Capstone 2: Final Report
Correlating web traffic to external events

Jonas Cuadrado

1. Problem to solve:

Knowing how different events affect the behavior of a company's clients can allow a company to improve its service, whether by offering specific products or by adapting to the clients' needs, as well as allowing the company to predict trends based on seasonal trends.

In this case study we will try to understand how the number of searches on Wikipedia correlates to general news articles from the media. If there is a correlation, we will explore what determines who searches what, and when it happens, and whenever there are no correlations, we will also explore the causes.

The Wikipedia database is selected due to its publicly available data, its popularity, and its ability to attract plenty of users from diverse backgrounds.

2. Client:

In this specific project, Wikipedia could use that information to predict higher demand of traffic and open more servers to reduce delays. This can be generalized to any company. Alternatively, a media outlet can understand what attracts the interest of its readers by following the traffic spikes in time.

In general, every company should be interested in how their users behave. Being able to track their interests and observe correlations between events is of upmost importance to determine how to improve and where to invest in the corporate model.

3. Datasets:

The Wikipedia project has readily available traffic data per article straight from the API. The description is available on https://dumps.wikimedia.org/other/pageviews/readme.html. There are two immediate options: directly accessing the tar.gz datafiles for each hour of the day, that contain the number of visits to every article per hour, and per day, or using aggregate counts straight to the API to obtain daily values.

I have opted for the daily approach to reduce the amount of data to handle, and to avoid hourly peaks that may be affected by the geographical location of the search. For instance, in the morning in Europe we may find searches on events related to what happened there rather than in the US, since in America it'd be the middle of the night.

There was a change in the data management, so we will only use data from after May 2015.

For the news, Kaggle has a public dataset containing articles from different sources for a period of time of over 1 year: https://www.kaggle.com/snapcrack/all-the-news

To better extract the data, some NLP will be required to obtain the topic, at least, for most of the articles. The date is immediately available, which facilitates the time correlation between events.

4. Initial findings: Wikipedia dataset

The Wikipedia dataset is well structured, essentially clean, and shows some of the expected behavior. Instead of accessing the daily views of all the articles and selecting the most relevant ones, it is possible to collect the 1000 articles most seen on each day. We can check if there are meaningful peaks in them, and if they are meaningful.

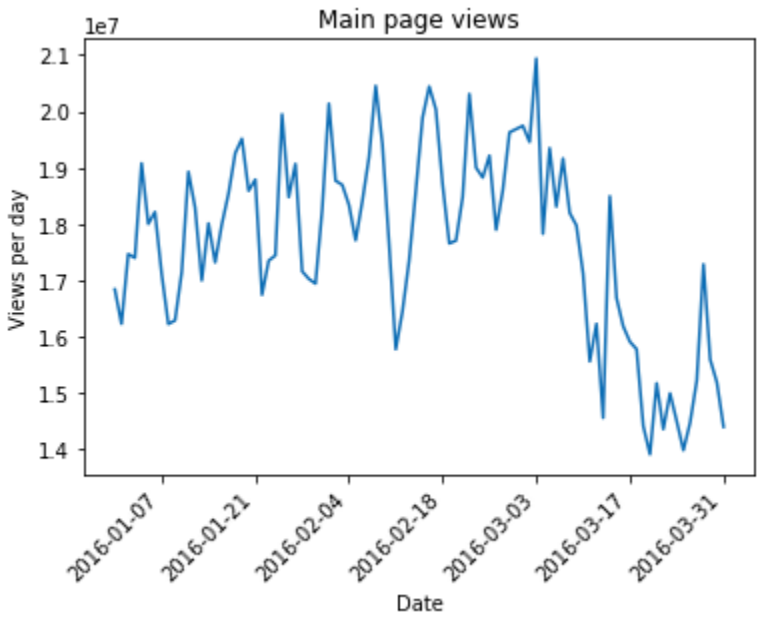

Figure 1: Daily views of the Wikipedia main page.

To perform an initial analysis, we can selected the months of January, February and March, 2016. The first noticeable property is that the main page is always the page with the most views. It has peaks, but it never grows or descends very importantly in percentage.

