

Franz Liszt in Denmark

Authors: Kristian S   Corvalan (studentID: 202208514) **and Clara Katrine Sydow** (studentID: 202204626)

This report is a product of a fully collaborative endeavor between the two authors.

Keywords: *Mediastream, newspapers, Franz Liszt, RStudio, text mining, sentiment analysis*

Introduction

In this project, we aim to examine how the Austro-Hungarian composer and pianist Franz Liszt was received in Danish media when he first came to Denmark in 1841. The virtuous pianist came not only with a spectacular way of composing and playing at concerts but was also seen as a symbol of the new ideas of liberalism that appeared after the July Revolution in France in the 1830s. The examination of how Danish newspapers described Franz Liszt and his concerts may thereby contribute to an understanding or discussion on how the new liberal ideas were received and expressed by the Danish people and media. In addition, we also want to investigate if there are any differences geographically in the emotions towards Franz Liszt. Our hypothesis on the geographical matter is that the articles from Copenhagen will appear more positive towards Franz Liszt than the rest of Denmark.

By using text mining in R Studio we will look at the words that were used to describe Franz Liszt and in which context they appeared alongside with a sentiment analysis that assesses the emotion of the words and sentences.

Problem Orientation and Background

Historical and cultural background

Franz Liszt came to Denmark in a period, where the July Revolution of France in 1830 had left its marks on Danish society. The liberal ideas of somehow political and cultural freedom in public, which led to a more commercial culture in France, were being expressed in some parts of the Danish media.¹ Mostly the part that belonged to the upper class in Copenhagen. These newspapers fought for more democratic reforms that would get rid of censure and

¹ 1. Ole Krogsgaard, "Offentlige Debatfora 1830-1849," Aarhus Universitet, Danmarshistorien.dk

political repression. But on the other hand stood the monarch with absolute power. So the Danish media had to find creative ways to criticize the rulers.²

One of these opportunities appeared when Franz Liszt came to Copenhagen in July 1841 where he gave three solo concerts at The Royal Theatre.³

Franz Liszt who was born in modern Austria but moved to Paris to study compositions and piano became famous not only for his unique piano playing but also for his attraction of public masses which changed the concert culture in parts of Europe including Denmark.⁴

Alongside as Franz Liszt became a popular culture phenomenon the music press evolved with reviews of concerts.⁵ Through concert reviews the Danish media now had the opportunity to wrap critics of society in reviews of concerts, music and artists.

Some of the newspapers that were critical of the absolute power the Danish monarch possessed used the concerts of Franz Liszt to express their revolutionary ideas and made Franz Liszt a symbol for the new liberal society and culture.

The field: related work and scripts

As recordings of Franz Liszt's concerts don't exist, what we, as historians, have to work with are secondhand sources in the form of for example reports, memoirs and reviews. This means that the media has played an essential role in how Liszt was and is viewed.

In his examination of the virtuoso Liszt, Lawrence Kramer, professor of English and Music at Fordham University, uses sources like the Danish author Hans Christian Andersen and other composers and pianists like Johannes Brahms, Robert Schumann and Frédéric François Chopin to look at how Liszt was viewed by people who knew him and his work and had seen him play.⁶

Peter. E. Nissen, mag.art, has examined Franz Liszt's impact on Danish musical life and has taken a closer look at the three concerts Liszt played in July 1841 in Denmark. Nissen believes that these concerts changed the musical life in Denmark and led to the Danish press experimenting more with the genre of reviews. The sources Nissen uses to investigate this are

²Nissen, Peter E. "Da Franz Liszt kom til byen : musik og revolution i København år 1841." In *Fortid og nutid* (2009). p. 2

³ Nissen p. 9

⁴ Kramer, Lawrence "Musical meaning: Toward a critical history" in *Franz Liszt and the Virtuoso Public Sphere* p. 69

⁵ Nissen p. 5

⁶ Kramer p. 67

reviews and discussions about Liszt in Danish newspapers in the short period following Liszt's concerts.⁷

Likewise, other academics have written about Liszt, his music, impact and life.

We want our paper to be a contribution to this field and want our examination of how Liszt was viewed by the Danish media to possibly make it easier to broaden the horizon sourcewise, by making an overview of all articles mentioning Liszt in the year 1841.

