# Implementing Natural Language Processing to Examine the Lens of News Media

Final Project Report in the course Introduction to Cultural Data Science

Anders Havbro Hjulmand (201910185)

Cognitive Science, School of Communication and Culture

University of Aarhus, Jens. Chr. Skous Vej 2, 8000 Aarhus C, Denmark

Lecturer: Adéla Sobotkova

December 20th, 2021

# Abstract

We learn about events in the world through the lens of news media. This lens is not neutral because news media adheres to principles that dictate the newsworthiness of an event. The aim of this paper is to investigate the prevalence of these principles by using natural language processing on a large corpus of news articles from two major newspapers. Results indicate that the two newspapers follow the principle of proximity, demonstrating that the lens of news media is far from neutral. This paper also demonstrates how natural language processing can be useful in the study of news media.

**Keywords**

Natural language processing, text mining, newspaper, news media.

**Github**

The code and data for the project can be found in the author's Github:
https://github.com/ah140797/news_media_NLP

**Reproducibility of Code**

A thoroughly commented version of the code in the format of an r-markdown notebook can be found in the author's Github. Instructions on how to open the notebook can also be found in the author's Github.

# TABLE OF CONTENTS

# Introduction

We learn about events in the world through the lens of news media. This lens is not neutral because news media adheres to principles that dictate the newsworthiness of an event. News media construct reality and has the potential to inflate less important events and deflate more important events (Gamson et al., 1992). The motivation of this paper is to investigate the prevalence of these principles and how widespread they are. The findings in this paper are important for understanding the lens of news media and how it is shaping our culture. The motivation of this paper is also to show how natural language processing can be useful in the study of news media.

## Theoretical Background

News articles are more likely to cover events that adhere to certain principles. A widespread criteria of news coverage is *proximity*, which consists of two components. 1) The geographical distance between an event and the office of the newspaper. News media select stories that are in near geographical proximity to the audience. 2) *Localization,* which is the extent to which an event has meaning for the audience. For example, if a country is engaged in a war on the other side of the world, the war will still be covered even though it is far away from the office of the newspaper. It has been shown that proximity increase the newsworthiness of an event. Therefore, proximity is a predictor of which events will be covered in the news (Devereux, 2007 p. 231-234).

Natural language processing (NLP) is the employment of computers to help understand natural language text or speech. The application of NLP is widespread and the usage of tools from NLP are increasing. (Chowdhury, 2003).

Named Entity Recognition (NER) is an NLP-technique used to classify entities into categories such as persons, locations, verbs etc. (Mohit, 2014). The NER-algorithm used in this paper has been trained on a corpus and learned to recognize entities and classify them correctly based on the context. There are many categories of entities. For the purpose of this paper the following entities are extracted: Countries, persons, organizations, nouns, verbs and adjectives.

Sentiment analysis is an NLP-technique which identifies the emotional tone in a text (Medhat et al., 2014). Two measures are interesting for the purpose of investigating the lens of news media. 1) *polarity* which indicates whether a text is positive, negative, or neutral and 2) *subjectivity* which indicates whether a text is subjective or objective.

## The Aim of the Paper

In order to investigate the principle of proximity and the lens of news media, a large corpus of news articles from two major newspapers will be compared. New York Times, which is an American newspaper based in New York and The Guardian which is a British newspaper based in London. In order to enable a more robust comparison between the newspapers, only news articles which relates to the Taliban conflict will be used. The Taliban conflict is used because both the United States and United Kingdom have been heavily engaged in the conflict for a duration of about 20 years.

Based on the theory it is hypothesized that:

**Hypothesis 1 (H1):** According to the principle of geographical distance it is expected that American presidents and United States appear more frequently in New York Times compared to The Guardian. It is also expected that British prime ministers and United Kingdom appear more frequently in The Guardian.

**Hypothesis 2 (H2):** According to the principle of localization it is expected that Afghanistan presidents, key terrorists and key organizations appear with similar frequency in New York Times and The Guardian.

**Hypothesis 3 (H3):** Both the United States and United Kingdom have been allies and shared many of the same agendas in the Taliban conflict (*Afghanistan War | History, Combatants, Facts, & Timeline | Britannica*, n.d.). It is therefore expected that there is no difference in polarity and subjectivity between New York Times and The Guardian.

Moreover, as an exploratory exercise the top 5 nouns, verbs and adjectives will be examined.

## Software Framework

The code for the project was written on the author’s Lenovo Ideapad S340, which runs Windows 10 operating system.

Two programming languages were used to write the code: R (v4.0.3; R Core Team, 2020) and Python (v3.6.6; Rossum, 2009). Each language provides different software packages that are used for different analysis. RStudio (v1.4.1103; RStudio Team, 2020) and Jupyter Notebook (v6.4.3; Kluyver, 2016) were used as integrated development enviroments (IDE). Jupyter Notebook was used in the enviroment of Anaconda (v4.10.3; Anaconda Software Distribution, 2020).

The following software packages were used for R: tidyverse (v1.3.0; Wickham et al., 2019), jsonlite (v1.7.2, Ooms J, 2014), plotly (v4.9.3; Sievert, 2020), tigris (v1.5, Kyle Walker, 2021), sf (v1.0-4; Pebesma, 2018), RColorBrewer (v1.1-2; Neuwirth 2014), shiny (v1.6.0; Chang, 2021), leaflet (v2.0.4.1; Cheng, 2021), htmlwidgets (v1.5.3; Vaidyanathan, 2020), popcircle (v0.1.0; Giraud, 2020), patchwork (v1.1.1; Pedersen, 2020) and wesanderson (v0.3.6; Ram, 2018).

The following software packages were used for Python: beautifulsoup4 (v4.10.0; Richardson, 2007), requests (v2.26.0; Reitz 2014), pandas (v1.3.2; McKinney, 2010), numpy (v1.20.3; Harris, 2020), spacy (v3.1.3; Honnibal, 2017), genism (v4.1.2; Rehurek, 2011) and pyldavis (v3.3.1; Sievert, 2014).

The accompanying code to the paper is presented as a notebook using r-markdown and the package bookdown (v0.21; Xie, 2020). This format makes it easy to read the code and therefore improves the reproducibility of the project. Throughout this paper, this notebook is referred to as *reproducibility notebook* and can be found in the author's Github.

# Data Acquisition and Preprocessing

Two main datasets were used. One dataset containing news articles from New York Times and another dataset containing news articles from The Guardian. In the following section, the acquisition and preprocessing of both datasets are outlined with reference to chapters in the reproducibility notebook.

## Acquisition of Data

Technical pipeline for New York Times:

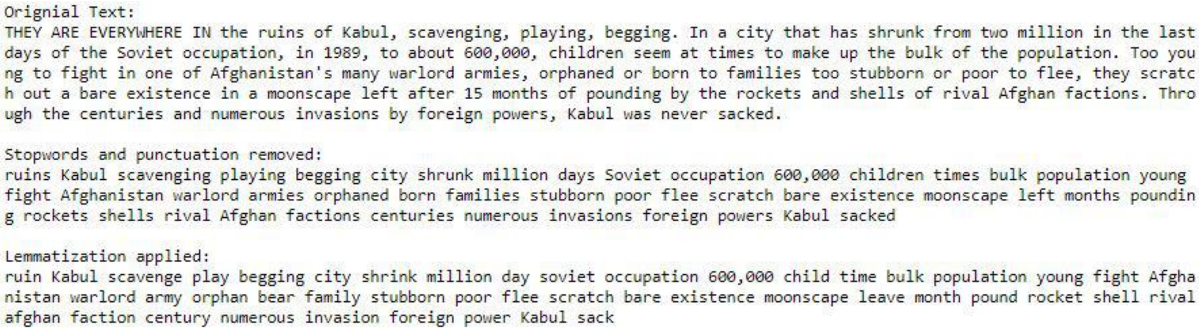
1. New York Times has developed an application programming interface (API). Follow this link to make an account and get started: https://developer.nytimes.com/. The "Article Search API" was used with the q query parameter "Taliban" to fetch all articles containing the keyword "Taliban". See chapter 1 in the reproducibility notebook.
2. The dataset was tidied using R and the package tidyverse. See chapter 2 in the reproducibility notebook.
3. The body texts of the articles were scraped using Python and the package beautifulsoup4. See chapter 3 in the reproducibility notebook.

Technical Pipeline for The Guardian:

1. The Guardian has developed an API. Follow this link to make an account and get started: https://open-platform.theguardian.com/access/. The query parameter "Taliban" was used to fetch all articles containing the keyword "Taliban". See chapter 4 in the reproducibility notebook.
2. The dataset was tidied using R and the package tidyverse. See chapter 5 in the reproducibility notebook.
3. The body texts of the articles were scraped using Python and the package beautifulsoup4. See chapter 6 in the reproducibility notebook.

## Preprocessing of Data

The body texts of all articles from both newspapers were preprocessed in three steps using the package spacy in Python. First, punctuation and special characters were removed. Then, stopwords were removed. Stopwords are those words that do not provide any useful information in the analysis. Lastly, all words were lemmatized. Lemmatization is the process of returning all words to their basic form. For example, the verb “to talk” may appear as “walk”, “walked”, or “walking”. Lemmatization returns all these forms to their base form “walk”. See figure 1 for an application of all three preprocessing steps. See also chapter 7 in the reproducibility notebook.



```
Orignial Text:
THEY ARE EVERYWHERE IN the ruins of Kabul, scavenging, playing, begging. In a city that has shrunk from two million in the last
days of the Soviet occupation, in 1989, to about 600,000, children seem at times to make up the bulk of the population. Too you
ng to fight in one of Afghanistan's many warlord armies, orphaned or born to families too stubborn or poor to flee, they scratc
h out a bare existence in a moonscape left after 15 months of pounding by the rockets and shells of rival Afghan factions. Thro
ugh the centuries and numerous invasions by foreign powers, Kabul was never sacked.

Stopwords and punctuation removed:
ruins Kabul scavenging playing begging city shrunk million days Soviet occupation 600,000 children times bulk population young
fight Afghanistan warlord armies orphaned born families stubborn poor flee scratch bare existence moonscape left months poundin
g rockets shells rival Afghan factions centuries numerous invasions foreign powers Kabul sacked

Lemmatization applied:
ruin Kabul scavenge play begging city shrink million day soviet occupation 600,000 child time bulk population young fight Afgha
nistan warlord army orphan bear family stubborn poor flee scratch bare existence moonscape leave month pound rocket shell rival
afghan faction century numerous invasion foreign power Kabul sack
```

*Figure 1: Preprocessing body text of articles using spacy.*

The pipelines resulted in two tidy datasets for New York Times (n=21.107) and The Guardian (n=9.609) respectively, containing columns for headline, date of publication, body text and the preprocessed body text.

