From *Zipf distribution* to *Universal Dependencies* – Interactive Notebooks for Swedish Text Analysis

Dimitrios Kokkinakis[1]

[1] *University of Gothenburg, Box 200, 405 30, Gothenburg, Sweden*

Abstract

Notebook-based environments are powerful (web-based) interactive development resources for conducting exploratory (textual) data analysis (EDA). These environments allow the embedding of code (code snippets in 'code cells') which can be easily executed with the results immediately presented into the user's window. This paper introduces some basic exploratory tools and techniques using *JupyterLab* notebooks, applied to Swedish using a subcorpus that address various topics related to the COVID-19 pandemic published during January-December 2021.

Keywords

[interactive] notebooks, Swedish, R, JupyterLab, text analysis, enhanced and active learning

1. Introduction

Notebook-based environments, such as *JupyterLab*[2] *(Jupyter)*, *Google Codelab*[3] (*Colab*) or the *Kaggle*[4] *Notebooks* are powerful (web-based) interactive development resources for conducting exploratory (textual) data analysis (EDA). The purpose of EDA is to find valuable insights in the data. Notebooks facilitate in-depth EDA by allowing collaborative research while promoting transparency and reproducibility in scholarly work by easily creating and sharing computational documents such as code, and data. These environments allow the embedding of code snippets ('code cells') which can be easily executed with the results immediately presented into the user's window. Formatted text or 'markdown cells' are used to supplement and explain the code. Moreover, code cells can be independently executed in an arbitrary order, edited between runs and iterations, share variables and functions and allow the experimentation with different methods, models, and tools. In addition, code cell outputs, which may include charts, maps, tables, and plots, are integrated within the notebook document, or saved locally as high-quality digital format (e.g., *.jpeg* or *.png* images).

Notebooks can and have been deployed in a variety of scientific contexts, ranging from educational, economic, engineering, data science and digital humanities [1-5].

2. Textual Data and Associated Resources

This paper introduces some basic exploratory tools and techniques using *JupyterLab* notebooks (v. 3.5.3), applied to Swedish. Jupyter is often used with the Python programming language or R scripting language, but other languages are also available. Here, the R language (v. 4.3.0) is used, and as the textual corpora in all the experiments we use a dataset of roughly 1,600 documents published on-line during 2021. This dataset is part of the *sv-COVID-19 corpus*, which contains published articles in Swedish assembled from the internet that address various topics related to the COVID-19 pandemic. The corpus is further divided into 8 stylistic genres depending on their original publication forum (AuTHoRiTieS; BLOG; MeDiCaL; NEWS; PuBLicMeDia; PeRiodiCaL; ReSeaRCH and SoCiaLMeDia) and can be searched and queried in SpråkBankenText's word research platform Korp[5].

Huminfra Conference 2024, Gothenburg, 10-11 January 2024.

dimitrios.kokkinakis@svenska.gu.se (D. Kokkinakis)

[2] https://jupyter.org/.
[3] https://codelabs.developers.google.com/.
[4] https://www.kaggle.com/docs/notebooks.
[5] https://spraakbanken.gu.se/korp/#?corpus=sv-covid-19.

3 Components and Tools

We present here an outline of various components that are implemented in the notebooks. These components not only serve as a comprehensive guide to the implemented methodologies but also offer a dynamic showcase of the diverse analyses conducted using R-scripts in JupyterLab. This transparency ensures the reproducibility of the research, allowing others to access and verify the data sources used.All resources, i.e., textual (raw text or URL's to the original textual document subcorpus), lexical or programmatic are available in GitHub[6]. The outline in this section presents a snapshot of some of the output of various R-scripts in the JupyterLab not ebooks, i.e., from basic *frequency analysis* to more advanced techniques such as *topic modeling* and the application of *universal dependencies'* Swedish models. These sophisticated analyses demonstrate the notebooks' capacity to accommodate intricate research methodologies, providing a valuable resource for scholars seeking to explore not only the breadth but also the depth of analytical possibilities within the JupyterLab environment.

3.1 A *smörgåsbord* of scripting results from Jupyter

Within this section, a number of diverse components implemented in the notebooks unfold, presenting users with a curated selection of outputs generated by various R-scripts in the environment. From basic frequency analysis to more sophisticated techniques such as topic modeling and the application of universal dependencies' Swedish models, the scripting results encapsulate a spectrum of analytical depths. This *smörgåsbord* of scripting outcomes not only showcases the flexibility of JupyterLab but also serves as an interactive guide for researchers navigating through the intricacies of data exploration.

