NAACP Media Research

CS 506 - Spark! Project

Yufeng Chen, yufeng72@bu.edu | Jiaqi Sun, sjq@bu.edu | Ruotian Liu, rtliu@bu.edu

I. Summary & Problem Statement:

The National Association for the Advancement of Colored People (NAACP) is a civil rights organization in the United States, formed in 1909 as a bi-racial endeavor to advance justice for African Americans. Their goal is to ensure the political, educational, social, and economic equality of rights of all persons and to eliminate race-based discrimination.

In this project, we are going to help our client, NAACP Boston, to understand the coverage of Boston Media in covering Boston's Black people and Black neighborhoods. We are going to assess the volume of coverage, the general sentiment of all reports, the topics covered, and how this has changed over time. To do so, we need to collect data from newspaper and radio websites (Boston Globe, as well as WGBH and WBUR), design algorithms to understand the data, then draw some conclusions. Our analysis will focus on traditionally Black communities which we will define using tract level census data. We think the results we get can provide a good view on how the media in Boston did in the past 5 years (from 2014 to 2018). Hopefully, these results will lead our client to some possible ways that can help the media to do better on the elimination of racial hatred and race-based discrimination.

II. Methodology & Algorithms:

Step 1: Collecting Data

The first step of the whole project is to collect all data (news) in the past 5 years from the website. To do so we need to get the links of all Boston Globe, WBUR and WGBH's website copies from an online database called the "Wayback Machine". Our first attempt was to get all links of 3 websites using the Wayback machine API for Scrapy. It went well in the test but when we started to really use it, several problems encountered like Windows system support

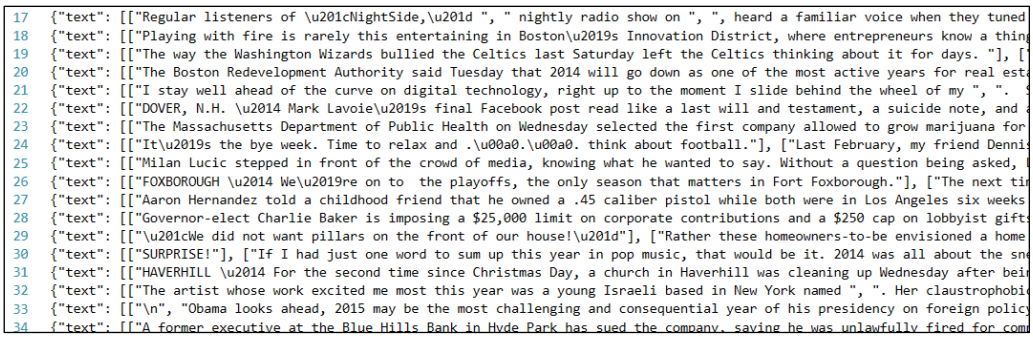
file missing, long website response time or various changing domains. We then tried some other methods, and finally chose a data science tool called Waybackpack, which successfully helped us get what we want.

After acquiring website links successfully, the next thing we need to do is to utilize a package named “HTMLsession” to get all article/news links we want to go through to scrape articles. Since all those links we want to pull always include some keywords (like “metro” and “sports” as topics in links of Boston Globe), we added some restrictions in to our algorithm to filter when scraping.

For example, a keyword of the general topic of an article is included in the link of this article, so we can filter links with these topic keywords to avoid getting useless links, like a register page or a help page.

Then, to scrape articles from website, we first tried a python module named “BeautifulSoup”. It could work correctly to collect articles after adding some restriction to it, but it’s just too slow (it took 5 hours to scrape Boston Globe’s news in only 30 days). Our speculation is that this tool may need to visit the website for every piece of news and is always waiting for the website’s response. It’s just not acceptable because we need to get all news in recent 5 years from 3 different websites.

