Bachelorarbeit am Institut für Informatik der Freien Universität Berlin

Human-Centered Computing (HCC)

Interactive Visualization Interface of Text Exploration and Annotation

Daniel Stachnik

Matrikelnummer: 5001807

danstach@zedat.fu-berlin.de

Betreuerin und Erstgutachterin: Prof. Dr. C. Müller-Birn

Zweitgutachter: Prof. Dr. E. Ntoutsi

Berlin, 22.09.2021

Eidesstattliche Erklärung

Ich versichere hiermit an Eides Statt, dass diese Arbeit von niemand anderem als meiner Person verfasst worden ist. Alle verwendeten Hilfsmittel wie Berichte, Bücher, Internetseiten oder ähnliches sind im Literaturverzeichnis angegeben, Zitate aus fremden Arbeiten sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

Berlin, den September 22, 2021

Daniel Stachnik

Abstract

Artificial intelligence (AI) has become increasingly present in everyday life, and it can be used wherever large amounts of data are available. The increasing complexity of AIs has led to research into methods that explain predictions of AIs (explainable artificial intelligence; xAI). When researchers work with large-scale text corpora, they often use algorithms and AIs to assist them in exploring these corpora. Clustering algorithms are used to classify similar texts into clusters. Dimensionality reduction algorithms like UMAP are used to visualize the results. The usage of such complex interfaces poses many questions about what explanations users want regarding the clustering, the visualizations, or the parameter tuning of those.

At the HCC Lab at FU Berlin, a text processing pipeline was developed to cluster and visualize large sets of YouTube comments. As part of this thesis, an interface was programmed to make the results of the text processing pipeline explorable. The interface also supports cluster annotation and visualizes the data using UMAP. Users can adjust the parameters of UMAP in the interface.

To find out what explanations users want about the functionality of such interfaces, especially concerning the visualization and clustering, a contextual inquiry was conducted with a social scientist working on the data produced by the text processing pipeline.

The contextual inquiry showed that the user did not desire explanations of how the clustering or visualization worked, although research on the emergence of explanation needs would have suggested otherwise. A thematic analysis of the data indicates that the specific perspective of interpretivist social scientists and the high complexity of exploring large-scale text corpora with many clusters may contribute to the low need for explanation. Reflexive technology is proposed as a possible way to increase explanation needs.

Zusammenfassung

Künstliche Intelligenz (KI) ist im täglichen Leben immer präsenter geworden und kann überall dort eingesetzt werden, wo große Datenmengen zur Verfügung stehen. Die zunehmende Komplexität von KI hat zur Erforschung von Methoden geführt, die Entscheidungen von KIs erklären (explainable aritifical intelligence; xAI). Wenn Forscher mit großen Textkorpora arbeiten, setzen sie häufig Algorithmen und KIs ein, um ihnen bei der Erforschung der Korpora zu helfen. Clustering-Algorithmen werden verwendet, um ähnliche Texte in Clustern zu gruppieren. Algorithmen zur Dimensionalitätsreduktion wie UMAP werden zur Visualisierung der Ergebnisse eingesetzt. Die Verwendung komplexer Methoden wirft viele Fragen darüber auf, welche Erklärungen (sog. explanations) die Benutzer in Bezug auf das Clustering, die Visualisierungen oder die Parametereinstellung derselben wünschen.

Am HCC Lab der FU Berlin wurde eine Textverarbeitungspipeline entwickelt, die große Mengen von YouTube-Kommentaren clustert und visualisiert. Im Rahmen dieser Arbeit wurde ein Interface programmiert, um die Ergebnisse der Textverarbeitungspipeline explorierbar zu machen. Das Interface unterstützt auch die Annotation von Clustern und visualisiert die Daten mit UMAP. Die Benutzer können die Parameter von UMAP im Interface anpassen.

Um herauszufinden, welche explanations die Benutzer sich über die Funktionalität solcher Interfaces wünschen, insbesondere in Bezug auf die Visualisierung und das Clustering, wurde ein contextual inquiry mit einem Sozialwissenschaftler durchgeführt, der mit den von der Textverarbeitungspipeline erzeugten Daten arbeitet.

Das contextual inquiry ergab, dass der Nutzer keine explanations zur Funktionsweise des Clusterings oder der Visualisierung wünschte, obwohl die Forschung zur Entstehung von explanation needs etwas anderes vermuten ließe. Eine thematische Analyse der Daten deutet darauf hin, dass die spezifische Perspektive interpretivistischer Sozialwissenschaftler und die hohe Komplexität der Exploration umfangreicher Textkorpora mit vielen Clustern zu den geringen explanation needs beigetragen haben könnten. Als möglicher Weg zur Steigerung der explanation needs wird reflexive technology erötert.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Context

Artificial Intelligence (AI) is widely used in many aspects of our daily lives. Complex AI models resemble black-boxes where it is not clear why the model made a specific prediction. Still, in many scenarios, users want or need to understand the reasoning of such a model.

