Project 2: AI for Global Health using Natural Language Processing (70 pts)

Sentiment analysis, also known as opinion mining, is a subfield of natural language processing (NLP) that involves using machine learning techniques to automatically identify and extract subjective information from text, such as opinions, emotions, attitudes, and sentiments.

Project 2 is divided into four parts in which you'll investigate different Natural Language Processing techniques to conduct a sentiment analysis of tweets. While the first milestone will focus on the development of a tweet pre-processing pipeline, milestone 2 will require the exploration and evaluation of a series of pre-trained word embedding and transformer approaches using a subset of the *TweetsCOV19* dataset. Finally, your ultimate objective will be to gain insights on a research question of your choosing with one of the NLP methods explored.

Each milestone requires the completion of several tasks detailed below. The completion of each task will grant you a defined number of points. Additional bonus points will be awarded to the teams which have demonstrated creativity in the design of their research question/analysis/visualizations or have performed the most thorough analysis of their research questions given the dataset and methods at hand.

Please provide **one document** with all answers, formatted in markdown (recommended, e.g. hackmd.io), latex or word. For each method mentioned below, follow the instructions and provide 1) a brief explanation of how the method works, 2) when explicitly requested, a code snippet showing the essential function of the algorithm, and 3) performance results as well as answers to specific questions. The *template markdown* file provides an example of how your report should be formatted. There are no restrictions regarding the usage of ChatGPT for coding or writing. However, for questions as well as code snippets, which have benefitted from ChatGPT, you should explicitly indicate that ChatGPT was used.

Dataset - TweetsCOV19 is a semantically annotated corpus of Tweets about the COVID-19 pandemic. A subset of the TweetsCOV19 dataset is made available on Moodle. **Please use this set of tweets throughout the project.** This dataset reflects the societal discourse about COVID-19 on Twitter in the period of October 2019 until May 2020. The dataset has been annotated for the purpose of sentiment analysis. Each tweet has a score for positive (1 to 5) and negative (-1 to -5) sentiment as well as author location and timestamp among others. To extract the dataset, a seed list of 268 COVID-19-related keywords was defined. Tweets in TweetsCOV19 contain at least one keyword from the set of seed terms, are written in

English and published throughout the aforementioned time period. Data cleaning and enrichment as described in TweetsKB has been applied.

Part 1: Data exploration and pre-processing (5 pts)

Q1: Preprocessing (2 pt)

After familiarizing yourself with the TweetsCOV19 dataset, explain your data pre-processing steps for the following parts of the project. Provide code snippets for each pre-processing step. Comment on possible challenges posed by using the raw Twitter data, such as irregular capitalization, variable declination of words, spelling mistakes, punctuation, urls, mentions, emojis, abbreviations and others. Describe whether these apply to this corpus, and how you handled them if so.

Hint: to ease the development of your pre-processing pipeline, you can use the NLTK library.

Q2: Exploratory data analysis (1 pts)

Perform a short data analysis of the TweetsCOV19 dataset. In particular, what are the most common uni- and bi-grams in the corpus before and after preprocessing? How does this word distribution vary as a function of the sentiment labels? What is the proportion of each sentiment? Think about how best to visualize this and summarize your findings.

Q3: Metric choice (1 pt)

Are the classes balanced? Choose and justify your evaluation metric(s) for the analysis of sentiment performed in the following parts of the project. Provide a code snippet used to produce performance measurements using the chosen metric.

Q4: Dataset splitting (1 pt)

Split your dataset into a training, validation and test set. Motivate your approach for this, comment on possible evaluation challenges (e.g. label shift over time).

Remark: All subsequent performance scores should be evaluated on your test set.

Part 2: NLP learning based methods (45 pts)

VADER (5 pts) Familiarize yourself with the VaDER sentiment analysis method (e.g. https://github.com/cjhutto/vaderSentiment). [1]

Q1: Briefly explaining how this method works (1 pt).

Q2: Provide a code snippet detailing how to use it for our task (2 pts). In light of what you have learned about this method, reflect on pre-processing steps that might be unnecessary when using VADER .

[1] Hutto, Clayton, and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text." *Proceedings of the international AAAI conference on web and social media*. Vol. 8. No. 1. 2014.

Q3: Apply this method to our TweetsCOV19 dataset and comment on the performance obtained (2 pts).