In addition to the main datasets, another collection of geographical datasets was acquired. These datasets are called shapefiles and were downloaded using the package tigris in R. These shapefiles contain information on the borders of countries and were used to make maps.

# Analysis

Two NLP analysis were conducted using the library spacy in Python: NER and sentiment analysis. NER was used to explore hypotheses H1 and H2. For a detailed walkthrough of the code associated with NER see section IV in the reproducibility notebook (chapter 8, 9, 10,11 and 12). The algorithm “spacytextblob” from spacy were used to conduct the sentiment analysis. Two measures were extracted for each article: Polarity and subjectivity. Polarity is an integer within the range [-1, 1] where -1 is very negative and 1 is very positive. Subjectivity is an integer within the range [0, 1] where 0 is very objective and 1 is very subjective. For a detailed walkthrough of the code associated with sentiment analysis see section V in the reproducibility notebook (chapter 13 and 14).

# Results

## Timeline of published articles

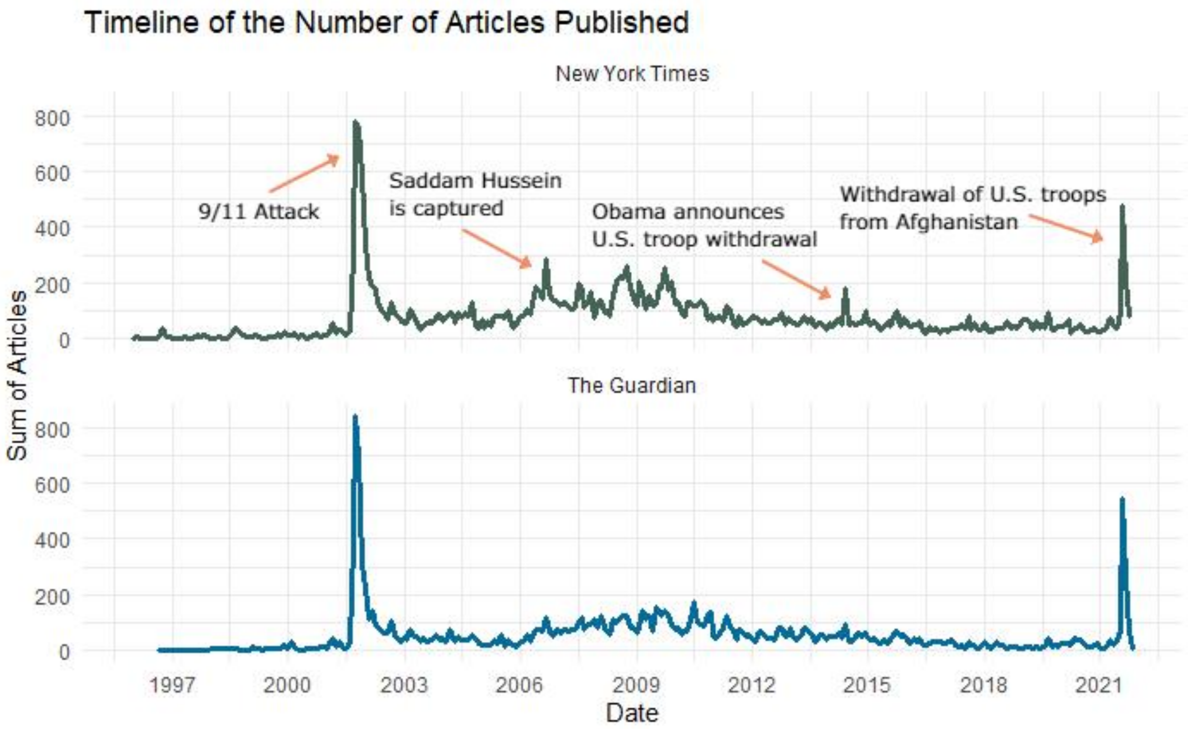


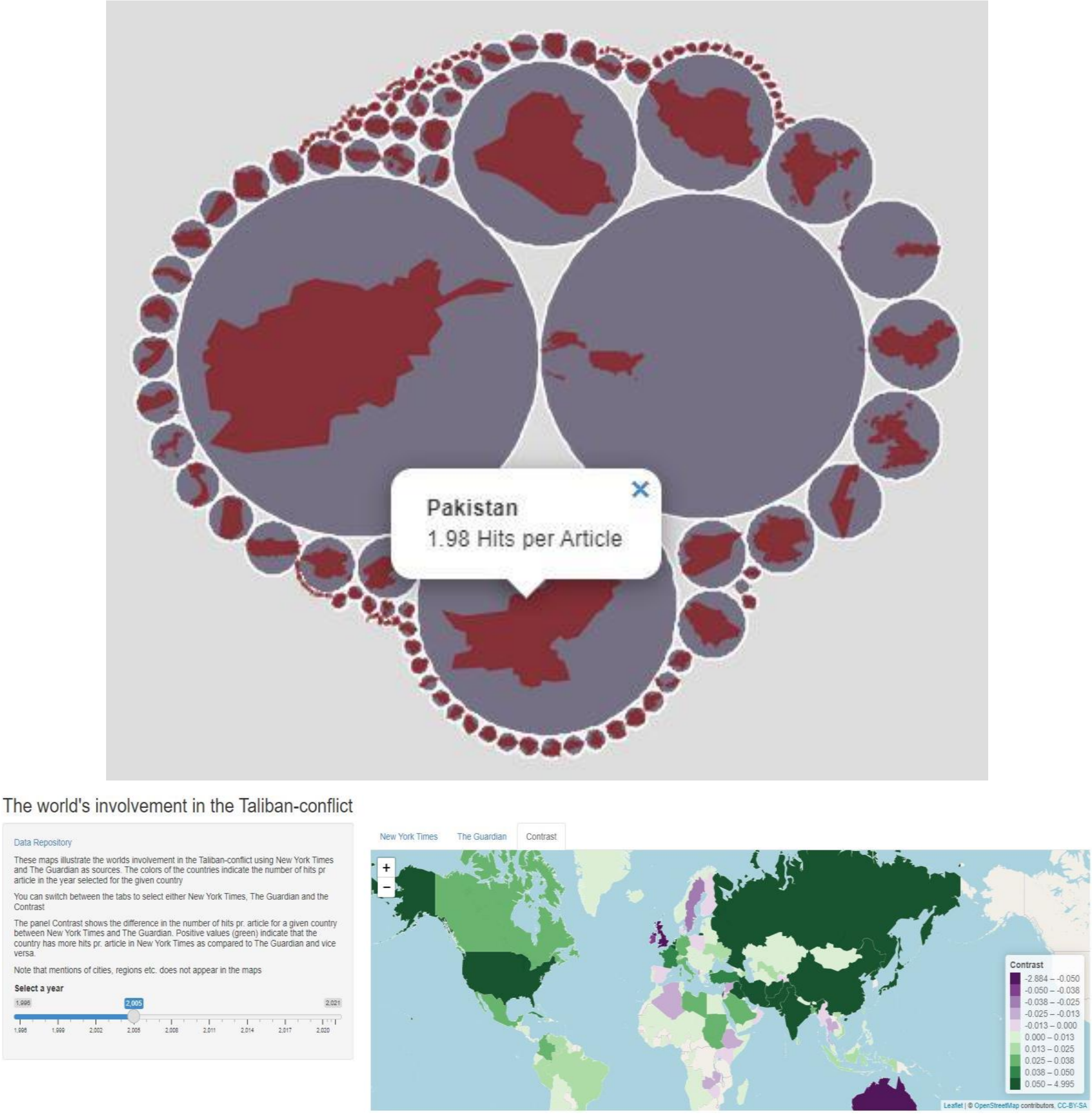
*Figure 2: TImeline of the number of articles published for New York Times and The Guardian. The trend lines show the sum of articles published per month.*

Figure 2 shows the sum of articles published per month as a timeline. Both newspapers have a quite similar timeline, showing an increased number of publications around events such as 9/11 that are relevant for the audience of both the United States and United Kingdom.

## Maps

Two interactive maps were created, to illustrate how the two newspapers focus on different countries. Figure 3 show examples of the interactive maps.

- A popcircle map showing the general focus on countries in the two newspapers. The map can be found in chapter 9.4 in the reproducibility notebook.
- A Shiny app showing the focus on countries in the two newspapers as a timeline. The app is hosted online: https://ah140797.shinyapps.io/shinyapp/. If the link doesn't work, please refer to the author's Github for a guide on how to run the app.



The world's involvement in the Taliban-conflict

Data Repository

These maps illustrate the worlds involvement in the Taliban-conflict using New York Times and The Guardian as sources. The colors of the countries indicate the number of hits pr article in the year selected for the given country

You can switch between the tabs to select either New York Times, The Guardian and the Contrast

The panel Contrast shows the difference in the number of hits pr. article for a given country between New York Times and The Guardian. Positive values (green) indicate that the country has more hits pr. article in New York Times as compared to The Guardian and vice versa.

