Final Report

Covid-19 News Analysis

Sam Sweere (i6231098), Alexander Reisach (i6197692)

KEN4135: Information Retrieval and Text Mining

Faculty of Science and Engineering
Dpt. of Data Science and Knowledge Engineering
Maastricht University
Netherlands

Abstract

We load and pre-process a substantial subsample of the Aylien Covid-19 news data set. We select the period from 01.02.20 to 05.04.20 as the most interesting and data rich subperiod for our analyses. For each day within this period we analyse the topics present. We find consistent topics that seem to accurately describe the trends in effects of the global pandemic. We further analyse entities of type country and person using a co-reference resolved version of the data. Any entities found are linked to their real-world counterparts as found in the DBPedia database. We observe that most entities are classified correctly and resolve the most common remaining disambiguations manually. Using aspect-based sentiments, we find the sentiment ascribed to each of the entities extracted, as well as that related to Covid-19 itself. We find the general sentiment towards entities to be negative and observe a negative but improving sentiment towards the virus itself. We use topic rivers, animated bar charts and choropleth maps to show the temporal and geographical trends in topics, entities and sentiments.[1]

[1]The full source code of the project can be found at `https://github.com/SamSweere/Covid19-News-Analysis`.

Contents

Chapter 1

Division of Work

We are continually discussing how to approach certain aspects and regularly do joint code reviews. Table 1.1 gives an indication of who spends the most time on which parts.

Sections	Alexander Reisach	Sam Sweere
Data Loading	X	
Processing Pipeline	X	X
Topic Analysis	X	
Entity Analysis - Coreference Resolution	X	
Entity Analysis - Entity Detection and Linking	X	
Entity Analysis - Computational Improvements		X
Sentiment Analysis - General Model Training/Benchmarking		X
Sentiment Analysis - ABSA Model Training/Benchmarking		X
Sentiment Analysis - Sentiment Tracking		X
Visual Presentation - Bar Chart Race	X	X
Visual Presentation - World Map		X
Visual Presentation - Topic River	X	

Table 1.1: Division of work

Chapter 2

Introduction

This project is a part of the course KEN4153 Text Mining and Information Retrieval. As such, we use the methods explained in said course to mine an extensive data set so as to be able to automatically answer content related questions about actors, locations, events, and trends. Given the recent global developments, we use a large publicly available text corpus on Covid-19 news coverage as the basis for our analysis.

2.1 Data Set

Our data set is a pre-aggregated selection of 500,000 news articles on the Covid-19 global pandemic[1]. It covers articles published between November 1st 2019 and April 6th 2020. The data set comprises publications from 400 prominent news sources with an Alexa [2] rank below 5,000. It contains English language publications only. Moreover, the data set has been pre-processed and enriched by entities recognised, topical category tags, a sentiment measure and summaries. Source information and date of publishing are included as well. The data set contains the following keys:

{*'author', 'body', 'categories', 'characters_count', 'entities', 'hashtags', 'id', 'keywords', 'language', 'links', 'media', 'paragraphs_count', 'published_at', 'sentences_count', 'sentiment', 'social_shares_count', 'source', 'summary', 'title', 'words_count'*}

Usage of the data set is open for researchers, scientist and any other form of non-commercial analysis.

[1] `https://aylien.com/coronavirus-news-dataset/`
[2] `https://alexa.com/topsites/category/Top/News`

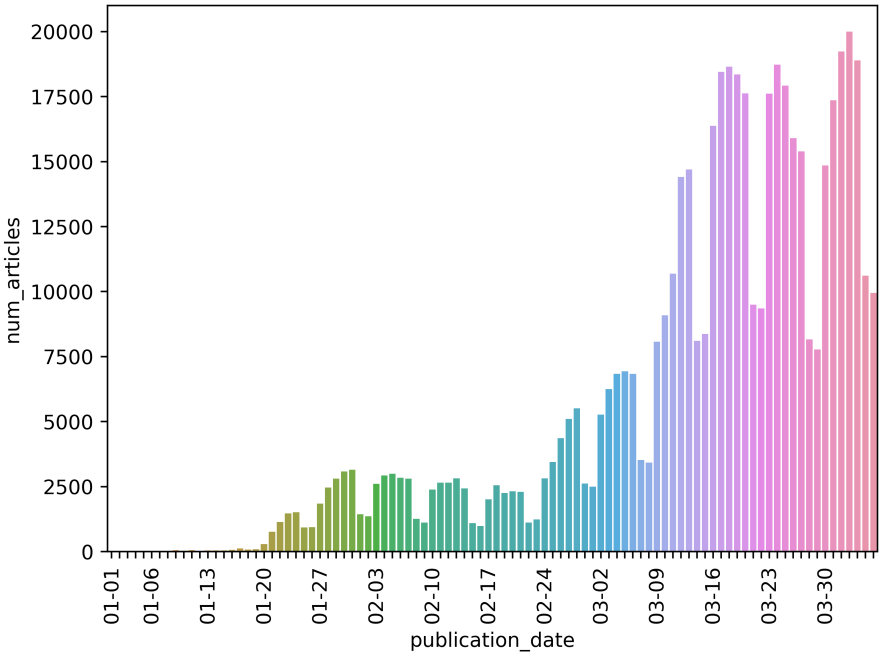

Figure 2.1: News output in number of articles per day

Figure 2.1 shows the number of articles published at each day within the period covered by our data set. We see a substantial increase of articles published on Covid-19, mirroring the spread of the pandemic. We also observe the regular news cycle with periodic lows on the weekend.

Chapter 3

Methodology

3.1 Data Loading

Our data set contains 500,000 news articles which take a total of 7.8 gigabytes of space. At this size we have to take computational and memory limitations into account. Therefore we implement a function to take a sample of the data based on publication dates. You can specify from and to which date you want to extract articles and how many articles you want from each specific day. Additionally, we have build functionality to get a random sample of articles from a given period. For testing purposes, we are working with samples from a subperiod to allow us to test prototypes of our text mining pipeline more quickly. For analysis we consider larger samples up to the whole data set.

3.2 Processing Pipeline

Before doing any text analysis we first have to do some pre-processing of the data. The data set already contains separate keys for headlines, body, authors, etc. This saves us a lot of time, since we do not have to text mine these articles ourselves. However, the text still consists out of plain text which we will have to structure to be able to run our data enrichment procedures. We use the Spacy [1] framework in Python since it is relatively easy to use, fast, and very versatile. By default, Spacy comes with a rudimentary pre-processing pipeline (figure 3.1). It includes pre-trained models but also gives us the flexibility to train models of our own and incorporate them into the pipeline.

