

DA 204o

Data Science in Practice



BIAS Detection in Indian News Media

Presented By:

Girish S N (girishsn@iisc.ac.in)

Bharath Karanth A (bharathk@iisc.ac.in)

Saravanakumar R (saravanakum1@iisc.ac.in)

Anmol Gupta (anmolg@iisc.ac.in)

PROJECT

Overview

Problem Statement and Motivation

- News outlets are often accused of biased reporting across various dimensions.
- With increasing polarization and misinformation, unbiased reporting is critical.
- With the rapid growth of digital news, readers are exposed to vast and diverse information, often with subtle or overt biases.
- Biased reporting can influence public opinion and societal harmony.
- There is a need for an automated, scalable solution that can analyze news articles for bias – offering transparent, multi-dimensional, and interpretable results.

Project Objectives

The goal was to achieve a two-fold objective:

- Firstly, we aim to develop a comprehensive **dataset** of Indian news articles with rich, multi-dimensional features.
 - This is achieved through:
 - Web Scraping of Indian News Articles, across languages and sources
 - Extensive Feature Engineering
- Secondly, we aim to detect, identify, categorize and quantify the **bias** in Indian news articles, across different categories.

Ultimately, this project seeks to provide researchers, fact-checkers, and the public with actionable insights into media bias in Indian journalism.

Project Overview

The goal of this project is to build an end-to-end **bias detection and analysis** pipeline for Indian news media articles.

- **Scrape** articles from multiple news organizations.
- **Preprocess** and **clean** text with utilities tailored to the Indian news context.
- Perform extensive **feature engineering**.
- **Detect** and **quantify** bias across categories – political, gender, religious, caste, regional, socioeconomic, age, disability, language.
- **Aggregate** and **visualize** long-term trends and media-wise patterns.

END-TO-END

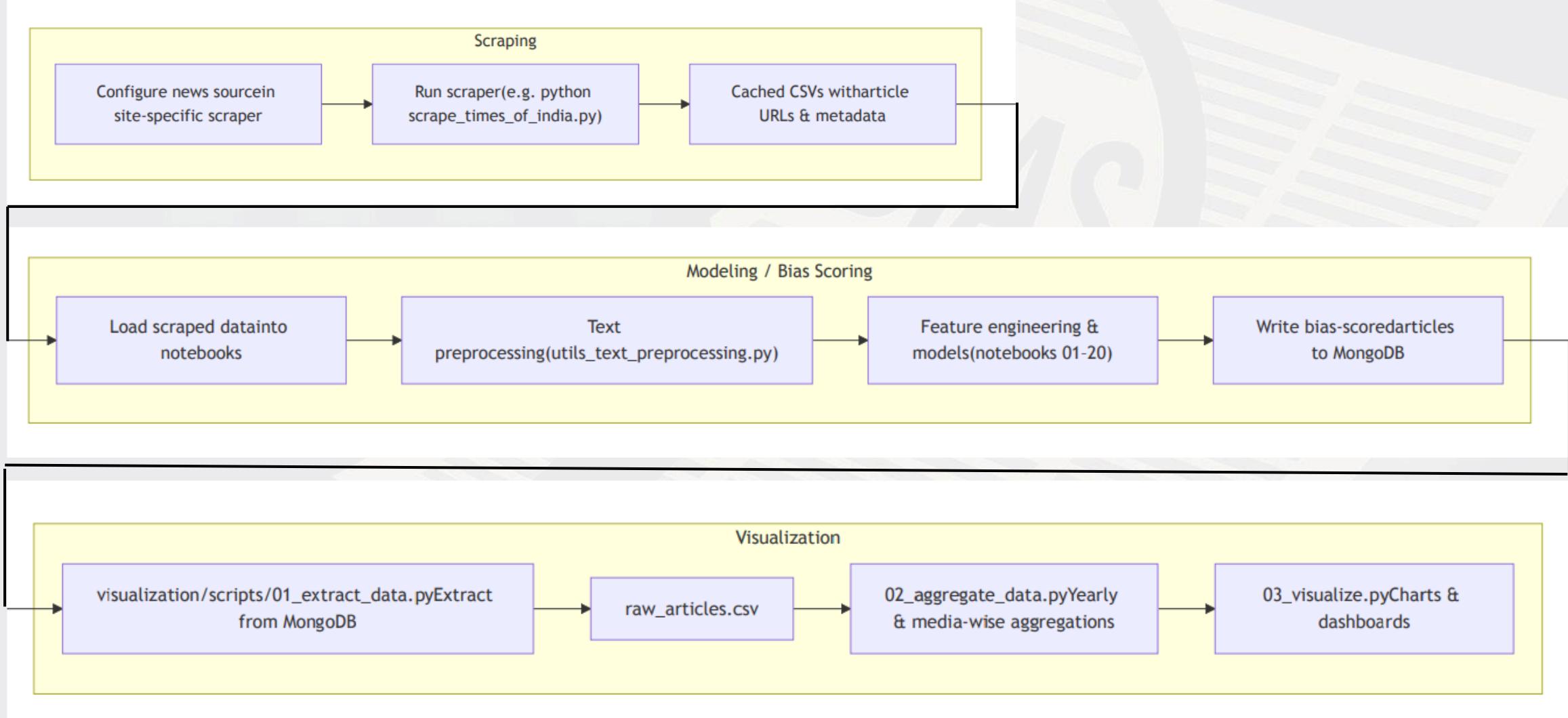
Workflow

Workflow

At the high level, the project workflow spans five phases:

- **Data Collection** – Scraping article archives into CSV files / MongoDB.
- **Data Cleaning** – Performing the pre-processing and cleaning.
- **Translation** – Translating non-English articles into English.
- **Feature Engineering** – Extracting useful features out of the cleaned and translated data.
- **Bias Modeling** – Assigning bias scores across different categories, through various methods and ensemble techniques.
- **Exploratory Visualization** – Using the visualization pipeline to generate trends and results through charts.

