google algorithm for ranking websites at it core is trying to sever up the website to search that both most closely meets your search query and provides you with the best possible users experience. That’s the short reason why slow sites rank low on Google.

Here are some tools you can use to test site speed and list of a few things you could do to improve profomance.

Use css instead of images anywhere you can

Take advantace of SVGs

Compless images

Use CDNs

Use less CSS and JS scripts

Minify css and js files

Use SVGs were possible

Use font icons where possible

Create image sprite

Micro Data

Search giants Google, Bing and Yahoo announced last summer a rare collaboration to support the use of microdata tagging to generate more relevant and more detailed search results. This offers business owners and other website publishers another opportunity to

improve their search engine optimization (SEO) by making a few changes to their websites.

<http://schema.org/>

<https://www.google.com/webmasters/markup-helper/u/0/>

<http://builtvisible.com/micro-data-schema-org-guide-generating-rich-snippets/>

Ask Google to Crawl your site

Because Google periodically crawls website based on the frequency of updates to that site eg. new sites and/or blogs that post regularly frequently have new content to be indexed. As a result of this Google spider usually finds new content every we it visits particular pages/sites. If that is not the case and content is not frequently updated you may want to ask Google to crawl your site after you make a large number of content changes or you have updated your SEO strategy. To do this click on the following link.

-- Google Web Master tools

Naming files and images

Search engings are getting smarter and smarter but they are still not smart enough to crawl images on your site for that you will have to give it some help. For that reason when it comes to SEO, it's important to chose image and file names carefully, use keywords to help your webpage rank on search engines. Creating descriptive, keyword-rich file names is absolutely crucial for image optimization.

Social Media

Twitter (Adding a Twitter card to your website)

Card Types

Summary Card: Default Card, including a title, description, thumbnail, and Twitter account attribution.

Summary Card with Large Image: Similar to a Summary Card, but with a prominently featured image.

Photo Card: A Card with a photo only.

Gallery Card: A Card highlighting a collection of four photos.

App Card: A Card to detail a mobile app with direct download.

Player Card: A Card to provide video/audio/media.

Product Card: A Card optimized for product information.

Twitter Cards Sample Code

Facebook

```
<meta property="og:title" content="The Rock" />
```

```
<meta property="og:type" content="video.movie" />
```

```
<meta property="og:url" content="http://www.imdb.com/title/tt0117500/" />
```

```
<meta property="og:image" content="http://ia.media-imdb.com/images/rock.jpg" />
```

The Open Graph protocol enables any web page to become a rich object in a social graph. For instance, this is used on Facebook to allow any web page to have the same functionality as any other object on Facebook.

Facebook Open Graph protocol

SEO MOZ Social media meta data templates

Pinterest

Product

```
<meta property="og:title" content="Name of your product" />
```

```
<meta property="og:type" content="product" />
```

```
<meta property="og:price:amount" content="1.00" />
```

```
<meta property="og:price:currency" content="USD" />
```

Article

```
<meta property="og:title" content="Title of your Article" />
```

```
<meta property="og:description" content="Description of what your article" />
```

```
<meta property="og:type" content="article" />
```

Recipe

Pinterest recipe rich pin documentation

Movie

Pinterest movie rich pin documentation

--Rich Pins Overview

Google Plus

```
<body itemscope itemtype="http://schema.org/Product">
```

```
<h1 itemprop="name">Shiny Trinket</h1>
```

```

```

<p itemprop="description">Shiny trinkets are shiny.</p>

</body>

Google Plus Snippets

Anchor tags

Back links used to describe your sites the more descriptive the better SEO is for you.
Descriptive links, title, alt.

em Tags

Header tags

h1 - Should be one page page

h2 - Subheader

h3

Keywords and keywords optimisation

Web page keywords should not be stuffed that is equivalent to spamming

Developing a good content strategy

Semantic HTML

Easy to read for both the programmer and the web crawler

5. Use Google analytics tools to implement the following

a. Conversion Statistics

The Conversion Tag Audit Tool is a Chrome extension that crawls a website and generates a gTag (Google Analytics, Google Ads and Floodlight event tags) report by monitoring network traffic from the page.

In this document, we will be outlining the installation, base functionality, features and way to use the Conversion Tag Audit Tool that may come up in most use cases.

