GNI Real-Time Newsroom Quality Index Playbook

NDTV Convergence

May 2024

INDEX

1. Purpose

NDTV, a leading Indian broadcaster, wanted to boost digital content consumption by growing new audiences and cultivating loyalty. They recognized that while trusted methodologies (like editorial judgment, readability formulas, and basic web analytics) were useful in measuring engagement, there was scope for improvement by leveraging real-time insights to refine content.

Hence, NDTV decided to create a framework for increasing engagement based on what content resonated most with readers and what they found valuable. To do this, NDTV partnered with Google and adopted a data-driven, iterative approach to enhance the quality of published content. This approach, The Newsroom Quality Index, involved a two-stage tool that provided editorial teams with real-time, data-driven insights.

2. Scope and methodology

Team NDTV worked with our data partners to predict the quality index of articles almost real-time, based on the article's pre-publishing parameters via CMS and the user's TimeOnPage and scroll over the article, real-time and archival. The Real-Time Newsroom Quality Index helps the editorial team to track how people respond to an article. The scores are predictive in nature and with ML, over time, become indicative of how users might interact with the content of the article.

With a predicted Newsroom Quality Index score in place, the editorial team can evaluate the content before and after publish. If the index score reduces after publishing, it will indicate less user interaction than expected, prompting the editorial team to review and improve the content. The whole process is divided into two stages, as explained below.