However, some specific events, like deaths or the release of a movie, can create a huge percentage increase in the traffic despite having a smaller total number of views. Indeed, the percentage change can go to over 10.000%. That percentage is artificial because there are usually no values before the peak, as the page is typically not amongst the top 1000 accessed the day before, so one needs to set the baseline, the number of views it has before the peak. We can look at different baseline values, and selected the one that contains the most articles. This also contains all the articles in most baselines, and is especially attractive since it contains articles of (probably) more impact like Queen Elisabeth II or Meghan Markle.

```
Is there no difference between baseline 1 and 1000? True
How many articles on baseline 1 are not on baseline 10.000? 1

Article not in b10k: ['Iman_(model)']

Articles not in b1: ['David_Bowie', 'Charles_Perrault', 'Pat_Bowlen', 'Alexander_Hamilton', 'Keanu_Reeve
s', 'Genie_(feral_child)', 'Rogue_One', 'Electronic_System_for_Travel_Authorization', 'Hertha_Marks_Ayrto
n', 'Jane_Jacobs', '404.php', 'Frankie_Manning', 'Harry_Potter', 'Elizabeth_I_of_England', 'MS_The_World',
'Meghan_Markle', 'Jagadish_Chandra_Bose', 'John_Glenn', "'Tis_the_Season", 'George_Michael', 'Chelsea_Mann
ing', 'Bessie_Coleman', 'Sally_Yates', 'Chance_the_Rapper', 'Alexander_Hamilton', 'Fazlur_Rahman_Khan', 'W
ikipedia:Contact_us', 'Gilbert_Baker_(artist)', 'Aaron_Judge', 'Jodie_Whittaker', 'Laverne_Cox', 'Joe_Arpa
io', 'Fridtjof_Nansen', 'Direct_and_indirect_realism', 'Meghan_Markle']
```

Figure 2: Results from comparing different baselines.

By applying an additional filter to select only peaks with over 500.000 views, we can narrow down to a reasonable selection of peaks with events that have attracted the eye of the public opinion, such as the death of Alan Rickman or Supreme Court Justice Antonin Scalia.

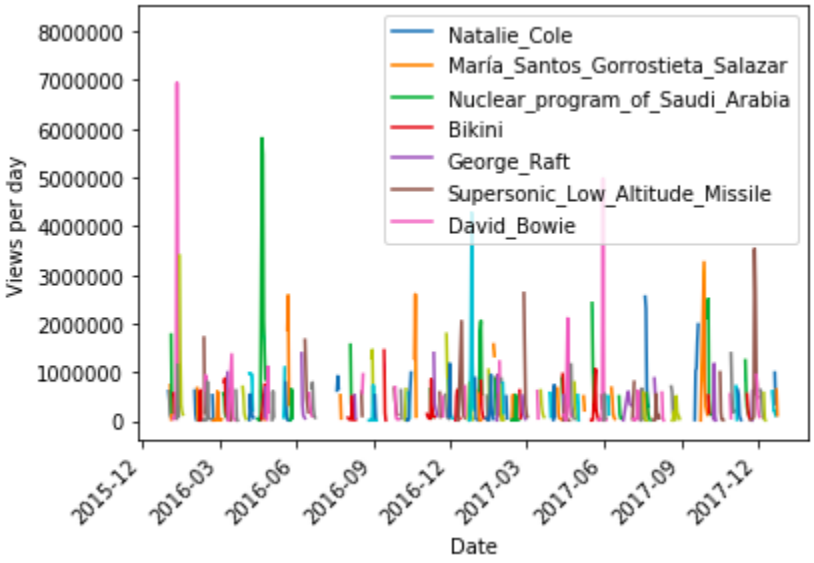

Figure 3: Views peaks.

```
Articles with most views
article
Charles_Darwin                                                  8145795.0
David_Bowie                                                     6948182.0
404.php                                                         6190956.0
Prince_(musician)                                               5808147.0
Wikipedia:Contact_us                                            4977887.0
Government_Secure_Intranet                                      4293586.0
George_Michael                                                  4275899.0
Meghan_Markle                                                   3536755.0
Meghan_Markle                                                   3536755.0
Alan_Rickman                                                    3394010.0
Hugh_Hefner                                                     3261496.0
Bill_Paxton                                                     2622501.0
Anterior_interventricular_branch_of_left_coronary_artery        2601855.0
Azúcar_Moreno                                                   2581231.0
Chester_Bennington                                              2550909.0
```

Figure 4: Articles with most views, and number of views in the best day.

The selected articles were downloaded from the Wikipedia and by means of regular expressions, all non-alphanumeric elements were removed.

5. Initial findings: News dataset

The news dataset from Kaggle contains over 100.000 articles from different sources, including New York Times, NPR, Business Insider or CNN. The data is mostly clean, contains plenty of advertisements from Breitbart, so I have deleted all Breitbart articles since their validity is under scrutiny recently.