Word Embeddings (20 pts)

Q1: Bag of Words (BoW) (2 pts): Implement a Bag of Words embedding approach to create embeddings of the TweetsCOV19 dataset. Explain the methodology and provide a code snippet of the function used to produce these embeddings.

Q2: TF-IDF (2 pts): Implement a TF-IDF embedding approach to create embeddings of the TweetsCOV19 dataset. Explain the methodology and provide a code snippet of the function used to produce these embeddings.

Q3: Word2Vec with CBOW or Skip-gram (2 pts): Implement a Word2Vec pre-trained embedding approach to create embeddings of the TweetsCOV19 dataset. Motivate your choice between CBOW or Skip-gram. Explain the methodology and provide a code snippet of the function used to produce these embeddings.[2]

Q4: GloVe (2 pts): Implement a GloVe pre-trained embedding approach to create embeddings of the TweetsCOV19 dataset. Explain the methodology and provide a code snippet of the function used to produce these embeddings.[3]

Q5: FastText (2 pts): Implement a FastText pre-trained embedding approach to create embedding of the TweetsCOV19 dataset. Explain the methodology and provide a code snippet of the function used to produce these embeddings.[4]

Q6: Visualization of embeddings (2 pts): Perform a qualitative comparison of the quality of the semantic content of each embedding approach. For example, visualize the location using dimensionality reduction techniques (e.g., PCA, t-SNE, UMAP[5][6]), of negative/positive samples, or tweets containing specific words, in the latent space.
Compare your visualization of each method and explain whether you can anticipate any differences in downstream performance as a result.

Q7: Tweet embeddings (2 pts): Propose three different approaches to combine word embeddings into an embedding for each tweet. Provide a code snippet for each function. How do all of the above methods deal with out-of-vocabulary words?

---

[2] Mikolov, Tomas, et al. "Advances in pre-training distributed word representations." *arXiv preprint arXiv:1712.09405*(2017).
[3] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014.
[4] Bojanowski, Piotr, et al. "Enriching word vectors with subword information." *Transactions of the association for computational linguistics* 5 (2017): 135-14
[5] L.J.P. van der Maaten and G.E. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2431–2456, 2008.
[6] McInnes, L, Healy, J, *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*, ArXiv e-prints 1802.03426, 2018

Q8: Classifier (3 pts): Implement three downstream classifiers to perform sentiment analysis of your tweets (i.e., predict both the positive (1 to 5) & negative (-1 to -5) sentiment score for each tweet). For each classifier, briefly explain how it works and make sure to tune hyper-parameters appropriately. Provide a results table showcasing the performance of all tested classifiers for each embeddings approach you implemented in Q1-5, and each aggregation method in Q7. For each family of classifiers tested provide a code snippet.

Q9: Performance comparison (3 pts): Using the summary table computed in Q8, compare the performance of all for methods on the sentiment analysis task using the TweetsCOV19 dataset. Also compare methods from a computational point of view. What embedding model, aggregation method and classifier would you select among all approaches? Give potential extensions that could help improve your performance.

Remark: you can use third-party libraries to help you with pre-trained word embedding implementations.

Transformers (20 pts)

Q1: Transformer-based language models[7] (4 pts). How do transformer-based large language models work? Discuss architecture choices and training details that are crucial to their performance. Comment on the difference between BERT[8], RoBERTa[9] and GPT[10] models.

Q2: Scalability (2 pts). Comment on the scalability of the embedding-based approaches compared to transformer-based language models (number of parameters). Comment on possible trade-offs in terms of computation resources and required dataset sizes.

Q3: Code (2 pts). Provide a code snippet detailing how to adapt a pre-trained language model for our task. Next, show how you can also fine-tune its last layers.
Run your code on one of the following pre-trained models:

a. BERT: https://huggingface.co/bert-base-uncased
b. RoBERTa, pre-trained on a similar task to ours: https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest

If you come across any issues with compute resources (note: use a GPU, for instance on Google colab), feel free to use other huggingface models with a smaller number of parameters. Follow instructions on huggingface for guidance and clearly state the model that you end up using.

---

[7] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).
[8] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).
[9] Liu, Yinhan, et al. "Roberta: A robustly optimized bert pretraining approach." *arXiv preprint arXiv:1907.11692* (2019).
[10] Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901.

Q4: Performance analysis (3 pts). Report and analyze the performance of this large language model after fine-tuning. Propose three different approaches you would investigate to improve performance, if compute resources were not a bottleneck.