1. Installation

Clone this repository using `git clone https://github.com/google/conversion-audit.git` or `git clone git@github.com:google/conversion-audit.git` if SSH has been setup. More details on how to set it up here -> Setup SSH. The code can also be downloaded as ZIP using the green Code button at the top.

In order to load the Chrome extension correctly, all the Angular files need to be built into bundles in a single folder with the required HTML, JS, CSS and manifest files that will

used by the extension. In case the tool is used out of the box, a dist folder has already been generated under the app folder. This is the folder that will be loaded in the Chrome Extensions page.

Open a Chrome browser window, navigate the extensions management page by browsing to: `chrome://extensions/`

On the top right of the page flip the "Developer Mode" switch to on.

At the top of the page, on the left, click the "Load Unpacked Extension ..." button.

Go to the app folder and then select the dist folder.

The tool should now be installed, and a new icon should show in the extensions toolbar on the top right corner of Chrome.

Finally click the icon in the extension toolbar to open the tool.

If the extension doesn't work due to chrome extensions restrictions in your organization you may need to generate a key, follow instructions here:
<https://developer.chrome.com/apps/manifest/key>

2. Installation for Developers

Follow these steps in case the tool has had some changes that want to be incorporated:

Open a terminal and check if npm is installed using `npm --version`. If not, install it following the instructions in the official npm docs [Install npm](#).

Check if the Angular CLI is installed using `ng --version`. If not, install it following the instructions in the official Angular docs [Install the Angular CLI](#).

Clone this repository using `git clone https://github.com/google/conversion-audit.git` or `git clone git@github.com:google/conversion-audit.git` if SSH has been setup. More details on how to set it up: [Setup SSH](#).

Once cloned, go to the app folder using `cd app` and install the required dependencies using `npm install`.

Then, in order to load the Chrome extension correctly, all the Angular files need to be built into bundles in a single folder with the required HTML, JS, CSS and manifest files that will be used by the extension. To build the files execute the build script `./build.sh`. After this, a new dist folder will be created. This is the folder that will be loaded in the Chrome Extensions page.

Once the files are built, the console will show something like this:

- ✓ Browser application bundle generation complete.
- ✓ Copying assets complete.
- ✓ Index html generation complete.

NOTE: If for some reason the code is changed, it needs to be rebuilt and reloaded in Chrome again to identify the changes.

Open a Chrome browser window, navigate the extensions management page by browsing to: `chrome://extensions/`

On the top right of the page flip the "Developer Mode" switch to on.

At the top of the page, on the left, click the "Load Unpacked Extension ..." button.

Select the dist folder created when the source code was built.

The tool should now be installed, and a new icon should show in the extensions toolbar on the top right corner of Chrome.

Finally click the icon in the extension toolbar to open the tool.

If the extension doesn't work due to chrome extensions restrictions in your organization you may need to generate a key, follow instructions here:
<https://developer.chrome.com/apps/manifest/key>

Add a new "key" field in manifest.json and set the value to your key.

3. User Interface

In this section we are going to outline the functionality of each element within the Settings panel.

Domain - Displays the top level domain for the website in the tab the tool was open in.

Depth (optional) - Determines how deep in the web page directory path you wish for the tool to scrape from the root domain

Load Time (seconds) (optional) - This setting determines how long the tool allows for a page to load before moving onto the next page. *It is critically important if using the tool in automated mode to choose a page load time that would be inclusive of when Google tags fire or use a load time that aligns with typical user navigation time.

URL Suffix - Optional field to add URL suffix to URL string

Enable Manual Mode - (defaults to off) - If checked, the tool will run the audit in manual mode meaning that it will not automatically visit and scrape web pages. Instead it will sit back passively and record any floodlight activity as the user navigates through the website on their Chrome tab. This allows a user to audit particular pages, completing actions (button click, sign up, test purchase) to record activity based.

Enable Global Site Tag Verification - (defaults to off) - If checked, it will enable the feature to capture Global Site Tag and cookie information on each visited page (compatible with manual and default automatic mode) which will be displayed in a separate table similar to the floodlight table.

Show Page with No Conversion Tags - (defaults to off) - If checked, tells the tool to add an entry in the Conversion Tag Report table for web pages that were visited and where no conversion tags were captured. If this feature is not activated, by default the tool will only

record entries on pages where conversion tags were present, leaving out pages with no conversion tags.