Note that mentions of cities, regions etc. does not appear in the maps

Select a year: 2,005

*Figure 3: Examples of the interactive maps created with popcircle and shiny.*

The maps show that New York Times has more hits per article for United States and The Guardian has more hits per article for United Kingdom. Nonetheless, the newspapers are quite similar in their focus on other key countries such as Afghanistan, Pakistan, Iraq, Iran and India.

## Persons

### United States Presidents

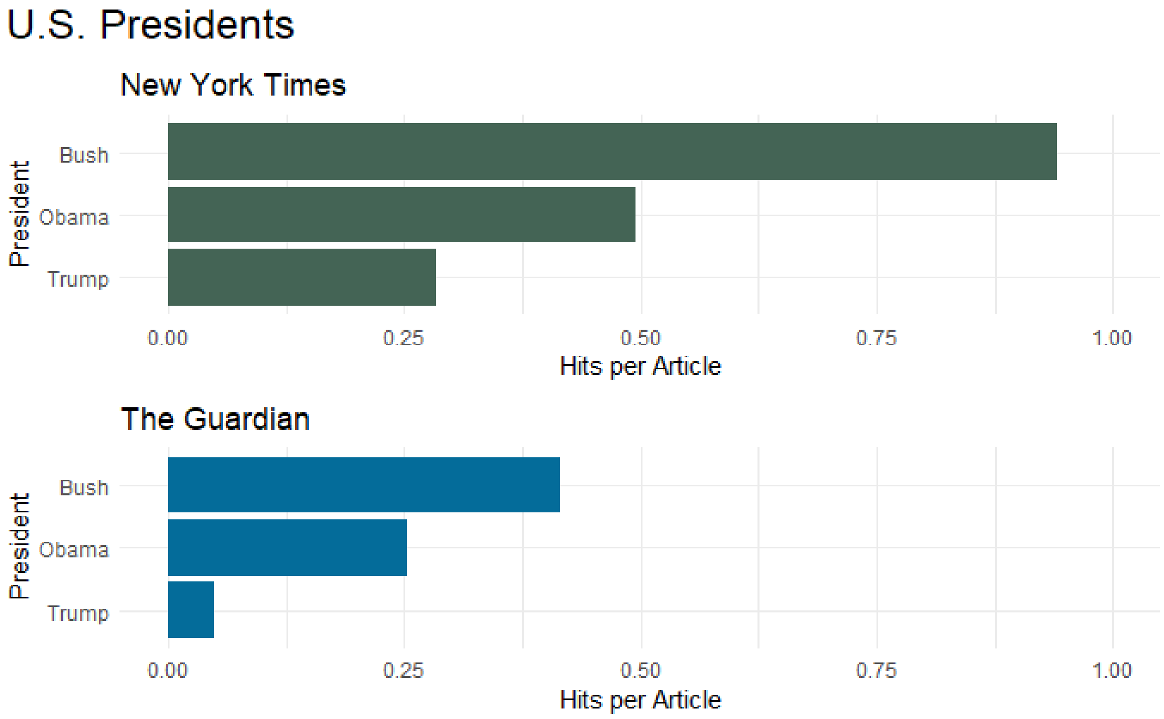


*Figure 4: The mean number of hits per article for each U.S. president and for each newspaper.*

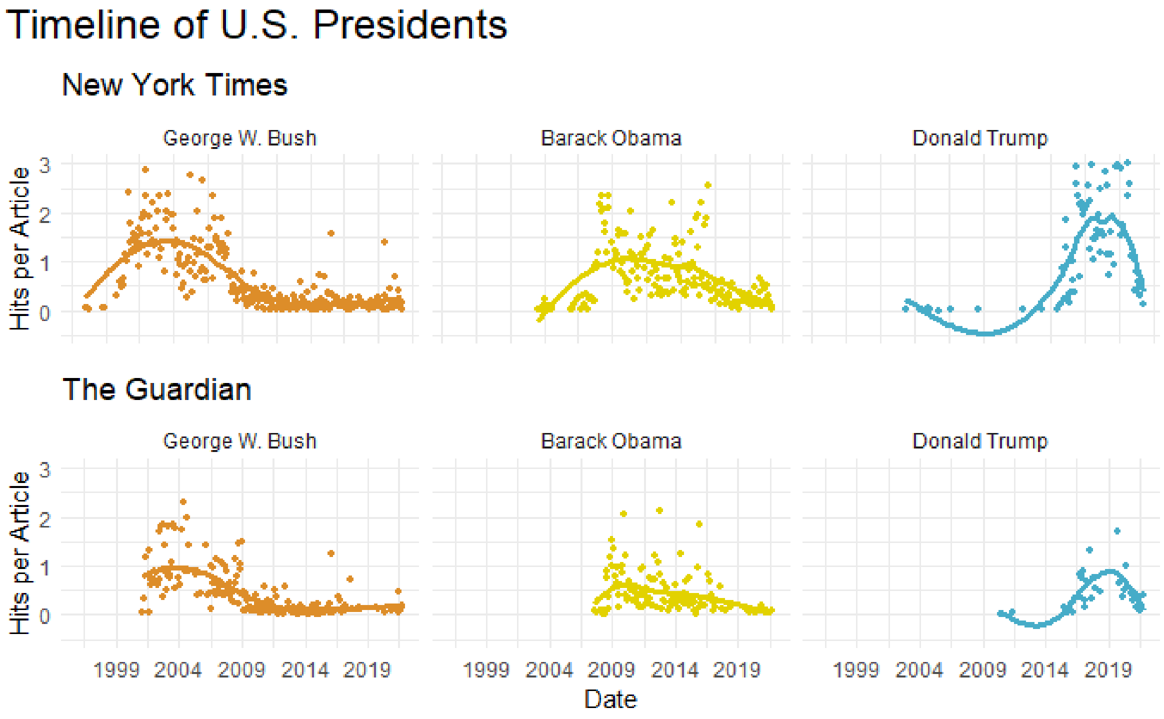


*Figure 5: Timeline of hits per article for each president and for each newspaper. The points represent the mean number of hits pr article per month.*

The three presidents of United States, George W. Bush, Barack Obama and Donald Trump have more hits per article in New York Times compared to The Guardian (see figure 4). However, figure 5 shows that New York Times and The Guardian show a similar interest in each president at corresponding dates.

## British Prime Ministers

Note that the NER from spacy did not grab Theresa May (British prime minister 2016-2019).

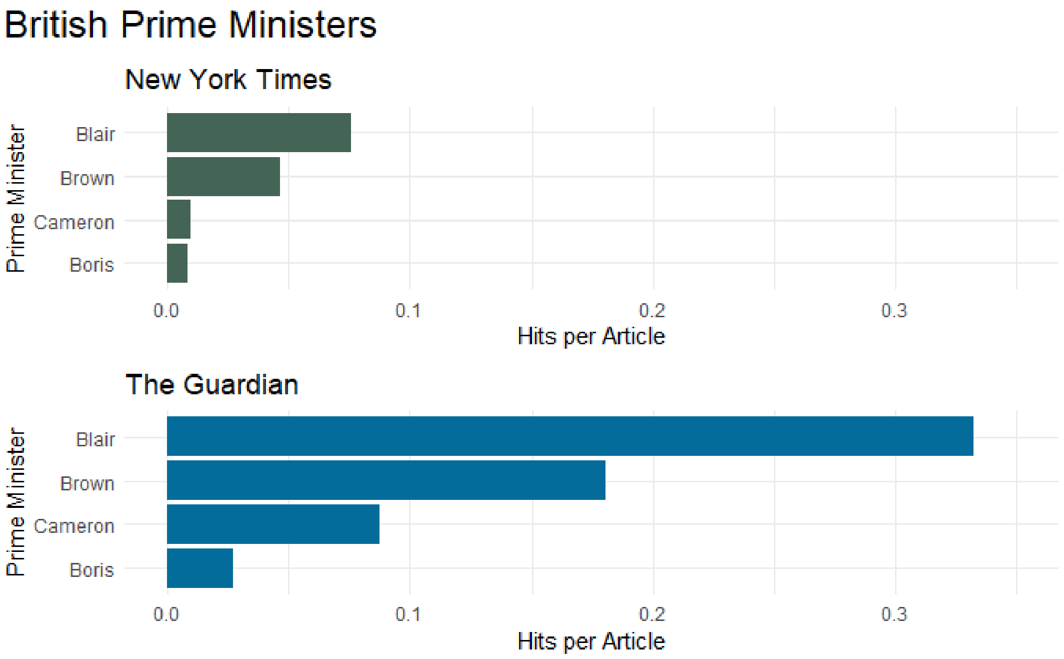


*Figure 6: The mean number of hits per article for each British prime minister and for each newspaper.*

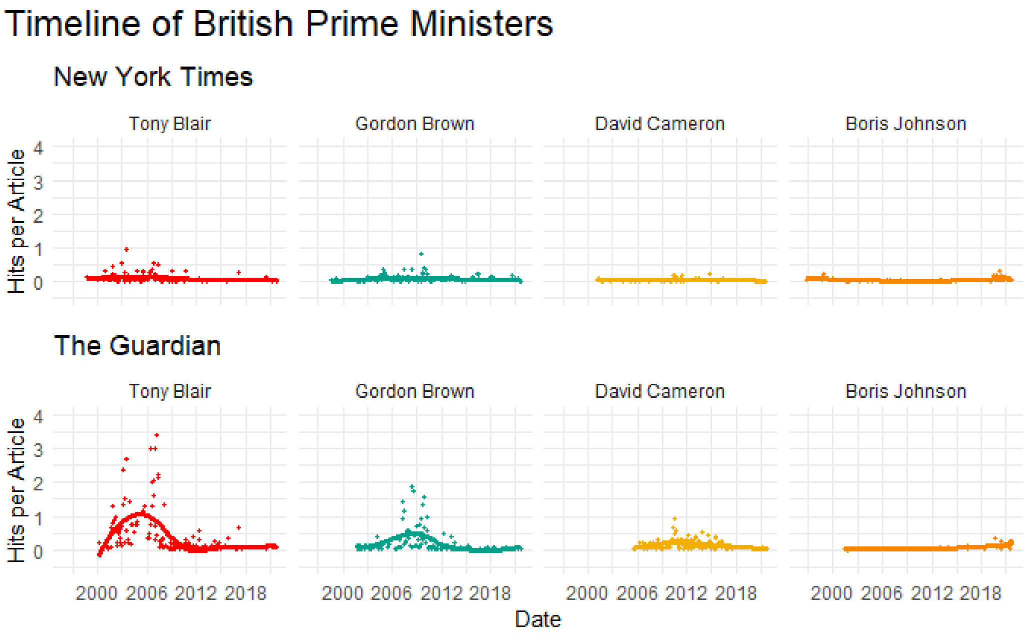


*Figure 7: Timeline of hits per article for each British prime minister and for each newspaper. The points represent the mean number of hits pr article per month.*

The four British prime ministers, Tony Blair, Gordon Brown, David Cameron and Boris Johnson has more hits per article in The Guardian compared to New York Times (see figure 6). However, figure 7 shows that New York Times and The Guardian show an interest in each prime minister at corresponding dates.