To make this possible we used text mining and sentiment analysis. We used the parts of coding that made sense for our examination from scripts made by Max Odsbjerg Pedersen to do the text mining part of our paper.⁸ To do the sentiment analysis we used a script and paper about Danish sentiment analysis.⁹

Software Framework

For this project, we used a newly bought Macbook Air (2020), 8 GB RAM, which runs macOS Ventura 13.4. We worked on the desktop version R (4.3.1) and RStudio (2023.06.2+561).

Data Acquisition and Processing

The data we used in this paper was extracted from Mediestream. As we wanted to extract the newspaper articles that mention Franz Liszt in the year 1841 we used the following regular expression:

```
Fran* Lis*t AND iso_date:[1841-01-01 TO 1842-01-01]10
```

the two * was added as we knew that the newspapers used different ways of spelling his name and because the OCR scan could have some problems recognizing the correct letters.

We used iso_date to get all articles in a specific period. It's important to write the date in the following format: yyyy-mm-dd.¹¹

This gave us 158 hits.

⁷ Nissen p. 3

⁸ Pedersen, Max Odsbjerg, *Åbne data - Ringe OCR-kvalitet*; Pedersen, Max Odsbjerg & Mulvad, Toke Jøns. *Text Mining St. Croix Avis*.

⁹ Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher. *SENTIDA*

¹⁰ Fran* Lis*t AND iso_date:[1841-01-01 TO 1842-01-01], in Mediestream

¹¹ Pedersen, Max Odsbjerg. *Snydeark til Mediestream*.

To load the data into RStudio, which is the program in which we worked, we retrieved a link from the experimental API via Swagger UI made by the Royal Danish Library.¹² We used the Lapsapi called GET: aviser/export/fields. Here we wrote our regular expression and specified that we wanted the data to be in CSV format. The API then executed an URL that we loaded into R.

Empirical Results

Word Clouds

The first figure shows the words used by newspapers in different cities to describe the virtuoso, Franz Liszt. The words shown in the word clouds are the most frequently used in that specific city. Words used frequently across all the cities have been removed as most of these words would be filler words, like prepositions and particles. This was done by calculating the individual word's frequency and by punishing the words that appear most across the newspapers in the different cities. We followed Max Odsbjerg's coding¹³ and some of the main principles from the book: *Text Mining with R: A Tidy Approach* (2022), chapter 3.¹⁴ Since we were interested in figuring out whether there was a difference in the way Liszt was viewed by different cities we used the `bind_tf_idf` function. The function multiplies the term frequency (tf) with the inverse document frequency (idf). By doing this, `bind_tf_idf` gave us the most important words per city in relation to the other cities. It gives more weight to words that are less frequently used in an attempt to find the important words and gives a value of zero to words that are the most common ones across all the cities.¹⁵ The following word cloud is the result of this:

¹² Swagger UI

¹³ Pedersen, Max Odsbjerg & Mulvad, Toke Jøns. *Text Mining St. Croix Avis*

¹⁴ Silge, Julia & Robinson, David. "Analyzing word and document frequency: tf-idf" in *Text Mining with R: A Tidy Approach*

¹⁵ Silge, Julia & Robinson, David. "Analyzing word and document frequency: tf-idf"

Importance determined by term frequency (tf) - inversed document frequency(idf)

Data from Mediastream Experimental API

This resulted in some new words appearing in the word clouds and others that became more visible but generally not a huge difference, as can be seen underneath in the next figure:

Important words describing Liszt ind Danish newspapers in different cities with new stopword list



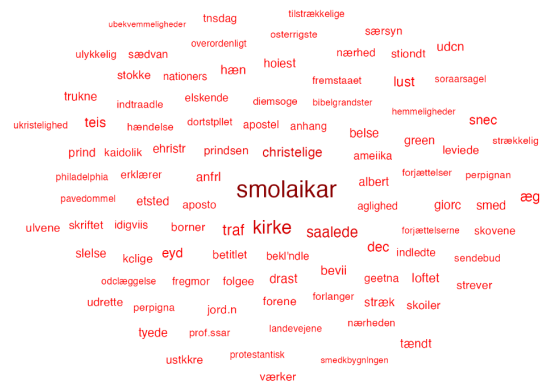
Data from Mediestream Experimental API

To take a closer look at some of the cities' word clouds, we made a new data frame consisting only of words from Roskilde, in the case below, by using the filter function that allowed us to pick out cities from our location column (named lplace).