End-to-end Workflow Diagram



SCRAPING

Overview

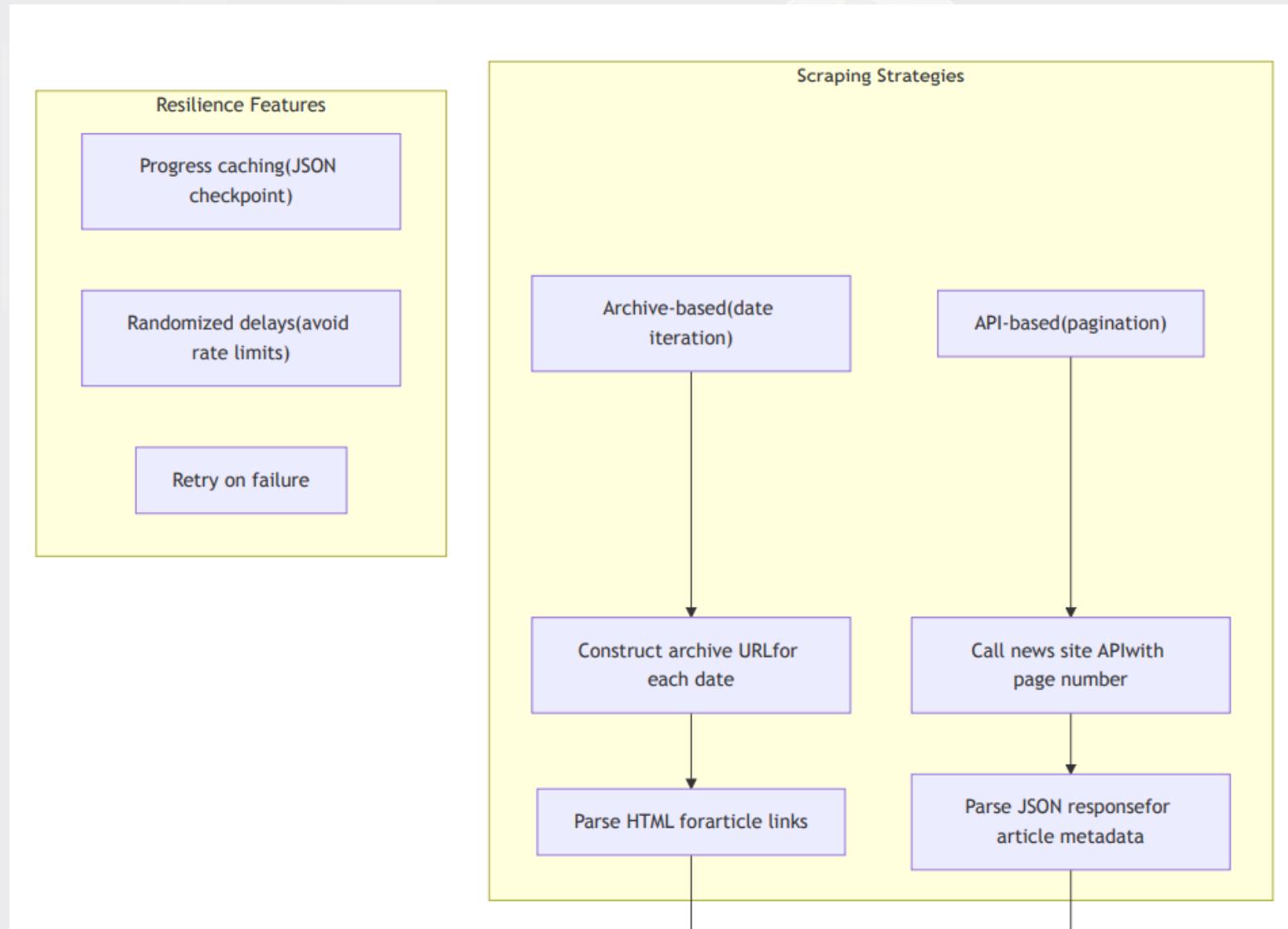
Scraping Overview

- As a first step, raw data was collected through the web scraping of online news articles.
- Multiple Indian news sources were scraped for content; some attempts were successful, while others were not.
- The outcome of the process was the creation of a raw dataset, comprising 3,96,739 articles.

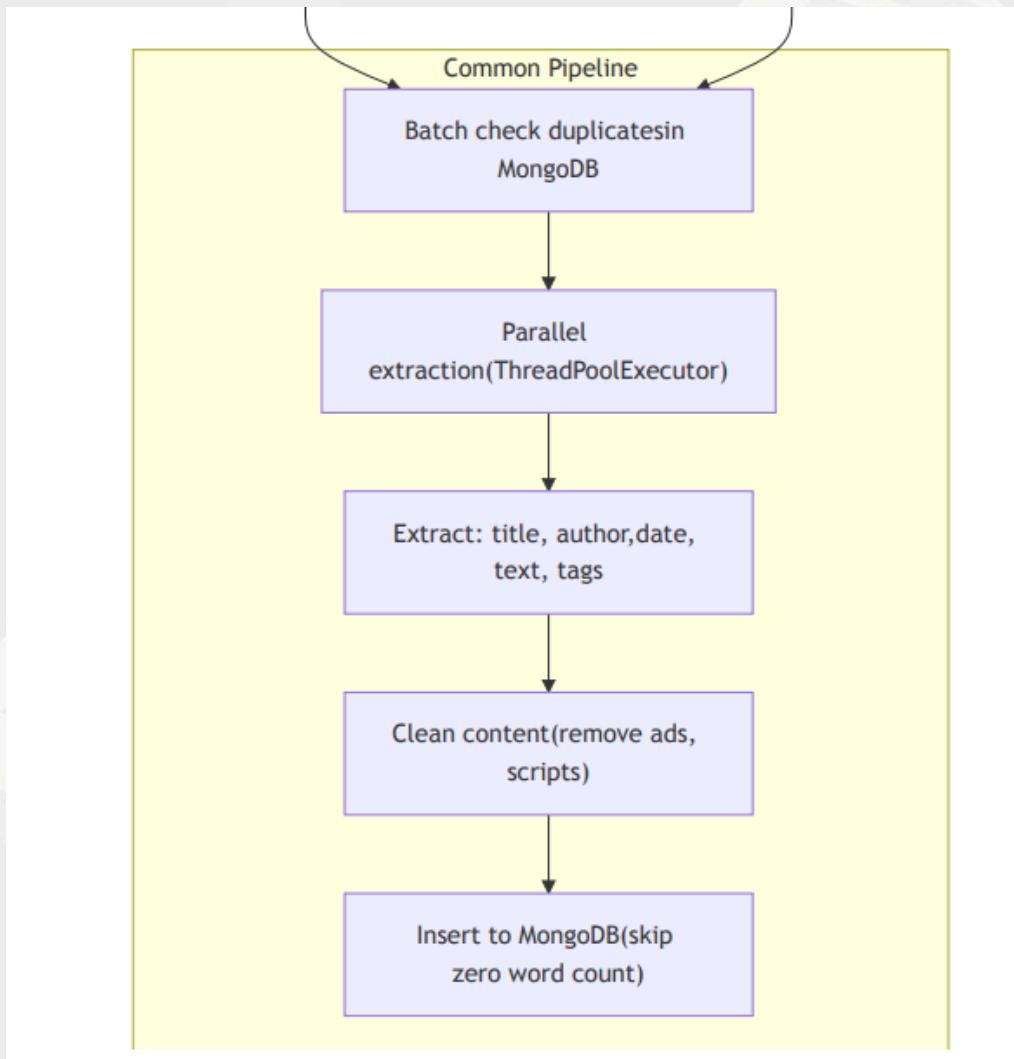
Scraping Statistics

News Source	Language	Articles Scrapped	Time Taken
The Times of India	English	2,023	80 hours *
The Indian Express	English	1,92,050	95 hours
The Economic Times	English	1,38,785	80 hours
News18	English	49,685	30 hours
Dainik Jagran	Hindi	4,268	3 hours
Public TV	Kannada	5,059	4 hours
Dinamalar	Tamil	4,869	3 hours
Total		3,96,739	295 hours

Scraping Architecture



Scraping Architecture



TRANSLATION

Strategy

Translation Overview

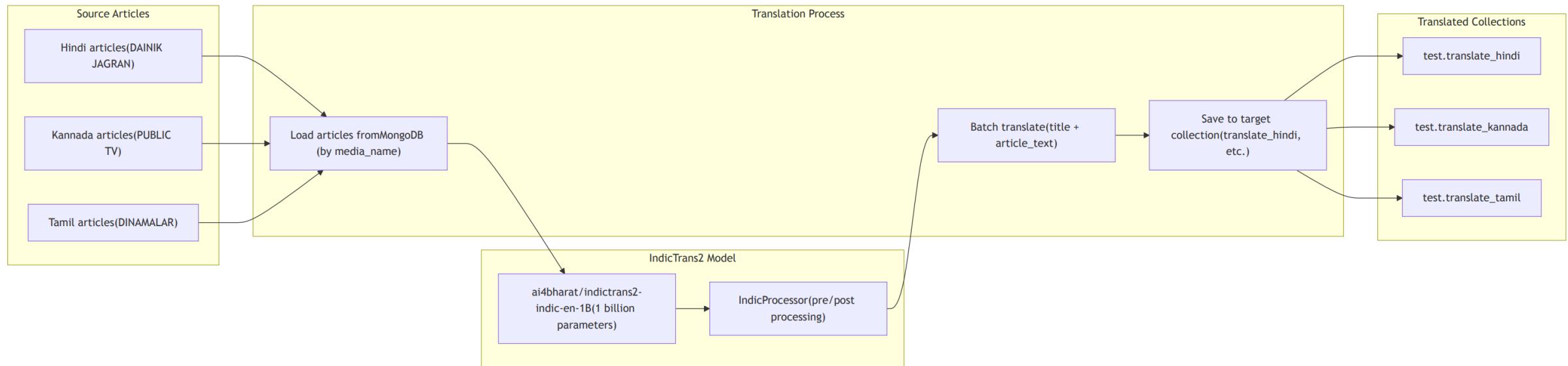
- The non-English articles (Hindi, Kannada, Tamil) were translated to English before further feature engineering and the bias analysis.
- Various models and APIs were tested.

Type	Name	Comment	Accuracy	Source
Model	OPUS-MT Marian	Did not support regional Indian languages	N.A.	Open source
API	LibreTranslate	Did not support regional Indian languages	N.A.	Open source
API	Google Translate	Worked very well for all regional languages tested, but high cost (About \$20 per 1 million characters)	>99%	Proprietary
Model	IndicTrans2	Works well for all regional languages tested	>95%	Hugging Face

Details of the Translation Model Used

Attribute	Value
Model	<code>ai4bharat/indictrans2-indic-en-1B</code>
Parameters	1 billion
Source Languages	22 Indic languages
Target Language	English (<code>eng_Latn</code>)
Toolkit	IndicTransToolkit (<code>IndicProcessor</code>)
Batch Size	2–5 articles
Max Length	384 tokens

Translation Pipeline



FEATURE

Engineering

Feature Engineering

8 feature blocks were added, each with its own number of columns.

- Temporal Features
- Keyword Features
- Linguistic Features
- Discourse Features
- Semantic Features
- NER (Named Entity Recognition) Features
- Intersectionality Features
- Implicit Bias Features

All these feature matrices were merged into a single feature table, successively adding the listed number of columns from each block.