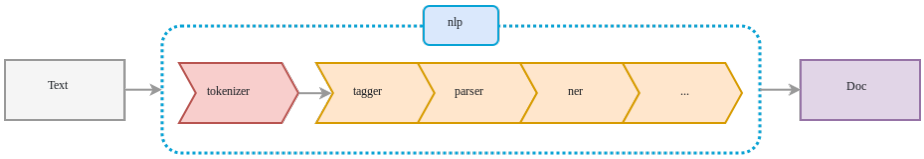

Figure 3.1: Spacy pipeline. (From: `spacy.io/usage/spacy-101`)

Tokenizer
Segment text into tokens. This segments the text into words, punctuation,

[1] `https://spacy.io/`

etc. It does this by first splitting the text based on white-spaces and checking whether the sub-string should be split further as in the case of *'s*.

Part-of-Speech Tagger
The Part-of-Speech (POS) tagger assigns its tags using the OntoNotes 5 [2] version of the Penn Treebank tag set.

Dependency Parser
The default dependency parser does within-sentence dependency parsing using a convolutional neural network based on [1] and inspired by [2].

Named Entity Recognition
Depending on the result of the previous steps, the named entity recognition assigns labels such as *Person* or *Location* to token it recognizes to be entities. Similar to the Parser, it is based on a convolutional neural network.

3.3 Topic Analysis

For topic analysis we apply Latent Dirichlet Allocation (LDA) and compare it to Non-Negative Matrix Factorization (NMF). We find that due to its additive nature, NMF results in topics with less overlap. Since we want to cover the distinct aspects of Covid-19 news reporting, we decide to use NMF for our topic analysis.

3.3.1 Pre-processing

We use a separate pre-processing pipeline specifically adjusted to the needs of Topic Analysis. We first clean our text of all non-ASCII characters such as non-breaking whitespaces and typographic quotes and remove all punctuation. In the next step, we tokenize and lemmatize each article. We also remove common English language stopwords as defined by the python NLTK corpus as well as the following custom stopwords of news agencies and other non-distinctive words common in news articles:

```
"say", "news", "reuters", "cbcca", "getty", "reuter", "get",
"am", "pm", "et", "go", "like"
```

3.3.2 Topic Fitting

For the purpose of finding our initial topics, we fit a vectorizer to arrive at a document-term matrix. We set the number of topics to be detected to 10. We then factorize the document-term matrix into a topic matrix $n\_articles x 10$ columns and a corresponding weight matrix of dimension $10 x n\_terms$. Due to the size of the data set we are analyzing, we detect topics on a randomly

[2] https://catalog.ldc.upenn.edu/LDC2013T19

sampled subset of articles from our data set. We define the name of the topic as the concatenation of the top three terms by weight associated with the topic.

3.3.3 Topic Detection

After we fitted our topics as described above, we select a certain number of articles per day and use them to calculate the given topic prevalence for each day. We use the NMF topic model trained on the random subsample to transform each of the daily samples and compile the absolute as well as the normalized prevalence for each topic. The resulting time series are visualized as "Topic River" in form of a stream graph.

3.4 Entities

We approach the topics & entities by asking the question "What are the key entities concerned with Coronavirus and how do they change over time?"

3.4.1 Default model

We do this analysis by using the default small version of the Spacy English model to reduce computation time. The performance difference between the small model and the bigger models is not substantial. Running our pipeline on the faster smaller models enables us to analyse a bigger part of our data set.

Syntax Accuracy	
Labelled dependencies (LAS)	89.71
Unlabelled dependencies (UAS)	91.62
Part-of-speech tags (POS)	97.05

Named Entities Accuracy	
F-score	85.55
Precision	85.89
Recall	85.21

Table 3.1: Performance of the small default Spacy model.

The smaller Spacy model consists of a multi-task CNN trained on OntoNotes [3]. Similar to every Spacy pipeline it starts with tokenization, it then does POS tagging, dependency parsing and finally it detects and labels named entities. The performance of this model is shown in table 3.1.

3.4.2 Coreference Resolution

In order to extract more valuable patterns from our corpus, we will resolve within- and between-sentence references to identical entities. For example, in the sentence "*On March 26, Johnson revealed he had tested positive and that he had been dealing with symptoms since that date.*" we see the name *Johnson* is referred to again later in the sentence by *"he"*. From the context we know that both refer to politician *Boris Johnson*. We would like to count both of them as a reference to the corresponding real world entity. The default dependency parser usually does not connect such relationships correctly, see figure 3.2.

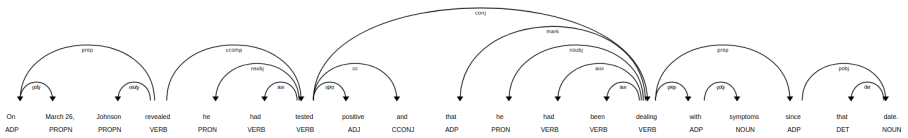

Figure 3.2: Spacy default dependency plot.

In order to resolve this issue, we first use an entity merger to ensure entities that consist of multiple words such as *Prime Minister Boris Johnson*, are treated as single entities. We extend the default model by using a neural network based coreference resolution before estimating the most frequent entities. For this purpose we apply *NeuralCoref*[3], an open source implementation of the reinforcement-learning based approach by [4].

Neural Coreference Resolution

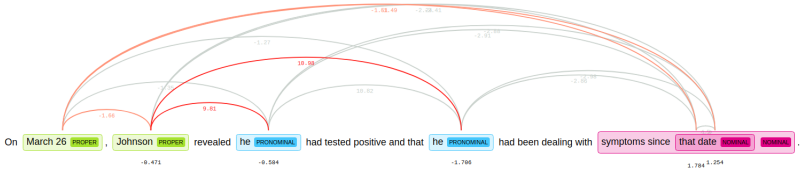

Figure 3.3: Coreference resolution using NeuralCoref

We resolve the clusters detected by *NeuralCoref* using the cluster name for all variants of direct or indirect mentions of the same instance, and run the entity recognition module on the resolved text corpus.

3.4.3 Entity Linking

Knowledge Base

After recognition and resolution of entities, we still need to ensure that we recognize variants that correspond to the same real-world entity as such. For this purpose, we make use of DBPedia Spotlight [5] as a knowledge base. DBPedia Spotlight is a tool for linking textual entities to their real-world counterparts as found in DBPedia [6], a structured database capturing the information gathered on Wikipedia. We are using the entity recognition capabilities of DBPedia Spotlight on the result of the coreference resolved version of the original text. We choose the caption of the corresponding Wikipedia article as the real-world name of our entity. This results in far more specific entity tags for most of the entities recognized. We search the position of the entity within the DBPedia

[3] `https://github.com/huggingface/neuralcoref`

schema for specific keywords to focus on certain types of entities such as for example "Person" or "Office holder".