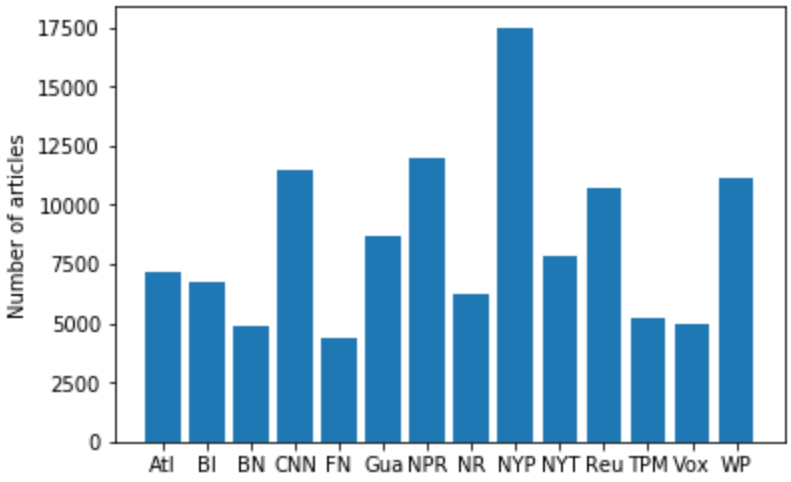

Figure 5:Number of articles per source.

The times in which they were written span different periods, mostly after 2016. To perform an initial analysis, we can select the articles from 2016 and 2017 (over 108.000). We observe that the distribution in time and in source is very uniform, having about 150 articles per day rather well mixed between all the sources.

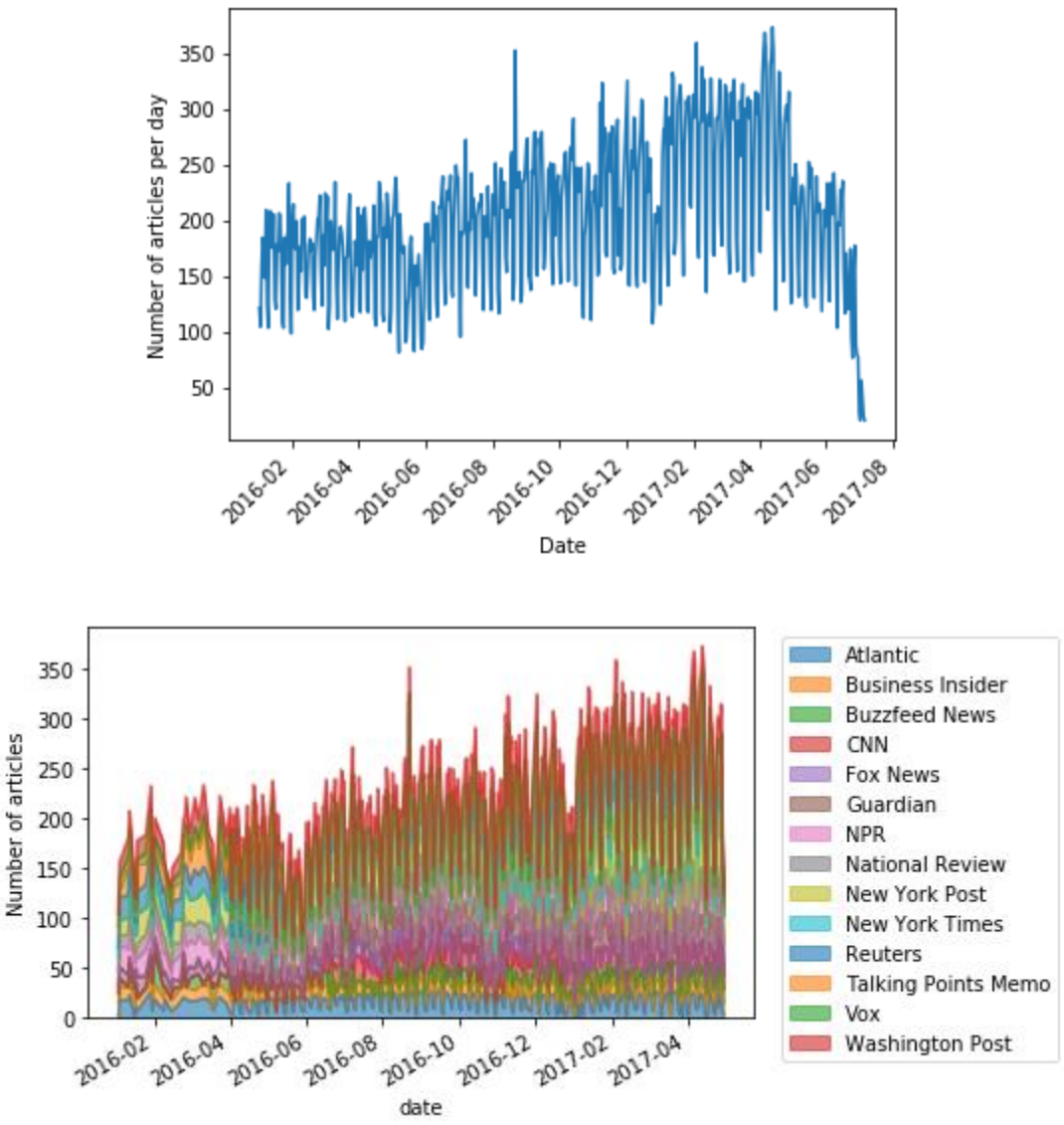

Figure 7: Number of articles per day.

