Data Science Tools and Techniques

ASSIGNMENT 3

Course Instructor: Dr. Mateen Yaqoob

Section: MS DS A

Submitted By:

Valeena Afzal
25I-8023

Contents

Preprocessing Steps

Primary column, content, contained unstructured text typical of web-crawl data. All text was lowercased, punctuation removed, and tokenized using nltk's word_tokenize. Stopwords from the provided file were removed, along with some URL-related words like "http", "www", and "com", which appeared frequently but carried no semantic meaning.

	content	tokens
0	Keyword article mapreduce cloud mapreduce sear...	[keyword, mapreduce, cloud, mapreduce, search,...
1	Navigation crawl learning cloud endpoint click...	[navigation, crawl, learning, cloud, endpoint,...
2	Extract analysis download search analysis tfid...	[extract, analysis, download, search, analysis...
3	Header request vector download nofollow anchor...	[header, request, vector, download, anchor, em...
4	Service endpoint download date request login s...	[service, endpoint, download, date, request, l...
5	Vector json api crawl css cloud article link s...	[vector, json, api, crawl, css, cloud, link, s...
6	Anchor time mapreduce neural cloud data page s...	[anchor, time, mapreduce, neural, cloud, data,...
7	Author stem embed vector mapreduce click heade...	[author, stem, embed, vector, mapreduce, heade...
8	Dataset data post navigation bow error xml sca...	[dataset, data, post, navigation, bow, error, ...
9	Cloud login html crawl view anchor user servic...	[cloud, login, html, crawl, view, anchor, user...

Stemming vs Lemmatization

Stemming (Porter Stemmer) truncated words to their base forms, often producing non-dictionary tokens like comput or analysi. Lemmatization (WordNet Lemmatizer) reduced words to valid lemmas (computing → compute, studies → study). The latter produced cleaner, more interpretable vocabulary and was better suited to web text, which already exhibits irregular grammar.

Bag-of-Words Analysis

A Bag-of-Words model was created using CountVectorizer with min_df = 5 and max_df = 0.85 to filter out overly rare and overly common tokens. The resulting Document-Term Matrix represented the frequency of each token per document. Analyzing the top 50 terms revealed key themes in the dataset. Common words provided an overview of popular topics, their counts helped identify dominant domains and the lexical diversity of the crawl.

	term	count
0	link	77474
1	service	77419
2	header	77136
3	anchor	77106
4	vector	77071
5	scala	77056
6	meta	77005
7	javascript	76974
8	time	76966
9	comment	76956
10	network	76934
11	navigation	76909
12	login	76893
13	request	76892
14	author	76863
15	spark	76863
16	error	76835
17	bow	76828
18	java	76811
19	keyword	76806
20	image	76785
21	endpoint	76760
22	data	76748
23	footer	76735
24	search	76730
25	token	76729
26	video	76727
27	model	76727
28	extract	76720
29	sidebar	76714
30	clean	76679
31	description	76674
32	content	76667
33	snippet	76655
34	hadoop	76650
35	html	76647
36	tokenize	76643
37	response	76633
38	analysis	76630
39	cloud	76604
40	api	76594
41	script	76582
42	tfidf	76549
43	date	76547
44	python	76536
45	stem	76529
46	download	76516
47	page	76512
48	neural	76506
49	post	76461