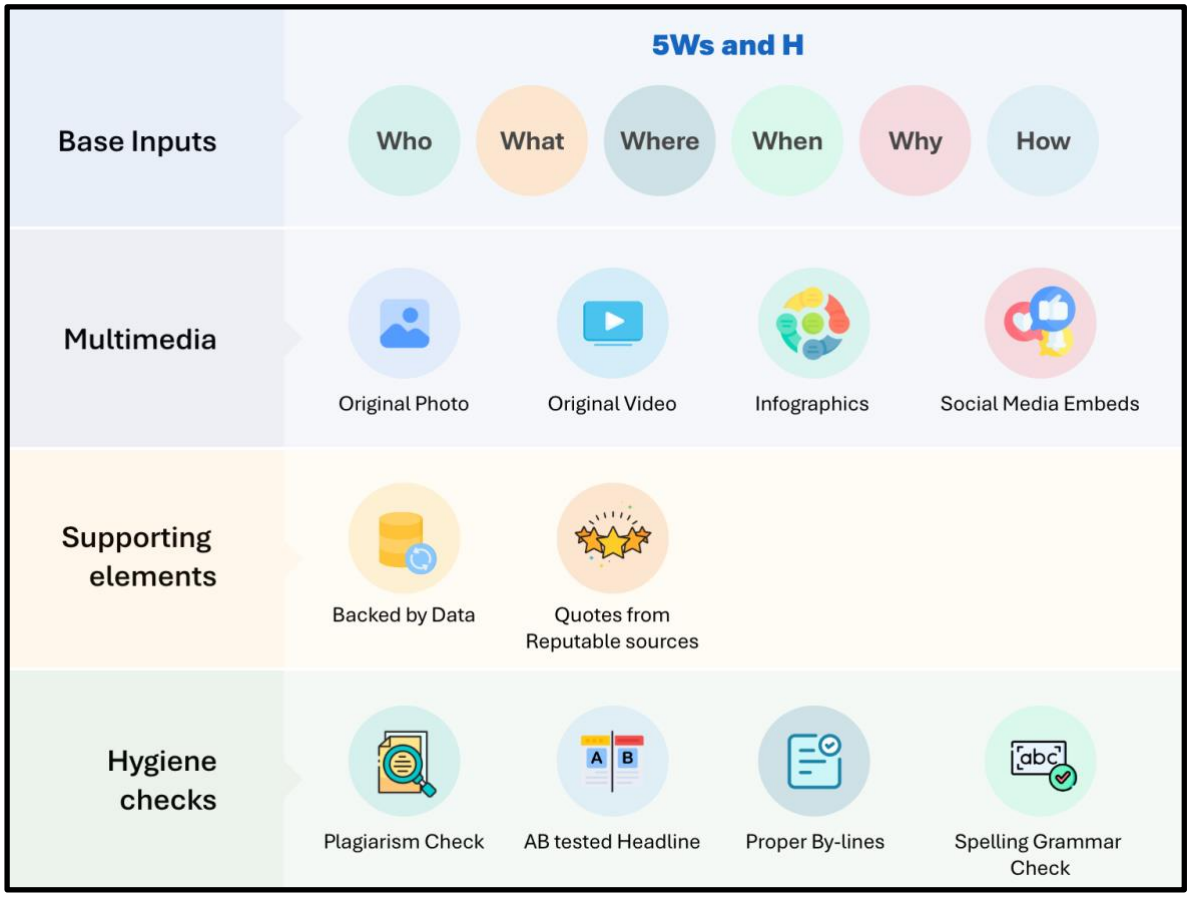
Stage 1: Pre-Publishing Quality Score

In the first stage, NDTV focused on strengthening the very foundation of article writing. To ensure that the articles are robust, original, and engaging, NDTV developed a comprehensive checklist of essential inputs.

Essential Elements for Quality Articles

1. Base Inputs (5Ws and H): Ensuring that the story answers all the basics – Who, What, Where, When, Why, How.
2. Multimedia: Using original photos, videos, infographics, and social media embeds for visual engagement.
3. Supporting Elements: Quotes and data for credibility and depth.
4. Core Hygiene: Meticulous checks for spelling, grammar, plagiarism, bylines, and headlines.

Once these elements are in place, the story is pushed into NDTV's Content Management System (CMS). The CMS generates a pre-publish quality score before the story goes live. Editorial teams must ensure the highest possible pre-publish score for every story.



An illustration of the Inputs that are used to build a story

Stage 2: Post-Publishing Predicted Engagement Score

In stage 2, NDTV combined several data sources such as event-based and user data via Google Analytics, real-time data from Datastream, Chartbeat's raw data pipeline and article data from our native CMS, by leveraging APIs and Google Cloud buckets to enable seamless extraction of data into Google BigQuery. The final unified data set allowed NDTV to effectively analyze and apply a quality score to each article.

NDTV leveraged Machine Learning to build a robust and accurate model which analyzed CMS details and real-time user interactions (consumption, time on site, etc.) to predict an article's engagement trajectory.

3. Data Sources

We have used the data source below to create this Solution.

1. Article CMS Data as primary score baseline
2. Chartbeat Datastream, (raw data pipeline) to feed into the ML algorithm
3. Google Analytics Data to source trendlines to feed into the ML algorithm

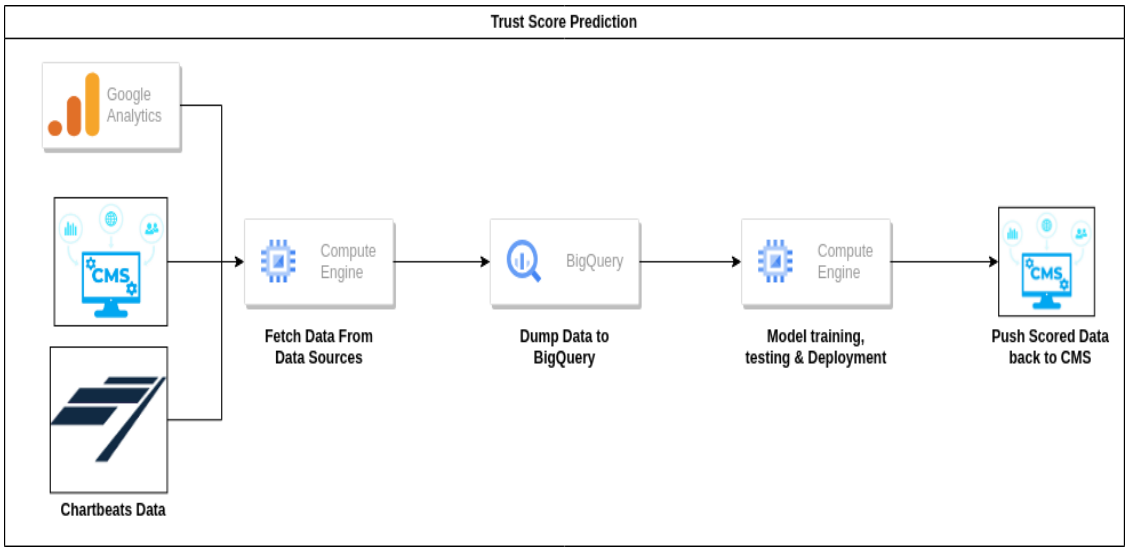
4. Approach

a. Data Extraction:

- CMS Data Extraction: We have built an API that enables seamless extraction of CMS data into Google BigQuery. This API allows the Team to push their CMS data, which is then securely stored in designated BigQuery tables.