To reduce the load and work on the topic modelling part, a random selection of 10.000 articles was performed.

6. Topic Modelling:

To extract the information from the articles and be able to compare them, we will use topic modelling. In natural language processing these are the steps linked to extracting the topic information:

1. It is common to treat a document as a bag of words, where the order of the words does not play a role. While a limitation in meaningfulness, a long enough document will not be totally built on rhetoric and have some key elements that make the approach reasonable.

2. We need to remove *stopwords*, or words without meaning. “a”, “so”, or “therefore” have grammatical meaning but not content meaning: these need to go. A list of 500 words is added to the topic modelling module before analysis.
3. We also want to filter rare and common words. If words appear in 95% of the documents, they do not give enough information to differentiate among articles, so it should be neglected. Same with very unusual words.
4. Finally, it is recommended to use a lemmatization component that merges words with a common origin into a single *token*. Then, words like ‘multiply’, ‘multiplication’ and ‘multiple’ are the same. We will not use lemmatization in this project.

While these steps are recommended, they are not required always. Both using and not using them brings up limitations that can be explored best on literature. The result, in any case, is a matrix where the rows are tokens associated to words, and the columns are the number of tokens per document. This way, we have transformed a set of words into a matrix.

If we use the sci-kit learn package to perform the work, we have to choose among a variety of algorithms and parameters to find the best topic modeler and tokenizer. In this project, we have compared Latent Dirichlet Allocation and Non-negative Matrix Factorization on the 10.000 article test set. But first, we need to compare their performance. What tools do we have?

The two easiest techniques to quantify how good a text processing is are the likelihood and the perplexity of the output. The likelihood lacks conceptual significance, it is essentially the probability of finding that distribution of topics given the documents. The perplexity is essentially a probability prediction, or how likely is the word to be associated to each topic. There is a measure that is more meaningful for human purposes: topic coherence. Topic coherence compares how meaningful are the words that have the largest weight per topic, and how they appear on other topics. We used coherence to compare LDA and NMF, and to find the best parameters for the topic modeler. LDA always outperforms NMF, so we explored the parameter space for that case

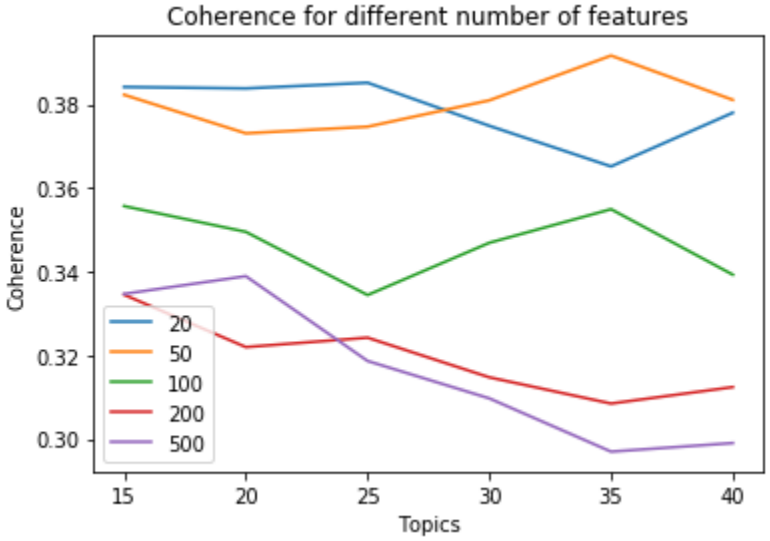

Figure 9: Topic coherence for different LDA parameters.

At then end, the best approach for the dataset was to take LDA with 50 features and 35 topics. The topics span politics, life, economy, or civil rights. In reality, every article is composed of a combination of topics, and we can take the most meaningful topic to classify it. For example, the Antonin Scalia article has a distribution of topics centered around topics 21 and 22, which are