Disambiguations

While we obtain a lot of very good results, this method is not free from disabiguation. For example, it seems that from the text it is not always obvious to the knowledge base whether a mention of "Trump" refers to the 45th president of the USA, or the "Trump" organization which includes hotels and other businesses. We also find that DBPedia Spotlight will tend to find a Wikipedia article for any word if there is one, and report high similarity rankings as long as the surface form is similar. For our purposes, this typically does not pose a problem since such false positives occur rarely enough for any given instance not to interfere with the more prominent entities we are interested in. There are however a few disambiguations for which we had to find manual solutions. In particular, "Washington" is often incorrectly linked to the first US president rather than the city, "Trudeau" refers most likely to Justin, not Pierre, and mentions of "Tesla" are probably not about the 20th century inventor. We resolve those special cases manually.

Linked Entity Resolution

In a final step, we resolve the mentions of each entity with its corresponding real-world entity as found by our entity linking algorithm. This way of unifying the variants of each entity allows us to easily detect mentions of our entities for aspect based sentiment mining.

Types of Extracted Entities

Persons
We choose the entity type person since they are the most general kind human's we can extract. In preliminary testing we also try the entity time *'office holder'* and find that the top extracted people are office holders. Extracting persons in general therefore gives us the additional insight into the common properties of persons dominating our data set
Countries
We extract countries to see which countries are mentioned the most and to be able to get the sentiment towards these countries. Note that if a county is mentioned we expect the news source to be foreign to that mentioned country, since we expect that most of the time if one refers to its own country they do not mention it by name. This therefore would also result in international sentiment towards countries regarding the coronavirus.
Coronavirus
Lastly we extract mentions of the coronavirus (including all its synonyms) to be able to extract the sentiment towards this virus over the time.

3.5 Sentiment Analysis

So to detect the change of general sentiment over the whole article over time, we track the sentiment of the articles for each published date. We expect that it will start more neutral, become more negative and might become take a more positive towards the end. This could be correlated to specific trends and events to see what possibly changed the sentiment. We also track the sentiment towards the extracted person and country entities. Finally, we also track the sentiment specifically towards the coronavirus. Our sentiment scores are values in the range of -1 to 1. Where -1 is most negative, 0 is neutral and 1 is most positive.

3.5.1 General Sentiment Detection

The Spacy NLP framework does not contain any sentiment analysis models. Therefore we train our own. We first train our model on the Large Movie Review data set [7]. This data set contains 25000 movie reviews for training, and 25000 for testing. The movie reviews are either positive (rate ≥ 7) or negative (≤ 4) such that there is less ambiguity, the data set is balanced and contains the same amount of both positive and negative reviews.

The model we fine tune for sentiment analysis is the state of the art XLNET [8] model. This model has been found to score an accuracy of 96.21% on the Large Movie Review data set and 96.8% on the Stanford Sentiment Treebank data set [9]. In order to get more fine-grained sentiments we calculate the sentiment based on the activation values of the last layer. I.e. if the model gives 0.6 certainty for the positive label (and thus 0.4 for the negative label) we give the article a sentiment of $(0.6 - 0.4) * 2 = 0.2$.

3.5.2 Aspect-Based Sentiment Analysis

Other than the general sentiment analysis it could be interesting to extract sentiments in relation to specific targets. This is also called aspect-based sentiment analysis. It could for example happen that the article is overall negative but at the same time positive towards a specific person. In a review of the literature, we find that most of the work in the field of target and aspect based sentiment analysis (ABSA) is done on services and consumer products sentiment analysis [10],[11]. However, our data set contains news and our extracted entities are mainly persons and countries. A sentiment analysis model trained on consumer products might not be the best fit. To alleviate this shortcoming, we train the model on the twitter, and laptop and restaurant data set combined.

Twitter Data Set One data set that is more general is the twitter data set [12]. This data set consists out of twitter comments. Where for each tweet the main target and sentiment is labeled. The sentiment has three values, 1 for positive, 0 for neutral and -1 for negative. The data set has 6248 tweets for training and 692 tweets for testing. Annotated by two people with an agreement

rate of 82.5%. Example sentence in the data set:

Sentence	*yeah this is true , i like* T *... i just hate microsoft 's marketing department ... ; -RRB-*
T (Target)	windows 7
Sentiment	1

Laptop and Restaurant Data Set Two other data sets focused on ABSA are the laptop and restaurant data sets [13]. These data sets have a similar format to the twitter data set except that the target is always in relation to laptops or restaurants. Combined these data sets consists out of 6086 train examples and 1600 test examples.

Models

General Sentiment

To find the general sentiment of articles, we use a BERT based model with Local Context Focus Mechanism called LCF-BERT [14], this model performs state-of-art on the Laptop, Restaurant and Twitter data set. However, LCF-BERT is only able to analyse sentences with a maximum length of 250 characters. Longer sentences require more than 16 gb of gpu memory which does not fit even on a Tesla V100. However, some sentences will be longer than 250 characters. In this case we trim the sentence such that the target entity is always in the middle of the sentence and that the trimmed sentence is 250 characters long such that as much of the information is conserved. If a target has multiple occurrences within a sentence we target the first occurrence.

Aspect-Based Sentiment

For aspect-based sentiment mining we use the ABSA-PyTorch [4] framework as a starting point. We extend this framework such that it can train on all the data sets combined and such that it can be integrated within our pipeline.

[4] `https://github.com/songyouwei/ABSA-PyTorch`

Chapter 4

Results

4.1 Topic Analysis

For our topic analysis we define our topics on a sample of 20,000 articles published between 01.02.2020 and 05.04.2020. Articles longer than 2000 characters are cut off at this mark. Using these topics, we then measure topic distribution for each day in the same period for 1000 articles per day at a maximum article length of 1000 characters. The resulting topics can be seen in figure 4.1.