- Chartbeat Data Extraction: Using Chartbeat real time data stream as a source, we store data in a secure Google Cloud Storage (GCS) Bucket. The program then performs regular data extraction from the GCS Bucket on a half-hourly basis. The extracted data is then seamlessly transferred and stored in Google BigQuery, ensuring its availability for analysis and insights

- GA Data Extraction: For Google Analytics (GA) data extraction, we capitalize on the existing integration with Google BigQuery. This eliminates the need for additional extraction steps, as the GA data is already available within the BigQuery environment. By preparing precise queries, we retrieve the desired GA data, enabling efficient analysis and reporting.



Architecture block diagram for data extraction process

b. Data Preparation:

To prepare the extracted data for analysis, we have followed a systematic approach using Python code. To start with, we retrieve the data from various sources, which includes NDTV native CMS, Datastream from Chartbeat, and data from Google Analytics. These are consolidated into dataframes. The joining key for all the data will be the article page link, which will allow us to merge and analyze the data effectively.

Here's an outline of the steps involved:

- Retrieve CMS data:
  - Execute the appropriate query using the API to fetch the CMS data.
  - Store the retrieved data in a dataframe.
  - The scores based on parameters highlighted under stage 1 above are retrieved from the CMS. These are a mix of Manual and Automated entries.
- Retrieve Chartbeat Data:
  - Access the stored Chartbeat data from the designated GCS Bucket.
  - Perform data extraction on a half-hourly basis
  - Store the extracted Chartbeat data in a dataframe.
  - Data retrieved from Chartbeat is TimeOnSite and Scroll Depth (scope here can increase subject to editorial and product team requirement)
- Retrieve GA Data:
  - As GA data is already available in Google BigQuery, we can directly prepare a query to fetch the desired data.
  - Execute the query and store the GA data in a dataframe.
  - Data retrieved from GA is PageViews (scope here can increase subject to editorial and product team requirement)
- Data Integration:
  - Utilize the article page link as the joining key to merge the CMS, Chartbeat, and GA dataframes.
  - Perform a left join operation to combine the Chartbeat and GA data with the CMS data.
  - This process will create a comprehensive and unified dataset for further analysis.
- Prepare Final Data for Scoring:
  - Once the data frames are merged, perform any necessary data transformations, cleaning, or feature engineering.
  - Ensure the data is in the desired format and structure for scoring or further analysis.

By following these steps, we can effectively prepare the final dataset for scoring or any subsequent analysis. The consolidated data will provide valuable insights by combining CMS information, Chartbeat metrics (Aggregated TimeOnPage and Scroll), and GA data (pageviews) through a seamless integration process. The set process also makes it easy to modify the inputs as deemed fit by the publisher per use case.

c. Model Data Processing:

To ensure accurate and reliable modeling, we need to perform various data processing steps. These steps will help us gain a better understanding of the data and its characteristics, identify dependencies between columns, handle missing or outlier values, and address biases in dependent columns.

A breakdown of the data processing steps is as shared below:

- Statistical Analysis:
  - Calculate descriptive statistics such as mean, median, and standard deviation for each column.
  - This analysis provides insights into the central tendency, spread, and variability of the data, aiding in data understanding.
- Correlation Analysis (Bivariate Analysis):
  - Conduct a correlation analysis to identify relationships between variables.
  - Determine the dependent and independent columns based on the correlation coefficients.
  - This analysis helps in understanding the interdependence of variables and identifying potential predictors.
- Preprocessing for Handling Missing and Outlier Values:
  - Handle missing values in the data by employing techniques such as imputation, deletion, or interpolation.
  - Detect and handle outliers
  - By addressing missing and outlier values, we ensure the integrity and quality of the dataset.
- Handling Dependent Columns:
  - Identify any dependent columns that may introduce bias into the model.
  - This step helps in building a more robust and unbiased model that avoids overfitting or underestimation.