We then entered the new data frame in our word cloud code and ran it:

Important words in Roskilde

Roskilde



Data from Mediestream Experimental API

Our new individual word cloud consisting only of words from Roskilde now has plenty of words for us to analyze.

We then made a word cloud for Aarhus/Århus:

Important words in Aarhus/Århus

Aarhus

familie
fial
forevares
publikum
len
te
tel
madame

Århus

foder
habil
trufne
djaervhed
beundre
budgettet
erempler
dcr
gylden
overstiger
franstmanden
gjennemsnitssummen
fordum
sabel
tilsidst
neutral
budgetter
erhvervede
franstmand
lovtaaler
indquarteret

Data from Mediestream Experimental API

A lot of the words in both these individual word clouds are still meaningless and not relevant for our examination but words such as “ukristelighed” or “elskende” in Roskilde and “beundre” and “publikum” in Aarhus/Århus are words that could be interesting to have a further look at.

Octograms

Therefore, we made octograms, following Max’s coding.¹⁶ The word clouds above have all focused on individual words that the function `bind_tf_idf` had calculated as the most important words per city. However, as we want to analyze a big chunk of text, in the form of multiple articles mentioning Franz Liszt in Denmark in the year 1841, we want to be able to analyze the context in which these important words appear. We have chosen to look closer at two important words from the Aarhus/Århus word cloud. We made this possible by using the

¹⁶ Pedersen, Max Odsbjerg, *Åbne data - Ringe OCR-kvalitet*.

function `unnest_tokens`, which we used earlier to divide the full text into separate words.¹⁷

The code we used can be seen here:

```
```{r}
liszt_denmark %>%
 unnest_tokens(octogram, fulltext_org, token = "ngrams", n = 8) -> octogram
```
```

To make the octograms, we added the option: `token = "ngrams"` and set the `n`-value to 8, which is the number of words we wanted each `n`-gram to contain. By setting the value to 8 it becomes octograms.

Then we used the separate function to separate the 8 words:

```
```{r}
octogram %>%
 separate(octogram, c("word1", "word2", "word3", "word4", "word5", "word6", "word7", "word8"), sep = " ") -> octogram_sep
```
```

This will make it easier to filter the words in the octogram and therefore make it easier to search for the context of specific words.

Hereafter we used the filter function to first tell R to give us a specific city and then to look after a specific word and where in the octogram the word should appear. We used the unite function to make sure the words were put back together so that the context would be visible to us:

```
```{r}
octogram_sep %>%
 filter(lplace == "Århus") %>%
 filter(word8 == "beundre") %>%
 unite(octogram, word1, word2, word3, word4, word5, word6, word7, word8, sep = " ") %>%
 select(octogram, everything()) -> beundre
beundre
```
```

We told R to filter Århus and then to look for the word `beundre` (admire) and that this word should appear as word number 8 in the octogram.

This gave us the following results:

A tibble: 2 × 16

octogram

<chr>

endnu i fordobler grad maa velsigne og beundre

sag frants li sir maa man derfor beundre

¹⁷ Silge, Julia & Robinson, David,. "Relationships between words: n-grams and correlations" in *Text Mining with R: A Tidy Approach*

From this, we can read in the first octogram, that the author says something about blessing and admiring, and in the second octogram that Franz Liszt, which due to the bad OCR scan has been written as “frants li sir”, is someone to be admired. The contents of these two octograms seem very interesting for our examination, and therefore we want to look at the articles in which they appear. In the RMarkdown we made it possible to see in which newspaper the octograms appear, the timestamp for the newspaper and a direct link to the article in Mediestream. The two octograms appeared in the two following quotes¹⁸:

*Et Træf som dette er saa characteristisk, at man, ved at fjende det, endnu i fordoblet Grad maa velsigne og beundre den Guddomsfunke, der aabenbarer sig under hans Spil, som kun kan have sit Udspring fra Lysets Væld.*¹⁹

*Frants List maa man derfor beundre, saavel som Kunstner som for sit Menneskeværd, og fra denne Side betragtet maa hans harmoniske Toner endnu kraftigere gjendkaldes i Erindringen.*²⁰

By making these octograms, and thereby being able to look at the context of the word “beundre”, it becomes very clear that the word appears in a context in which Franz Liszt is viewed with extremely positive feelings. He is being praised for his worth as an artist and it seems as though the newspaper Jyllands Posten in Århus views Liszt very positively.