File Upload - Optional field to upload a csv list of URLs for the tool to crawl (no URL volume limit)

Run Button - Will trigger the audit process once it is clicked. After the first click, it will be replaced by a Stop button which will terminate the audit.

Download Button - Allows the user to download the audit results as a csv file matching the information displayed in the UI. It will download Floodlight results and Global Site Tag (if enabled by user) results as separate CSV files. Can be clicked at any point during the audit process.

4. How to Use It

Navigate to the page from which you want to start with in Chrome, usually the websites home page;

Open the tool by clicking the icon from the chrome toolbar;

The Domain is pre-populated based on the domain on the page from which you started, you can change it to narrow down the pages that should be crawled;

(OPTIONAL) Check “Enable Manual Mode” you wish to run the audit in manual mode. If checked you as the user will need to navigate through the website manually.

(OPTIONAL) Check “Enable Global Site Tag Verification” to enable and record GST and cookie data during the audit.

(OPTIONAL) Check the “Show Pages with No Conversion Tags” in case you want the report to include pages that are visited but do not cause floodlight tags to be fired. This is particularly useful if you want to determine pages that are not being tracked.

Click the Run button, and wait as the crawler starts to visit your site. Note, keep the tool popup open, if you close it by clicking anywhere on Chrome the process will stop, and you will only get a partial report.

Once the crawling is over and the number of pages visited is the same as the number of pages found then the audit will be marked as completed. At this point you can click the Download button to export a CSV version of the final Floodlight and Global Site Tag report (if enabled).

5. Output

Page - URL that was crawled for that result

Tag Type - Floodlight, Google Ads Conversion Tags, Google Analytics Conversion Tags

Account ID - Config ID of the associated Global Site Tag

gTag (Y/N) - Flag to confirm associated gTag was observed*

Network Call - Network call of the observed tag

Floodlight ID - Floodlight Activity ID

Floodlight Activity Tag - Floodlight Activity Tag. "Cat=" Parameter value.

Floodlight Activity Group - Floodlight Activity Group. "Type=" Parameter value

Floodlight Sales Order - Order ID or cachebuster random number, depending on whether the tag in question is a Sales Tag or a Counter Tag

Floodlight uVariables - Custom uVariables associated with the floodlight in question and whether they pulled in values for that Floodlight fire

Warnings - Some warnings (like calling out empty uVariables) may be expected. We are just highlighting this for you to look into if you wish.

Errors - Any implementation errors we observe

6. Notes

*If you are seeing "False" for the "OGT" Column in the Conversion Tag Report section:

Check that the Global Site Tag (gTag) includes the Config ID associated to the conversion tag

Ensure the gTag is implemented properly and is firing immediately on each page. If there is a delay, the output could show pages as not being tagged

Validate that the specific Conversion or Remarketing actions are deployed using GTM or a gTag Event Snippet

Google Analytics calls are captured with google-analytics.com domains. If it is a newer GA4 implementation the calls will not be captured if they are hitting analytics.google.com instead of google-analytics.com/g/collect.

5b. Use Google analytics tools to implement Visitor Profiles

use this widget, you'll first need to set up a Google API project and attach it to the Google Analytics profile you wish to monitor.

1. Create and download a new private key for Google API access.

1. Go to <https://code.google.com/apis/console>
2. Click 'Create Project'
3. Enable 'Analytics API' service and accept both TOS's
4. Click 'API Access' in the left-hand nav menu
5. Click 'Create an OAuth 2.0 Client ID'
6. Enter a product name (e.g. Dashing Widget) - logo and url are optional
7. Click 'Next'
8. Under Application Type, select 'Service Account'
9. Click 'Create Client ID'
10. Click 'Download private key' NOTE: This will be your only opportunity to download this key.