TF-IDF Results

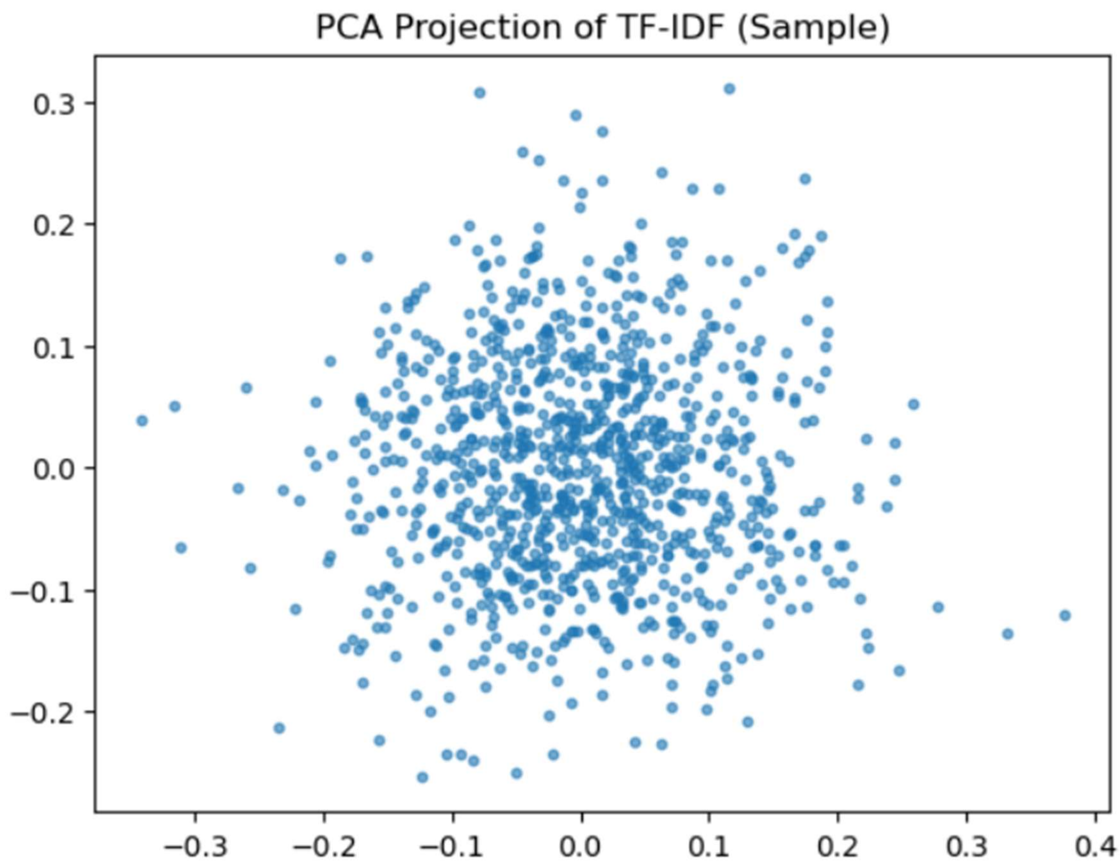
TF-IDF weights were computed to highlight words that were distinctive to specific documents rather than globally frequent. Experiments with K = 50, 100, 200 showed consistent patterns. High-TF-IDF terms were domain-specific, indicating meaningful variation among pages. Files containing top K terms were saved as CSV. While common tokens were widespread, discriminative words carried stronger contextual value for topic identification and clustering.

	term	tfidf
0	link	4697.043837
1	service	4692.703442
2	request	4686.977832
3	scala	4685.039356
4	javascript	4683.881297
5	header	4683.411335
6	navigation	4683.063385
7	java	4681.924199
8	search	4681.245705
9	endpoint	4679.048616

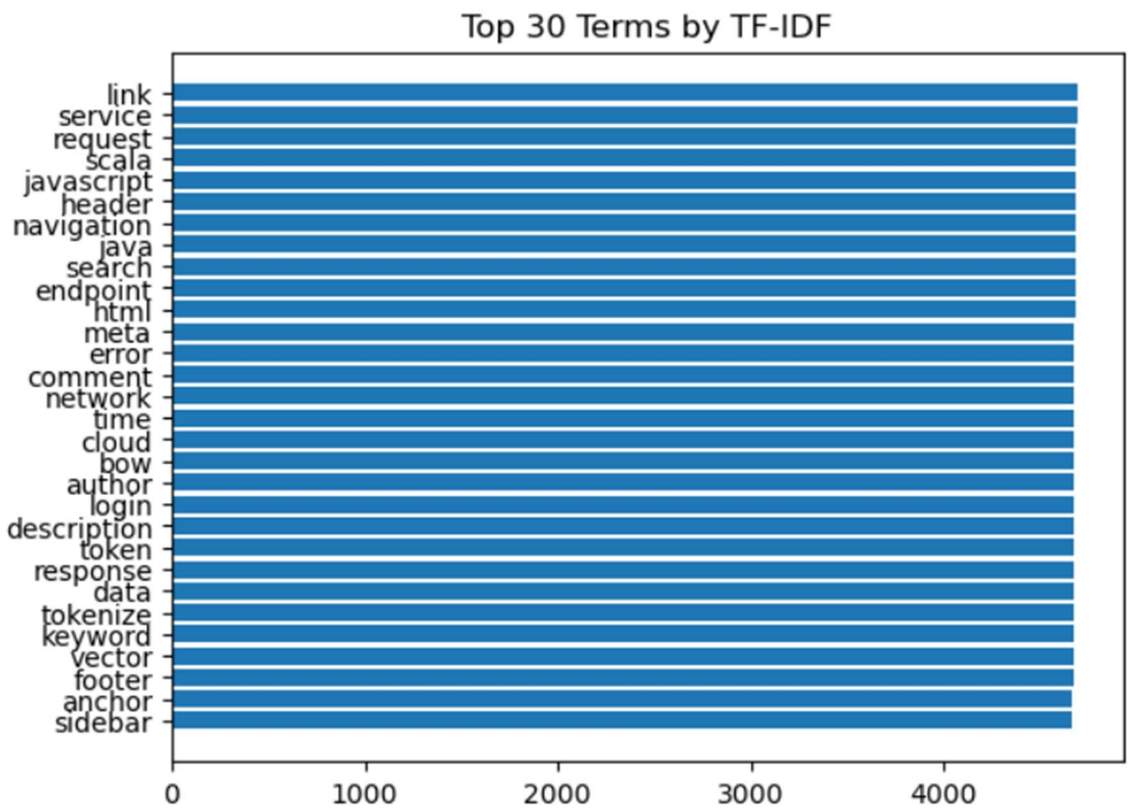
Visualization Insights

PCA demonstrated clear document grouping by textual similarity, confirming that TF-IDF features effectively captured content structure. Cluster map grouped documents with similar word distributions. Bar chart summarized the most relevant terms. Co-occurrence heatmap uncovered latent semantic relationships among top tokens.