Feature Engineering

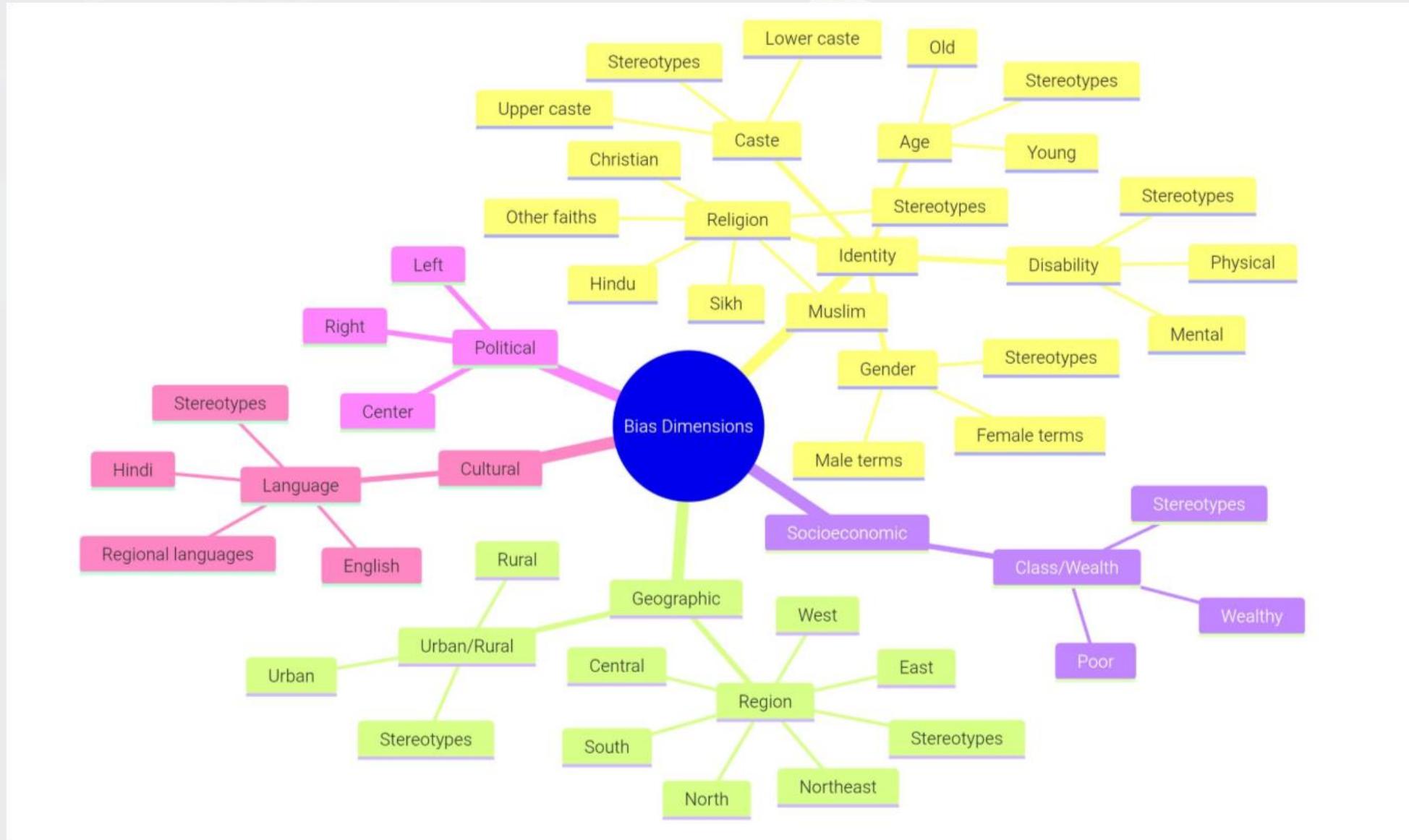
- Temporal features: 44 columns
- Linguistic features: 35 columns
- Keyword-based features: 121 columns
- Discourse features: 30 columns
- Semantic features: 2 columns
- NER features: 22 columns
- Intersectionality-related features: 59 columns
- Implicit bias features: 98 columns

Total combined feature space: **411 columns**

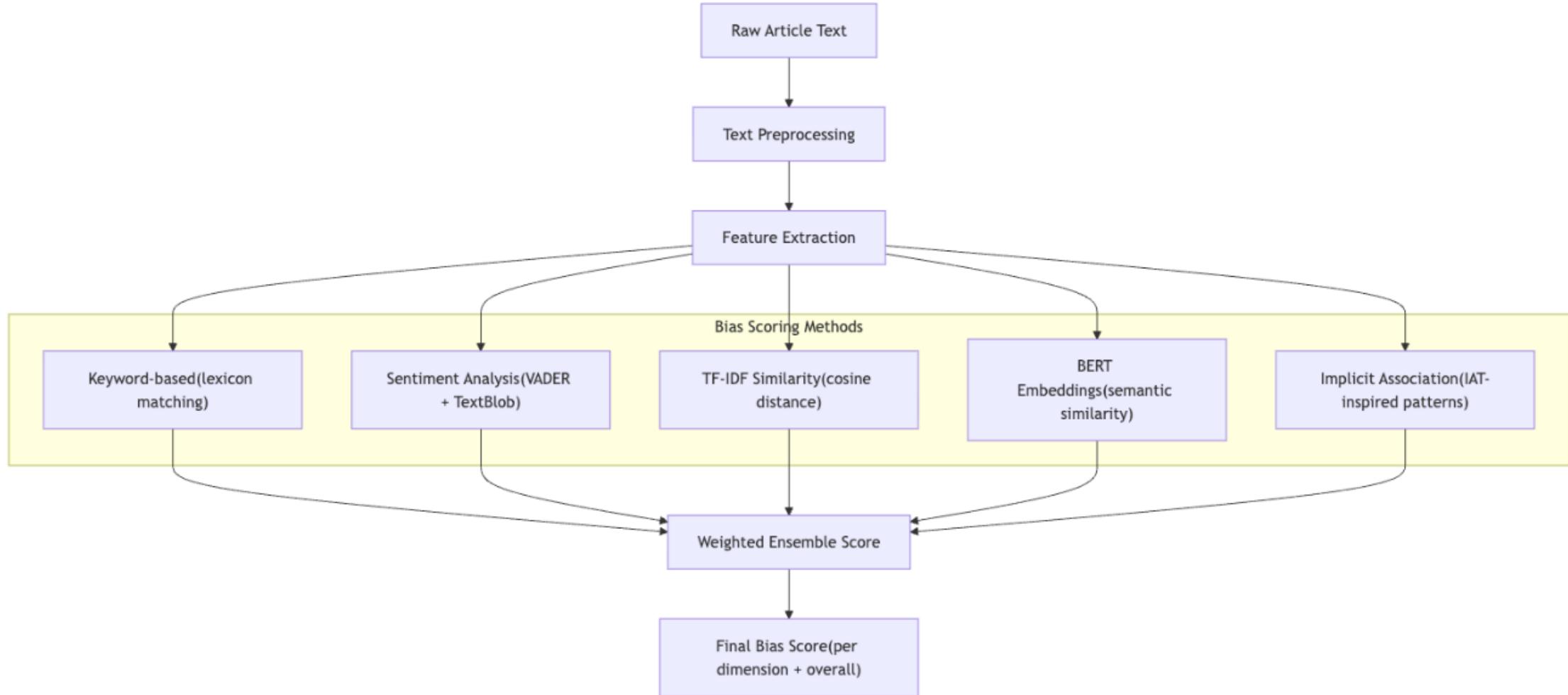
BIAS SCORING

Methodology

Bias Dimension Categories



Bias Scoring Methodology Overview



Bias Scoring Methods

- Keyword Counting
- Stereotype Density
- Co-occurrence of Dimensions
- VADER Sentiment
- TextBlob Subjectivity & Polarity
- Emotion Detection
- Discourse Analysis
- Implicit Association
- ML Ensemble
- Topic Classification
- Ensemble Bias
- Overall Bias Score Aggregation

Overall Bias Calculation

The overall bias score is a weighted average of individual dimension scores:

Dimension	Weight
Gender	0.20
Religious	0.20
Political	0.15
Caste	0.15
Regional	0.15
Socioeconomic	0.15

$$\text{overall_bias_score} = \sum (\text{dimension_score} \times \text{dimension_weight})$$

Here, *dimension_score* is obtained after the ensemble model training, while *dimension_weight* is a tunable parameter.