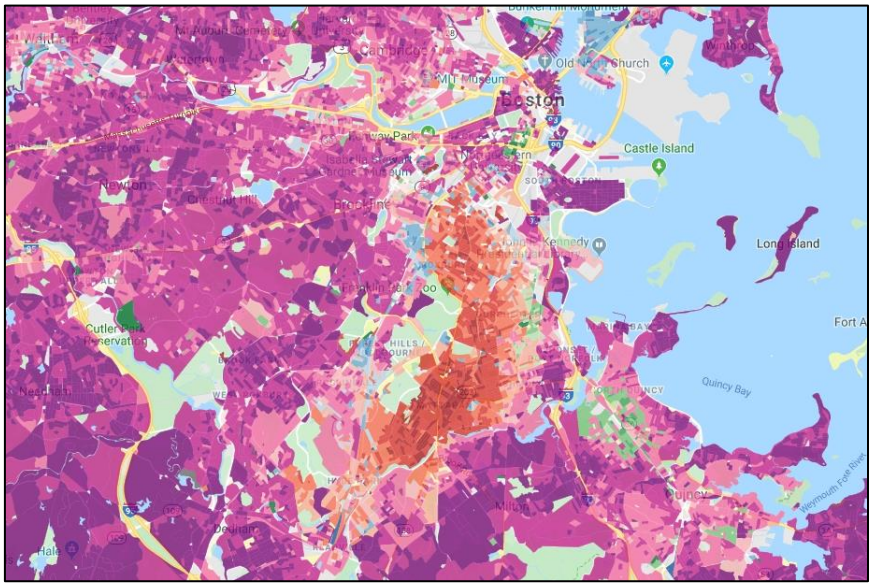
Our next try is to use Scrapy instead to solve this problem, and it turned out Scrapy is an excellent choice for the job. With relatively little code, Its Python interface can execute complex instructions for "scraping" web pages as described above. In its simplest form, it can be used as a general-purpose web crawler, visiting links and dumping the HTML pages back to the user. It can also be used in more precise ways, such as reading the DOM (Document Object Model) or structure of HTML pages and writing certain fields back to the local environment. In fact, if our goal is gathering as much information as possible in a structured way, a web browser is a painfully slow and imprecise avenue to take. It would be much more reliable to write our own code to send requests, process the responses, and store the output. Then, we could wrap that code in a script, and command it to visit and catalog websites automatically. Here our core mission was to understand the structure of the certain website we want to scrape, find out which class the article part lies in, and write down that class in our code. After doing this, Scrapy will help us locate in that class and scrape the article in an efficient way.



This is what our raw data file (.json) looks like

Step 2: Filtering News

Step two is to select all news about black people and black communities, which we care about. After meeting with our client, we all agreed that using black neighborhoods' name as a filter should be a good way. There are less than one hundred neighborhoods or sub-neighborhoods in Boston, so we just collect the census data manually from several websites (American FactFinder, Justice Map and Google Places), then get a list of black neighborhood names.



Justice Map - Boston

Because only neighborhood names are not enough to get us as much news as possible about black people, we then added some other words like "black man" or "Latino" into our filter-word list which could help us get more results. After filtering, we had a list of articles about the black. We classified these articles by year so that we can do the analysis later. We also filtered articles again with a different set of word list (which contains all neighborhood names) to calculate the volume of the coverage of the black neighborhoods.

Step 3: Sentiment Analysis

Now it's time to see what has the media said about the black people in the past 5 years. At the beginning we planned to train a model just like what we did in the midterm to judge an article's sentiment. However, we then realized that we couldn't find a very good train dataset, so the results were just not very satisfying. Then we planned to use some existing sentiment analysis tools to do the job. Finally, we decided to use 'VADER-Sentiment-Analysis' to help us determine if a piece of news is positive or negative after testing on some sample articles. This tool can analyze sentences with a high speed and is able to correctly judge those tricky sentences that confuse other sentiment analysis tools. The result contains four scores: pos, neg, neu and compound, and we used the compound score to determine whether if an article is overall positive or negative. We run it on all articles about the black, then run it again on the whole dataset (all news), getting two sets of results to see what's their difference.

Step4: Getting Popular Keywords/Topics

After comparing the sentiment results, we wanted to find out what causes the difference. We looked into both datasets (black news vs all news) to find out what topics are popular. First we did some analysis based on word count: we initially attempt to extract topics by simply counting the number of occurrence for each word. However, this method didn't offer us good results because the category of different words is too numerous. We tried to use stop-words list but even that didn't work because there are too many words should be put into it and we just cannot cover them all. For all meaningless words, we apparently need find some way to clear it, and even for these meaningful words, we still need some algorithm to classify them or we will get too many topics in the end.