Then we continued by using the same coding with the word “publikum” (audience) in Aarhus (same city, just spelt differently). But this time we chose that the word should appear as the first word in the octogram:

¹⁸ Translation of the first quote: “A coincidence like this is so characteristic that, by hating it, one must still doubly bless and admire the spark of divinity that reveals itself during his playing, which can only have its origin in the power of light.”

Translation of the second quote: “Frants List must therefore be admired, both as an artist and for his human dignity, and from a consideration of this perspective, his harmonic tones must be even more forcefully recalled in the memory.”

¹⁹Jyllandsposten (1838-1841). Mediestream, August 6, 1841.

²⁰ Jyllandsposten (1838-1841). Mediestream, August 6, 1841.

```

```{r}
octogram_sep %>%
 filter(lplace == "Aarhus") %>%
 filter(word1 == "publikum") %>%
 unite(octogram, word1, word2, word3, word4, word5, word6, word7, word8, sep = " ") %>%
 select(octogram, everything())-> publikum
publikum
```

```

It gave us the following four octograms:

A tibble: 4 × 16

octogram
<chr>

```

publikum ak spille dein alle igjennem han bad
publikum cm at klappe naar han spille en
publikum taug endelig begyndte han marseillaisen der strap
publikum at man ikke vilde give slip paa

```

4 rows | 1-1 of 16 columns

We looked closer at the columns telling us which newspapers had written the articles and when. All the octograms originate from the same article published in Aarhus Stifts-Tidende on the 30th of July 1841. The two first octograms appear in the same quote²¹:

*L. fremtraadte med en heel Pakke fuld, men mente dog, at det vilde trætte Publikum at spille Dem alle igjennem, han bad derfor meget galant Publikum om at klappe, naar han spilte en Melodi, som man ønskede, han skulle vælge; paa den Maade vilde han tage 3 Themaer.*²²

This octogram shows us that Liszt asked his audience to participate in his performance by clapping, and thereby choosing the melodies he should play. This could be seen as an example of how the reviewers were surprised by the way Liszt held his concerts and how he contributed to a new way to engage with music.

We believe that the creation of octograms, or other forms of n-grams, makes it possible to quickly get an idea of the context in which important words appear in these articles. This

²¹ Translation: “L. came forward with a whole packet full but thought that it would tire the audience to play them all through, he therefore very gallantly asked the audience to clap when he played a melody that they wanted him to choose; by doing this he chose 3 themes”

²² Den Til Forsendelse Med Brevposterne Kongelig Allernaadigst (Alene) Privilegerede Aarhus Stifts-Tidende (1821-1844). Mediestream, July 30, 1841.

method can easily be applied to bigger datasets and other kinds of written materials. By using the octograms we broaden the view of our examination by not just looking at what words pop up in the word cloud, but by looking at what context they appear in, and thereby being able to see how Liszt was viewed by newspapers in different cities.

Sentida Analysis

To investigate the overall sentiment of the articles about Franz Liszt we used the tool SENTIDA which analyzes the general sentiment of words in Danish. To install the tool in R we followed the instructions in Github²³ made available by the developers of Sentida.

To be able to download the Sentida tool from Github we had to install the “devtools” function and with this function, we could download the Sentida Tool directly from Github:

```
library(devtools)

if(!require("devtools")) install.packages("devtools")

devtools::install_github("Guscode/Sentida")

library(Sentida)
```

To include the Danish letters æ, ø and å we used the following code which was also available in the Sentida Github.

```
# Code to include the danish letters æ, ø and å
Sys.setlocale(category = "LC_ALL", locale = "UTF-8")
```

Before running the Sentida tool we made a dataframe with a single column consisting of all the words of all the articles.