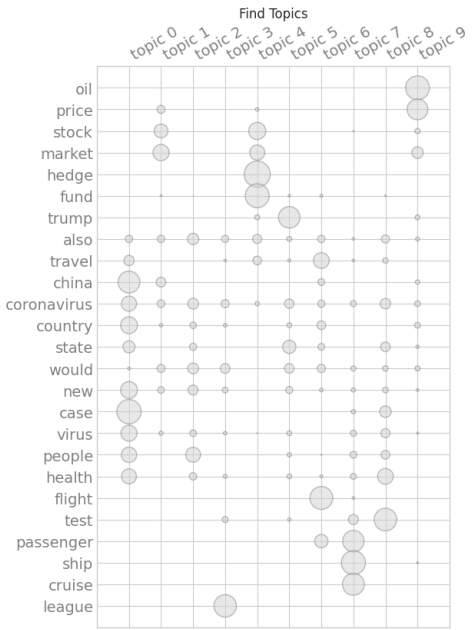

Figure 4.1: 10 NMF topics as found on our randomly sampled subset

4.1.1 Absolute Topic Prevalence

In figure 4.2[1] we can see the total number of Covid-19 news articles growing dramatically from mid-February onwards. We find that the topics identified by our algorithm seem to be internally consistent and seem to denote different areas of concern related to the Covid-19 medical crisis and the co-occurring economic crisis. We also observe an overall increase of news output from the end of February onward. In the period until March we also observe a periodic contraction indicating the regular news cycle with highs during the week and lows during the weekend. From there on, the number of articles on Covid-19 explodes to such an extent that the maximum number of samples we can process given our computational resources becomes the limiting factor and the news cycle is no longer visible.

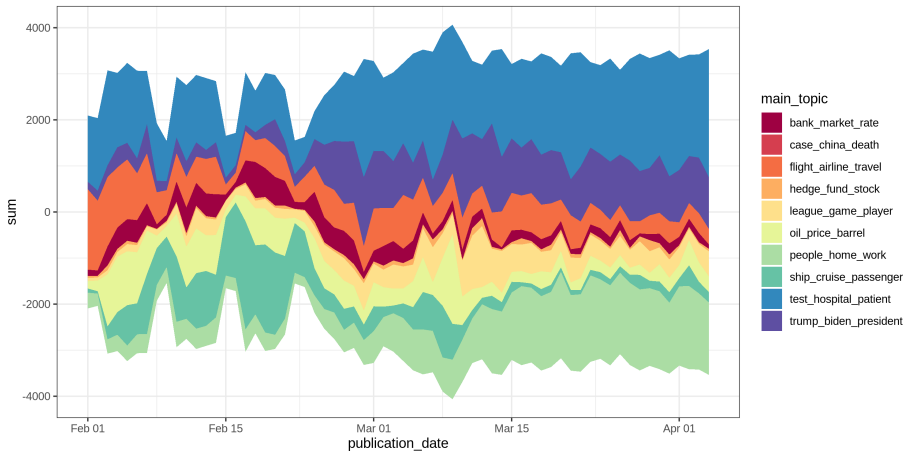

Figure 4.2: Topic river of absolute topic prevalence

4.1.2 Relative Topic Prevalence

In figure 4.3[2] we can see the development of the topics normalized for the number of articles published on any given day. Earlier during the time period, the news coverage is dominated by travel restrictions and concentrated outbreaks on board of cruise ships. With the arrival of the pandemic in the United States on the 29. in February, the US presidential race suddenly grows in importance. Over time, the lockdown and people working from home slowly grows to be a more prevalent topic, second only to the core of the crisis, namely the global medical crisis that is Covid-19.

[1] See figures/interactive/TopicAnalysisStreamgraph_sum.html in the repository for an interactive version.

[2] See figures/interactive/TopicAnalysisStreamgraph_mean.html in the repository for an interactive version.

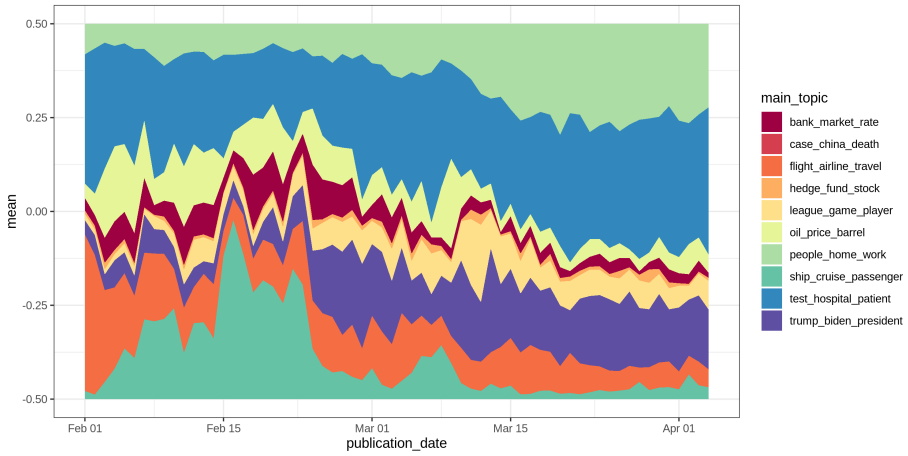

Figure 4.3: Topic river of relative topic prevalence

4.2 Entity Analysis

4.2.1 Person Entities

For our entity analysis we use a sample of our data over the period from the 01.02.20 to the 05.04.20, which is the period containing the majority of articles as seen in 2.1. From each day within the period we sample 1000 articles and cut them at a maximum length of 500 characters. Figure 4.4[3] shows a screenshot of the visualized named entities recognized by our models with the detected entity type set to "Person". Despite the rather general filter, we see that most of the persons recognized are major figures of international politics. As the Covid-19 crisis develops, we see politicians of the most affected countries rise to the front. This is most notable for Donald Trump, who, after the US has its first major recorded outbreaks by the end of February, takes a commanding lead and is by far the most mentioned person in English language Covid-19 news articles.

[3]See figures/interactive/entities_counts.mp4 in the repository for an interactive version.

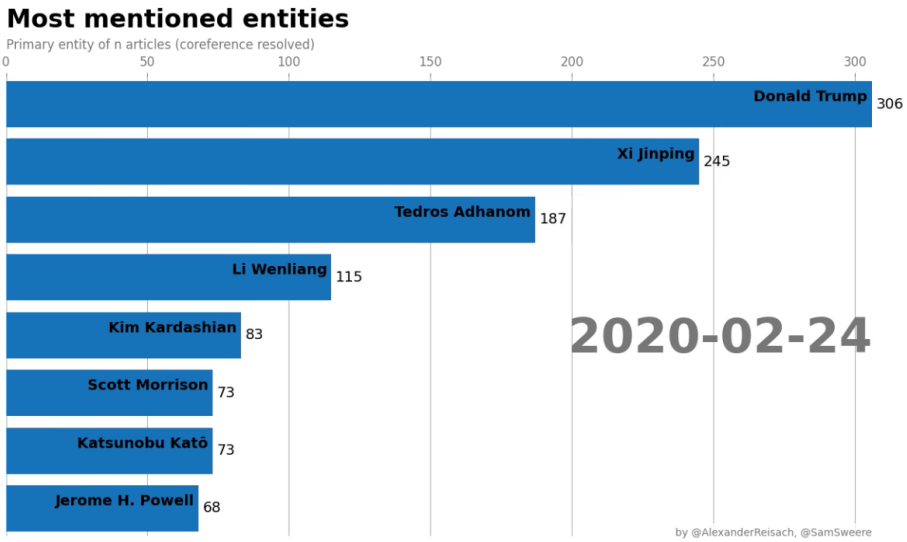

Figure 4.4: Cumulative entity mentions on the 24.02.20, just before the pandemic unfolds in the US. Sample period: 01.02.20-05.04.20, 1000 samples per day, max article length 500.