3.1.1. Word distributions and frequencies

The first three figures below, show different ways to depict word distributions and frequencies in the examined data. The first plot of the left image 'Rank frequency', shows the Zipfian distribution of the word frequencies in the data; where few words occur very often, and many words occur very rare. The 'green' range, second plot of the left image 'log-Rank-frequency', marks the meaningful terms in the data. Here, stopwords and low frequent words (≤10) are removed. The middle image shows the frequencies of 8 selected words over a monthly period; while the most frequent keywords in two of the genres are shown at the far right image.

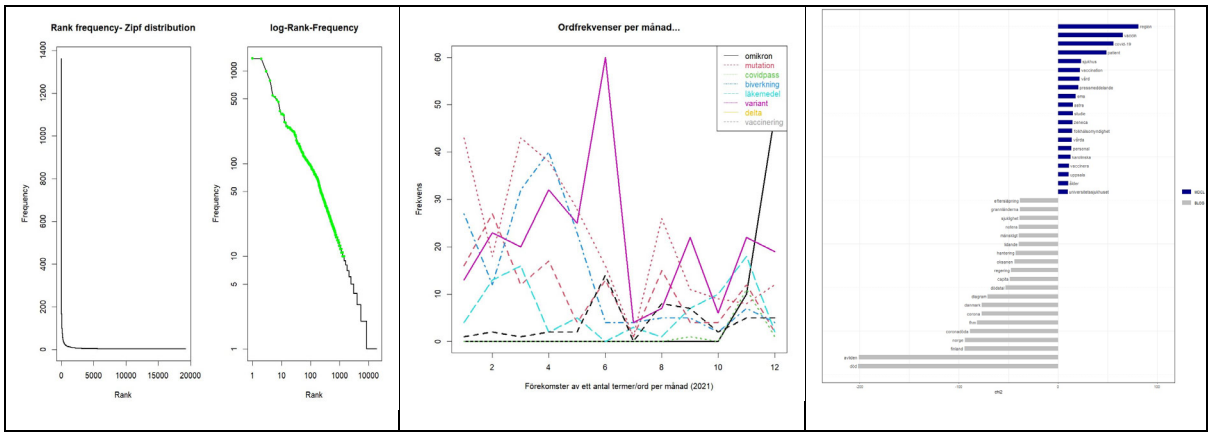

Figure 1: Word distributions and frequency analysis.

3.1.2. Significant word associations and heatmaps

The images below show different views for depicting significant words associated with a single term, here we chose the word 'omikron'. The left image shows terms strongly collocating with 'omikron'. While the middle image shows a network graph with words associated with 'omikron' in the dataset.

[6] The URL links of the data can be found here: https://github.com/Research-at-SBXtext/sv-JupyterLab-examples/blob/main/textual-resources/url-links-swedish-dataset.txt.

The image to the right shows a heatmap which depicts values for a variable (here pronouns) across the genres in the dataset. Each cell is colored in a way that darker colorings imply a more frequent occurrences of a specific pronoun per genre; e.g., the personal pronoun *jag* ('I') has much higher frequency (marked in dark red) in texts that belong to the genre marked as "SCLMD", i.e., texts from social media.

