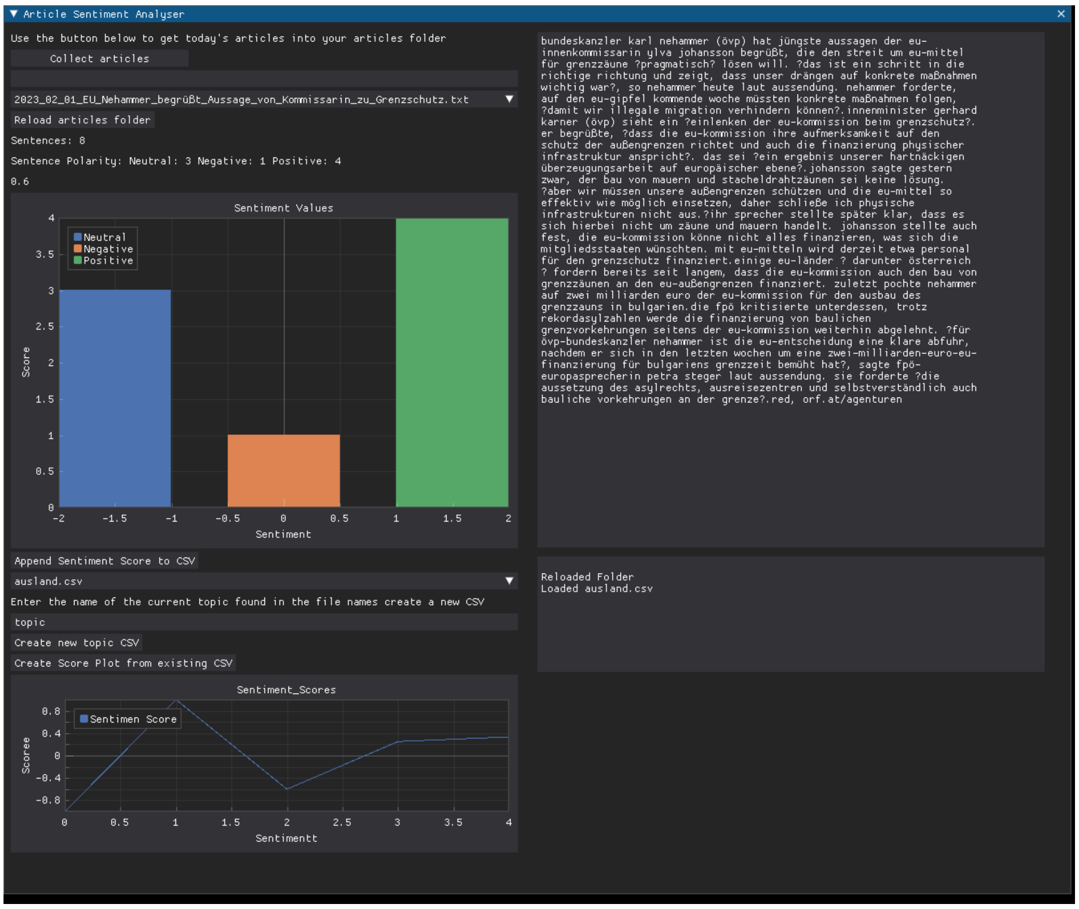
Lukas Waldhofer – Sentiment Analysis Tool



```
Use the button below to get today's articles into your articles folder
2023_02_01_EU_Nehammer_begrüßt_Aussage_von_Kommissarin_zu_Grenzschutz.txt
Sentences: 8
Sentence Polarity: Neutral: 3 Negative: 1 Positive: 4
0.6
ausland.csv
Enter the name of the current topic found in the file names create a new CSV
topic
```

```
bundeskanzler karl nehammer (övp) hat jüngste aussagen der eu-
innenkommissarin ylva johansson begrüßt, die den streit um eu-mittel
für grenzzäune ?pragmatisch? lösen will. ?das ist ein schritt in die
richtige richtung und zeigt, dass unser drängen auf konkrete maßnahmen
wichtig war?, so nehammer heute laut aussendung. nehammer forderte,
auf den eu-gipfel kommende woche müssten konkrete maßnahmen folgen,
?damit wir illegale migration verhindern können?.innenminister gerhard
karner (övp) sieht ein ?einlenken der eu-kommission beim grenzschutz?.
er begrüßte, ?dass die eu-kommission ihre aufmerksamkeit auf den
schutz der außengrenzen richtet und auch die finanzierung physischer
infrastruktur anspricht?. das sei ?ein ergebnis unserer hartnäckigen
überzeugungsarbeit auf europäischer ebene?.johansson sagte gestern
zwar, der bau von mauern und stacheldrahtzäunen sei keine lösung.
?aber wir müssen unsere außengrenzen schützen und die eu-mittel so
effektiv wie möglich einsetzen, daher schließe ich physische
infrastrukturen nicht aus.?ihr sprecher stellte später klar, dass es
sich hierbei nicht um zäune und mauern handelt. johansson stellte auch
fest, die eu-kommission könne nicht alles finanzieren, was sich die
mitgliedsstaaten wünschten. mit eu-mitteln wird derzeit etwa personal
für den grenzschutz finanziert.einige eu-länder ? darunter österreich
? fordern bereits seit langem, dass die eu-kommission auch den bau von
grenzzäunen an den eu-außengrenzen finanziert. zuletzt pochte nehammer
auf zwei milliarden euro der eu-kommission für den ausbau des
grenzzauns in bulgarien.die fpö kritisierte unterdessen, trotz
rekordasylzahlen werde die finanzierung von baulichen
grenzvorkehrungen seitens der eu-kommission weiterhin abgelehnt. ?für
övp-bundeskanzler nehammer ist die eu-entscheidung eine klare abfuhr,
nachdem er sich in den letzten wochen um eine zwei-milliarden-euro-eu-
finanzierung für bulgariens grenzzeit bemüht hat?, sagte fpö-
europasprecherin petra steger laut aussendung. sie forderte ?die
aussetzung des asylrechts, ausreisezentren und selbstverständlich auch
bauliche vorkehrungen an der grenze?.red, orf.at/agenturen
```

```
Reloaded Folder
Loaded ausland.csv
```