By following these data processing steps, we can enhance the quality and reliability of the dataset. Performing statistical analysis, correlation analysis, and preprocessing techniques allows us to gain insights into the data, handle missing or outlier values, and address any biases that may exist. These steps lay the foundation for building a robust and accurate model based on a well-processed dataset.

d. Model Training and Tuning:

Once the data has been prepared, the next step is to train and tune the machine learning models. This process involves training multiple models on the prepared dataset and evaluating their performance using a suitable evaluation metric, such as RMSE (Root Mean Square Error).

The following steps outline the model training and tuning process:

- Model Training:
  - Select and train different machine learning models suitable for the problem at hand, such as regression, classification, or ensemble models.
  - Split the prepared dataset into training and validation sets to assess model performance.
  - Train each model on the training set using appropriate algorithms and techniques.
- Model Evaluation:
  - Evaluate the performance of each trained model using the validation set.
  - Calculate the RMSE score for each model to quantify the prediction error.
  - Compare the RMSE scores to identify the models that perform better in terms of accuracy and predictive power.
- Model Tuning:
  - Perform model tuning by adjusting hyperparameter values to further optimize model performance.
  - Utilize techniques such as grid search, random search, or Bayesian optimization to systematically explore different hyperparameter combinations.
  - Evaluate the models with tuned hyperparameters and select the best-performing model based on the improved RMSE scores.
- Saving the Best Model:
  - This step involves serializing the trained model into a file format (e.g., pickle or joblib) that can be easily loaded and utilized in production environments.

By following these model training and tuning steps, we can identify the most accurate and reliable model for the given task. The iterative process of training, evaluation, tuning, and saving the best model ensures that the deployed model is optimized and ready for production use.

e. Model Deployment:

To deploy the trained model and make real-time predictions based on the data extraction and preparation steps, we can follow the following deployment steps:

- Finalize Data Extraction and Preparation Script:
  - Combine the data extraction and preparation steps into a comprehensive script.
  - This script should include the logic to extract and prepare the data at regular intervals, such as every half-hour.
  - Ensure that the script retrieves the latest data from the relevant sources and preprocesses it using the established techniques.
- Load the Saved Model:
  - Load the previously saved best-performing model into memory.
  - This step allows us to utilize the trained model for real-time predictions.

- Make Real-Time Predictions:
    - Utilize the loaded model to make predictions on the real-time data extracted and prepared in step 1.
    - Apply the trained model to the incoming data to generate predictions based on the established features and patterns.
- Save Predicted Data to BigQuery:
    - Store the predicted data in Google BigQuery for further analysis and integration with other systems.
    - Ensure that the appropriate table structure and schema are followed to maintain consistency and ease of access.
- Send Predicted Audience to NDTV's CMS Database:
    - Establish a connection to NDTV's CMS database or API.
    - Send the predicted audience information to the CMS database in real-time.
    - This step ensures that the predicted audience data is readily available for content customization or any other required purposes within NDTV's systems.

By following these model deployment steps, we establish a streamlined process for extracting and preparing real-time data, making predictions using the trained model, and storing the predicted data in BigQuery for further analysis. Additionally, integrating the predicted audience with NDTV's CMS database allows seamless utilization of the predictions within NDTV's content management and customization workflows.