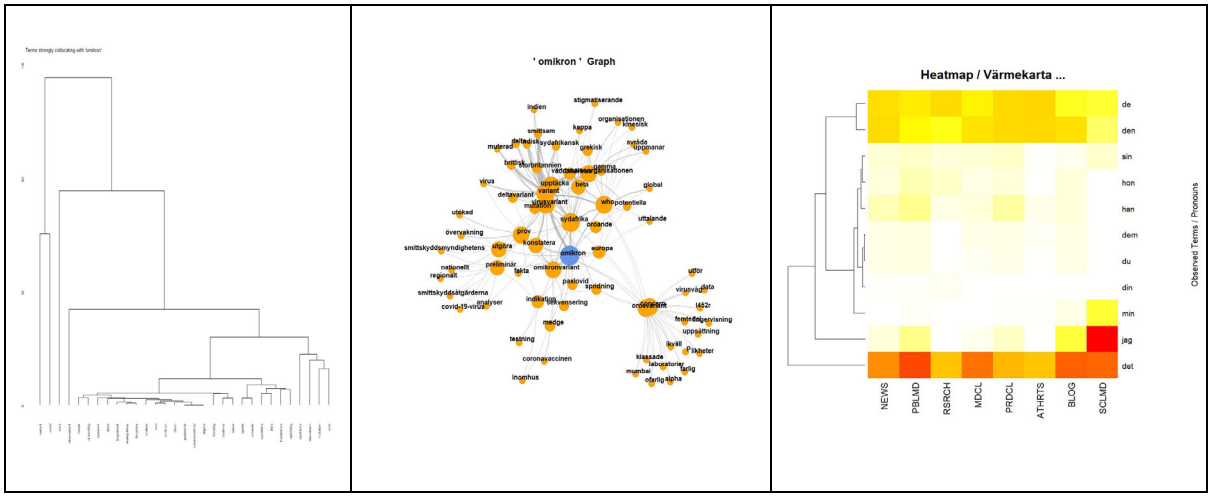

Figure 2: Significant words and words over genre distribution.

3.1.3. Sentiment analysis and universal dependencies

The left image below shows the aggregated results for sentiment analysis per corpus genre. Here we use a lexical-based approach to sentiment analysis by incorporating a large list of words with a pre-assigned sentiment value. The first bar of the plot shows that texts of category "BLOG" are much more negative than any other text genre. The image to the right shows the distribution of the Universal part-of-speech tags in the dataset (here 'NOUNS' are the most frequent type of part-of-speech). The counts originate from the application of the universal dependencies model 'swedish-talbanken-ud-2.5-191206.udpipe'.

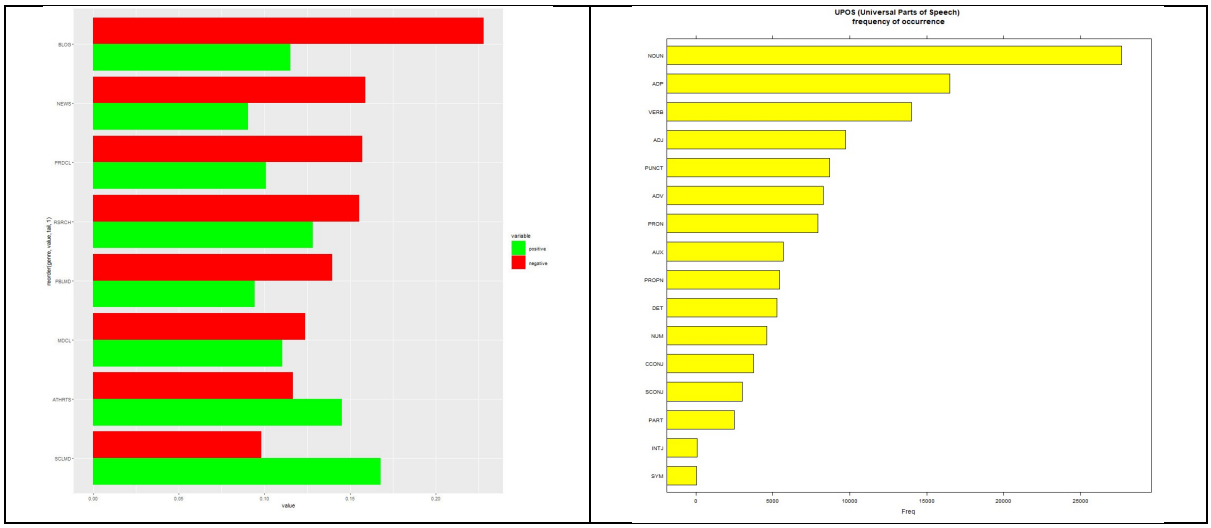

Figure 3: Sentiments and part-of-speech distribution.

3.1.4. (Flavours of) Topic modelling

Topic modeling can be used to automatically cluster and organize large document collections based on their content. There exist various 'flavours' of topic modelling techniques. The image to the left, uses a vanilla *Latent Dirichlet Allocation* (LDA), and the 40 most frequent words in one of the generated topics are shown as a word cloud. The image to the right shows the topic distribution per month (with the number of topics set to 9).