So, we came out with another solution: we vectorized the articles and tried to get rid of the influence of meaningless words for drawing conclusion like "said", "Boston" or "Monday" from two aspects. The first one was to remove all the words in our keywords used for filtering and in stop-word list since these are a common part of relevant articles. The second one was to apply tf-idf algorithm on them. As we learnt in class, tf-idf was born to deal with this problem. After these steps, what remained was a classification problem. We tried different models we used before and choose Latent Dirichlet Allocation (LDA) to do the job because it worked pretty well with text classification problems. And the package also provides a nested dimensionality reduction API, so that we can make our result easy-to-understand, which was quite clear to evaluate and show our results.

III. Findings & Results & Observations:

Data:

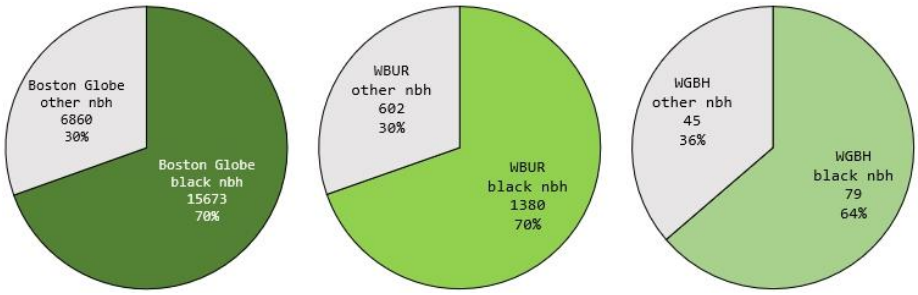
We've collected all data we need from all three websites (Boston Globe, WGBH and WBUR). The whole dataset contains all articles (news) on their website from 2014 to 2018. The total number of articles is about 230,000 and the data size is about 1GB. There are about 200,000 pieces of news collected for Boston Globe, about 30,000 articles for WBUR and only 1500 for WGBH because the Wayback Machine didn't keep much copies of it and some of its links are broken.

Coverage:

We used only neighborhood names, not other words as keywords to calculate the coverage. For Boston Globe, the coverage (number of articles about black neighborhoods / number of articles about all neighborhoods) is about 70% and hasn't changed much during the last 5 years. For WUBR, the average coverage is also about 70%, but it seems to drop quite a lot in recent 2 years (from 78.9% in 2016 and 58.9% in 2018). Because there are only about 1500 articles from WGBH, so we didn't classify them with years, and WGBH has an overall coverage of 63.7%.

	black nbh	all nbh	all news	coverage
bg14	1509	2181	18597	0.691884
bg15	2276	3273	27112	0.695386
bg16	2636	3873	35761	0.680609
bg17	3486	4932	42114	0.706813
bg18	5766	8274	69646	0.696882
wbur14	194	261	4872	0.743295
wbur15	242	310	3814	0.780645
wbur16	300	380	6558	0.789474
wbur17	338	512	7213	0.660156
wbur18	306	519	6987	0.589595
wgbh	79	124	1506	0.637097