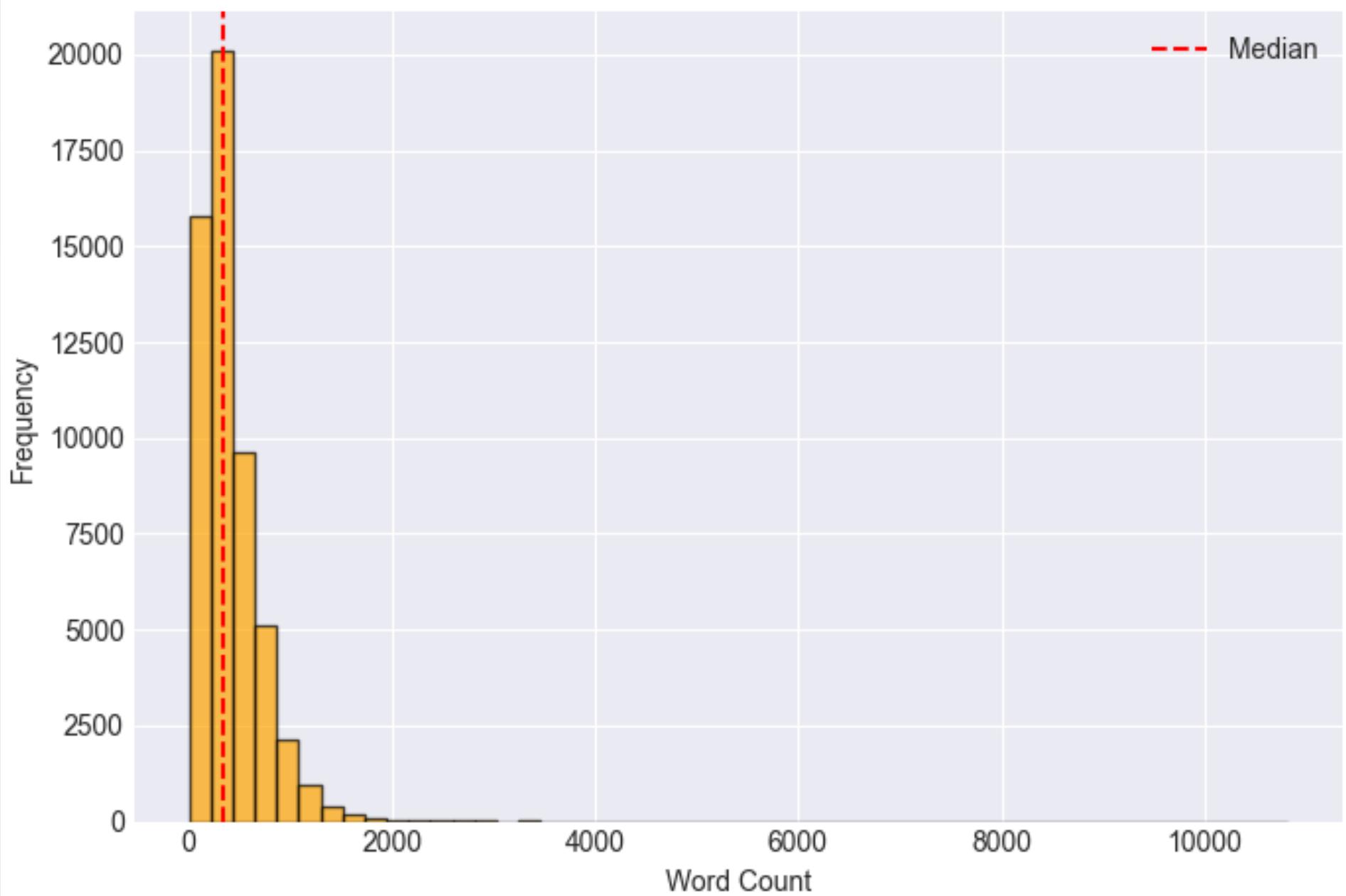
VISUALIZATION

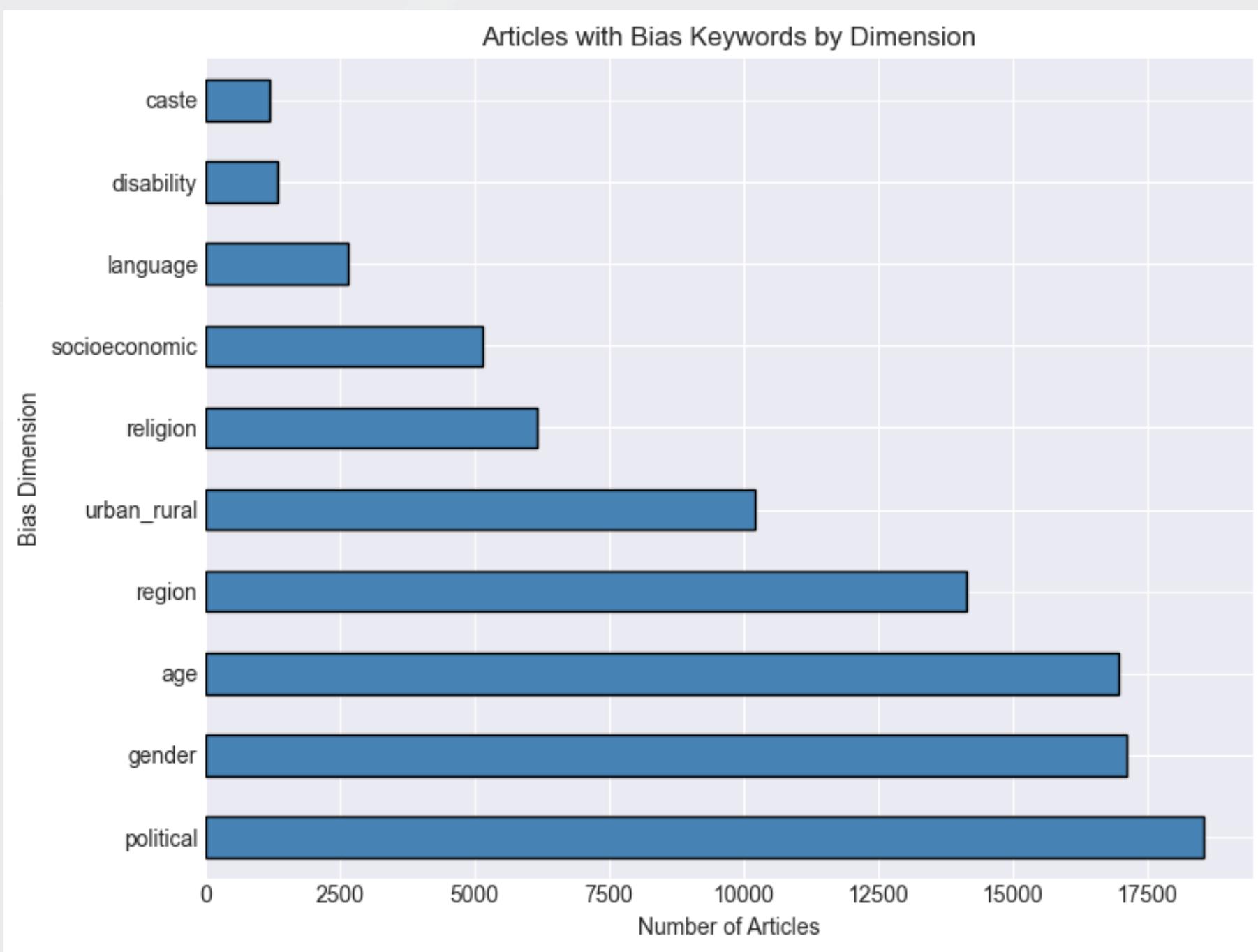
and Key Observations

Some Key Observations

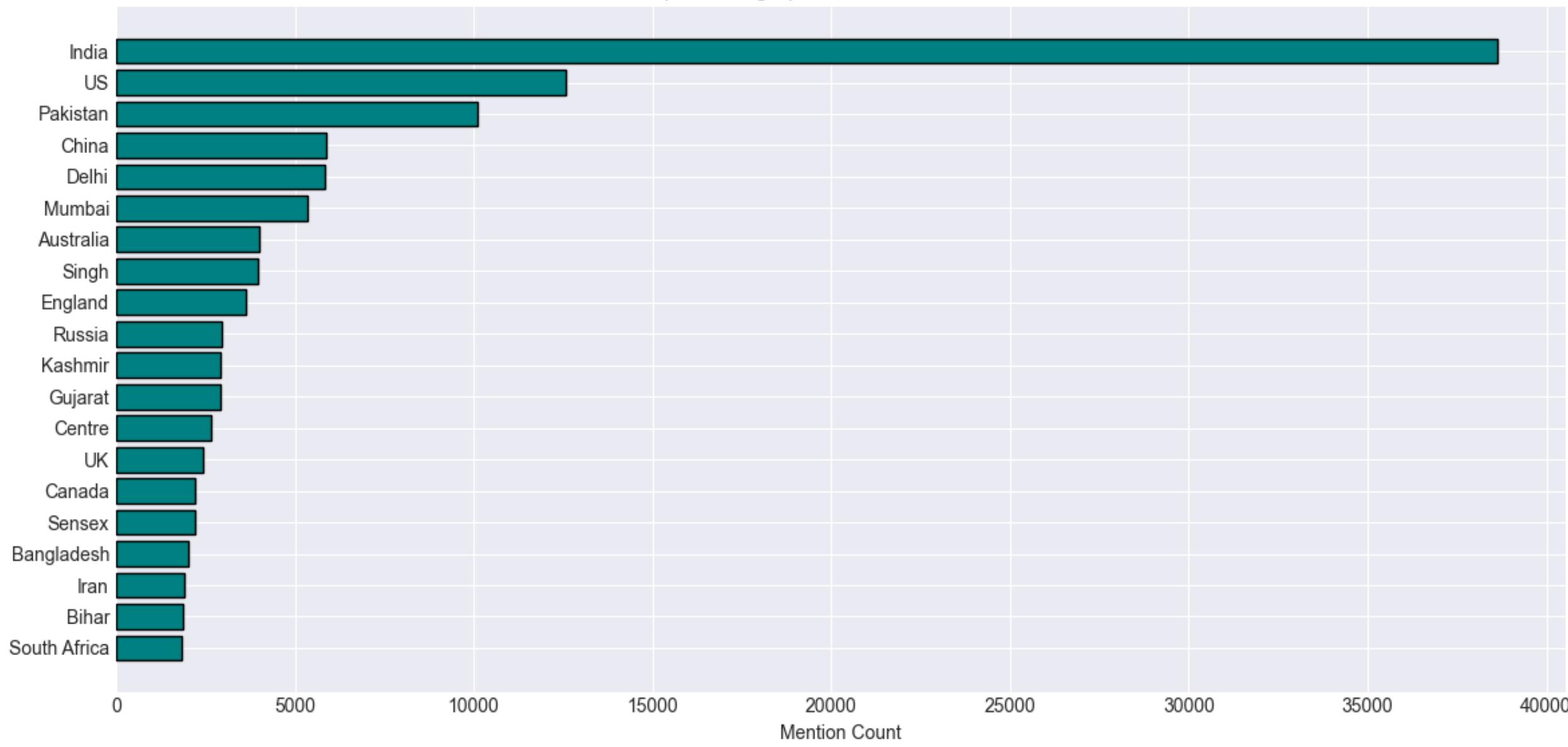
- Across topics and years, bias is persistent – especially in politics and crime – with many articles containing political, gender, age and regional bias keywords, and stereotype density highest for religion, gender and socioeconomic status.
- Coverage heavily centers on India (with the US and Pakistan some way behind), relies much more on authorities than on experts, and repeatedly uses negative implicit frames such as “Elderly Weak”, “Muslim Violence” and “Dalit Poor”.
- Overall sentiment is mostly positive, but representation is unequal: men and older people are mentioned far more than women and youth, and most articles are short-to-medium in length.