We then entered the new data frame in the “sentida” function.

```
sentida(liszt_only_words, output = "mean")
```

As the code illustrates we also had to choose which output we wanted from the Sentida sentiment analysis. Either the argument output = “total” which would provide an accumulated score for all words in our data frame or the output argument output = “mean” which would result in a mean sentiment score of all the words.

The “mean” argument was the most relevant for our investigation and by running the code above we got the sentiment score:

²³ Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher. *SENTIDA*

[1] 0.1861439

This score indicates a very weak positive emotion in average in all the articles about Franz Liszt. Sentida's rating system goes from -5 (most negative) to 5 (most positive).²⁴

Since we from the start of the project had a hypothesis of Copenhagen likely being more positive towards Franz Liszt than the rest of Denmark, we wanted to make a separate sentiment analysis of the articles from Copenhagen and the rest of the cities.

Before making a sentiment analysis of articles from cities excluding Copenhagen we had to make a new data frame without the words from Copenhagen.

We used the following code (found on R-bloggers²⁵)

```
liszt_no_cph <- liszt_new_nostop %>%  
  filter(!lplace %in% c("Charlotte Amalie", "Christianssted", "København" ))
```

To make sure that all the names of the cities did not affect the sentiment analysis we made a data frame consisting only of a single column containing words from all cities excluding Copenhagen.

We then entered the new data frame in the Sentida function and chose the output argument "mean" which resulted in the score:

```
sentida(liszt_word_no_cph, output = "mean")
```

[1] 0.2408006

We then used the same method to analyze the sentiment of the words from articles from Copenhagen which gave us the score:

[1] 0.1430159

To judge from these Sentida sentiment analysis scores the articles from around the country excluding Copenhagen had a slightly more positive emotion in their articles about Franz Liszt which was the opposite of our hypothesis from the start.

²⁴Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher (2019). SENTIDA: A New Tool for Sentiment Analysis in Danish. *Journal of Language Works - Sprogvidenskabeligt Studentertidsskrift*, 4(1), p. 51

²⁵ Jim, "How to Use 'Not in' Operator in Filter," R-Bloggers (blog), July 17, 2022.

Critical evaluation

When processing data from Mediastream to an API we asked Mediastream to collect all the articles where Franz Liszt was mentioned in a 1 year period from 1841 to 1842. The way Mediastream identifies the articles which contain Franz Liszt is by scanning the page for rubrics, subheadings and body text because that is how most modern articles are built. But newspapers from the 1840s did not quite use the same structure.²⁶ There are fewer rubrics and subheadings which makes it more difficult for Mediastream to identify the articles. This means that our dataset also includes text from other articles that Mediastream scanned as part of the ones about Franz Liszt. So not all the words that we analyzed have been from texts about Franz Liszt which constitute an uncertainty on some of the results.

Another uncertainty about using text from the 1840s is that they are written in gothic written language which the OCR scanner has some difficulties with and also for RStudio to register the words afterwards. A lot of the words in our dataset have been meaningless which is visible in our word clouds and octograms.

For example, the quite important word, Franz Liszt, is split in the octogram underneath:

octogram

<chr>

endnu i fordobler grad maa velsigne og beundre

sag frants li sir maa man derfor beundre

A way to avoid these mistakes from the OCR scanner and also the registration of the words in R would be to transcribe all the text from gothic to Danish but that was too time-consuming for our time frame.

We made the word clouds with the expectation that they immediately would show us the most frequently used words in each city. However, this was not exactly the result we got. As mentioned above the OCR scan caused us problems with the words deemed important by R. Furthermore, we used the `bind_tf_idf` with the expectation that it would only punish filler words but this function might also have punished words like “Franz Liszt”, “koncert” (concert) or “Bifald” (applause), as words like these likely could have appeared in most articles. If we were to expand this examination we could look at how the word clouds would look without calculating the words’ frequency. Despite the problems we encountered, the frequency function still made it possible for us to get an idea of what kinds of words were

²⁶ Pedersen, Max Odsbjerg, *Åbne data - Ringe OCR-kvalitet*.

most important in the different cities. The addition of the stopwords list helped make this a bit clearer.