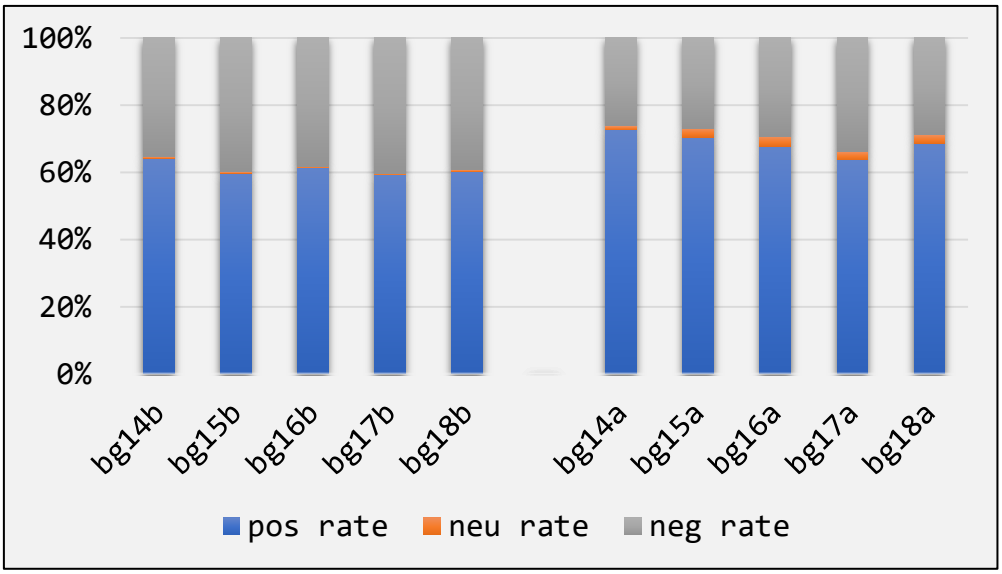
Statistics – coverage data

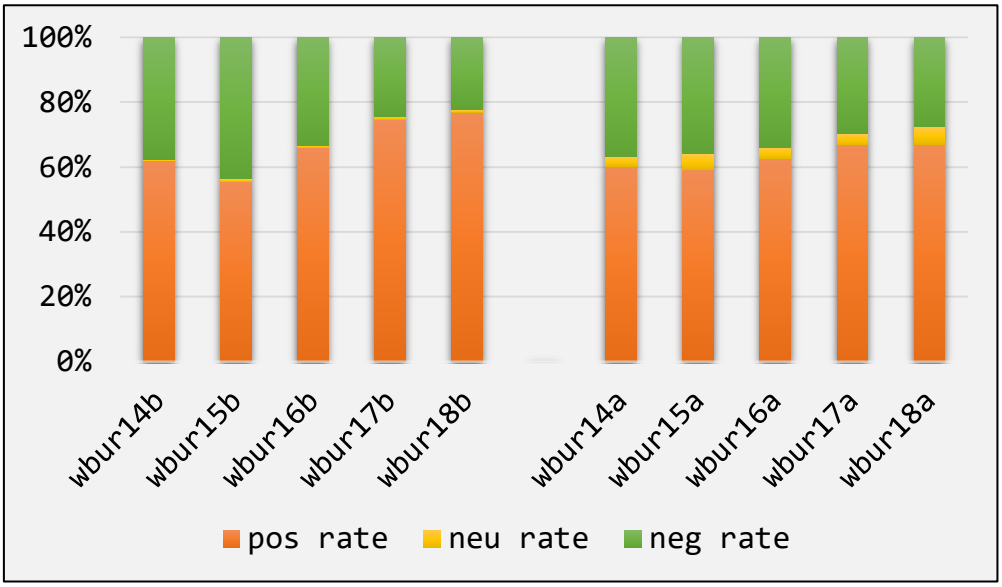


Pie chart – coverage results

From our data, although the coverage of black neighborhoods may change a little between different years, but the coverage percentage seems to stay at a high level (more than 50%). Considering the black population in Boston, we think the black neighborhoods are well covered by the Boston media.

Sentiment:





Bar chart – sentiment analysis results (Boston Globe and WBUR), left: black news, right: all news

As we can see from the charts above, Boston Globe's articles about black people has a lower positive rate and a higher negative rate than other articles. And it hasn't changed much over the past 5 years. For WUBR, the positive rate and negative rate for black news are at the same level as other news from 2014 to 2016, and the sentiment about black news become more positive when it comes to 2017 and 2018, higher than average. We didn't do sentiment analysis on WGBH because we thought its article number is not enough to draw any conclusion.