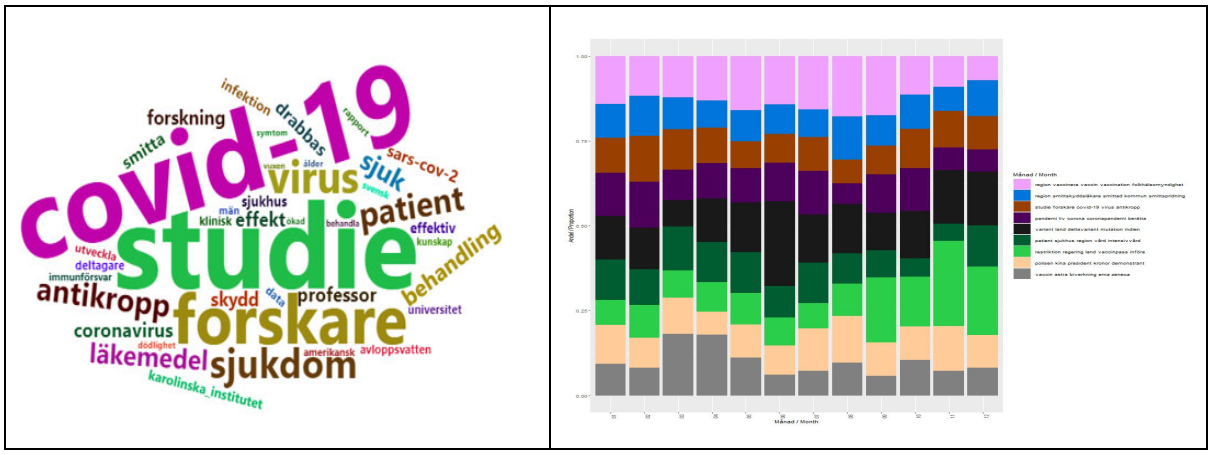

Figure 4: Topic modelling using *vanilla* LDA.

The two images below show some visualizations of *Structural Topic Modeling*, an approach which allows to incorporate document-metadata into the model; for instance, you can calculate the extent to which topics are more or less prevalent over *time* by incorporating the publication date of each document in the dataset. The left image shows the model diagnostics, *exclusivity*, *heldoutout likelihood* and *semantic coherence* (as before, 9 topics have been chosen); the right image, shows the words with the highest probabilities for each topic in the data.

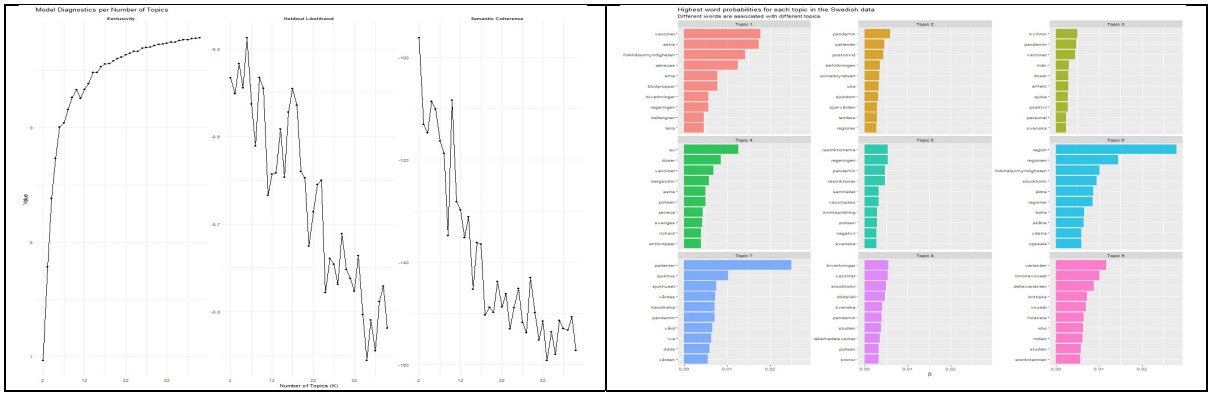

Figure 5: Structural topic modelling (diagnostics and word distributions per topic).

4 Conclusions

Interactive notebooks can be a powerful tool for researchers in e.g., digital humanities, both as a pedagogical, analytical, and scholarly tool, offering a flexible and efficient environment with rich textual documentation alongside the code, ease of collaboration and code interactivity and data visualization capabilities to help convey findings and insights. With the ability to execute code independently, researchers can engage in iterative experimentation, adjusting parameters and visualizing results instantly. This interactivity proves especially beneficial in digital humanities, where the exploration of datasets and analytical methods demands a responsive and iterative approach. Beyond the research phase, the notebooks serve as a versatile tool for report preparation, allowing scholars to generate polished documents in various formats such as PDF or HTML directly from their notebooks.

