Mastering Natural Language Processing with Python

JAN BISCHOFF
DATA SCIENCE & VISUALIZATION

Objective
Natural Language Processing (NLP) is a rapidly growing field that enables computers to understand, interpret, and generate human language. It is widely used in academic research across disciplines—from digital humanities to social sciences and biomedical text analysis. This course focuses on advanced NLP techniques using the Python programming language, equipping researchers with the tools to analyze, model, and draw insights from textual data with clarity and precision.

Through hands-on coding exercises and real-world datasets, you will develop practical skills for processing and analyzing natural language. You will explore foundational and advanced methods, from preprocessing and vectorization to machine learning models for text classification and generation.

Description
This course is designed for researchers and practitioners with prior experience in Python programming. It delves into core and advanced NLP techniques commonly applied in academic and research contexts.

After successfully completing this course, you will be able to:

- Process and clean raw textual data for analysis
- Transform text into numerical representations using techniques like TF-IDF and word embeddings
- Build and evaluate models for tasks such as sentiment analysis, topic modeling, and named entity recognition
- Use Python libraries such as NLTK, spaCy, scikit-learn, and Hugging Face Transformers
- Apply best practices for interpretability, reproducibility, and responsible use of NLP in research

Methods
- The course will alternate between short introductions to concepts or methods and small do-it-yourself coding exercises.
- In between the theory sessions, you are encouraged to work on exercises that further deepen your understanding.

Conditions
- **Prior coding experience is strongly recommended.** The course will dive deeper into concepts of advanced visualization in Python. Prior experience in data science with Python (pandas, basic pyplot, basic seaborn) is recommended to follow the content. However, some reading material will be shared before the course to repeat the basics.
- **A Google Account** to work with Google Colab as a simple and ready-to-go development environment. Alternative: a locally installed Jupyter Notebook environment. Due to time constraints, we will not go through the installation and setup, so we highly encourage using Google Colab.
- **Please bring your own laptop** to code along during the application exercises.