We enrich our entity count data by aspect-based sentiments. Our sentiments are in the interval $[-1, 1]$, with -1 as negative, zero as neutral and 1 as positive. We see that throughout the time period the general sentiment towards any entity seems to be quite negative (most of the bars appear as red). This does not come as a surprise, given that all articles are related to a major global crisis. Nonetheless, we do see positive sentiments as well, but they tend to be rare and short-lived. In figure 4.5[4] we observe Li Wenliang, the Chinese doctor who first warned of Covid-19 to be one of the few entities towards whom the average sentiment is positive.

[4]See figures/interactive/entities_sentiment.mp4 in the repository for an interactive version.

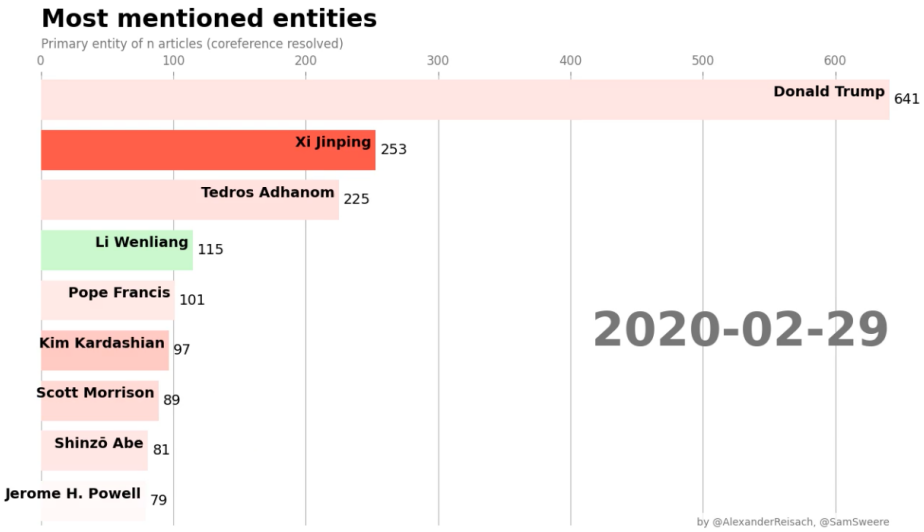

Figure 4.5: Average sentiment towards entities on the 29.02.2020. 1000 samples per day, max article length 500.

4.2.2 Country Entities

We find in 4.6[5] that the number of mentions varies greatly between countries. China, as the earliest country affected by the pandemic, accumulates by far the most mentions, with other badly affected countries such as the US, India and UK coming next. We can also see that some countries, in particular African ones are not mentioned at all, while Europe, Asia and North America know no such gaps.

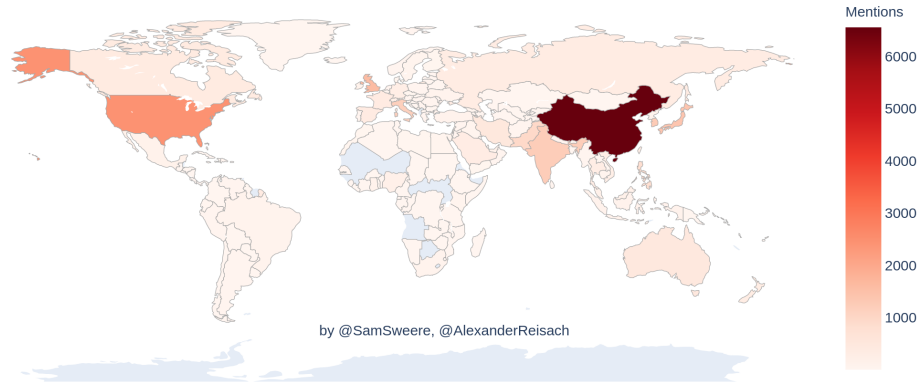

Figure 4.6: Mentions of counties over the period from 01-02-2020 until 05-04-2020. 1000 samples per day, max article length 500.

[5] See figures/interactive/country_sentiment.html in the repository for an interactive version.

4.3 Sentiment Analysis

4.3.1 General Sentiment Analysis

Sentiment	Accuracy
F-score	92.3
Precision	93.6
Recall	91.0

Table 4.1: Sentiment Accuracy of the XLnet based model on the Large Movie Review Dataset.

We trained the XLnet model for a few iterations and for a longer period of time. We found that the XLnet model that trained longer got a higher accuracy, F-score, precision and recall but performed worse on our news dataset. This could be because it started to over-fit on the Movie Review Dataset. We choose the model that trained for a shorter time to be in our pipeline. The results of this model are shown in table 4.1. For every analyzed article we determine the general sentiment using the XLnet model. The average sentiment of all the analysed articles can be seen in figure 4.7.

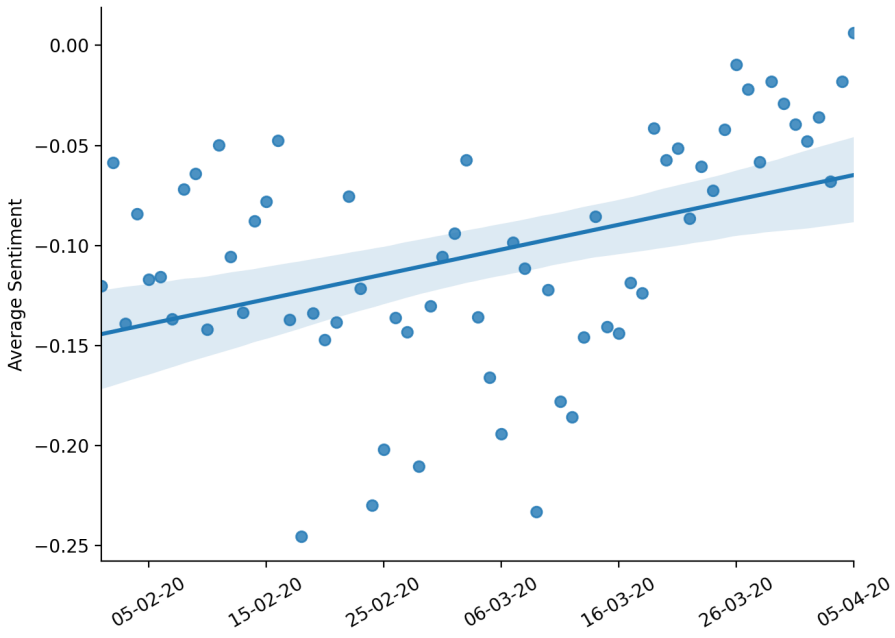

Figure 4.7: Average general sentiment of the articles in the period from 01-02-2020 until 05-04-2020. 1000 samples per day, max article length 500.