We were positively surprised by how much we ended up achieving by making the octograms. This part of our digital project ended up being important for our analysis as it broadened the way we could look at and use the important words from the word clouds. The only thing we encountered problems with, regarding the octograms, was that the knitted version of our Rmarkdown would show neither the links to Mediestream nor which newspaper the octogram appeared in. Because of this, there are some differences between the knitted version and the RMarkdown as we chose that the tibble showing the octograms looked rather confusing containing all the information that the RMarkdown version did while not even being useful. Therefore we used the function: `head(publikum$octogram)`, which instead only showed the contents of the octograms.

In our sentiment analysis, we used the tool Sentida which looks at the words of a given text and calculates a score that indicates the sentiment of the text on a scale of -5 to 5.

The individual words' sentiment in Sentidas' word or lemma lexicon is rated by three authors by the assumption that the words have a sentiment.²⁷

First, we used the Sentida tool on all the words of all articles and afterwards on words from Copenhagen and then the rest of the cities collectively.

The first Sentida analysis of all words provided a score of 0,1861 which indicates a very weak positive emotion. But as the Sentida tool has a slight bias towards positive words with 56 percent of the words in their lexicon rated positive it is difficult to conclude that the texts are weakly positive when it's so close to neutral emotion.²⁸

We also made a comparison of the sentiment of words from Copenhagen and the rest of the world. We predicted a more positive outcome in Copenhagen but our Sentida scores gave us another picture. Copenhagen with a score of 0,1430 and the rest of the cities with 0,2408 which means that the words from Copenhagen actually have a more negative emotion than the rest of the cities collectively.

It is hard for us to tell how much we can say about the text's emotion from these scores as the uncertainty from the OCR also plays a role since a lot of the words have not been recognized

²⁷ Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher. s. 39

²⁸ Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher. s. 42

correctly. However, we believe that the Sentida tool could be very useful for the right texts and has been interesting to work with and therefore still has its right in this project.

We used R and RStudio to make our digital project. These tools worked well for our examination. Our method turned out to be some sort of patchwork of coding from the scripts by Max Odsbjerg, the book *Text Mining with R: A Tidy Approach* (2022) and the Sentida analysis tools made by Gustav Aarup Lauridsen, Jacob Aarup Dalsgaard and Lars Kjartan Bacher Svendsen. This made sense for how we wanted our digital project to be built and what we wanted to achieve with it.

Conclusions

By making word clouds, octograms and using the Sentida analysis we examined 158 articles from the year 1841 in which the Austro-Hungarian composer and pianist Franz Liszt was mentioned. The word clouds showed us the most important words mentioned in newspapers in different cities. Even though the OCR scan and the disadvantages of using the `bind_tf_idf` function in this instance caused us some trouble, we got an overview of the important words. Words that could be of interest were examined more closely by making octograms, in which we could do a close reading of the context where the important words appeared. This made it visible that Jyllands Posten's article about Franz Liszt from the 6th of August 1841 had a very positive view of Franz Liszt. The article from 30th of July 1841 from Aarhus Stifts-Tidende showed us a description of how Liszt engaged the audience (Publikum) at his concert, and it seemed that this made an impression on the reviewer. Perhaps this can be seen as an example of how these new liberal musical ideas were received and described by the Danish media. We ended our digital project by doing a sentiment analysis using the tool Sentida which allows us to analyze Danish text. The results were quite vague probably also because of the uncertainty from the OCR scanner but the tool remains useful and recomendable.

Acknowledgements

We would especially thank Max Odsbjerg Pedersen for his extremely useful advice, help and scripts that made this paper possible. Furthermore, we would like to thank Gustav Aarup Lauridsen, Jacob Aarup Dalsgaard and Lars Kjartan Bacher Svendsen for making a Danish Sentiment analysis and a paper about this and making this public and available for everyone to use.

References

Den Til Forsendelse Med Brevposterne Kongelig Allernaadigst (Alene) Privilegerede Aarhus Stifts-Tidende (1821-1844). Mediestream, July 30, 1841. <http://hdl.handle.net/109.3.1/uuid:df189803-2a74-44f3-a89b-90f5ffd1e4ea>.

Fran* Lis*t AND iso_date:[1841-01-01 TO 1842-01-01], in Mediestream. (last accessed October 27, 2023). https://www2.statsbiblioteket.dk/mediestream/avis/search/Fran*%20Lis*t%20AND%20iso_date%3A%5B1841-01-01%20TO%201842-01-01%5D

Jim. “How to Use ‘Not in’ Operator in Filter.” *R-Bloggers* (blog), July 17, 2022. <https://www.r-bloggers.com/2022/06/how-to-use-not-in-operator-in-filter/>.