In essence, interactive notebooks emerge as a flexible, collaborative, and efficient solution, providing researchers in digital humanities with a robust platform for analysis, documentation, and communication of their scholarly endeavors. Moreover, the interactive nature of notebooks fosters a dynamic learning environment in educational settings, enabling instructors and students to actively engage with course material. This real-time interaction facilitates a deeper understanding of complex concepts and encourages hands-on exploration, making it an invaluable resource for not only teaching digital humanities but also fostering creativity, and preparing students for the rapidly evolving landscape of technology and information in the digital age in all humanistic disciplines [6].

References

[1] David L. Alderson. 2021. Interactive Computing for Accelerated Learning in Computation and Data Science. INFORMS Transactions on EducationVol. 22:2. https://doi.org/10.1287/ited.2021.0261
[2] Lorena A. Barba, Lecia J. Barker, Douglas S. Blank, et al. 2019. Teaching and Learning with Jupyter. URL: https://jupyter4edu.github.io/jupyter-edu-book/index.html and https://jupyter4edu.github.io/jupyter-edu-book/
[3] Brian E. Granger, and Fernando Pérez. 2021. Jupyter: Thinking and Storytelling With Code and Data. Computing in Science Eng. 23(2), 7–14. https://doi.org/10.1109/MCSE.2021.3059263
[4] Cécile Hardebolle. 2023. Online interactive textbooks with Jupyter Notebooks. URL: https://www.epfl.ch/education/educational-initiatives/jupyter-notebooks-for-education/teaching-and-learning-with-jupyter-notebooks/online-interactive-textbooks-with-jupyter-notebooks/
[5] Quinn Dombrowski, Tassie Gniady, and David Kloster. 2019. Introduction to Jupyter Notebooks. Programming Historian 8. https://doi.org/10.46430/phen0087
[6] Jon Chun and Katherine Elkins. 2023. The Crisis of Artificial Intelligence: A New Digital Humanities Curriculum for Human-Centred AI. Journal of Humanities and Arts Computing, Volume 17:2, Page 147-167. https://doi.org/10.3366/ijhac.2023.0310

R Packages

quanteda: *Quantitative Analysis of Textual Data*, https://quanteda.io/ (v. 3.3.1)
quanteda.textstats: *Textual Statistics for the Quantitative Analysis of Textual Data*, https://cran.r-project.org/web/packages/quanteda.textstats/index.html (v. 0.96.4)
quanteda.textplots: *Plots for the Quantitative Analysis of Textual Data*, https://cran.r-project.org/web/packages/quanteda.textplots/index.html (v. 0.94.3)
udpipe: *Universal Dependencies pipeline*, https://lindat.mff.cuni.cz/services/udpipe/ (v. 0.8.11)
topicmodels: *Topic Models*, https://cran.r-project.org/web/packages/topicmodels/index.html (v. 0.2.14)
stm: *Estimation of the Structural Topic Model*, https://cran.r-project.org/web/packages/stm/index.html (v. 1.3.6.1)

Matrix: *Sparse & Dense Matrix Classes*, https://cran.r-project.org/web/packages/Matrix/index.html (v. 1.5.4)
FactoMineR: *Multivariate Exploratory Data Analysis and Data Mining*, https://cran.r-project.org/web/packages/FactoMineR/index.html (v. 2.9)
factoextra: *Extract and Visualize the Results of Multivariate Data Analyses*, https://cran.r-project.org/web/packages/factoextra/index.html, (v. 1.0.7)

dplyr: *A Grammar of Data Manipulation*, https://cran.r-project.org/web/packages/dplyr/index.html (v. 1.1.3)

ggplot2: *Data Visualisations Using the Grammar of Graphics*, https://cran.r-project.org/web/packages/ggplot2/index.html (v. 3.4.4)
ggdendro: *Create Dendrogr. & Trees*, https://cran.r-project.org/web/packages/ggdendro/index.html (v. 0.1.23)
ggraph: *Graphics for Graphs & Networks*, https://cran.r-project.org/web/packages/ggraph/index.html (v. 2.1.0)
igraph; *Network Analysis*, https://cran.r-project.org/web/packages/igraph/index.html (v. 1.5.1)

wordcloud2: *Create Word Clouds***,** https://cran.r-project.org/web/packages/wordcloud2/index.html (v. 0.2.1)
reshape2: *Reshape Data*, https://cran.r-project.org/web/packages/reshape2/index.html (v. 1.4.4)
pals: *Color palettes and colormaps*, https://cran.r-project.org/web/packages/pals/index.html (v. 1.7)