Distribution of Article Word Count

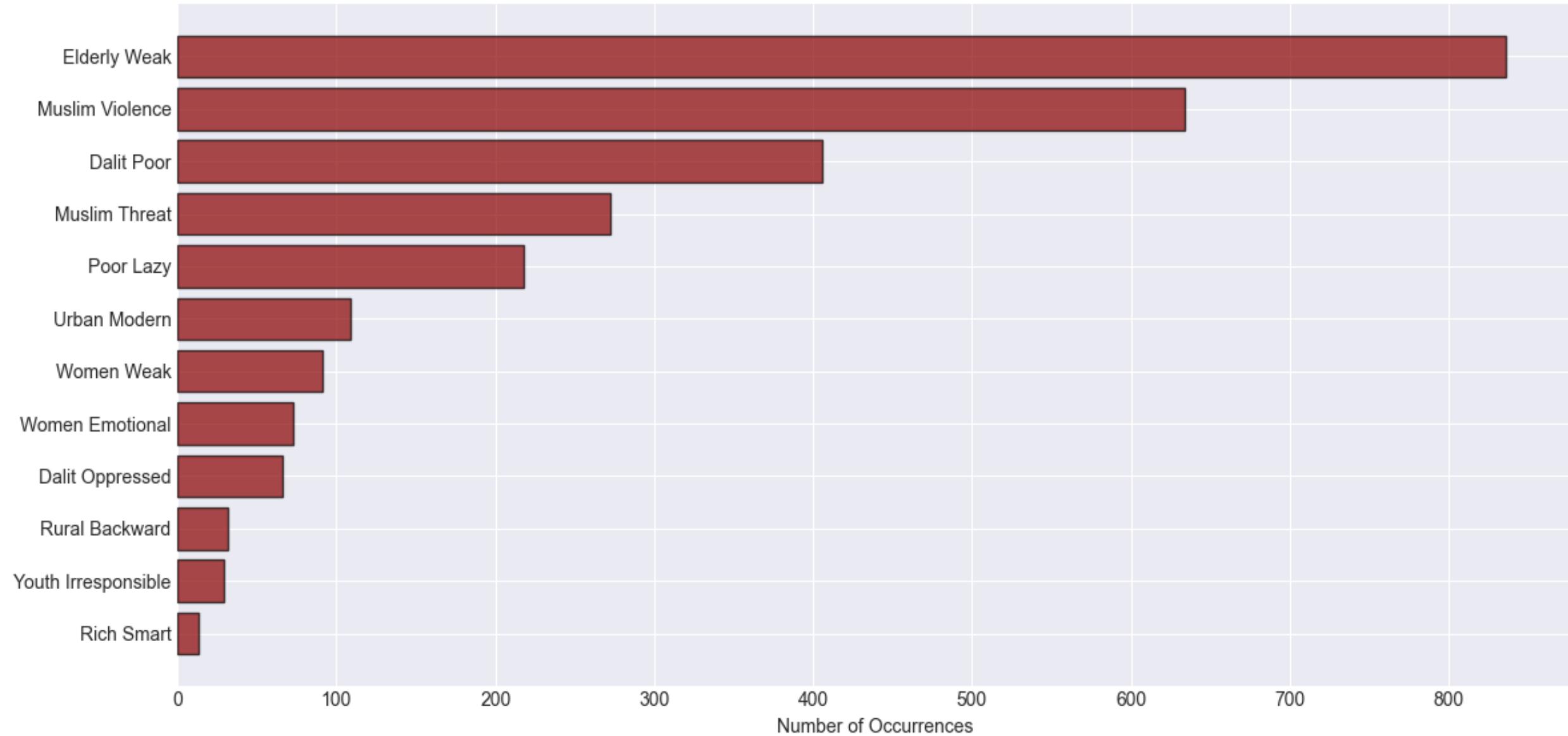




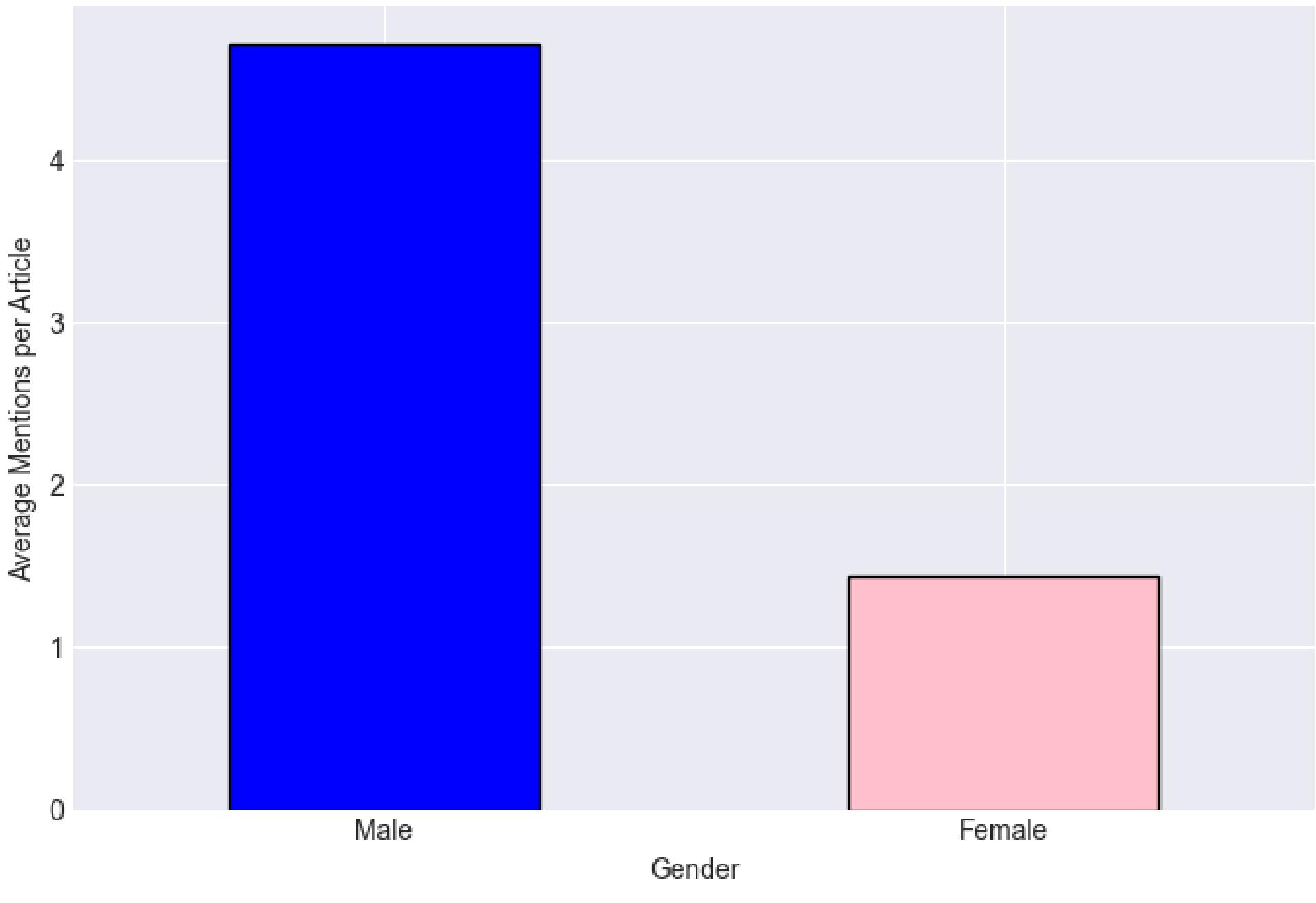
Top 20 Geographic Locations Mentioned



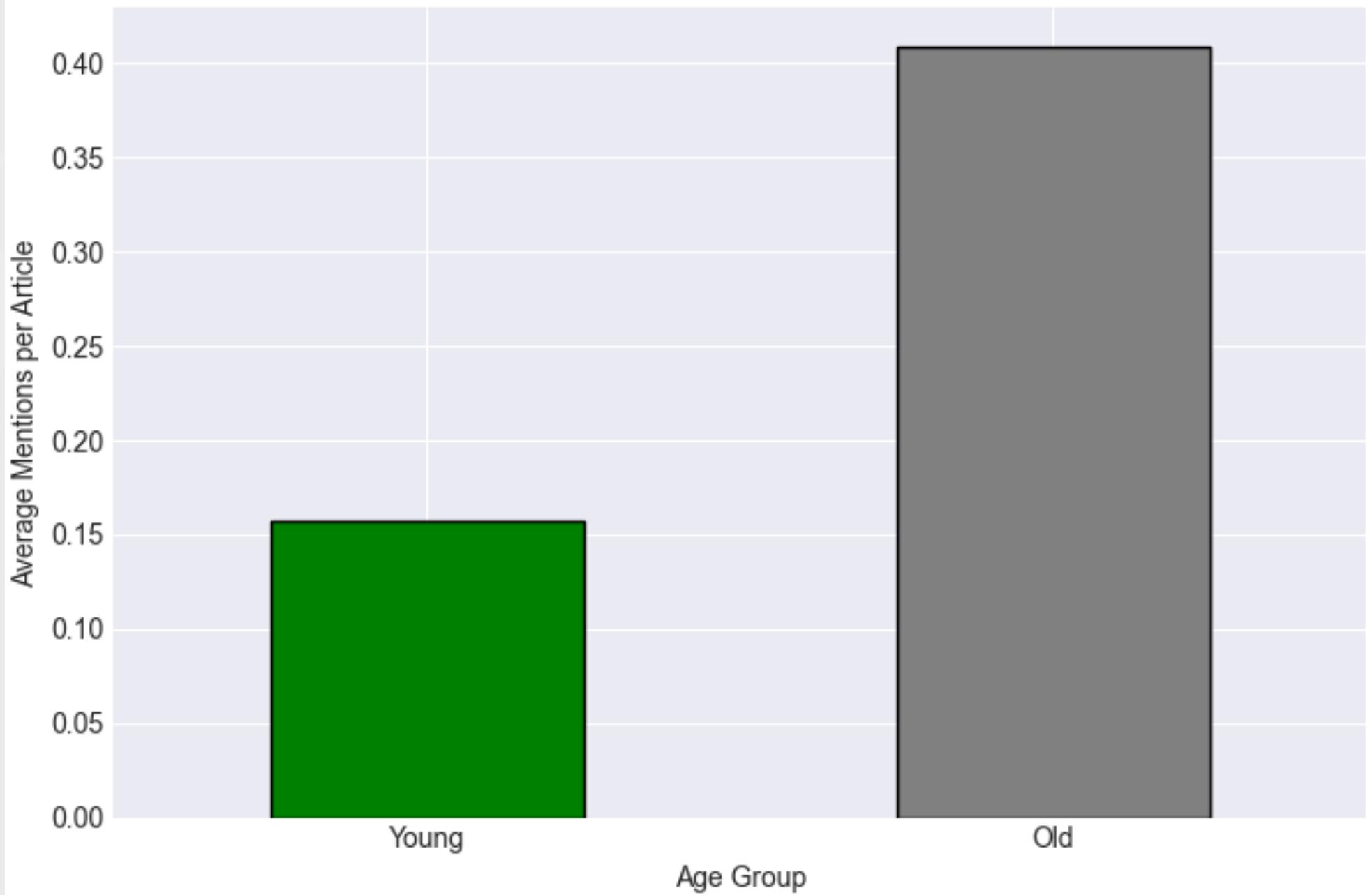
Implicit Bias Pattern Frequency



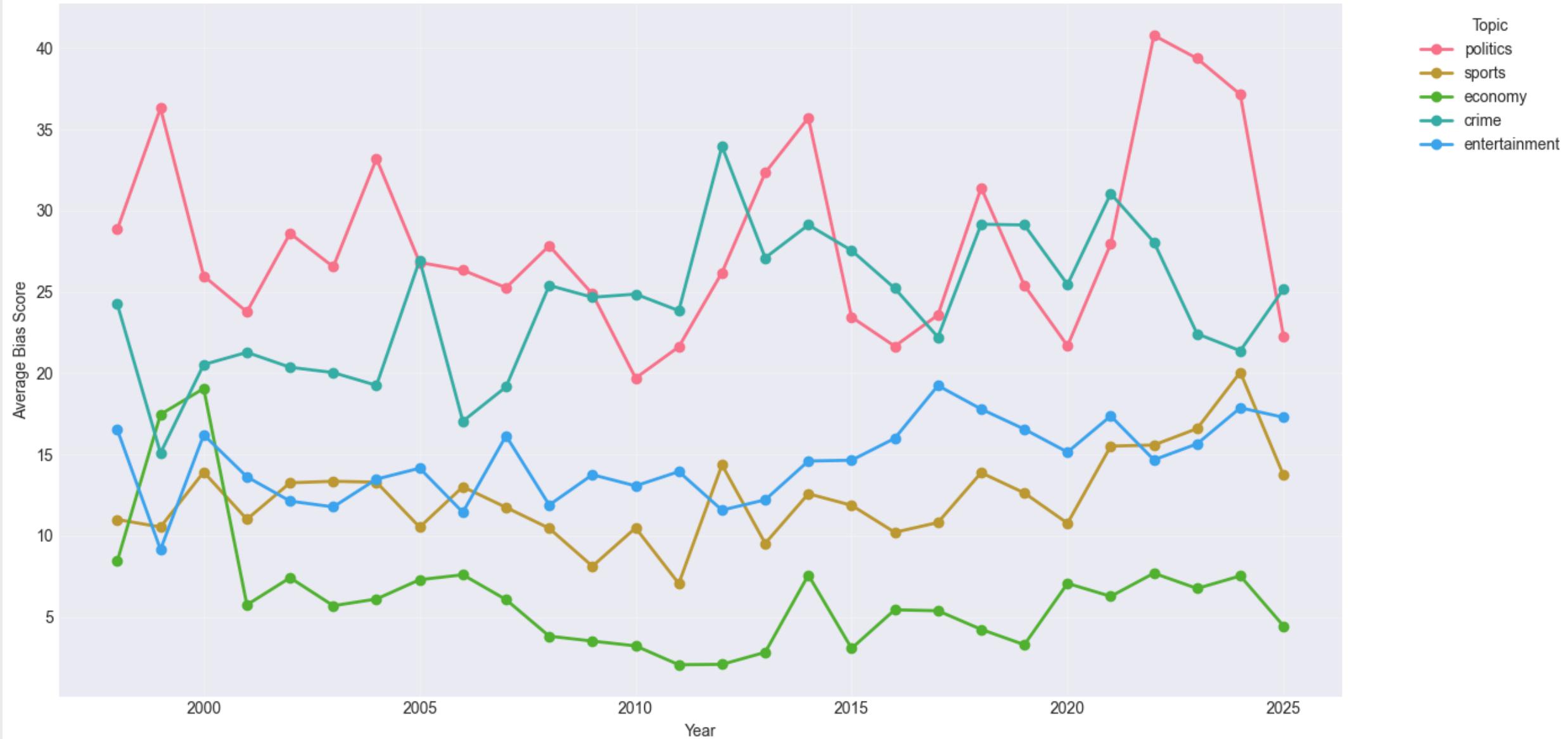
Gender Representation



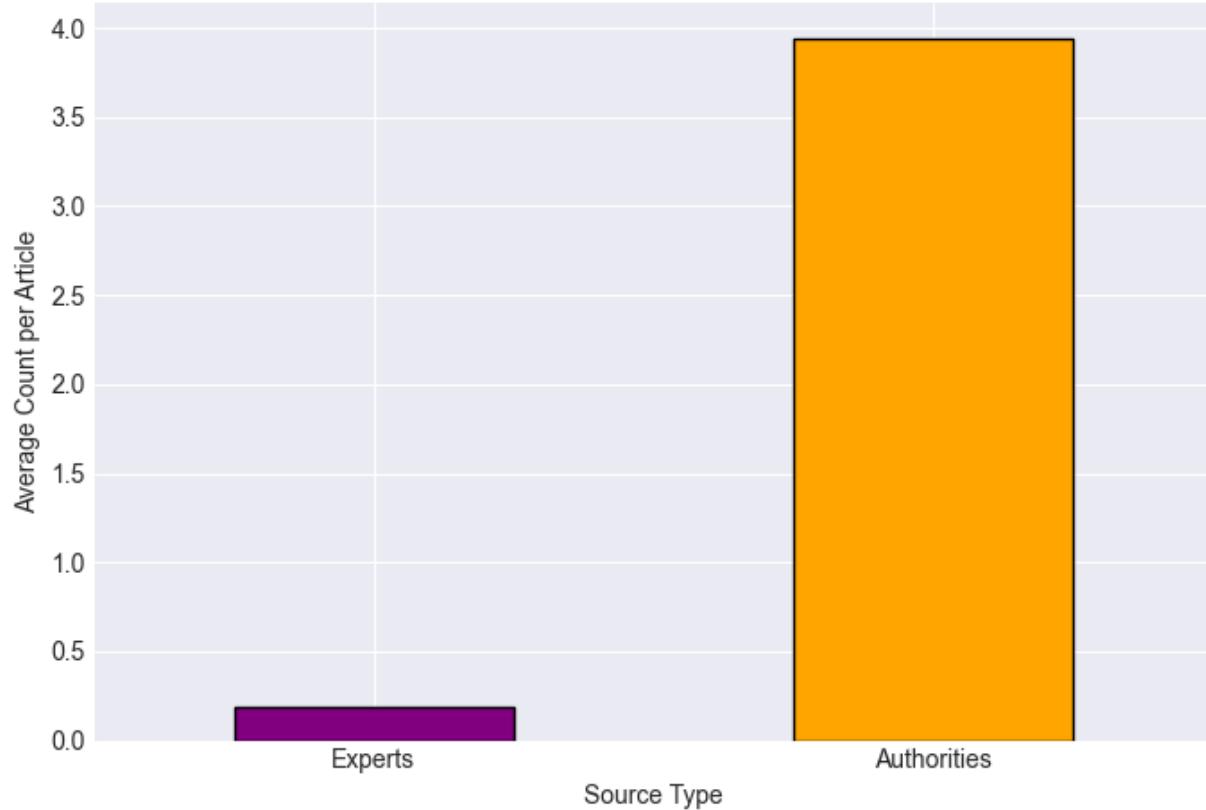
Age Representation



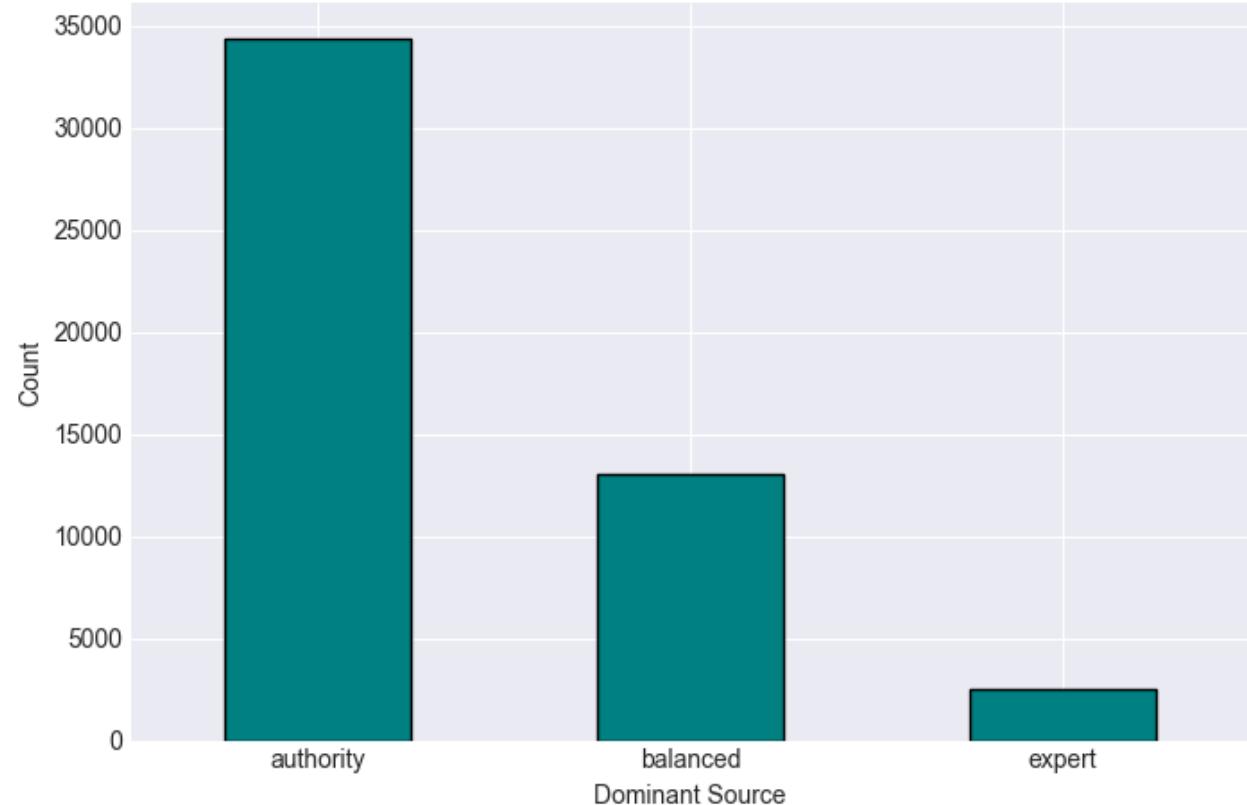
Bias Trends Over Time by Topic



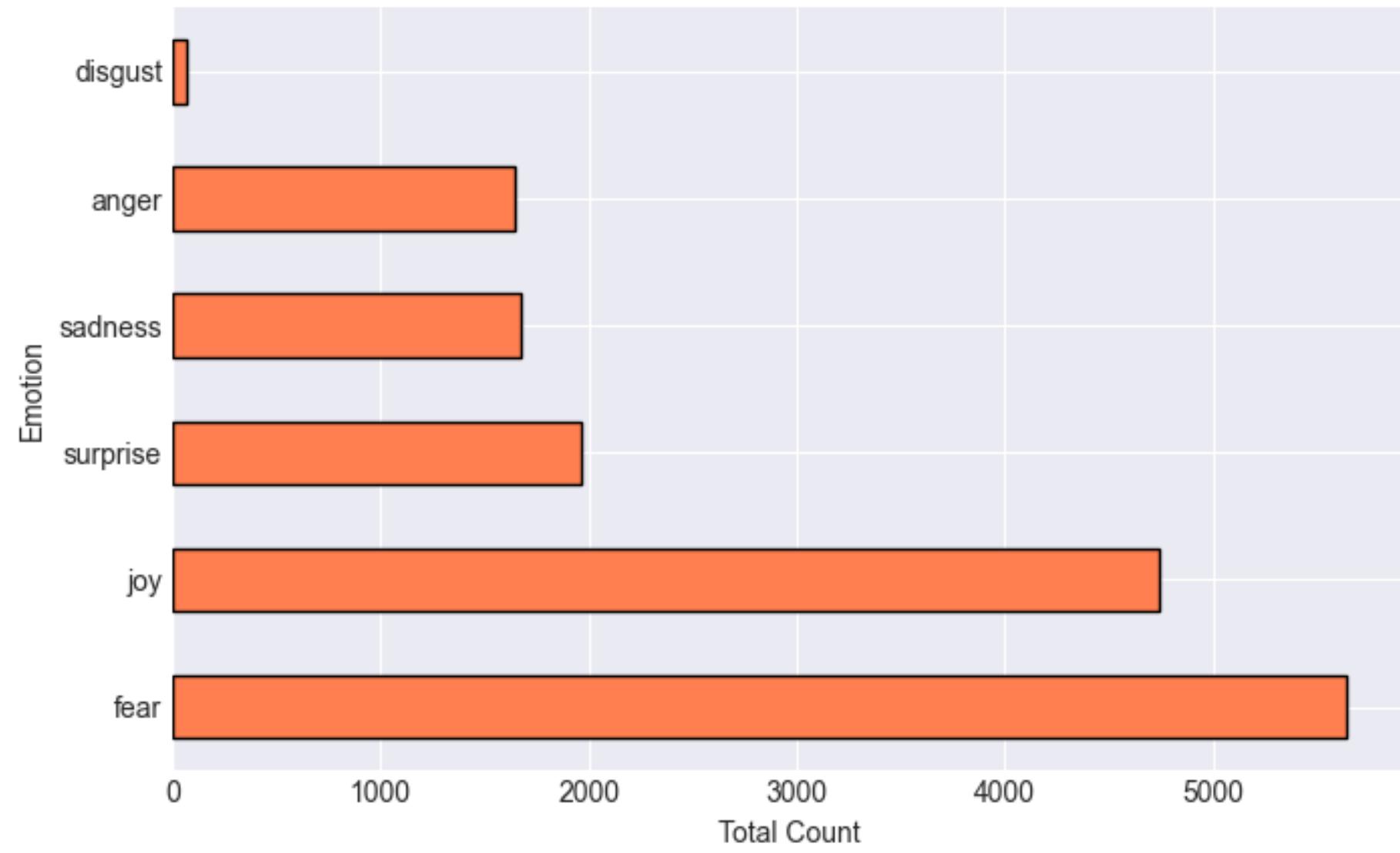
Average Source Mentions



Dominant Source Type Distribution



Total Emotion Words Across All Articles



CHALLENGES

and Future Roadmap

Challenges

- Identifying scrape-friendly news sites.
- Dealing with IP blocking.
- Improving the scraping performance.
- Ensuring coverage of regional languages.
- Translation of the articles in other languages to English.
- Choosing between native keywords vs translation-first.
- Selecting a suitable language model.
- Deepening NLP/feature engineering skills.
- Constructing a comprehensive India-specific bias lexicon.

Future Roadmap

- **Source Expansion:** More Indian news outlets and support for more regional languages.
- **Increase Bias Categories:** Add new numeric bias score fields and corresponding type categorical labels.
- **Richer Dashboard Visualizations:** Web-based dashboards with interactive graphs and detailed analytics.
- **Further Research:** Use more models and techniques, and also retrain models.
- **Community Feedback:** Incorporate impact measurements with feedback from the users.

DATA SCIENCE

Canvas

Data Science Canvas

Project:	Bias Detection in Indian News Media
Team:	Girish S N, Bharath Karanth A, Saravanakumar R, Anmol Gupta