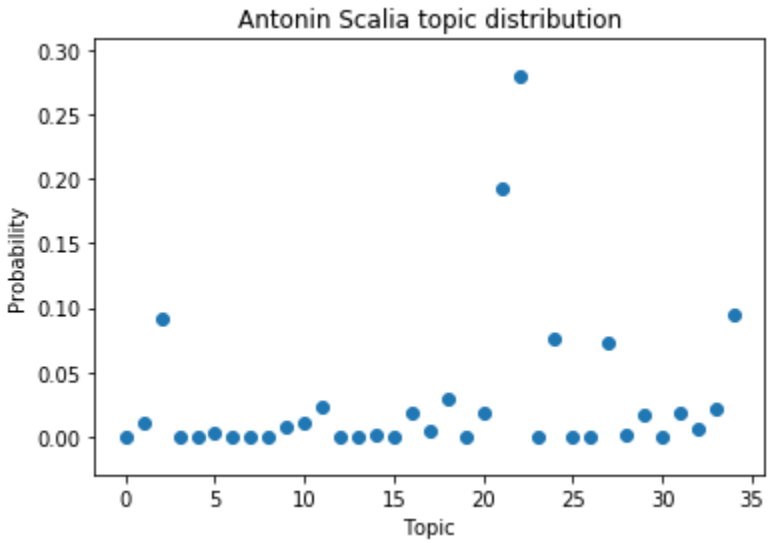

Figure 10: Antonin Scalia's article topic distribution.

- Topic #21: states united president country including world american work year make
- Topic #22: court law public including state year long time called president

For the articles in Wikipedia, the most significant topics are only a few of them, while for the whole dataset they are more distributed.

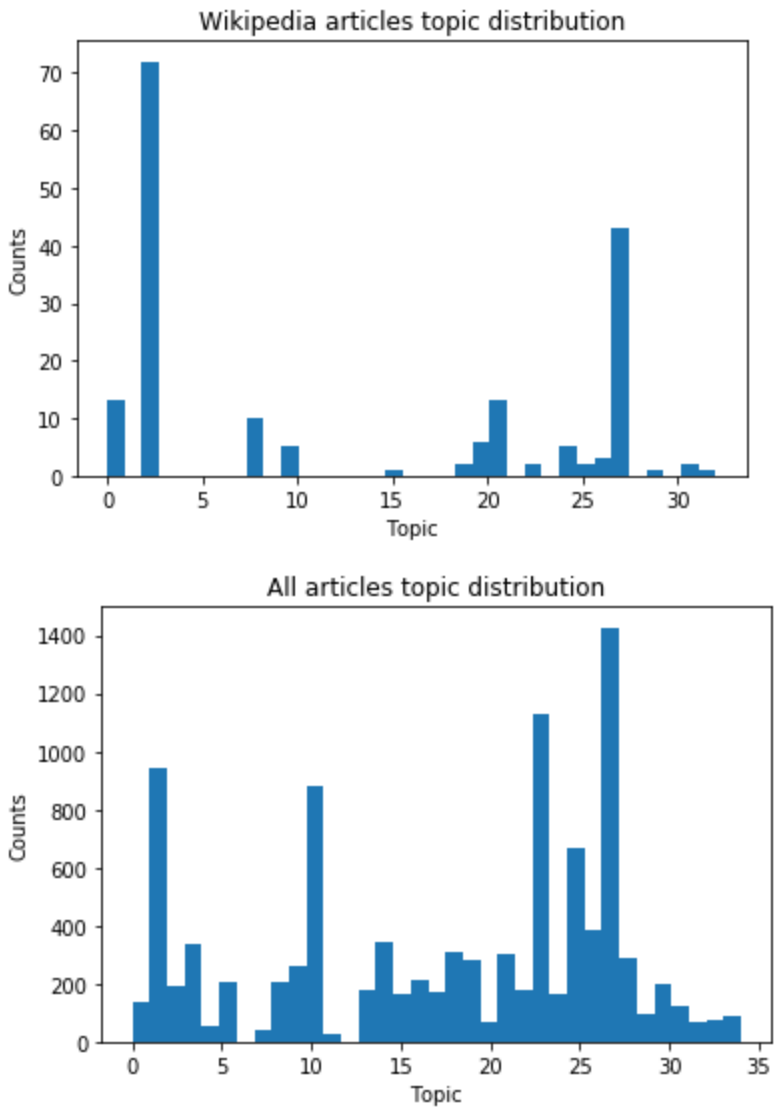

Figure 11: Histograms for the number of articles whose main topic is N, for the Wikipedia subset and the whole 10.000 news artciles

The most significant topics are composed of the words:

- Topic #0: campaign times political including trump people government time make told
- Topic #1: trump president donald house campaign government national political time obama
- Topic #2: news work times people good american including told president week
- Topic #10: people work years time life make don called long group
- Topic #21: states united president country including world american work year make
- Topic #23: trump donald republican campaign don people make called york told
- Topic #27: year time years work life long called make day including

It seems the topics are political and about life. By choosing some articles well defined, like the deaths of important public people, we can gain some insight. Topic 2 is associated to most of them (not for Scalia

at its peak, but higher than most, and dominant for Alan Rickman and David Bowie). This is indicative of a trend we want to explore more.

```
Topic for:  David_Bowie  :  2
Topic for:  Alan_Rickman  :  2
Topic for:  Antonin_Scalia  :  22
```

Figure 12: Main topic for these three articles.