4.3.2 Aspect-based Sentiment Analysis

Initially we trained the LCF-BERT model only on the twitter dataset. This achieves close to state-of-the-art results on the twitter dataset as can be seen in table 4.2. However when using it in our pipeline we discovered that in the far majority of the cases it would give a neutral sentiment to the target, where in a lot of cases it should not. This could be because the twitter dataset is very vocal and uses a lot of extreme words. Our dataset sonsists of news articles many of which strike a neutral and fatual tone. We therefore decided to also train the dataset on the twitter, laptop and restaurant dataset combined hoping that it will perform better on our dataset. The results of the LCF-BERT model trained on the combined data set is visible in table 4.3.

Sentiment	Accuracy
F-score	71.41
Accuracy	72.83

Table 4.2: Sentiment Accuracy of the LCF-BERT model trained on the Twitter data set. The accuracy is calculated on the test set.

Data set	Accuracy	F-score
Combined	0.7563	0.7270
Twitter	0.7023	0.6808
Restaurant	0.7920	0.6842
Laptop	0.7524	0.7182

Table 4.3: Performance of the LCF-BERT model trained on the combined data set. The accuracy is calculated on the test set.

There are three entity types we analysed the sentiment towards: person, country and coronavirus.

Entity Type: Person The sentiment towards the person entities are visualised in the bar chart race, figure 4.5. The most prevalent person is Donald Trump, we see that the sentiment towards him is on average slightly negative. This stays roughly the same throughout time.

An interesting person to look at is Li Wenliang (the doctor who blew the whistle on COVID) starts off with having a negative sentiment towards him, especially after he just died on 7'th of February. This could be explained by that a lot of news articles could be referring to him as being deceased, which would probably give a negative sentiment. However, as time progresses we see the sentiment towards him become more positive after which he disappears as the main person entity in a lot of news articles.

Entity Type: Country The average sentiment towards referred countries is visible in figure 4.8[6]. We can see that most of the countries have a near neutral sentiment except for some exceptions where the sentiment is more extreme. This can be explained by noting that not every country is mentioned as often increasing the standard deviation (how many times a country is referred to can be seen in figure 4.6). Comparing the countries that are the main mentioned country entity we see that China has the most negative sentiment with average sentiment of -0.36 (main country 6553 times), compared to USA -0.28 (main country 2456 times), United Kingdom -0.24 (main country 1612 times), India -0.19 (main country 1239 times) and Italy -0.33 (main country 1164 times). We can also observe that the countries in the African continent have the most negative sentiments towards them, however, note that these countries are not often the most mentioned country in a news article, therefore we expect the standard deviation to be higher. Nicaragua has the most positive sentiment with 0.85, but is only the main mentioned country 2 times.

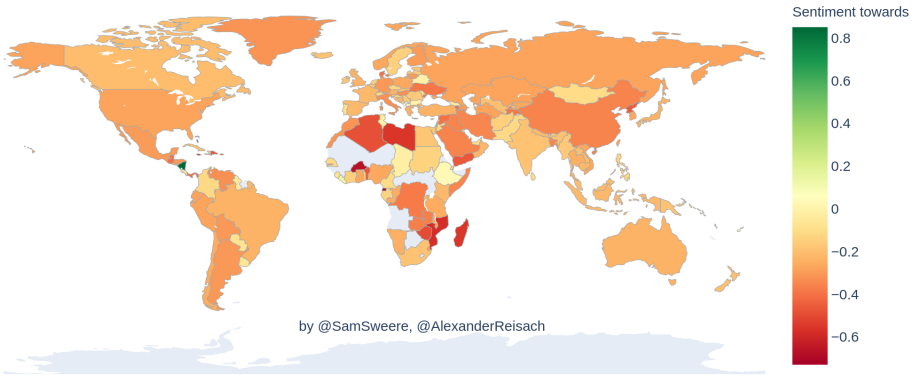

Figure 4.8: Average sentiment towards referred countries in the period from 01-02-2020 until 05-04-2020.

Entity Type: Coronavirus We also extracted the sentiment towards the coronavirus itself (and all the different naming variations).

The average sentiment towards coronavirus can be seen in figure 4.9.

[6]See figures/interactive/country_mentions.html in the repository for an interactive version.

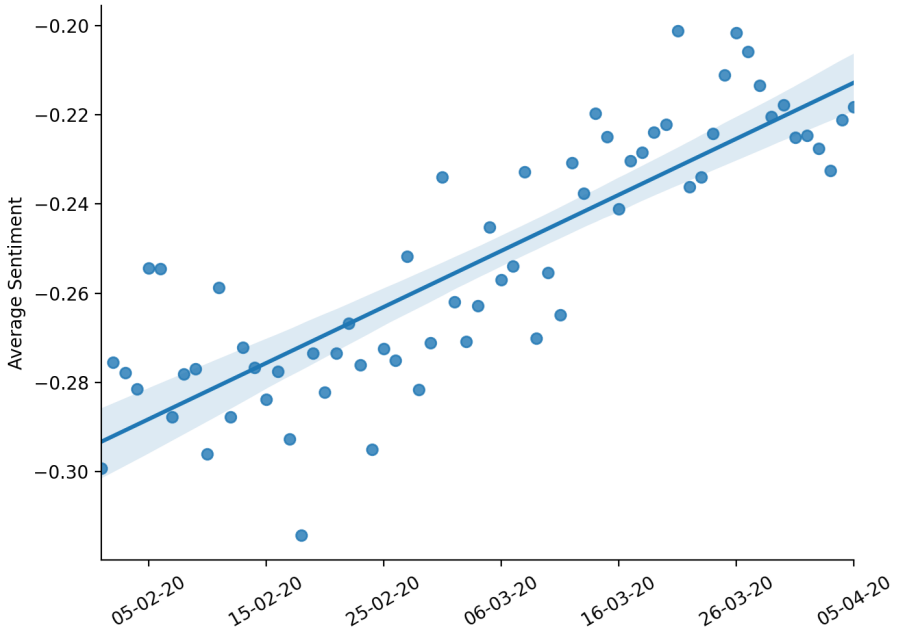

Figure 4.9: Average general sentiment towards coronavirus in the period from 01-02-2020 until 05-04-2020.