Manual:

Collect articles button: (callback = initialiseWebscraper) – Button that triggers the initial web scraping function and saves all articles of the day as .txt. files in the "articles" folder.

Progress bar (tag = "download_progress"): This is a simple bar to show the progress of downloading. A bug that might occur here is that the bar either ends on 90% or 110% completion, this is due to invalid links that may be passed through. But most of the time it stops perfectly on 100%

Article Combobox (tag = "cbo_article", callback = analyse): This is a list box containing all the files that lie in the "articles" folder. It has a callback assigned to the analyse() function which will be triggered on selecting an item from the list.

Reload article Folder Button (callback = reloadFolder): Button to reload the entries in the "article" folder.

Sentence Count – Sentence Polarities – Overall Polarity: These are plain texts to display the stats of the selected article.

Sentiment Barplot: This is a simple 3 column Barplot which displays the positive, negative, and neutral sentences of the article. It will be live-updated as soon as you change the article in the Combobox.

Append Sentiment Score to CSV Button: (callback = csv_export) This button appends a pandas data frame consisting of the article name and overall sentiment score to the selected CSV file.

CSV Combobox: This is a list box containing all the .csv files that lie in the "csv" folder. The here selected .csv file will be used to append and load data.

Topic input text + Create new topic button (callback = create_csv): Here you can define the name for a new CSV file which will be created once you push the "Create new topic CSV" button.

Create score Plot from existing CSV Button: (callback = load_csv) This button will update the line plot below to read from the currently selected .csv file.

Development and background:

For my final project I developed a Sentiment Analysis Software to analyse news articles from the Austrian news website orf.at. My main question was: Will I be able to create a software that can be used as a proper sentiment analysis tool?

Due to me doing a sentiment analysis in another course I had to change my previous question which was about actually analysing the articles to a more, development orientated goal.

Modules:

- I used the module *Dearpygui* as my UI and plotting framework.
- For the NLP and Sentiment Analysis part I used *nltk* (for the stopwords) and *textblob_de* (which is the German version of *textblob*)
- For file writing and reading from designated folder I've used a combination of the *os* and *sys* and *datetime* (for better filenames) modules.
- Text filtering and formatting (for the text output field) I've used *Regex* and *textwrap.*
- Web scraping was done using the *requests* and *BeatufiulSoup* modules.
- Finally, the CSV import and export was managed through a combination of *pandas* data frames and the *os* and *sys* modules I've used before.

Development process:

Before starting to work on the project I've set up a GitHub repository, so I'll be able to work on the project from my desktop PC and my laptop at the same time. At first, I've set up the UI framework with Dearpygui. Then I added two functions for web scraping, one to run a scrape through the overall orf.at website to get all links that are displayed on the main site. This function passes the "page" objects (after a validation check) which are retrieved through a HTTP request to the second scraping function which then scrapes the html contents of the single articles. The scraped articles are then sent to the sentiment analysis function which splits the overall text into sentences and calculates their sentiment value and an overall sentiment score. The values are then added as input parameters for the Barplot to visually display them. The Dearpygui (dpg) object are altered with either dpg.setvalue("item_tag", value) or dpf.configure_item("item_tag", parameter=value). The UI elements are also linked with callbacks to the functions of the script. These callbacks are triggers for declared functions as soon as a UI Element does something that is assigned with it (e.g. a button press, index change in the list box etc.)

At the bottom I've added a CSV import and export section. Using pandas I've made three separate functions for appending, creating, and loading a .csv file. Here I've created a two-column data frame to display the article title and the overall sentiment score. Here it was quite important to only use header=True on creating the .csv file, because on appending it would always append the headers to a new row too. The finished .csv can the be loaded to be displayed in a line plot on the bottom left.

I've created two text fields on the right site, the article text, and a debug console. Every important action is additionally logged into the debug console to check if it was properly executed.

Possible anomalies:

I've tested the program on several days and it should be working just fine. What I've encountered is that the website I'm scraping from sometimes uses weird html formatting on some articles which may result in them being invalid and thus being skipped. The scraper also draws empty links sometimes which may be deleted from the article folder by searching for files with 0 kb file site. The only 'real' bugs I've encountered come from drawing data from the website. The rest of the program should work just fine every time.

Conclusion:

I've managed to put together a quite interesting application with a lot of potential. To be used as a professional tool I think I would have to integrate at least one or two more sentiment analysis algorithms to give the user a selection to try out and compare. Thanks to the intuitive way to work with Deapygui I actually had a great time designing and developing this tool. This project also showed me that python really doesn't lack anything when it come to UI design in comparison with something like a Visual C# Windows Forms Application (which I'm normally used to work with UI wise).

For further information about the program, you can contact me anytime at: lukas.waldhofer@edu.uni-graz.at