To look at the dependence between media and Wikipedia articles we need to establish a distance between them. Now that they are in vector form, we can use the cosine similarity. For the Antonin Scalia article, we see that there is a peak on the cosine similarity to articles published around the peak day.

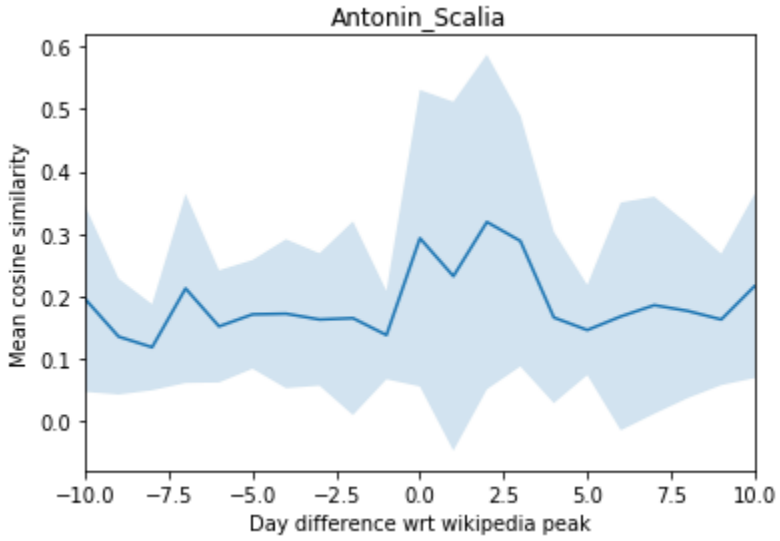

Figure 13: Mean and standard deviation of the cosine similarity between Scalia's articles and all others, plotted as time difference between Scalia's peak and the publication date of the others.

This indicates that after the peak, a lot of articles were similar to what Wikipedia said about the Supreme Court Justice. If we look in a longer scale, there are plenty of fluctuations: it was a very hot topic, especially in the election of the replacement.

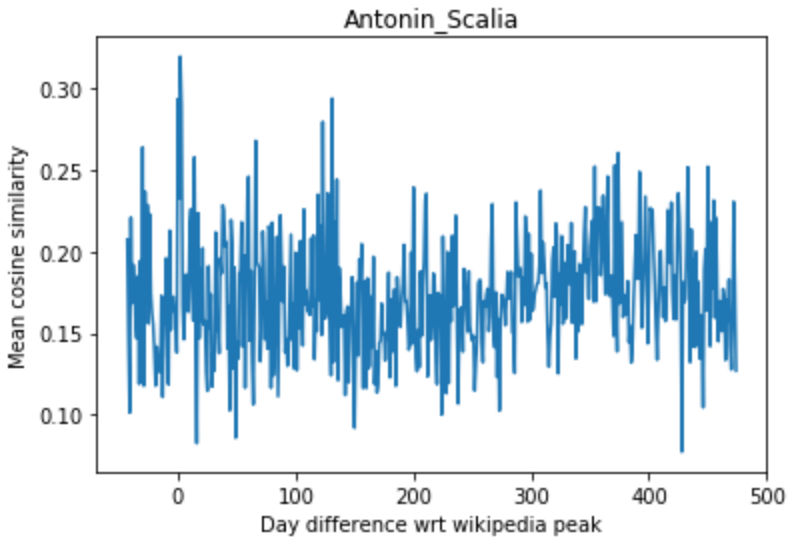

Figure 14. Same than figure 13, but without std area and on a longer time span.

Such an increase is not consistent through articles, as it washes out as we include all articles in a single topic.

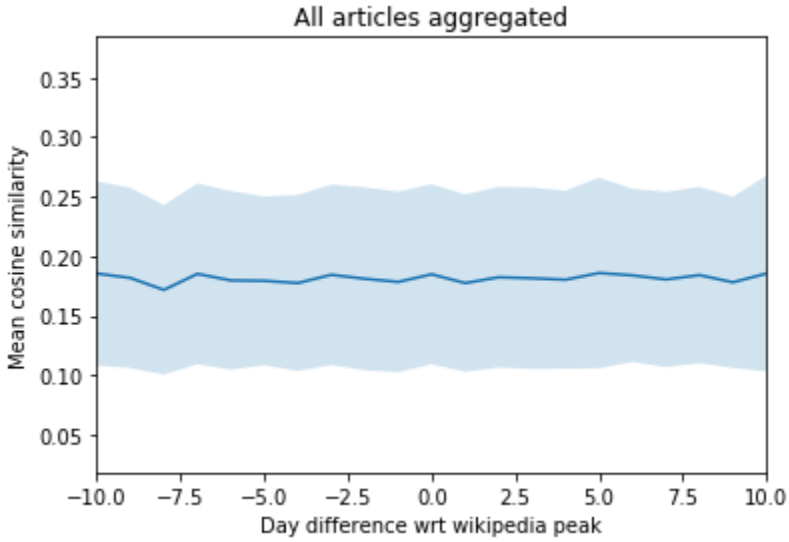

Figure 15: Same as figure 13, but aggregating over all the Wikipedia articles.