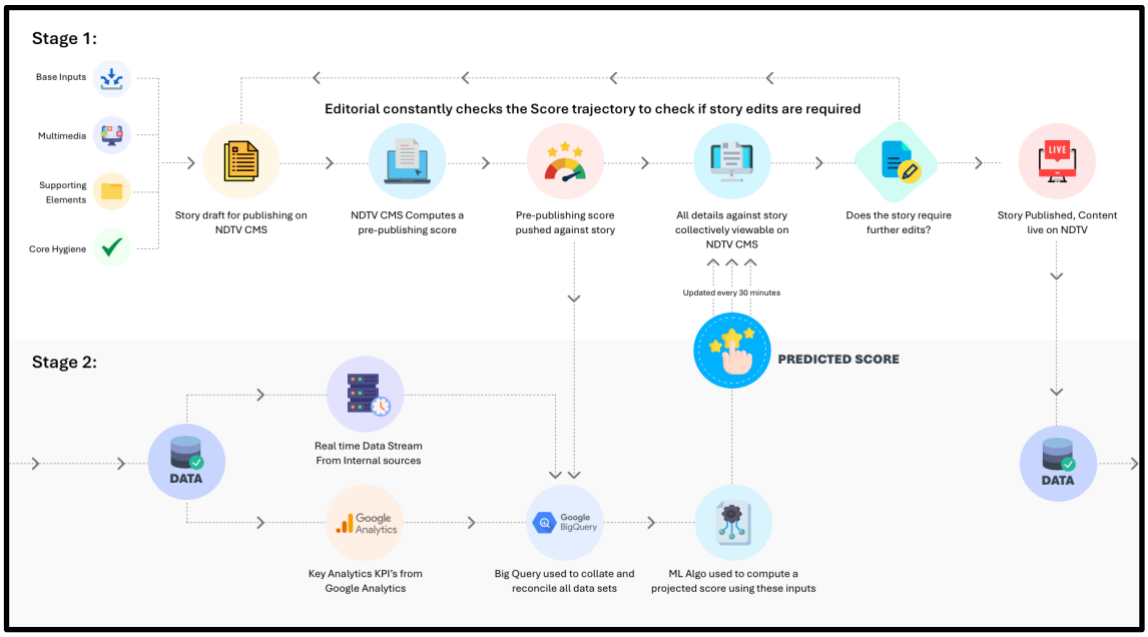
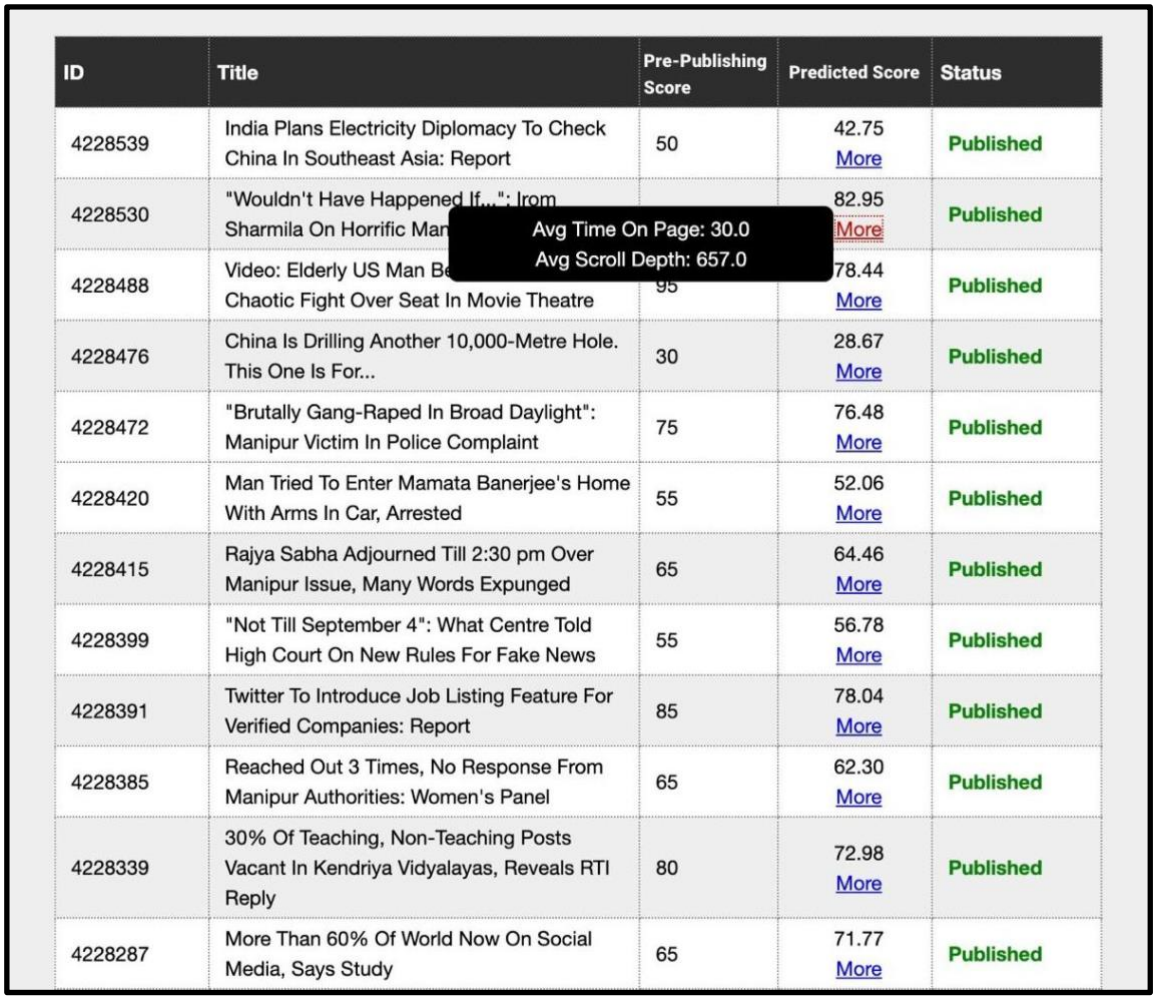