Topics:

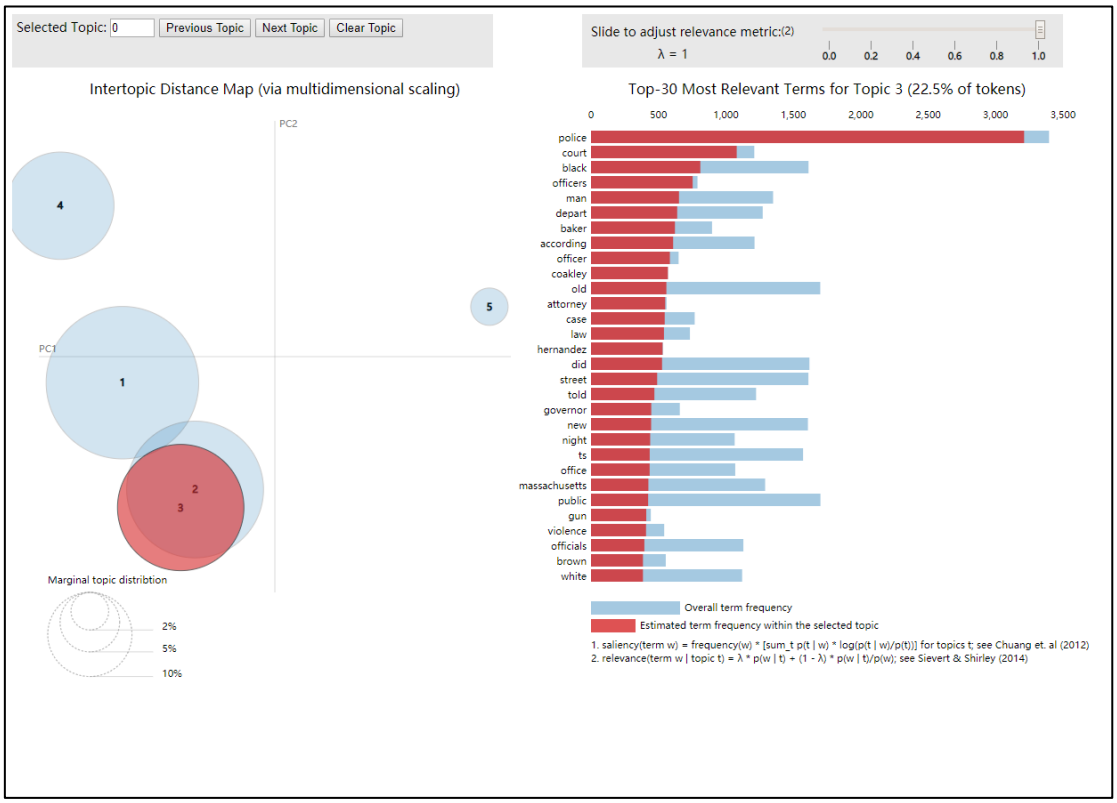
As we mentioned above, Boston Globe's news about black people has a lower positive rate and a higher negative rate than other news, which is not very good. What kind of news cause that situation? We tried to answer this question by collecting the most popular topics for each year.

2014 top 5 topics	2015 top 5 topics	2016 top 5 topics	2017 top 5 topics	2018 top 5 topics
police court black officers man	snow 2024 Olympic line south	school students black schools community	street team sox day	black white trump president American
percent public health students	school students percent schools public	police officers man old street	trump president white Jackson black	police street officers man court
council wall Keolis contract councilor	police officers street man court	Massachusetts public housing million	life children family father mother	school students schools public percent
world black new American African	black life white Hernandez day	day world new way old	school students percent million schools	Massachusetts health depart public office
school Menino mayor day family	work new building food home	Trump Clinton voters campaign president	police depart court officers according	life day world way