### Afghanistan Presidents

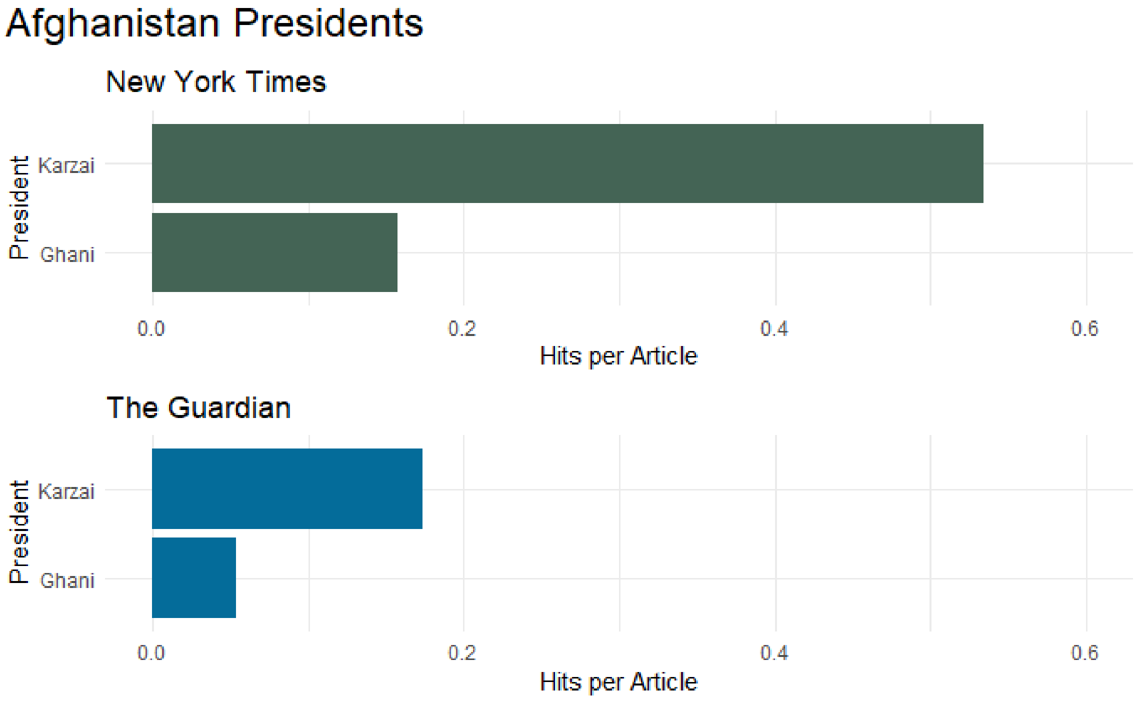


*Figure 8: The mean number of hits per article for each Afghanistan president and for each newspaper.*

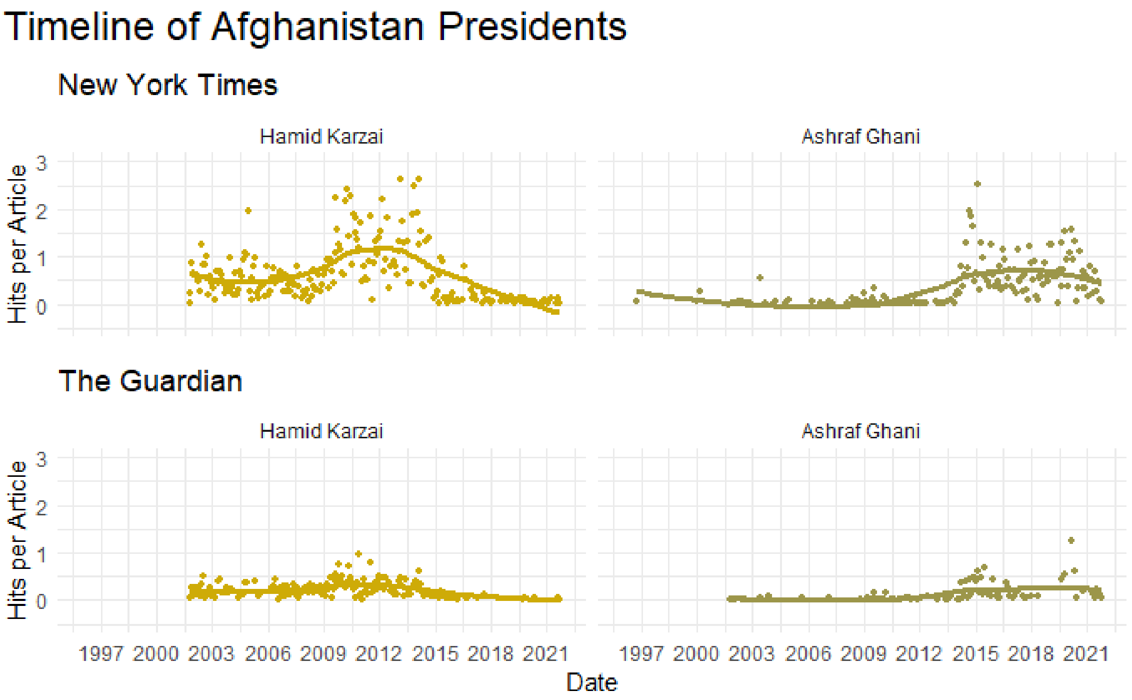


*Figure 9: Timeline of hits per article for each Afghanistan president for each newspaper. The points represent the mean number of hits pr article per month.*

The two Afghanistan Presidents, Hamid Karzai and Ashraf Ghani has more hits per article in New York Times compared to The Guardian (see figure 8). However, figure 9 shows that New York Times and The Guardian show an interest in each president at corresponding dates.

## Terrorists

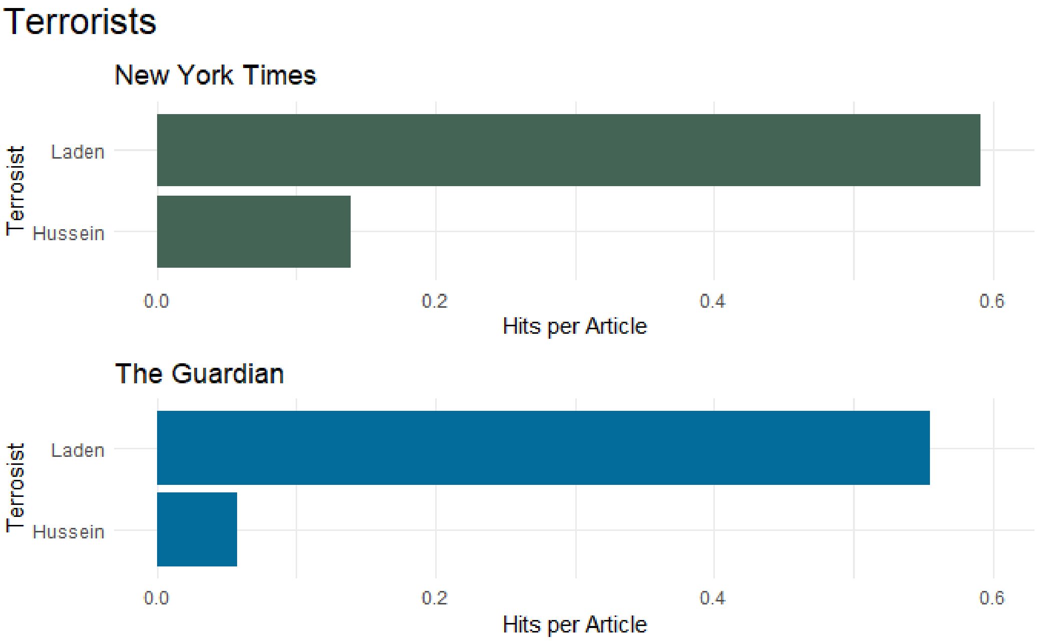


*Figure 10: The mean number of hits per article for each terrorist and for each newspaper.*