Jyllandsposten (1838-1841). Mediestream, August 6, 1841. (Accessed October 27, 2023). <http://hdl.handle.net/109.3.1/uuid:2d9e4809-f876-4d83-a06b-9f17e4a005a5>.

Krogsgaard, Ole. “Offentlige Debatfora 1830-1849.” Aarhus Universitet. (Accessed October 27, 2023). <https://danmarkshistorien.dk/vis/materiale/offentlige-debatfora-1830-1849>.

Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher (2019). SENTIDA: A New Tool for Sentiment Analysis in Danish. *Journal of Language Works - Sprogvidenskabeligt Studentertidsskrift*, 4(1), 38–53. Retrieved from <https://tidsskrift.dk/lwo/article/view/115711>

Lauridsen, Gustav Aarup, Dalsgaard, Jacob Aarup, & Svendsen, Lars Kjartan Bacher (Accessed October 27, 2023). Sentida [Danish sentiment analysis tool]. <https://github.com/Guscode/Sentida.git>

Nissen, Peter E. “Da Franz Liszt kom til byen : musik og revolution i København år 1841.” *Fortid og nutid* (2009).

Pedersen, Max Odsbjerg. (Accessed October 27, 2023). *Åbne data - Ringe OCR-kvalitet*. [script for OCR scanned articles from Mediestream].

https://library.au.dk/fileadmin/www.bibliotek.au.dk/LibLab/Eksempel/old_newspapers_1849_korr.html?fbclid=IwAR0yIfPs2O5mLcM5krmIgdMqQ59dOxoZVeyypLgjnIK1rilcOpNiQB9xkB8#Tidy_Text_formatet

Pedersen, Max Odsbjerg. (Accessed October 27, 2023). *Snydeark til Mediestream* [cheatcodes for Mediestream, regular expressions]. <https://gist.github.com/maxodsbjerg/e2dd484d3c9dcaa9c422a861d6a93f6e>

Pedersen, Max Odsbjerg & Mulvad, Toke Jøns. (Accessed October 27, 2023). *Text Mining St. Croix Avis* [introduction to text mining, R]. <https://github.com/maxodsbjerg/TextMiningStCroixAvis.git>

Silge, Julia & Robinson, David. (last built on 2022-11-02). *Text Mining with R: A Tidy Approach*. <https://www.tidytextmining.com/>

Swagger UI. (Accessed October 27, 2023). <https://labs.statsbiblioteket.dk/labsapi/api//api-docs?url=%2Flabsapi%2Fapi%2Fopenapi.yaml>

Required Metadata

Table 1 – Software metadata

| Nr | Software metadata description | Please fill in this column |
|----|---|---|
| S1 | Current software version | R (4.3.1) and RStudio (2023.06.2+561) |
| S2 | Link to project | Kristian:
https://github.com/Digital-Methods-HASS/au728045_Corvalan_Kristian/tree/5f1e8b258d84244b870e1346b037cd6a2512a0cd/Final_Project
Clara:
https://github.com/Digital-Methods-HASS/AU721527_Sydow_Clara/tree/708f76e19ae2a62b2d9f909a8d7b3db69a89269a/Final_Project |
| S3 | Computing platform/ Operating System | Macbook Air (2020), 8 Gb RAM, macOS Ventura 13.4. |
| S4 | Installation requirements & dependencies for software not used in class | For RStudio:
SENTIDA |

| | | |
|----|--|---|
| S5 | If available Link to software documentation for special software | https://github.com/Guscode/Sentida/blob/master/README.md |
| S6 | Support email for questions | Kristian: 202208514@post.au.dk
Clara: 202204626@post.au.dk |
| S7 | Link to Homework | Kristian:
https://github.com/Digital-Methods-HASS/au728045_Corvalan_Kristian/tree/abc3260409aed4ae9829ad39c055261d2209b7dd/Homework

Clara:
https://github.com/Digital-Methods-HASS/AU721527_Sydo_w_Clara/tree/b16d17becfca1692e9fa70824fb7f4a5ca4c4682/Homework |