Diagram explaining the Newsroom Quality Index playbook.

5. Outcome

The outcome of the data extraction, preparation, and model deployment process is the predicted score for each article. This score is derived from the analysis and integration of various features extracted from CMS, Chartbeat, and GA data.



ID	Title	Pre-Publishing Score	Predicted Score	Status
4228539	India Plans Electricity Diplomacy To Check China In Southeast Asia: Report	50	42.75 More	Published
4228530	"Wouldn't Have Happened If...": Irom Sharmila On Horrific Man		82.95 More	Published
4228488	Video: Elderly US Man Be... Chaotic Fight Over Seat In Movie Theatre	95	78.44 More	Published
4228476	China Is Drilling Another 10,000-Metre Hole. This One Is For...	30	28.67 More	Published
4228472	"Brutally Gang-Raped In Broad Daylight": Manipur Victim In Police Complaint	75	76.48 More	Published
4228420	Man Tried To Enter Mamata Banerjee's Home With Arms In Car, Arrested	55	52.06 More	Published
4228415	Rajya Sabha Adjourned Till 2:30 pm Over Manipur Issue, Many Words Expunged	65	64.46 More	Published
4228399	"Not Till September 4": What Centre Told High Court On New Rules For Fake News	55	56.78 More	Published
4228391	Twitter To Introduce Job Listing Feature For Verified Companies: Report	85	78.04 More	Published
4228385	Reached Out 3 Times, No Response From Manipur Authorities: Women's Panel	65	62.30 More	Published
4228339	30% Of Teaching, Non-Teaching Posts Vacant In Kendriya Vidyalayas, Reveals RTI Reply	80	72.98 More	Published
4228287	More Than 60% Of World Now On Social Media, Says Study	65	71.77 More	Published

Illustration of Newsroom Quality Index as seen by editors on a CMS

The predicted Newsroom Quality Index score serves as a valuable metric that indicates the quality of an article based on users' interactions and online data. By leveraging insights from Chartbeat and GA, we can assess user engagement, pageviews, and other relevant metrics to gauge the credibility and impact of the article.

This Newsroom Quality Index score is being utilized in several ways, including:

- Content Customization:
  - The predicted Newsroom Quality Index score is integrated with NDTV's CMS database to dynamically customize content based on the engagement of each article.

- Quality Assurance:
    - The Newsroom Quality Index score can be used as a quality assurance metric to assess the performance of articles and identify areas for improvement.
    - It helps in identifying articles that resonate well with the audience and drive higher engagement.
- Decision-Making:
    - The Newsroom Quality Index score serves as a valuable input for decision-making processes such as editorial calls, content promotion strategies, and resource allocation.
    - It enables data-driven decision-making, ensuring that resources are assigned to articles with higher scores and audience engagement potential.

Overall, the predicted Real-Time Newsroom Quality Index score provides valuable insights into the user engagement with the article. By incorporating these insights into decision-making processes and content customization workflows, any publisher can enhance content for user experience, optimize strategies, and drive engagement effectively.