Problem Statement				Execution & Evaluation		Data Collection & Preparation	
Business Case & Value Added <p>News media outlets are often accused of <u>biased reporting</u> across various categories. With the rapid growth of digital news, readers are exposed to vast and diverse information at their fingertips, often with subtle biases.</p> <p>There is a need for a solution that can <u>analyze</u> news articles for bias – offering transparent, multi-dimensional, and interpretable results.</p> <p>This project seeks to provide researchers, fact-checkers, and the public with actionable <u>insights</u> into media bias in <u>Indian journalism</u>.</p>	Model Selection <p>Various bias detection methods were employed.</p> <p>Keyword-based lexicon matching for different bias categories (gender, religion, caste, region, etc.).</p> <p><u>Sentiment analysis</u> using VADER and TextBlob.</p> <p><u>Emotion detection</u> via lexicons.</p> <p><u>Discourse analysis</u> (active/passive voice, agency).</p>	Model Requirements <p>Models must work with multilingual content and the <u>Indian news context</u>.</p> <p>Bias scoring requires proper <u>feature engineering</u> including careful preprocessing.</p> <p>Models should combine interpretable <u>lexicon features</u> with more complex <u>semantic</u> and ML ensemble scores for robustness.</p> <p>The overall bias score is a <u>weighted aggregation</u> across multiple bias categories, implying the model must output category-wise scores reliably.</p> <p>The system requires consistent updates as it depends on <u>evolving news data</u>, so resilience in scraping and model retraining is crucial.</p>	Skills <p>Expertise in <u>Python programming</u>.