#MACHINE LEARNING

Like many other digital humanities terms, #MACHINE LEARNING appeared as neologism specifically in the field of computing during the post-war period. It focuses on "the capacity of computers to learn and adapt without following explicit instructions, by using algorithms and statistical models to analyse and infer from patterns in data." * The definitions by the mixed methods projects sketch broader and deeper fields of issues. They point to different processes of learning in the area of computing in comparison with human learning and trace the structure as well as process of the learning, thus connecting machine learning to #HUMAN-IN-THE-LOOP.

First, they differentiate between supervised and unsupervised learning: "The supervised learning maps an input – based on example input-output pairs– to an output. In the unsupervised learning, there are no pre-existing labels and it tries to find previously unknown patterns in data set." Different strategies of #MACHINE LEARNING address different information or research interests when the "algorithms use manually engineered features or automatically learnt representation" (Rhythmicalizer). As such, the decision for a specific alignment of the algorithm forms a frame for the possibility and impossibility of results.

In addition to it, some projects underline that the application "of differentiable and parameterized mathematical functions" for "pattern recognition is effective at highly specific pre-formulated tasks of high labour intensity, but fails at transferring gained insights to different non-formulated contexts" (ANCI). Here, different roles of the human element emerge, ranging from the creation of the data and the algorithms to supervising, to potentially re-ordering, and finally, to interpreting the learning process.

* "machine, n.". in: Oxford English Dictionary (OED), Third Edition, March 2000; most recently modified version published online June 2021, https://www.oed.com/ [accessed 25.10.2022].

Title: anci – analysing networked climate images

Team: Birgit Schneider, Frank Heidmann, Thomas Nocke, Janna Kienbaum, Paul Heinicker

Corpus: We created our own research database from scratch via automated sampling of thousands of climate change images from the web. We filtered these images by certain keywords because we were interested in so called catch images, as specific recurring visual manifestations, like the Hockey Stick graph, the burning earth or protest photography.

Field of Study: Visual Studies, Media Studies, Human Cartography, Environmental Humanities, Interface Design

Institution: University of Potsdam, University of Applied Sciences Potsdam, Potsdam Institute for Climate Impact Research

Methods: The project searches for methods to conquer the sheer mass of climate imagery on the web to technically gather and process data. By quantitative means, we focus on approaches of visual analytics with the help of various visualisation techniques but also purely algorithmic approaches ranging from web scraping, computer vision, natural language processing and their extension via machine learning. These algorithmic processes were critically reflected and are in interaction with qualitative approaches to analyse the content. Hence, we used formal-stylistic or iconic, iconographic, iconological analysis methods as well as diagrammatic approaches. In addition, framing analysis was transferred from communication science to visual climate communication in order to analyse image content and its contextualisation for frames or specific statements and arguments.

Tools: There are various tools used in the project. Our toolset includes a web scraping application called *anci miner* – which we developed and programmed as main-outcome of our research endeavor, a visualisation interface for high-dimensional data sets with an additional cluster feature and a data visualisation interface which allows quick insights to main themes within a text corpus.

Technology: For the implementation of natural language processing, computer vision and machine learning algorithms we relied on open-source Python frameworks, such as OpenCV or Google's tensorflow. The resulting tools are custom-made web-based applications built with JavaScript and also enriched by open source add-ons