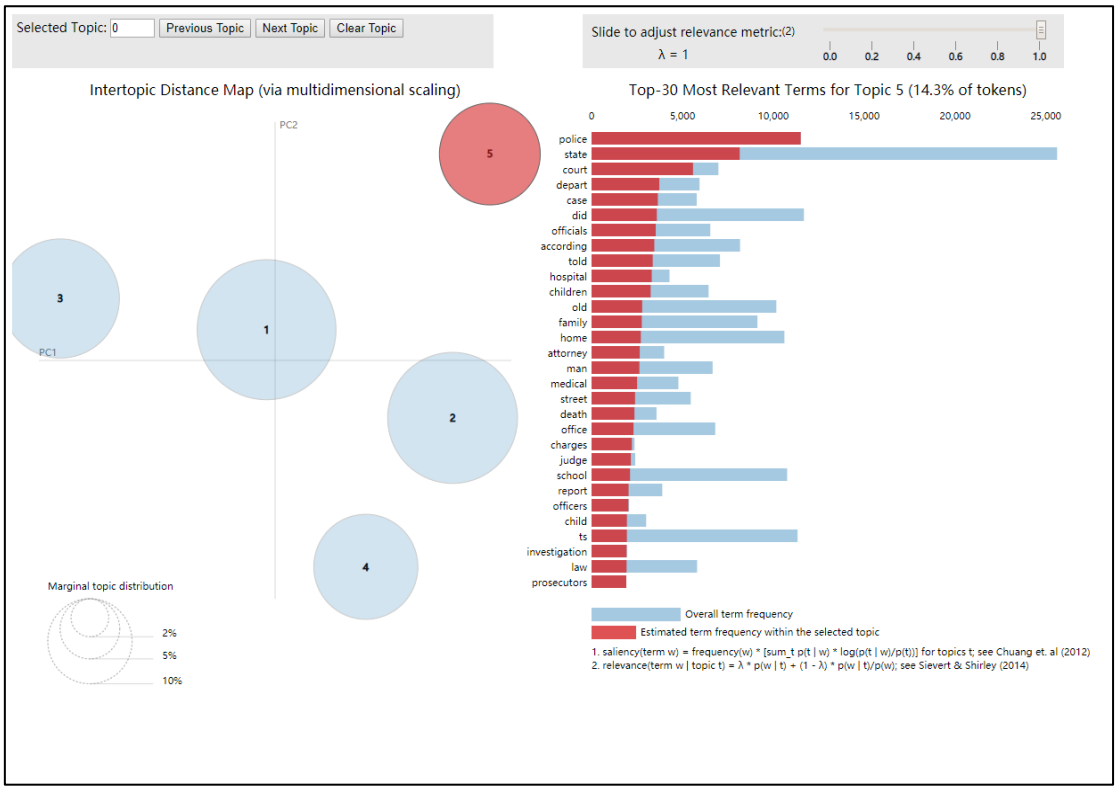
Top 5 topics (generated by LDA) of Boston Globe's black news, 2014~2018

As we can see from above, among these popular topics, the topics constructed by word "police", "court", "man" and some other words seems like the "crime" topic which we usually consider

negative. Then we looked the popular topics of all news and find “crime” topic too:



“Crime” topic in black news



“Crime” topic in all news

As we can see from these two figures, although topics about crime is popular both in black news and all news, there are still some difference. First one is the weight. The crime topic has a bigger proportion (22.5% of tokens) among all black news, while for all news this number is only 14.3%. You can see this from the area of the red circle too. The red circle in the top figure has a larger area, which means crime news appears more often in black news (like about 20 in 100 pieces of news) than in all news (like about 12 in 100 articles). And the keywords connected with the topic are different too. When look at these two figures, words like “gun”, “violence” and “night” can only be found in the top figure. These “bad” words may indicate a more serious crime, which will result in a more negative sentiment for the article. We shared our findings with the other team, and they got similar results, which is negative words (in their research about Boston Herald, negative words include “murder”, “shot”, “gun”, “killed” and “weapon”) appear more often in news about black neighborhoods.

IV. Conclusions:

In this project, we dug into Boston’s media data about black people, and tried to help our client, NAACP Boston, to understand the coverage of Boston Media in covering Boston’s Black people and Black neighborhoods. We collected data, designed algorithms to understand & analyze them, and tried to see how things have changed in the past 5 years. Then, we drew some conclusions that may answer our client’s questions. In the end, we generated some visualization results that makes our analysis visible and easy to understand.

All our data collected can be downloaded at:

https://drive.google.com/open?id=1B82_YZmT1HhK2BNwLuGcZZ-ZhcN06ESn

All our code and file results can be found at:

https://github.com/AllenChenGH/NAACP_MEDIA_RESEARCH

We sincerely thank Tanisha, John and Ziba for their patience and help.

Thank you!