Q5: Transfer learning details (3 pts). Study the effect on downstream performance of freezing different layers and a different number of layers during fine-tuning.

Q6: Embedding analysis (6 pts). Use the last layer of the pretrained model and visualize the representations obtained under different inputs. (2 pts) How does fine-tuning affect your results? (2 pts)
Compare your results to what you obtain with embedding methods in Part 2, Q6. Does your visual analysis support the performance differences in Part 3, Q4? (2 pts)

Part 3: Downstream Global Health Analysis (20 pts)

Q1: Research question (2 pts). Among the following papers, identify a research question they address that you could explore using the TweetsCOV19 dataset. For example, set out to analyze sentiments towards Covid-19 as a function of time or geographical location, or of any sub-topic related to the pandemic. Motivate the relevance of your question in the context of the pandemic. Feel free to use another peer-reviewed paper analyzing COVID tweets for public health insights for inspiration. Ensure to provide a correct citation.

- How epidemic psychology works on Twitter: evolution of responses to the COVID-19 pandemic in the U.S.
- COVID-19 Public Sentiment Insights and Machine Learning for Tweets Classification
- Comparing tweet sentiments in megacities using machine learning techniques: In the midst of COVID-19,
- A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets
- Evaluation of Twitter data for an emerging crisis: an application to the first wave of COVID-19 in the UK
- Public Perception of COVID-19 Vaccine by Tweet Sentiment Analysis
- “Thought I’d Share First” and Other Conspiracy Theory Tweets from the COVID-19 Infodemic: Exploratory Study

Remark: for spatial analysis, please refer yourselves to third-party libraries such as Nominatim from Geopy which uses lexicon matching. An example of how to apply this tool is provided in this Medium post.

Q2: Method choice and design (5 pts). Among the methods you have explored in Part 2, select one approach to tackle this research question using the TweetsCOV19 dataset and motivate your choice. Detail any necessary modifications you implement to achieve this analysis and performance metrics to measure the success of your approach. Provide a code snippet used to perform this analysis.

Q3: Results & Analysis (6 pts). Analyze your results and provide numerical evaluation and visualizations showcasing your findings. (3 pts) Explain what conclusions can be drawn from these, as well as key takeaways which answer (or partially answer) your research question. (3 pts)

Q4: Comparison to literature (3 pts). Discuss whether your results support or disagree with the paper you have chosen for inspiration. Provide plausible reasons for different findings or any performance discrepancies.

Q5: Discussion (3 pts). Discuss the pros and cons of your approach compared to the one used in the paper.

Q5: Summary & Conclusion (1 pt). Provide a summary of your analysis and the insights it provides about your research question.

Bonus: Topic & Emotion Analysis (+5 pts)

This last part of the project will count for bonus points. In this section, you will explore a novel NLP task which will allow you to tackle more complex research questions. **Please choose one out of the two proposed task below and complete the task at hand to obtain bonus points**:

- **Topic Modeling**: Perform a topic modeling using the Latent Dirichlet Allocation (LDA) approach. Explain the foundations of this approach and provide the code snippet used to apply this method. Finally provide a visualization of your results and detail the conclusions you can draw about the COVID-19 epidemics using topic modeling. Compare your results with this Nature article which performs a similar analysis. Hypothesize on why results might be different.

  Remark: you can use third-party libraries to help you such as the Gensim library.

- **Emotion Analysis**: The sentiment analysis performed in the first parts of this project focused on the classification of tweets according to a negative-positive grouping of tweets. In this section, we wish to explore the classification of tweets according to a set of distinct emotions. The *bonus_covid_sa.csv* dataset gathers 10000 tweets all labeled according to a range of emotions (Optimistic (0), Thankful (1), Empathetic (2), Pessimistic (3), Anxious (4), Sad (5), Annoyed (6), Denial (7), Surprise (8), Official report (9), Joking). The dataset was shared with you on Moodle. This is an example of a dataset you could use to train an multi-class multi-label emotion analysis model, but feel free to use another approach (e.g. lexicon-based). Thoroughly explain the methods used to perform this novel task, the design choices made as well as the associated code snippets. Provide a visualization of your results and detail the conclusions you can draw about the COVID-19 epidemics using emotion analysis. Insights about how to conduct an emotion analysis can be found in this Nature article.