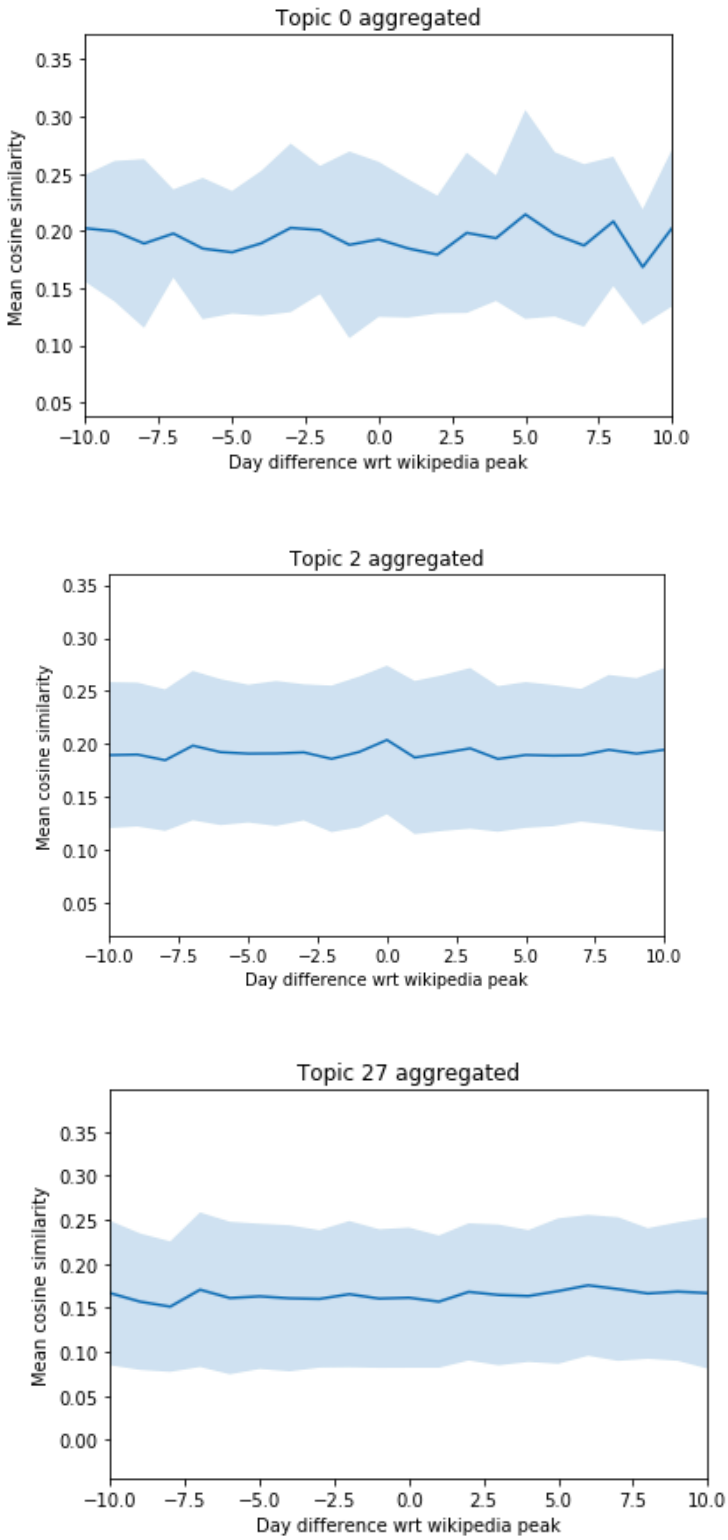

Figure 16: Same as figure 13, but aggregating over all articles on topics 0,2, and 27

This is not caused by the selection of 10.000 articles, as we can use the 100.00 to generate a smoother curve which does not have any peak.