</p> <p>Experience in <u>web scraping</u> and <u>MongoDB</u> database management.</p> <p>Understanding of <u>ensemble modeling</u> and semantic embeddings.</p> <p>Familiarity with bias detection concepts and <u>Indian socio-political contexts</u> for meaningful feature engineering and interpretation.</p> <p>Visualization skills to create <u>insightful charts</u> for bias trends.</p>	Model Evaluation <p>Sentiment and emotion scores also provide validation layers to distinguish <u>tone differences</u>.</p> <p>Explainability is addressed by <u>lexicon-based features</u> and <u>clear bias type classifications</u>.</p>	Data Storytelling <p>Target users might require clear visualizations showing <u>bias trends over time</u> and among various media sources.</p> <p>Charts were generated to <u>highlight key narratives</u> and support exploratory analysis.</p>	Data Selection & Cleansing <p>Initially scraped raw data included only a csv of URLs and timestamps.</p> <p><u>Pre-processing</u> and <u>cleaning</u> of the raw data was done, so that it can be further used for training and analysis.</p>	Data Collection <p>As a first step, raw data was collected through the <u>web scraping</u> of online Indian news articles.</p> <p>The non-English articles (Hindi, Kannada, Tamil) were <u>translated</u> to English.</p> <p>Extensive <u>feature extraction</u> techniques were applied to eventually create a robust and comprehensive dataset.</p> <p>Creation of the dataset itself was one of the two major objective of our project.</p>