Both the general sentiment (figure 4.7) and the sentiment towards corona (figure 4.9) become more positive over time. Even when taking the uncertainties of our sentiment models into account this points to a general trend. This could indicate that the news about corona became more neutral over time.

4.3.3 Evaluation

To see how our model performs we first analyse it on made up test examples. A known drawback of BERT models is that they are not always very good at detecting negations. We therefore made up test examples that are positive, negative and negated negative (and thus positive). A full collection of the examples is visible in appendix A.1. We observe, that for strong sentiments the ABSA model evaluates positive, negative, and even negated positive statements correctly. However, in cases where the sentiment is not as strong, we find that it is less likely to correctly evaluate negations.

For the general XLnet sentiment model we obtain less consistent results and sometimes even see negative sentiments assigned to strictly positive statements. Despite these drawbacks, we find that the model usually gets the tendency and relative ordering of sentiments right.

Article Body	*Turkmenistan bans word 'coronavirus' ASHGABAT: By banishing the word "coronavirus" from the Turkmen vocabulary in a radical move to suppress all information about the pandemic, Turkmenistan's government is putting its citizens in danger, Reporters Without Borders (RSF) says. The state-controlled media are no longer allowed to use the word and it has even been removed from health information brochures distributed in schools, hospitals and workplaces, according to Turkmenistan Chronicle, one of t*
Extracted Person	None
Extracted Country	Turkmenistan
General sentiment	-0.93
Person sentiment	None
Country sentiment	-0.473
Corona sentiment	-0.305

News articles examples Our model correctly extracted the country and determined a negative sentiment towards Turkmenistan. We agree that this sentiment is correctly negative. The sentiment towards *coronavirus* is only slightly negative, this also is in line since the coronavirus is only referred to in a neutral sense.

Article Body	*MANILA - The local government of Navotas is filing charges against 459 residents for violating the enhanced community quarantine imposed in the entire island of Luzon to prevent the spread of COVID-19. Navotas Mayor Toby Tiangco said the local government filed charges against the 459 for violating Article 151 of the Revised Penal Code on resistance and disobedience to a person in authority, which is punishable by a fine not exceeding P100,000 and imprisonment of up to 6 months. Among those arr*
Extracted Person	Toby Tiangco
Extracted Country	None
General sentiment	0.03
Person sentiment	-0.62
Country sentiment	None
Corona sentiment	-0.75

In the above example our model correctly extracted the person *Toby Tiangco*. However, the sentiment it gave towards *Toby Tiangco* is quite negative while we would argue that he is only stating things and should therefore have a more neutral sentiment. The sentiment of the whole article is neutral, we think this is correct since it is formally written, only stating facts.

Chapter 5

Discussion

5.1 Topic Analysis

We currently label topics by simply connecting their top terms. While this gives an idea of what a topic is about, the meaning of a topic could be more accurately captured by a method that takes all topic constituents into account. This could be combined with linking some of the topic words to their real-world entity. Moreover, an analysis of the topics for different given topic numbers could give additional insights into which topics tend to dominate the news discourse on Covid-19 to what degree. Further insights could be gained by a more thorough detailed analysis of the relationships of the different topics to one another. For example, it could be interesting to see how the top terms for each topic change as we decrease or increase the number of topics.

5.2 Entity Detection

In effect, we run two completely separate pipelines for entity detection, one for the purpose of coreference resolution (spacy) and one for the purpose of identifying underlying real-world entities (DBPedia). We find that both of the named entity recognition methods have very different strengths and weaknesses. The spacy NER module allows fine tuning as to when to split entities. It tends to classify entities by the way they are used in a sentence, which results in detections that make intuitive sense. On the flip side however, a word like "Coronavirus" can end up being classified as a person because it is frequently used as the subject in sentences with an active verb such as *threatens*, or *prevents*. DPPedia Spotligt's NER module on the other side, plucks entities apart such that for example "Washington state Gov. Jay Inslee" will be seen as mentions of both "Washington", and the politician "Jay Inslee". We are currently only using both models for separate purposes, however there might be a more structured way to use both models in combination to resolve disambiguations and obtain more accurate results.

5.3 Aspect Based Sentiments

We find that our sentiments do capture the general sentiment in connection with an entity, but a closer look reveals that the same sentiment can be triggered by

very different causes. For example, in 4.5 we see that both, US politician Mike Pence and actor Tom Hanks have a very negative sentiment. We know however, that this is during the time when Tom Hanks contracted Covid-19, so it seems likely that the negative sentiment refers to his health and does not constitute an opinion towards him as a person. On the same day, Mike Pence gave a speech at a NATO summit that was not well-received internationally. This example shows the ambiguity of sentiment. An annotation with additional data such as the topics of the articles that mention the most prominent entities could help put the sentiment into context.

5.4 Computational Limitations

Our pipeline is computationally very expensive, in part due to multiple big deep-learning models. This forced us to limit the amount of articles we could analyse within a time period and to trim the article length if they were too long. We ended up trimming the articles to a maximum character length of 500. Almost all articles are of course longer than 500 characters, we thus do not analyse a big part of the data set. Even with these restrictions it took over 24 hours on a machine with 8 cores and all the deep learning sentiment models being gpu accelerated to retrieve the results. For future work the pipeline could be optimised more and be run on stronger hardware for a longer period to get more (accurate) results.

5.5 General Sentiment Training Data Set

The data set we used to train the general sentiment model is the Large Movie Review Data set [7]. This data set however is quite different from our data set, which consists of news articles. We could therefore expect that the results of a model trained on a data set that is more in line with the writing of news articles would be better. For future work we could retrain the model on the Stanford Sentiment Treebank data set [9] in the training process. This data set consists out of 215,000 phrases with fine-grained sentiment labels. This data set would be more in line with our data.

5.6 Twitter Dataset Quality

In the twitter dataset [12] which we use to train our aspect-based sentiment model we discover some inconsistencies or annotations which we consider mistakes. Examples are:
In our opinion this should be labeled as being positive towards Hillary Clinton.
Another example:
In our opinion this sentiment should be negative towards Sarah Palin.

Sentence	*holy shit ! i would fucking kill to see* T ...
T (Target)	hillary clinton
Sentiment	-1

Sentence	*" Obama deserves the Nobel prize for the simple reason that if it were n't for him* T *would be in the White House . "*
T (Target)	sarah palin
Sentiment	0

It is unclear who did the annotation and perhaps they were not native English speakers. Expressions such as *"i would fucking kill to see"* could be interpreted wrongly if you only formally learned English. This also emphases how hard this challenge is. If even humans have trouble annotating the data, how well can we expect a machine learning model to learn it?