11. Note the password for your new private key ('notasecret')
12. Close the download key dialog
13. Find the details for the service account you just created and copy its email address which will look something like this: 210987654321-3rmagherd99kitt3h5@developer.gserviceaccount.com - you'll need it in your ruby code later

2. Attach your Google API service account to your Google Analytics profile

Note: you will need to be an administrator of the Google Analytics profile

1. Log in to your Google Analytics account: <http://www.google.com/analytics/>
2. Click 'Admin' in the upper-right corner
3. Select the account containing the profile you wish to use
4. Select the property containing the profile you wish to use
5. Select the profile you wish to use
6. Click the 'Users' tab
7. Click '+ New User'
8. Enter the email address you copied from step 13 above
9. Click 'Add User'

3. Locate the ID for your Google Analytics profile

1. On your Google Analytics profile page, click the 'Profile Settings' tab
2. Under 'General Information' copy your Profile ID (e.g. 654321) - you'll need it in your ruby code later

4. Start coding (finally)

1. Copy the visitor_count.rb file in to your dashing jobs\ folder.
2. Update the service_account_email, key_file, key_secret and profileID variables

```
service_account_email = '[YOUR SERVICE ACCOUNT EMAIL]' # Email of service account
key_file = 'path/to/your/keyfile.p12' # File containing your private key
key_secret = 'notasecret' # Password to unlock private key
profileID = '[YOUR PROFILE ID]' # Analytics profile ID.
```

3. Add the widget HTML to your dashboard

```
<li data-row="1" data-col="1" data-size="1" data-sizey="1">
  <div data-id="visitor_count" data-view="Number" data-title="Visitors This Month"></div>
</li>
```

Notes

If you want to modify this plugin to pull other data from Google Analytics, be sure to check out the Google Analytics Query Explorer.

6. Use Google analytics tools to implement the Traffic Sources.

Measuring your social media traffic will help you determine which marketing tactics are working for you and which are coming up short. The traffic that comes from Facebook, Twitter, LinkedIn, YouTube, or other social media sources funnels into content on your site and then triggers some sort of completion such as a lead, a purchase, or whatever you're trying to accomplish with that traffic.

Your social media traffic will come from both paid and unpaid sources. To illustrate, Facebook traffic can come from paid ads, shared posts from your page, and maybe even posts from a group. The same can be true with Twitter, LinkedIn, and YouTube.

Social media traffic paid and unpaid sources graphic.

You can also look at social media traffic on a more granular level. On YouTube, for instance, traffic might come from specific areas of the site such as cards, the backend, or description links.

You want to measure how all of this social media traffic converts into content and ultimately into your completion goal. You can do that with Google Analytics and UTMs.

#1: View Data About Your Social Media Traffic in Google Analytics

The Source/Medium report in Google Analytics is where you'll find all of the relevant details about your social media traffic. In this one report, you can see the identity of each traffic source, how much of an audience you're getting from that source, how that audience is engaging with your site, and the results of those actions.

Here's how to get started using this report.

Access the Source/Medium Report

To access the report, open Google Analytics and go to Acquisition > All Traffic > Source/Medium.

Open Google Analytics and go to Acquisition > All Traffic > Source/Medium.

Scroll down the page to see the list of traffic sources for your site. This data is divided into several different sections. For this walk-through of the report, we'll look at some data from the Google Merchandise Store demo account.

Unlock Your Social Marketing Superpowers! 🚀

Social Media Marketing Society

Feeling overwhelmed by the ever-changing world of social marketing? Say hello to the Social Media Marketing Society! 🎯

Dive into exclusive live and on-demand expert led training on Instagram, YouTube, Facebook, TikTok, LinkedIn, content marketing, video creation, Google Analytics, and more.

Effortlessly stay ahead of the social marketing game and deliver outstanding results with ease. Join the Society today and become the marketing superhero for your company or clients! ✨

JOIN THE SOCIETY TODAY

The far-left column of the Source/Medium report identifies the traffic source and the medium. You can think of the “source” as the brand of the traffic that's coming through and the “medium” as the type of traffic.

To visualize this, the first traffic source listed below is google/organic. In this case, Google is the brand of traffic and organic is the type of traffic. For google/cpc, the traffic also comes from Google and the type of traffic is CPC, which is paid traffic.

For google/organic, Google is the brand of traffic and organic is the type of traffic.

The next part of the report, Acquisition, tells you about the quantity of traffic from that source. You can see the number of users, new users, and sessions.

In the Acquisition section of the Source Medium report, you can see the number of users, new users, and sessions.

The third section, Behavior, tells you about the actions people are taking. You can see the bounce rate, pages per session, and average session duration for this audience.