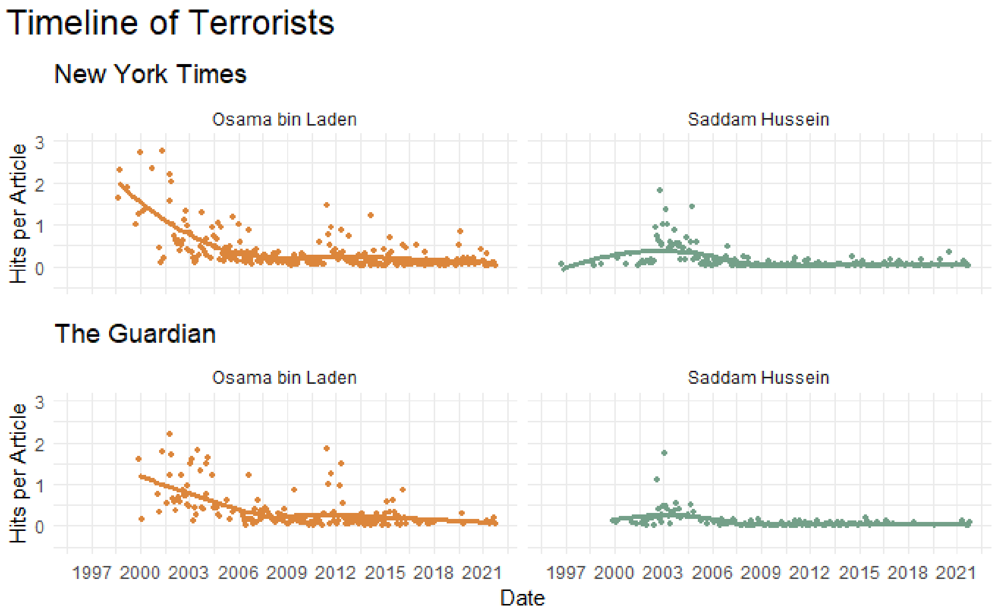


*Figure 11: Timeline of hits per article for each terrorist for each newspaper. The points represent the mean number of hits pr article per month.*

The two Terrorists, Osama bin Laden and Saddam Hussein has a similar amount of hits per article in New York Times and The Guardian (see figure 10). Moreover, figure 11 shows that New York Times and The Guardian show an interest in each terrorist at corresponding dates.

## Organizations

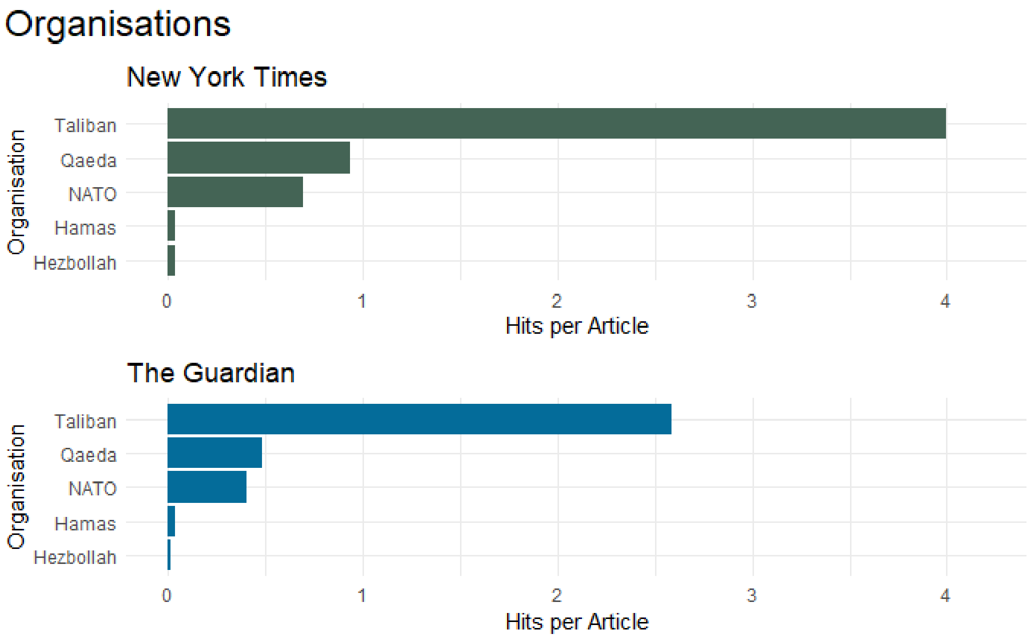


*Figure 12: The mean number of hits per article for each organization and for each newspaper.*

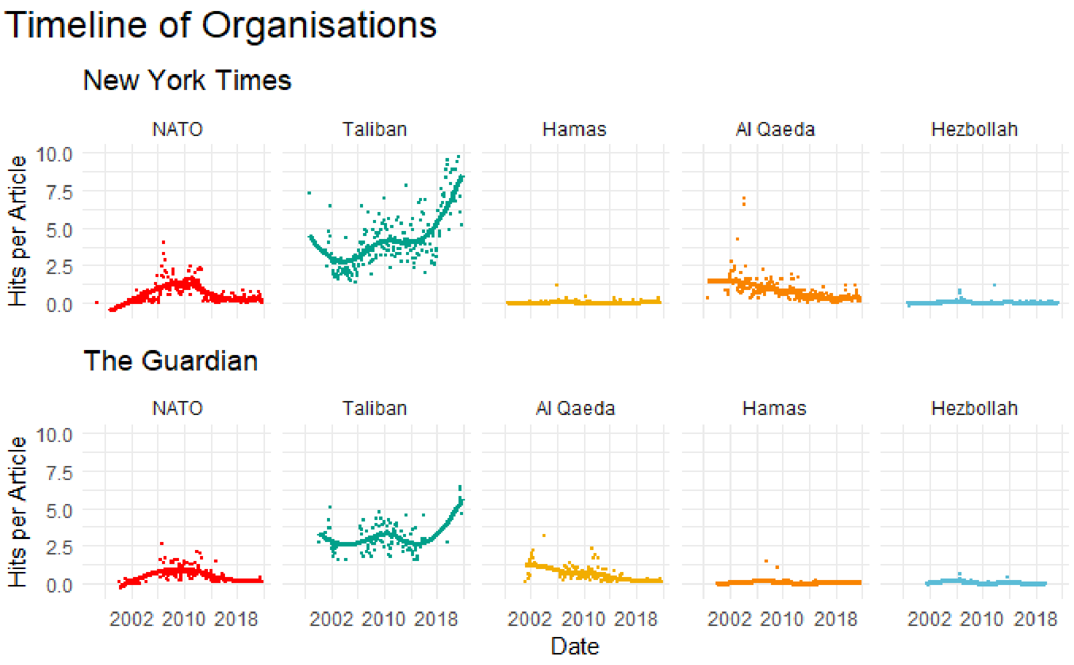


*Figure 13: Timeline of hits per article for each organization for each newspaper. The points represent the mean number of hits pr article per month.*

The three organizations, Taliban, Al Qaeda and NATO has more hits per article in New York Times compared to The Guardian (see figure 12). However, figure 13 shows that New York Times and The Guardian show an interest in each organization at corresponding dates.

## Top 5 Nouns, Verbs and Adjectives

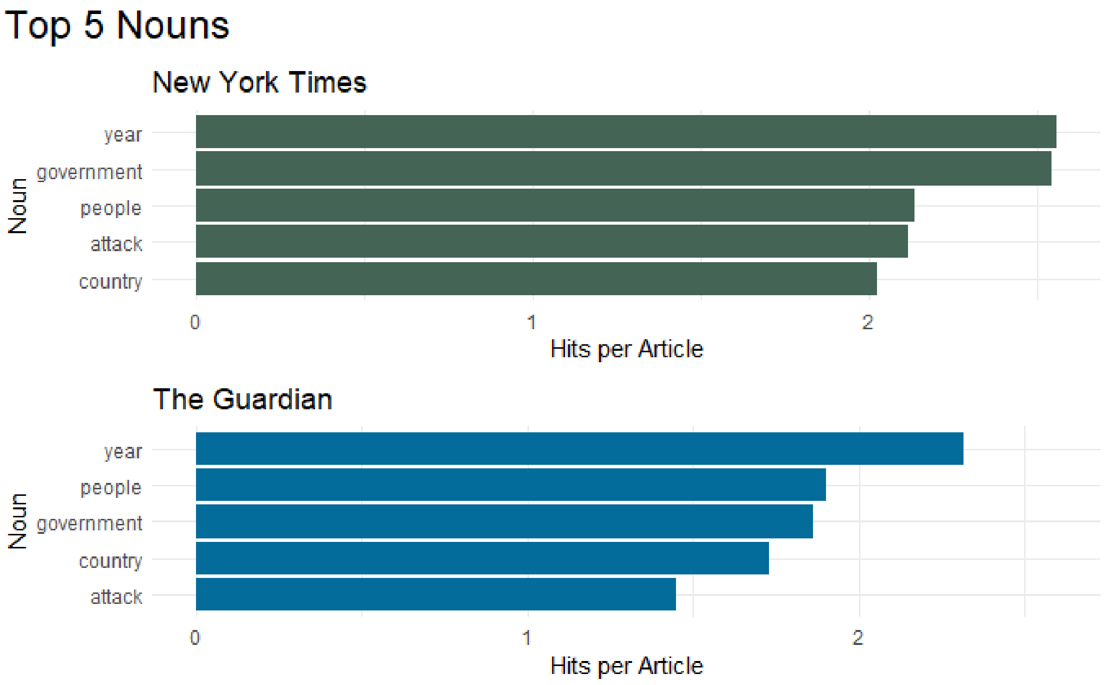


*Figure 14: The mean number of hits per article for the 5 most prevalent nouns for each newspaper.*

Figure 14 shows that the top 5 Nouns are very similar for New York Times and The Guardian.