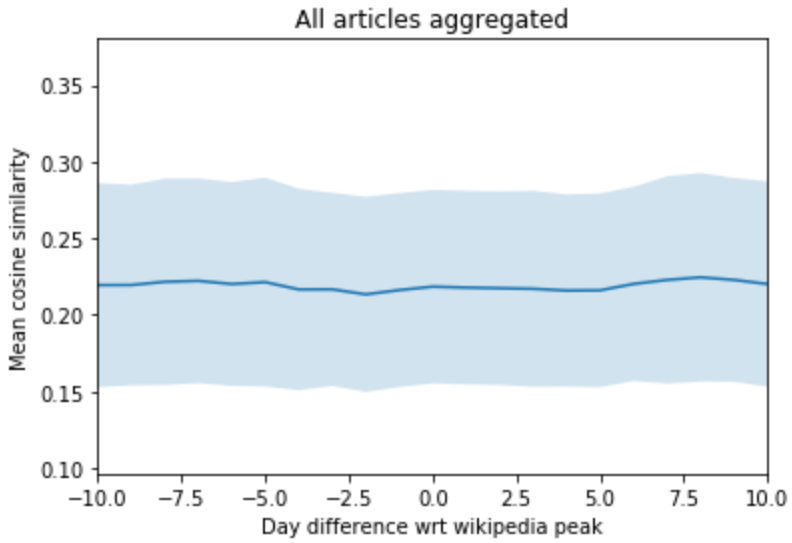

Figure 17: Same as figure 15, but on 100.000 Wikipedia articles.

We can also select the articles with the highest and lowest correlation on 0, and on 1. We observe a diversity of topics and artciles:

```
Articles with highest media repercussion

Antonin_Scalia ( Topic 22 ) ---  1.122920939911423
Jesus_nut ( Topic 27 ) ---  0.8925623985316848
Pierre_Brassau ( Topic 27 ) ---  0.8270671778995227
José_Fernández_(pitcher) ( Topic 2 ) ---  0.7343471297114634
Merrick_Garland ( Topic 22 ) ---  0.7003123062622494
Mary_(elephant) ( Topic 21 ) ---  0.6654081748011484
Alan_Thicke ( Topic 2 ) ---  0.6176231233540612
Muzdalifah ( Topic 27 ) ---  0.5533614530726585
Leonard_Cohen ( Topic 2 ) ---  0.518054574686845
Fidel_Castro ( Topic 2 ) ---  0.4728181036742789
Nuclear_program_of_Saudi_Arabia ( Topic 2 ) ---  0.4690601092609108
Alexander_Hamilton ( Topic 21 ) ---  0.4311465078836383
Kamehameha_I ( Topic 27 ) ---  0.3973156549749848
Government_Secure_Intranet ( Topic 26 ) ---  0.39625627829111365
VX_(nerve_agent) ( Topic 21 ) ---  0.38009215389085327
```

```
Articles with lowest media repercussion

Blendo (Topic 19 ) ---  -0.5472519071287285
George_Michael (Topic 2 ) ---  -0.45409047881495623
Lupe_Fuentes (Topic 2 ) ---  -0.3767248587152442
Tanghulu (Topic 10 ) ---  -0.35018516908178887
Pat_Bowlen (Topic 2 ) ---  -0.3443064372645511
Meldonium (Topic 2 ) ---  -0.33982641428616267
Issus_(genus) (Topic 2 ) ---  -0.33358537786703635
Psychosis (Topic 27 ) ---  -0.3249566356458933
Whale_fall (Topic 27 ) ---  -0.2994812893261487
Alan_Rickman (Topic 2 ) ---  -0.2892498161353291
Web_performance (Topic 27 ) ---  -0.24969376583323144
Lincoln_Logs (Topic 19 ) ---  -0.23811642256339527
GBU-43/B_Massive_Ordnance_Air_Blast (Topic 27 ) ---  -0.23527552957338382
Supersonic_Low_Altitude_Missile (Topic 21 ) ---  -0.21149151908014063
Load_testing (Topic 27 ) ---  -0.20960071971640137
```

Figure 18: Articles with highest and lowest correlation on the news.

These articles do not correlate to those with the highest peaks, but they do have something in common: they are about content not such well understood. For instance, does everybody know the names of all supreme court justices? Who knows what is a Jesus nut? It's a piece of a helicopter. And Pierre Brassau? Sounds either as a painter or a monkey. It turns out it's both.

On the other hand, most people have heard of Alan Rickman, Psychosis or George Michael. Those have less impact, but they became trendy at some point.

7. Conclusion:

In this project we have provided some insight in the relationship between views of Wikipedia articles and information published on newspapers and media. We have obtained two independent datasets with the articles, selected the most relevant documents, and modelled them into a set of quantitative vectors using natural language processing. We have compared different NLP approaches and used a topic coherence to come up with the model that best describes the data. We have also used transformed the data to compare the articles and see how media news and Wikipedia searches are not correlated on average, but there are some cases in which they are.

The picture we can extract from this analysis is that the Wikipedia searches evidence what people are interested in, irrespective of where that interest is coming from. In some cases, like politics or technical concepts, the interest may come from a breaking news article that gets developed afterwards. In other, like events regarding entertainment, they may come other source like social media. This information can be used to measure the impact of news articles in the people's interests.