In the Behavior section of the Source Medium report, you can see the bounce rate, pages per session, and average session duration for this audience.

Looking at the Acquisition and Behavior data together will give you an idea of the quality of the traffic from that source. For instance, you may have a source that drives a ton of traffic to your site, but those users don't take the actions you want or leave quickly. And you might also have a source that doesn't send you a ton of traffic but those users really engage with your message and your content. That second source is a little higher quality.

The last section of the Source/Medium report shows you the results. If you've set up goals in Google Analytics to measure actions like leads or purchases, this is where you can see those results. Select one of your goals from the drop-down menu to compare traffic sources for different results.

Select one of your goals from the drop-down menu to compare traffic sources for different results.

Analyze the Data in the Report

Now that you're familiar with what's in the report, let's look at how to analyze this data. When you review the data, don't get caught up in the numbers. Instead, look for trends.

If you look at the Behavior data below, you can see that the traffic sources with the lowest bounce rates are [mall.googleplex/referral](#) (11.05%) and [sites.google.com/referral](#) (13.31%). This data indicates the audiences from those two sources are more engaged than the audiences from the other sources.

The same two traffic sources also stand out from the others in pages per session and average session duration. These audiences viewed more pages on average during a session (8.28 and 6.58, respectively), and spent more time on the site (4:28 and 4:13, respectively).

When you review the data in the Source Medium report, look for trends.

Now that you've determined the audiences from these two sources are really engaged, you need to find out if that translates to results. On the ecommerce side, you can see that mall.googleplex had 93 transactions for a total of \$8,839, but sites.google.com had only 2 transactions for a total of \$248.

While the engagement levels from the two sources are similar, the first source sent you 93 transactions and the second source only 2. That tells you the second source isn't working as well for you as the first one. If that first source was Facebook, and the second source was YouTube, you'd want to put more of your efforts toward Facebook.

While the engagement levels from the two sources are similar, the first source sent you 93 transactions and the second source only 2.

Now that you have a general understanding of how to use this report in Google Analytics, you're ready to start tagging your own traffic.

#2: Track Your Social Media Traffic Sources With UTMs

UTM parameters are tags that you add to the links you share on social media so you can get more detailed information about your traffic in Google Analytics.

Tagging your links with UTM parameters lets you determine which source of social media traffic brings the most visitors to your site, what pages or content they're interested in, and even more details such as how much they purchase, what they do after they purchase, where they drop off your funnel, and more.

UTM parameters in URL.

Suppose you have a Facebook campaign and use multiple ads to send visitors to the same piece of content on your site. To determine which ad gets the most clicks, it's easy to look at the analytics from your Facebook account to determine this metric. However, which ad gets you the most page views after the initial click? Which ad turns the clicks into subscribers or customers?

Google Analytics can show you this information if you tag your traffic. When it comes to tagging, think of the structure like this:

Product/service: The product or service you're ultimately promoting or sending traffic to
Brand: The brand of traffic you're using (Facebook, YouTube, Twitter, etc.)
Type: The type of traffic that brand provides, such as paid or shared traffic, or organic
Headline: The headline (or the subject line if it's an email)
Details: The details about the traffic source
Structure of UTM tag graphic.

To understand how this structure translates to your social media marketing efforts, let's look at a Facebook ad example. Here are the details for this ad:

Tools Resource Guide

Looking for something to make your life easier?

Discover the tools we recommend to drive engagement, save you time, and boost sales across your entire marketing funnel or business.

Whether you need help planning content, organizing social posts, or developing your strategy, you'll find something for every situation.

FIND YOUR NEXT FAVORITE TOOL

Product/service: Measurement Marketing Academy

Brand: Facebook

Type: Paid

Headline: "Know Your Numbers"

Details: Retargeting blog readers – laptop image

Structure of UTM tag for Facebook paid campaign.

You want to include the "laptop image" identifier in the details because you're testing different images in otherwise identical Facebook ads and want to see the results of using the different images in your ads in Google Analytics. Tagging your traffic this way allows you to see details about a specific ad and what type of actions people take after clicking that ad.

So how do these details translate to UTMs? Your product or service is the "campaign," the brand is the "source," the type of traffic is the "medium," the headline is the "term," and the details are the "content."