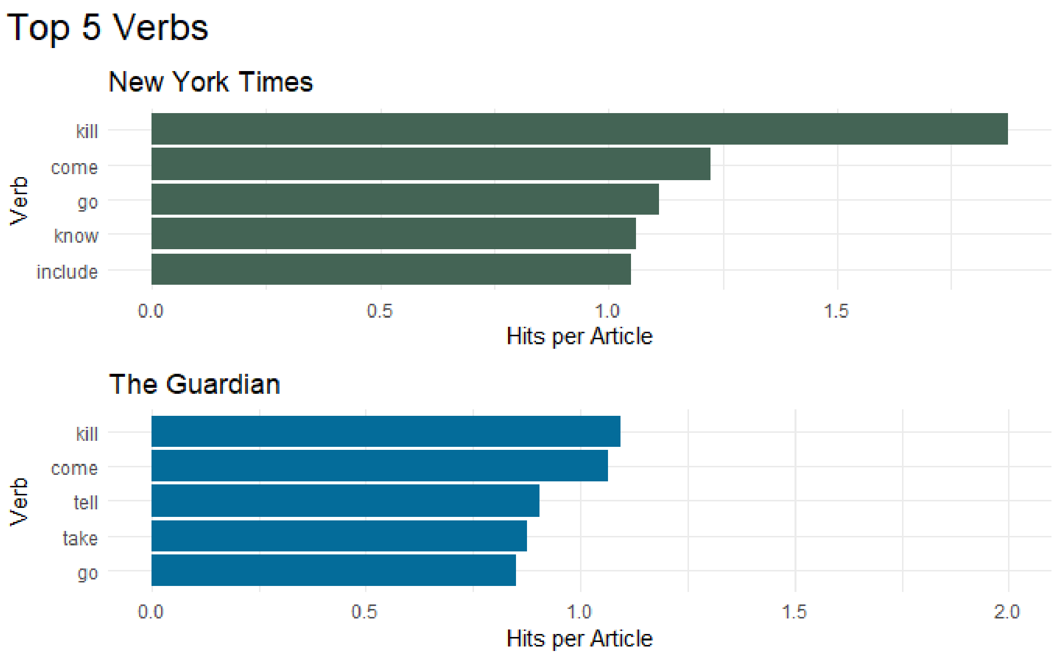


*Figure 15: The mean number of hits per article for the 5 most prevalent verbs for each newspaper.*

Figure 15 shows that the top 5 verbs are somewhat similar for New York Times and The Guardian. Nevertheless, New York Times show an increased usage of the verb "kill" compared to The Guardian.

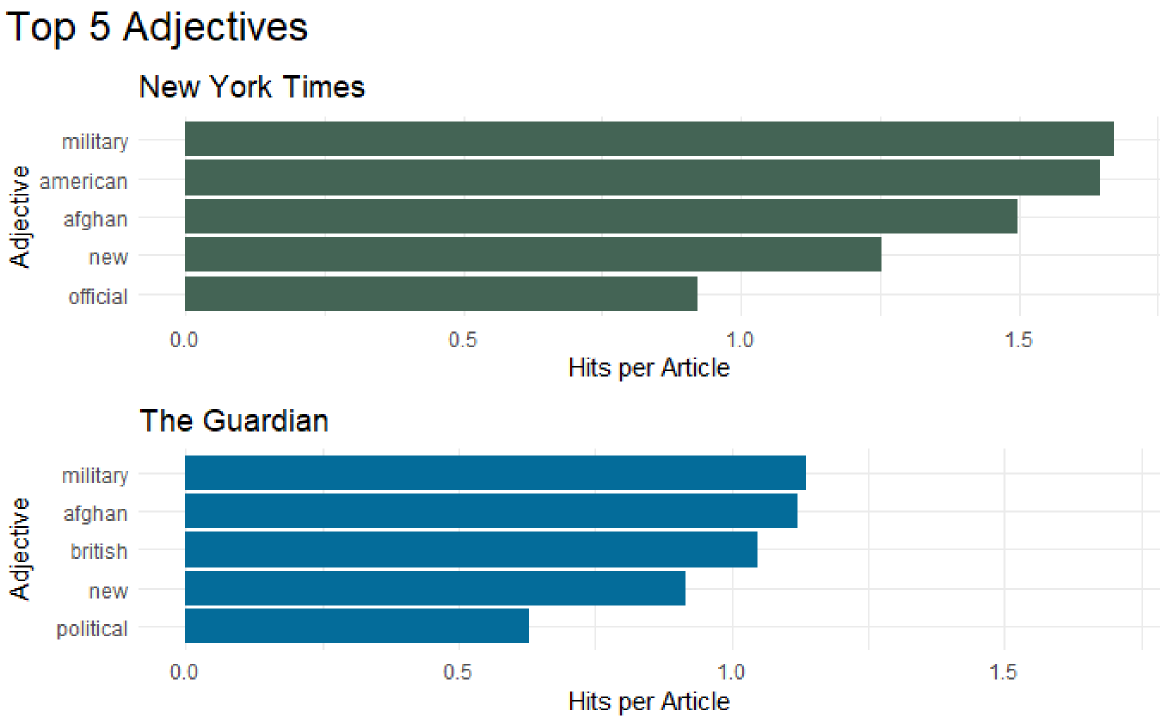


*Figure 16: The mean number of hits per article for the 5 most prevalent adjectives for each newspaper.*

Figure 16 shows that the top 5 adjectives are somewhat similar for New York Times and The Guardian. Nonetheless, each newspaper shows an increased usage of the adjective for their corresponding country.

## Sentiment Analysis

### Polarity

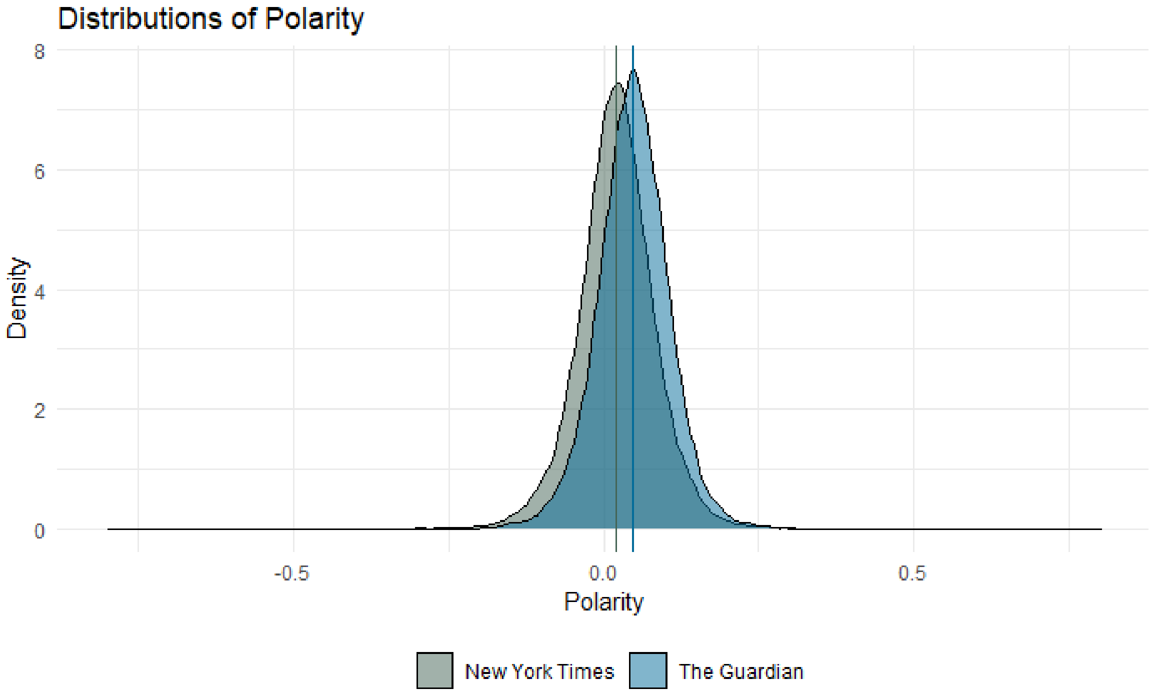


*Figure 17: Distributions of polarity for each newspaper respectively. Both distributions approximate a gaussian distribution.*

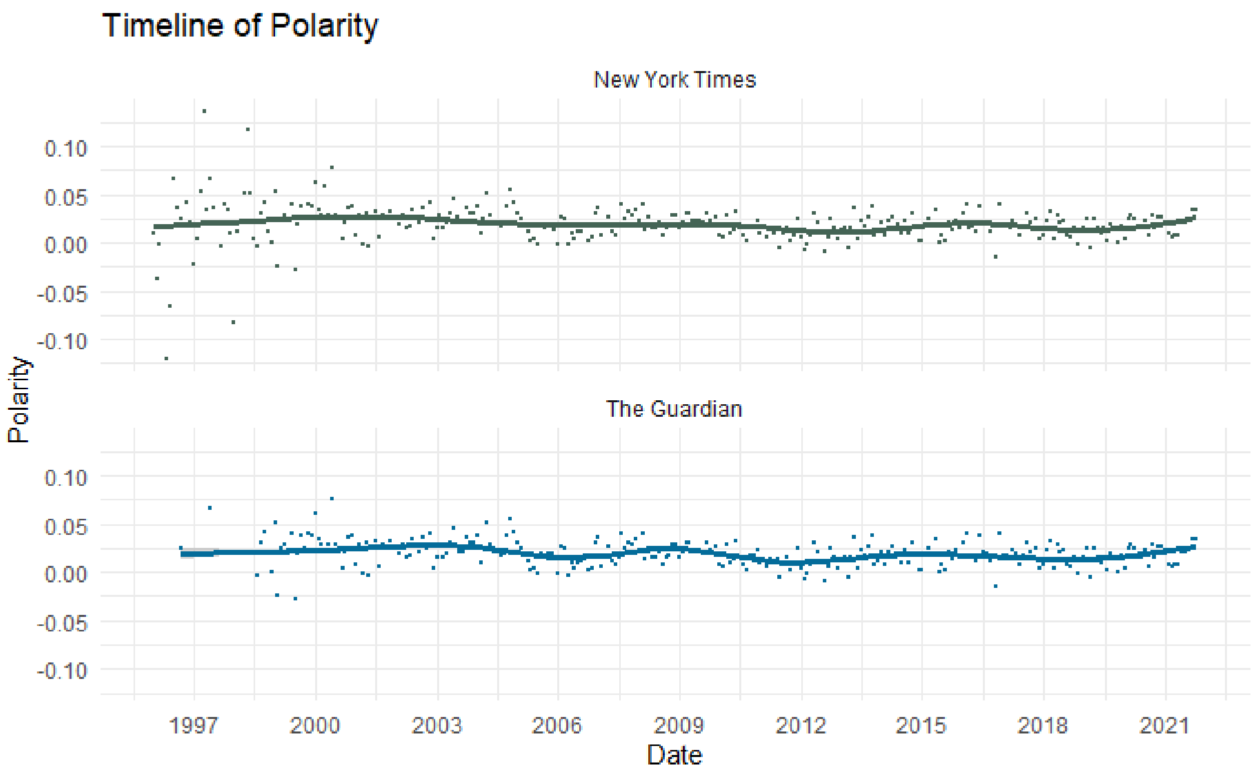


*Figure 18: Timeline of the polarity for each newspaper. The points represent the mean polarity per month. The line is fitted using the method "gam".*

The mean polarity of the articles in New York Times is 0.0204. The mean polarity of the articles in The Guardian is 0.0471. Both newspapers show a positive polarity. However, there is no substantial difference between the polarity of the newspapers (see figure 17). Figure 18 shows that the polarity of each newspaper corresponds quite well over time.