One way to improve our ABSA model would be to let it train on a dataset generated from a multi-domain text corpus, such as the *ZyLAB-Targeted-Sentiment-Reviews* [15]. Since the vocabulary regarding multiple domains could positively affect the model to be able to extract the sentiment better in more diverse sentences.

Chapter 6

Conclusion

We load and pre-process the Aylien Covid-19 news data set. For our analyses, we focus on the period with most news coverage from the 01.02.20 to the 05.04.20. For topic analysis, the text is cleaned and a number of custom-defined stopwords are removed. We observe that the phrases making up the topics are consistent with one another and the trends in topics are in line with the development of the pandemic. A more thorough analysis of the topic constituents, the relationships between topics, and an analysis for different numbers of topics could help identify further insights into the nature, robustness and interaction of the topics. For entity analysis, we first apply a standard pipeline of tokenization, POS-tagging, dependency parsing, and entity recognition. The entity recognition in this first run is a CNN based model. On this basis, we resolve co-references to the same entity. This allows us to find a lot more patterns in the data and results in entity counts that are in line with the real-world geopolitical importance of the entities identified. To identify these real-world entities we use DBPedia Spotlight. The entity matches are generally of high quality but we do find disambiguations and mistakes in the way entities are split up. A suitable combination of the neural network based and knowledge-based approaches for entity recognition could help improve these results. For general sentiment analysis, we fine-tune XLnet, a state-of-the-art sentiment model and apply it to our news data. We find that the general sentiments identified are not entirely consistent with our evaluation of text samples but have the right relative order. For aspect-based sentiment mining, we apply LCF-BERT and train it on a diverse mix of data sets to improve its generalization and applicability to our data set. We find that aspect-based sentiments are typically negative and only some prominent examples exhibit prolonged positive sentiments. Both the global sentiment and the sentiment towards COVID-19 are negative on average but become more positive over time. In examples we see that strong sentiments are typically evaluated correctly more often than subtle sentiments. A training data set more similar to our data set could help improve classification performance. We use topic rivers, animated bar charts and choropleth maps to show the temporal and geographical trends in topics, entities and sentiments. We provide our results in different formats that are suitable for interactive exploration and web-based presentation.

Bibliography

[1] Y. Goldberg and J. Nivre, "A dynamic oracle for arc-eager dependency parsing," in *Proceedings of COLING 2012*, pp. 959–976, 2012.

[2] E. Kiperwasser and Y. Goldberg, "Simple and accurate dependency parsing using bidirectional lstm feature representations," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 313–327, 2016.

[3] e. a. Weischedel, Ralph, "Ontonotes release 5.0," *Philadelphia: Linguistic Data Consortium, 2013*, 2013.

[4] K. Clark and C. D. Manning, "Deep reinforcement learning for mention-ranking coreference models," *arXiv preprint arXiv:1609.08667*, 2016.

[5] J. Daiber, M. Jakob, C. Hokamp, and P. N. Mendes, "Improving efficiency and accuracy in multilingual entity extraction," in *Proceedings of the 9th International Conference on Semantic Systems*, pp. 121–124, 2013.

[6] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer, *et al.*, "Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015.

[7] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, (Portland, Oregon, USA), pp. 142–150, Association for Computational Linguistics, June 2011.

[8] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in neural information processing systems*, pp. 5754–5764, 2019.

[9] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

[10] H. H. Do, P. Prasad, A. Maag, and A. Alsadoon, "Deep learning for aspect-based sentiment analysis: a comparative review," *Expert Systems with Applications*, vol. 118, pp. 272–299, 2019.

[11] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning based text classification: a comprehensive review," *arXiv preprint arXiv:2004.03705*, 2020.

[12] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent twitter sentiment classification," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, (Baltimore, Maryland), pp. 49–54, Association for Computational Linguistics, June 2014.

[13] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, *et al.*, "Semeval-2016 task 5: Aspect based sentiment analysis," in *10th International Workshop on Semantic Evaluation (SemEval 2016)*, 2016.

[14] B. Zeng, H. Yang, R. Xu, W. Zhou, and X. Han, "Lcf: A local context focus mechanism for aspect-based sentiment classification," *Applied Sciences*, vol. 9, no. 16, p. 3389, 2019.

[15] Z. Gerolemou and J. C. Scholtes, "Target-based sentiment analysis as a sequence-tagging task*," 2019.

Appendix A

Appendix

A.1 Model Evaluation - Examples

Article Body	*The professor Yann LeCun hates it when his students have purple hair.*
Coreference Resolved	*The professor Yann LeCun hates it when The professor Yann LeCun students have purple hair.*
Extracted Person	Yann LeCun
General sentiment	-1.0
Person sentiment	-0.94

Article Body	*The professor Yann LeCun does not like it when his students have purple hair.*
Coreference Resolved	*The professor Yann LeCun does not like The professor Yann LeCun when The professor Yann LeCun students have purple hair.*
Extracted Person	Yann LeCun
General sentiment	-0.9
Person sentiment	-0.89

Article Body	*The professor Yann LeCun likes it when his students have purple hair.*
Coreference Resolved	*The professor Yann LeCun does likes it when The professor Yann LeCun students have purple hair.*
Extracted Person	Yann LeCun
General sentiment	-0.71
Person sentiment	0.78

Article Body	*People in Greenland do have the burden of the coronavirus, since they are isolated from the rest of the world they do have to worry about the disease and all its negative implications.*
Coreference Resolved	*People in Greenland do not have the burden of the coronavirus, since People in Greenland are isolated from the rest of the world People in Greenland do not have to worry about the disease and all its negative implications.*
General sentiment	0.16
Country sentiment	-0.84
Corona sentiment	-0.76

Article Body	*People in Greenland do not have the burden of the coronavirus, since they are isolated from the rest of the world they do not have to worry about the disease and all its negative implications.*
Coreference Resolved	*People in Greenland do have the burden of the coronavirus, since People in Greenland are isolated from the rest of the world People in Greenland do have to worry about the disease and all its negative implications.*
General sentiment	0.25
Country sentiment	-0.51
Corona sentiment	-0.51

Article Body	*People in Greenland have a great time, even though the coronavirus is terrible for the rest of the world they keep their positive vibe.*
Coreference Resolved	*People in Greenland have a great time, even though the coronavirus is terrible for the rest of the world People in Greenland keep People in Greenland positive vibe.*
General sentiment	1
Country sentiment	0.86
Corona sentiment	-0.63