Structure of UTM tag graphic.

To track this information in Google Analytics, you add UTM parameters to your links:

The campaign (your product/service) becomes utm_campaign.

The source (the brand) becomes utm_source.

The medium (the type of traffic) becomes utm_medium.

The term (the headline) becomes utm_term.

The content (the details) becomes utm_content.

Structure of UTM tag graphic.

For the Facebook ad example, here's how to add the UTM parameters to the link.

First, identify the source (the brand), which is Facebook in this case:

UTM source parameter graphic.

Next, identify the medium (the type of traffic). In this case, you're using CPC, which stands for cost per click:

UTM medium parameter graphic.

Follow this up with the campaign (product/service). It's Measurement Marketing Academy, but we'll use Academy for short:

UTM campaign parameter graphic.

Then add the term (headline/subject), which is Trust Your Numbers:

UTM term parameter graphic.

Finally, provide the content (details). You're retargeting blog readers and using an image of a laptop in the ad, so you write it like this:

UTM content parameter graphic.

Now you need to add these parameters to the link itself. Note that the UTM parameters can be used in any order and only source/medium/campaign are required.

For this example, when users click the Facebook ad, it takes them to the home page at <https://measurementmarketing.io>. That's the main link.

Now add a question mark to the end of the main link and then the individual UTM parameters. Separate each parameter with an ampersand. Here's what the final URL will look like:

Separate each UTM parameter with an ampersand.

Now let's look at how you'd use this URL when you set up the Facebook ad. In Ads Manager, type your main link in the Website URL box.

In Ads Manager, type your main link in the Website URL box.

Then add your tracking parameters (everything after the question mark) to the URL Parameters box.

In Ads Manager, add your tracking parameters (everything after the question mark) to the URL Parameters box.

Now when somebody clicks on your Facebook ad, that information will come through your Google Analytics.

If you open the Source/Medium report, you can see where the traffic is coming from (Facebook), what specific ad it's coming from (the "Trust Your Numbers" ad with the laptop image that's retargeting blog readers), what the users' actions are, how much traffic is being sent from that traffic source, and ultimately what the results of that traffic are.

#3: Create Your Own UTMs With the UTM Builder Tool

The good news is that there's an easier way to create UTMs for your campaigns. The UTM Builder tracking tool will keep your UTMs structured and ensure all of your information is organized and in one place.

To use this method, open the UTM Builder and then choose File > Make a Copy to create your own copy so you'll be able to edit it.

Open the UTM Builder and then choose File > Make a Copy to create your own copy.

On the first tab, UTM Building Tips, you'll find a recap of the UTM information discussed earlier.

On the first tab, UTM Building Tips, you'll find a recap of the UTM information discussed earlier.

To start customizing this sheet, open the Traffic Tag Settings tab to set up your core traffic tag settings. In the Source column, list the “brands” of traffic sources you use (Facebook, YouTube, etc.). In the Medium column, add the types of traffic you use (share, CPC, email, etc.). In the Campaign column, list the products or services you offer.

Open the Traffic Tag Settings tab to set up your core traffic tag settings.

The sources, mediums, and campaigns you list on this tab will show up in drop-down lists on the other tabs of this sheet, as you'll see in a second.

Once you've filled in that information, you're ready to start creating your UTMs. To understand how to use this tracking tool, let's use it to create the UTM for the Facebook ad example from earlier. Start by opening the Facebook-CPC tab.

Open the Facebook-CPC tab of the UTM Builder spreadsheet.

In the URL column on this tab, type in the URL for the ad's landing page. Then in the Source column, select the social media traffic source (Facebook, in this case) from the drop-down list.

Select the social media traffic source (Facebook, in this case) from the drop-down list.

In the Medium and Campaign columns, select the medium (CPC) and campaign (academy) from the drop-down lists.

Here's what your sheet looks like at this point:

Fill in the URL, source, medium, and campaign in the UTM Builder.

Next, type in your term and add the details about your content.

Add your term and content information in the UTM Builder.

As you define the different parameters, the spreadsheet will automatically generate the URL for you in the Code column. Click the code in the spreadsheet to test it and make sure it opens to the correct landing page.

The UTM Builder spreadsheet will automatically generate the URL for you in the Code column.