## Subjectivity

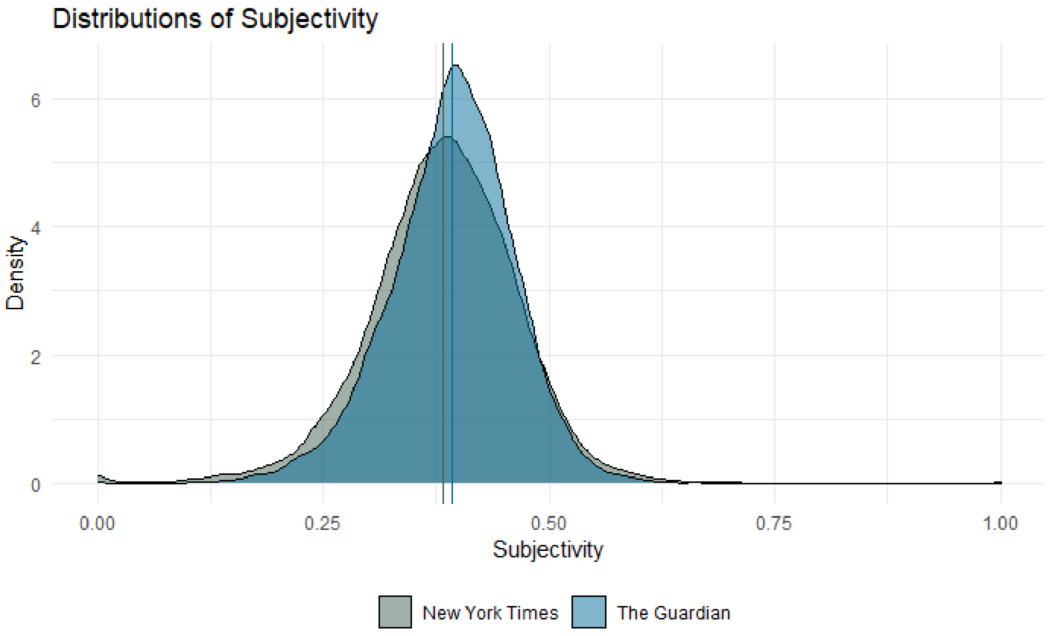


*Figure 19: Distributions of subjectivity for each newspaper respectively. Both distributions approximate a gaussian distribution.*

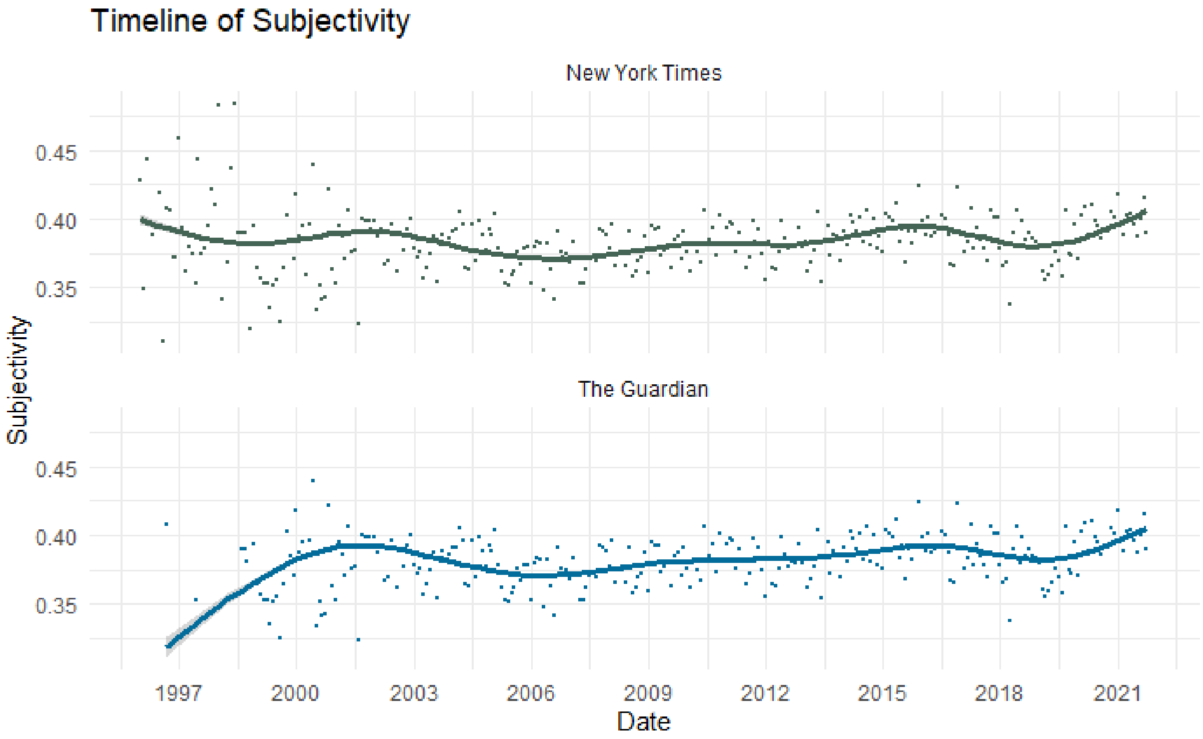


*Figure 20: Timeline of the subjectivity for each newspaper. The points represent the mean subjectivity per month. The line is fitted using the method "gam".*

The mean subjectivity of the articles in New York Times is 0.383. The mean subjectivity of the articles in The Guardian is 0.392. Both newspapers show a small to medium amount of subjectivity. However, there is no substantial difference between the subjectivity of the newspapers (see figure 19).

Figure 20 shows that the subjectivity of each newspaper corresponds quite well over time. Moreover, both newspapers show more subjectivity in the periods around the 9/11 attack and the withdrawal of U.S. troops from Afghanistan.

# Critical Evaluation

## Evaluation of hypotheses

Substantial evidence was found for hypothesis 1. Both newspapers show increased coverage of countries and persons in near geographical proximity to the newspaper's headquarters.
Some evidence was found for hypothesis 2. Two key terrorists appear with the same frequency in both newspapers. However, Afghanistan presidents and key organizations appear with higher frequency in New York Times. There is considerable evidence of localization, but the interests of the two newspapers are different in that regard.
Substantial evidence was found for hypothesis 3. No difference in polarity or subjectivity was found between the newspapers.

## Critique of research methods

There are some potential issues with the datasets. First, many articles were lost during the scraping of body text. This is a technical issue where improvements can be made so that more articles are scraped. Second, using the search query "Taliban" in the API can fetch articles that are not primarily about Taliban, but where Taliban only appear as an insignificant sidenote.

Moreover, the validity of the sentiment analysis can be put into question. The sentiment analysis was conducted using the algorithm "spacytextblob" which is quite simple and is not customized to analyzing news articles. A more sophisticated sentiment analysis would improve the reliability of the results (Balahur et al., 2013).

# Conclusion

This paper used NLP on a large corpus of news articles from two major newspapers to investigate the lens of news media. Substantial evidence was found for the principle of proximity, indicating that the two newspapers adhere to a principle that skew the representation of events in relation to the Taliban conflict. This demonstrates that the lens of news media is far from neutral. Future directions of the project could be to construct a topic model and also to investigate whether there are differences depending on the gender of the journalist. It would also be useful to incorporate a non-western newspaper such as Al Jazeera.

# References

*Afghanistan War | History, Combatants, Facts, & Timeline | Britannica*. (n.d.). Retrieved December 14, 2021, from https://www.britannica.com/event/Afghanistan-War

Anaconda Software Distribution. (2020). Anaconda Documentation. Anaconda Inc. Retrieved from https://docs.anaconda.com/

Balahur, A., Steinberger, R., Kabadjov, M., Zavarella, V., van der Goot, E., Halkia, M., Pouliquen, B., & Belyaeva, J. (2013). Sentiment Analysis in the News. *ArXiv:1309.6202 [Cs]*. http://arxiv.org/abs/1309.6202

*Bookdown citation info*. (n.d.). Retrieved December 13, 2021, from https://cran.r-project.org/web/packages/bookdown/citation.html

Chowdhury, G. G. (2003). Natural language processing. *Annual Review of Information Science and Technology*, *37*(1), 51–89. https://doi.org/10.1002/aris.1440370103

Devereux, E. (2007). *Media Studies: Key Issues and Debates*. SAGE.

Erich Neuwirth (2014). RColorBrewer: ColorBrewer Palettes. R package version 1.1-2. https://CRAN.R-project.org/package=RColorBrewer

Gamson, W. A., Croteau, D., Hoynes, W., & Sasson, T. (1992). Media Images and the Social Construction of Reality. *Annual Review of Sociology*, *18*(1), 373–393. https://doi.org/10.1146/annurev.so.18.080192.002105

Harris, C.R., Millman, K.J., van der Walt, S.J. et al. Array programming with NumPy. Nature 585, 357–362 (2020). DOI: 10.1038/s41586-020-2649-2. (Publisher link).

Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

Joe Cheng, Bhaskar Karambelkar andYihui Xie (2021). leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library. R package version 2.0.4.1. https://CRAN.R-project.org/package=leaflet

Karthik Ram and Hadley Wickham (2018). wesanderson: A Wes Anderson Palette Generator. R package version 0.3.6. https://CRAN.R-project.org/package=wesanderson

Kenneth Reitz (2014). Requests. https://docs.python-requests.org/en/latest/dev/authors/#keepers-of-the-crystals

Kluyver, T., Ragan-Kelley, B., Fernando P&#x27;erez, Granger, B., Bussonnier, M., Frederic, J., … Willing, C. (2016). Jupyter Notebooks – a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Eds.), Positioning and Power in Academic Publishing: Players, Agents and Agendas (pp. 87–90).

Kyle Walker (2021). tigris: Load Census TIGER/Line Shapefiles. R package version 1.5. https://CRAN.R-project.org/package=tigris

McKinney, W., & others. (2010). Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference (Vol. 445, pp. 51–56).

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, *5*(4), 1093–1113. https://doi.org/10.1016/j.asej.2014.04.011

Mohit, B. (2014). Named Entity Recognition. In I. Zitouni (Ed.), *Natural Language Processing of Semitic Languages* (pp. 221–245). Springer. https://doi.org/10.1007/978-3-642-45358-8_7

Ooms J (2014). “The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects.” arXiv:1403.2805 [stat.CO]. https://arxiv.org/abs/1403.2805.

Pebesma, E. (2018). Simple Features for R:Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, https://doi.org/10.32614/RJ-2018-009

R Core Team (2020). R: A language and environmentfor statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Ramnath Vaidyanathan, Yihui Xie, JJ Allaire, Joe Cheng, Carson Sievert and Kenton Russell (2020). htmlwidgets: HTML Widgets for R. R package version 1.5.3. https://CRAN.R-project.org/package=htmlwidgets

Rehurek, R., & Sojka, P. (2011). Gensim–python framework for vector space modelling. NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic, 3(2).

Richardson, L. (2007). Beautiful soup documentation. April.

RStudio Team (2020). RStudio: Integrated Development for R. RStudio, PBC, Boston, MA URL http://www.rstudio.com/.

Sievert Carson, Shirley Kennets (2014). Pyldavis. https://github.com/bmabey/pyLDAvis

Sievert, C. (2020). *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC. https://plotly-r.com

Thomas Lin Pedersen (2020). patchwork: The Composer of Plots. R package version 1.1.1. https://CRAN.R-project.org/package=patchwork

Timothee Giraud (2020). Popcircle: Get proportional circles and polygons on a compact layout. Package version 0.1.0 https://github.com/rstudio/bookdown.

Van Rossum, G., & Drake, F. L. (2009). Python 3 Reference Manual. Scotts Valley, CA: CreateSpace.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T., Miller, E., Bache, S., Müller, K., Ooms, J., Robinson, D., Seidel, D., Spinu, V., … Yutani, H. (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686.

Winston Chang, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert and Barbara Borges (2021). shiny: Web Application Framework for R. R package version 1.6.0. https://CRAN.R-project.org/package=shiny

Xie, Y. (2020). bookdown: Authoring Books and Technical Documents with R Markdown. https://github.com/rstudio/bookdown.

# Appendix: Required Metadata

## Table 1 – Software metadata

| Nr | **Software metadata description** | |
|---|---|---|
| S1 | Current software version | R (v4.0.3), Python (v3.6.6), RStudio (v1.4.1103) and Jupyter Notebook (v6.4.3). |
| S2 | Permanent link to executables of this version (your Github repo URL) | https://github.com/ah140797/news_media_NLP |
| S3 | Legal Software License | Creative Commons 4.0. |
| S4 | Computing platform / Operating System | Microsoft Windows 10. |
| S5 | Installation requirements & dependencies for software not used in class | Package management and installer for R: pacman (v0.5.1). Package management and installer for Python: pip (v21.3.1). |
| S6 | If available Link to software documentation for special software | |
| S6 | Support email for questions | au651039@post.au.dk |

## Table 2 – Data metadata

Please note that some datasets are split into two or three datasets, because they are too large for GitHub. The names of these datasets end with "_1", "_2" etc. These datasets are always concatenated at the beginning of the scripts.

| Nr | Name | Description |
|---|---|---|
| | | **Main Datasets (A)** |
| A1 | NYT_clean.csv | Dataset containing news articles from New York Times from 1995-2001 with the keyword "Taliban". Each article has an associated 9 columns: headline, url, date, bread_text, bread_text_preprocessed, polarity, subjectivity, dominant_topic and topic_contribution. |
| A2 | guardian_clean.csv | Dataset containing news articles from The Guardian from 1996-2001 with the keyword "Taliban". Each article has an associated 9 columns: headline, url, date, bread_text, bread_text_preprocessed, polarity, subjectivity, dominant_topic and topic_contribution. |
| | | **Datasets for Shiny App (B)** |
| B1 | gpe_shiny_NYT.rds | Dataset used to make the tab "New York Times" in the shiny app. It consists of 8 columns: ISO3, country, count, year, geometry, sum_articles_yearly, penalized_count and penalized_count_round. The format of .rds is needed to keep the column geometry. |
| B2 | gpe_shiny_guardian.rds | Dataset used to make the tab "The Guardian" in the shiny app. It consists of 8 columns: ISO3, country, count, year, geometry, sum_articles_yearly, penalized_count and penalized_count_round. The format of .rds is needed to keep the column geometry. |
| B3 | gpe_shiny_contrast.rds | Dataset used to make the tab "Contrast" in the shiny app. It consists of 6 columns: ISO3, country, year, contrast, contrast_round and geometry. The format of .rds is needed to keep the column geometry. |
| | | **Datasets for Popcircle (C)** |
| C1 | gpe_popcircle_NYT.rds | Dataset used to make the popcircle for New York Times. It consists of 5 columns: country, ISO3, count, count_per_article and geometry. The format of .rds is needed to keep the column geometry. |
| C2 | gpe_popcircle_guardian.rds | Dataset used to make the popcircle for The Guardian. It consists of 5 columns: country, ISO3, count, count_per_article and geometry. The format of .rds is needed to keep the column geometry. |

| | | |
|---|---|---|
| **Geographical Datasets Shapefiles (D)** | | |
| D1 | TM_WORLD_BORDERS_SIMPL-0.3.dbf | Dataset used for drawing country borders. |
| D2 | TM_WORLD_BORDERS_SIMPL-0.3.prj | Dataset used for drawing country borders. |
| D3 | TM_WORLD_BORDERS_SIMPL-0.3.shp | Dataset used for drawing country borders. |
| D4 | TM_WORLD_BORDERS_SIMPL-0.3.shx | Dataset used for drawing country borders. |
| **Datasets for NER (E)** | | |
| E1 | GPE.csv | Dataset used for maps. It contains 5 columns: word, count, GPE, date and article_index. |
| E2 | person.csv | Dataset used for persons. It contains 5 columns: word, count, person, date and article_index. |
| E3 | ORG.csv | Dataset used for Organizations. It contains 5 columns: word, count, ORG, date and article_index. |
| E4 | noun.csv | Dataset used for nouns. It contains 5 columns: word, count, noun, date and article_index. |
| E5 | verb.csv | Dataset used for verbs. It contains 5 columns: word, count, verb, date and article_index. |
| E6 | adjective.csv | Dataset used for adjectives. It contains 5 columns: word, count, adjective, date and article_index. |
| **Datasets not mentioned in the paper, but used in the reproducibility notebook** | | |
| **Datasets for Plotly (F)** | | |
| F1 | GPE_plotly_NYT.csv | Dataset used to make the plotly map for New York Times. It consists of 8 columns: ISO3, country, count, year, penalized_count, sum_articles_yearly, penalized_count_round and hover. |
| F2 | GPE_plotly_guardian.csv | Dataset used to make the plotly map for The Guardian. It consists of 8 columns: ISO3, country, count, year, penalized_count, sum_articles_yearly, penalized_count_round and hover. |
| **Additional Datasets for NER (G)** | | |
| G1 | FAC.csv | |

| **Datasets for Early Chapters (H)** | | |
|---|---|---|
| H1 | NYT_clean_cp2.csv | Dataset containing all the articles for New York Times, before body text is scraped. It contains 3 columns: headline, url and date. Cp2 refers to chapter 2 in the reproducibility notebook. |
| H2 | guardian_clean_cp2.csv | Dataset containing all the articles for The Guardian, before body text is scraped. It contains 3 columns: headline, url and date. Cp2 refers to chapter 2 in the reproducibility notebook. |
| H2 | NYT_clean_cp3.csv | Dataset containing all the articles for New York Times, including articles with missing body text. It contains 4 columns: headline, url, date and bread_text. Cp3 refers to chapter 3 in the reproducibility notebook. |
| H4 | guardian_clean_cp3.csv | Dataset containing all the articles for Guardian, including articles with missing body text. It contains 4 columns: headline, url, date and bread_text. Cp3 refers to chapter 3 in the reproducibility notebook. |
| **Dataset for Gender Analysis (I)** | | |
| I1 | NYT_author.csv | Dataset used for making analysis based on gender. |
| **Datasets for Topic Model Tuning (J)** | | |
| J1 | tuning_results_NYT.csv | Used to choose the number of topics in the topic model for New York Times. It consists of 2 columns: topics and coherence. |
| J2 | tuning_results_guardian.csv | Used to choose the number of topics in the topic model for The Guardian. It consists of 2 columns: topics and coherence. |