Data Landscape <p>There was no publicly available dataset for Indian news articles with enough features that could have been used to meet our objectives.</p> <p>Hence, we <u>created our own dataset</u> through <u>web scraping</u> of online Indian news articles.</p>	<p><u>Implicit association</u> tests inspired regex patterns.</p> <p><u>Semantic similarity</u> using TF-IDF and BERT embeddings.</p> <p><u>Topic classification</u> (LDA, K-means) to contextualize bias across topics.</p>		<p>Software & Libraries</p> <p>Python was utilized as the <u>main software language</u>.</p> <p><u>Key libraries</u> included <u>pandas</u> and <u>numpy</u> for data handling, <u>nltk</u> for NLP preprocessing, <u>requests</u> and <u>beautifulsoup4</u> for web scraping, <u>pymongo</u> for database connectivity, <u>python-dateutil</u> for date handling, and <u>matplotlib</u> and <u>seaborn</u> for plotting and visualization.</p> <p>MongoDB served as the primary <u>database</u> for storing scraped and processed data.</p> <p>A <u>Docker setup</u> was provided for running MongoDB and Mongo Express.</p>		<p>Data Integration</p> <p>Raw data, scraped from different online news websites and across different languages, was ultimately integrated into a single <u>comprehensive dataset</u>. Python scripts were used to scrape as well as integrate the data.</p> <p>The total number of articles in the final dataset was <u>3,96,739</u>.</p>	<p>Explorative Data Analysis</p> <p>The data includes articles scraped from multiple media outlets and languages, with <u>preprocessing to normalize text</u>.</p> <p>Outliers and structure are considered through statistical metrics and by aggregating bias scores with <u>5-year moving averages</u> to smooth trends.</p> <p>Calendar and event metadata were integrated to relate bias trends to <u>major social and political events</u>, helping to identify patterns and anomalies.</p>	

THANK YOU!