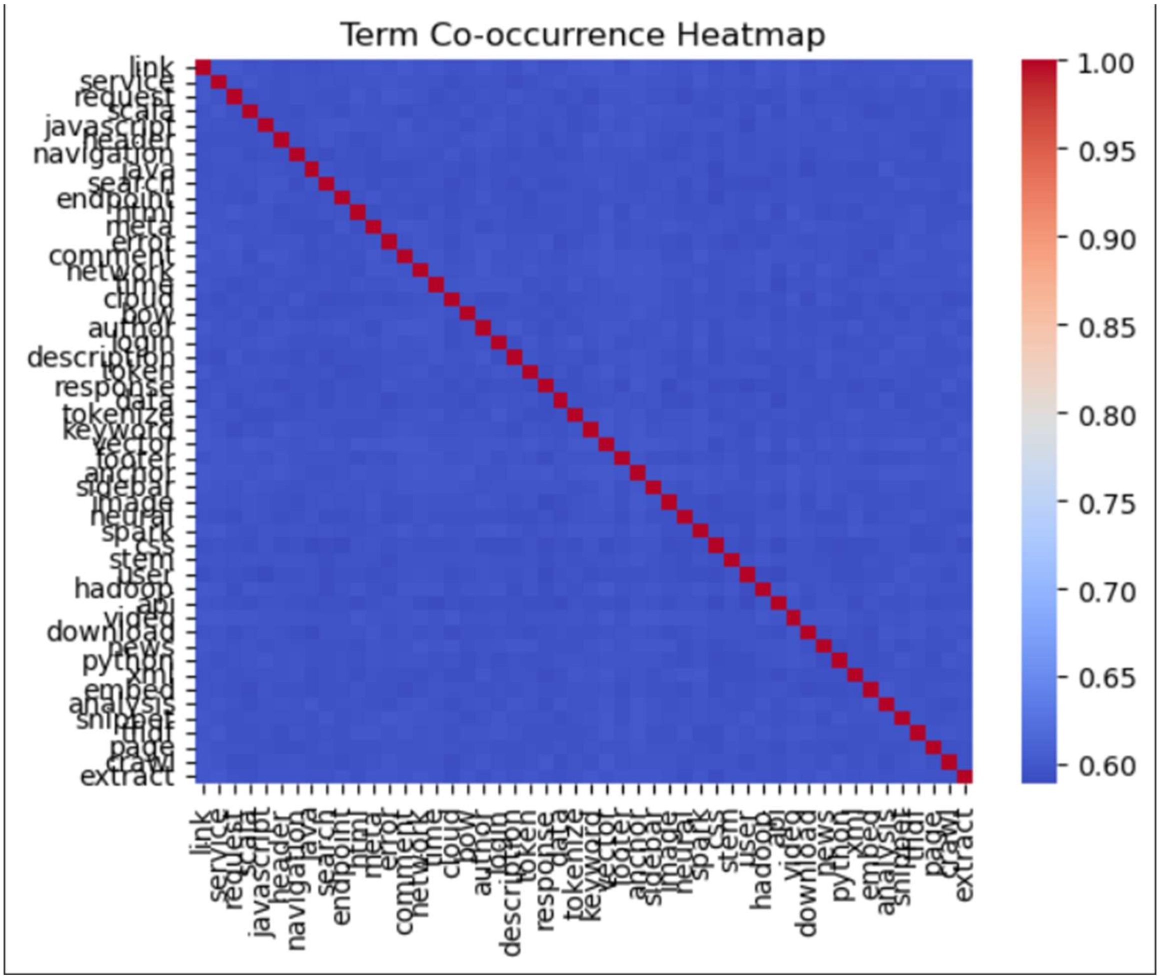
- The **PCA scatter plot** shows how similar or different documents are in 2D form.
    - Each point = one document.
    - Similar documents appear close together.
    - Helps quickly spot groups (topics or types of websites).



- The **hierarchical cluster map** shows how documents relate by word usage.
  - Rows = documents, columns = top 50 words.
  - Darker colors = words used more often.
  - Vertical lines mean some words appear in many pages.
  - Clusters show which docs or words belong to similar topics.
- The **bar chart** of top 30 TF-IDF terms highlights the most important words.
  - Top words = unique or domain-specific (not common words).
  - Shows which subjects dominate (like tech, policy, education).
  - Sharp drop in scores → few words are very common, rest less frequent.



- The **term co-occurrence heatmap** shows how often top words appear together.
    - Bright diagonal = same words matching themselves.
    - Bright off-diagonal spots = related words that appear together.
    - Useful for finding related concepts or topics.



These visuals together help understand Main topics, Important keywords and Relationships between words turning complex TF-IDF data into clear, meaningful insights.