# **Liberia: Analysis of Media Coverage by Development Indicators During Times of Ebola**

**By Nathan Danielsen**

### **Research Question**

Does the level of development and demography of a location in Liberia (county, municipality and lower levels like village) affected the quantity and quality of media coverage (local, national and international)?

### **Hypothesis**

I hypothesize that timely and accurate media coverage of significant events in Liberia is determined by population density but also by the availability (and funcationality) of telecommunications infrastructure and localized political centers of gravity and governance. Furthermore I hypothesize that there is a signficiant politicization of media by political parties which is likely fueled by ethnic and linguistic divisions.

### **Strategy**

Since the quantity of Ebola media coverage has been extensive, I hope to test my hypothesis by contrasting Ebola media coverage content with geographical, demographic, other development indicators.

**Software Project Design**

I designed each section of the data project to be able to operate independently of each others and to be able to operate in an asynchronous environment – ie different worker scheduled to operate independently but cooperatively at different times in the ingestion, extraction/ cleaning, analysis and visualization pipeline.

**Data Collection (Scraping)**

Built a basic scraper to iterate through 9 Liberia online news content sites. These sites allowed a numeric post-fix to a url to easily allow iterate through content available.

Example:

[http://www.liberianobserver.com/node/**101**](http://www.liberianobserver.com/node/101)

[http://www.liberianobserver.com/node/**102**](http://www.liberianobserver.com/node/102)

…

[http://www.liberianobserver.com/node/**1500**](http://www.liberianobserver.com/node/1500)

[http://www.liberianobserver.com/node/**1501**](http://www.liberianobserver.com/node/1501)

Html data for all papers requested was saved to a txt document if the request response was a 200. The name for the file was the timestamp at time of request.

52995 requests to the 9 websites were made over the course of about a week. All requests and selected information was logged to file to monitor the collection and for later analysis.

25105 requests returned a 200 status code. 27890 requests failed or produced a code other than 200.

Quantity of Articles Available

--------------------------------------------------

Dailyobserver : 4524

Frontpageafricaonline : 4359

Gnnliberia : 1749

Golministryofinformation : 2884

Post1847 : 6023

Theanalyst : 304

Thenewdawn : 5262

\*The GNN Liberia and the New Dawn in all cased returned 200 even in no article was available. For future note, I could develop a spam/ham type model for identifying which pages to save.

Log File Data Dictionary:

url\_request - the website requested ie http://www.liberianobserver.com/node/101

url\_status\_code – The response from received website ie 200, 404,

header\_len – Length of the header response

response\_len – Length of the html code received (if received)

name – User defined name of publication / content site

time – Datetime stamp

message – User defined message ie 'Testing', 'Production', etc

All header responses received from the sites where saved to a JSON file specific to each content site for future analysis.

**Data Model for Analysis**

With analysis in mind and the eventual goal of scaling up to larger data, a data model and supporting database was designed during the cleaning and design process. Unicode was selected as the encoding with the eye for future multi-language requirements.

Sqlalchemy with SQLite was selected because of it's pythonic nature, ease of use on the local machine, and Sqlalchemy's ability to easily drop in another database such as postgres.

Data Model Dictionary

publication\_name – User Defined Name of Content Provider

true\_url - True published URL of content

category – Content Provider Defined Category of Content

title – Title of Content Piece

datetime – Date Content Published

author – Content Piece By line (None value where not identified)

clean\_content – Main text content extracted from html

image\_url – Lead image (None where no image published)

**Data Extraction**

Due to time constraints, only the Daily Observer's content extracted because it is a prominent publication in Liberia. Beautiful soup was used for text data extraction and cleaning. Arrow was used for converting publish date to python datetime objects.

When extracted, data file was saved to a pickle file format to preserve data-types which in ways mimics a database environment.

Of the 4524 html files cleaned, the unique number of values for each category is below.

| **'Liberian Observer'** | **true\_url** | **category** | **title** | **datetime** | **author** | **clean\_content** | **image\_url** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **unique** | 4524 | 192 | 4489 | 4453 | 439 | 4293 | 3383 |

From looking at the raw data and reviewing many of the pages on the live webpage, many pieces of content lack author's attributed, images or content/ images. There are several editorials or cartoons that do not supply a 'By' attribute.

**Feature Extraction**

Using Pandas and NLTK, several features were extracted for analysis.

Lead – First four sentences that normally represent a concise summary of story.

ebola\_title – 1 if 'ebola' or 'ebv' in title, 0 if not

ebola\_lead - 1 if 'ebola' or 'ebv' in lead, 0 if not

ebola\_content - 1 if 'ebola' or 'ebv' in lead, 0 if not

location\_title – List of counties or county capitals referenced.

location\_lead – List of counties or county capitals referenced.

location\_content – List of counties or county capitals referenced.

I still need to pull out the dummy variables to tag each row with 1 or 0 if it contains the geographic location.

Initial findings:

Ebola in title: 458

Ebola in lead: 908

Ebola in main content: 1,229

**Modeling**

In progress to look at infection rates by county and contrast with ebola news coverage by county.

**Challenges**

***Visualization***

I found it incredibly hard to find resources to do pandas time series visualizations.

I wished to contrast weekly number of 'ebola' stories per calendar week with 'ebola' stories per region in a timeseries plot and contrast that with infections per week per region.

**Lessons Learned**

Building a model for database engine took too much time. Considering the small size of the data, Pandas was a much better fit for data processing and analysis. But I'm glad that I thought deeply about the data types to extract.

Writing unit tests for each section of the project would have made debugging much easier, but as this was a learning projects, I didn't know which tests to write or how to test capability.

As is often said, data science is mostly about data cleaning. In this project, it took me approximately 90% of the time to collect and clean the data. I'm still rushing to analyze it.

Life gets in the way of being a weekend data science warrior. Also moving to a new apartment ate up a lot of my time.

I need to work through the NLTK book and get a better understanding of some of it's amazing tools that would save me time.

**Plans to Scale Up to 'Bigger' Data Models**

Raw web pages should be stored to a database. It would be easier to access and categorize them.

I think that I'm in a good stage that if I decided to collect data for another target country, I would relatively easily refactor my code to make it more generalizable.

Writing a better scraper that re-requests 3 times to a site if a non-200 is received is on the to-do list.

**Business Applications**

My international nonprofit organization does work in education, media development and civic development around the world.

This data product would be adapted to assess the impact of our interventions on media content producers and even journalists/authors. Furthermore, it would be tailored to identify trends in civil society and used to identify propaganda.