Technical experts require explanations to better understand models [MZR21]. AI models were introduced to many other areas and to an audience of non-technical experts, too. A high need for explanations is presumed in fields like law and medicine (e.g. [MZR21], [WYAL19], [LYAW19]). Samek and Müller attribute the high need for explanations to the possible impact of wrong predictions in law or medicine [SM19]. Even though, or possibly because, a high explanation need is presumed in those fields, recent xAI research lacks the involvement of users and their perspectives into the usefulness of explanations [GR21]. Górski and Ramakrishna note that " <1% of papers in the area of case-based reasoning" involve user evaluation [GR21].

Similarly, there has been only a small number of user research into what explanation needs arise in applications specifically targeting social scientists. One text exploration interface for social scientists was developed by Baumer et al. [BSM+20]. In their publication, they explored many design needs of social scientists working with clustered text corpora [BSM+20]. In contrast to the high explanation need presumed in other fields, they note that for social scientists doing interpretivist research, "rather than providing an explanation [...], there is valuable space for techniques that support individuals [...] forming their own interpretation" [BSM+20]. This work extends the research of Baumer et al. and explores what explanation needs interpretivist social scientists may have in the context of text exploration.

A typical application use case for social scientists is the analysis of large text corpora [RSB+17]. Clustering is often used to structure the data. To further aid the understanding of the clustering, visualizations are commonly employed. Visualizations of high dimensional data like text embeddings require prior dimensionality reduction with algorithms like UMAP [MHM20]. Such dimensionality reduction algorithms remain challenging to evaluate regarding their accuracy and usefulness [MHM20]. As dimensionality reduction algorithms usually require parameters that influence the resulting visualization, interactive parameter tuning opportunities can be used. However, it remains unclear

what explanations needs exist in interfaces facilitating text exploration and annotation, in general, and, more specifically, what factors visualizations play in users' need for explanations.

At the HCC lab at FU Berlin, a text processing pipeline was developed to cluster and visualize a large textual dataset consisting of YouTube comments of climate change related videos. The pipeline was built as part of the research of a social scientist into commenting patterns on such videos.

1.2 Goals

As part of this work, an interactive interface integrating the results of the text processing pipeline was developed. This work further explores explanation needs users have of such interfaces. The following research questions are explored:

1. What explanation needs arise in social scientists exploring and annotating clusters of texts?
    a) How are visualizations used while interacting with an interface for text exploration and annotation?
    b) What effects do parameter tuning opportunities have on the perceived usefulness of visualizations?
    c) What triggers possible explanations needs?

1.3 Methods

A contextual inquiry was conducted with one social scientist to answer the research questions.

The contextual inquiry was used as a data collection method because it lets the interviewer observe the interviewee while they are working in their natural setting [RF96]. The interviewer can interrupt the interviewee; both are considered partners, and they build a shared understanding [RF96]. Thus, contextual inquiries are open by design and, compared with structured interviews or surveys, allow the user to explore an interface freely. Still, the interviewer can set a focus on explanation needs. As it produces rich qualitative data, only one inquiry was conducted. Thus, the results of this work are not generalizable. Considering the open-ended research questions, the deep insights of qualitative research into the reasoning of users working with interfaces for text exploration and annotation are preferable.

The contextual inquiry was designed to have two phases. In the first phase, the user was asked to get familiar with the interface. In the second phase, the user was asked to use the interface to get an overview of the comments. If additional questions arose regarding explainability, the user was interrupted and asked about their reasoning. The contextual inquiry took 2:07 hours and was conducted in German. It was recorded and transcribed.

The transcript was inductively analyzed using thematic analysis. In the first step, all non-explainability-related themes were removed. In successive steps, all explainability-related themes were gathered and aggregated which resulted in the topics of the *Results* chapter.

1.4 Structure

This remainder of this work is structured as follows:

1. First, all background topics necessary to understand this work are explained.
2. The requirements and design decisions considered during the development of the interface for text exploration and annotation are presented and the interface itself.
3. The results of the contextual inquiry are presented.
4. The results are discussed.

The structure of this work can be seen in figure 1.1. As the development of the prototype was relatively independent of the development of the text processing pipeline and the contextual inquiry, parts labeled as "*Concerns the Interface Implementation*" in the figure are not strictly necessary to understand the xAI related contributions of this work.

All interview data is in German. To improve accessibility, all quotes in this work were translated by the author at his best abilities to reflect the tone and message of the original. To respect the anonymity of the social scientist interviewed, he or she is referred to as they throughout this work.

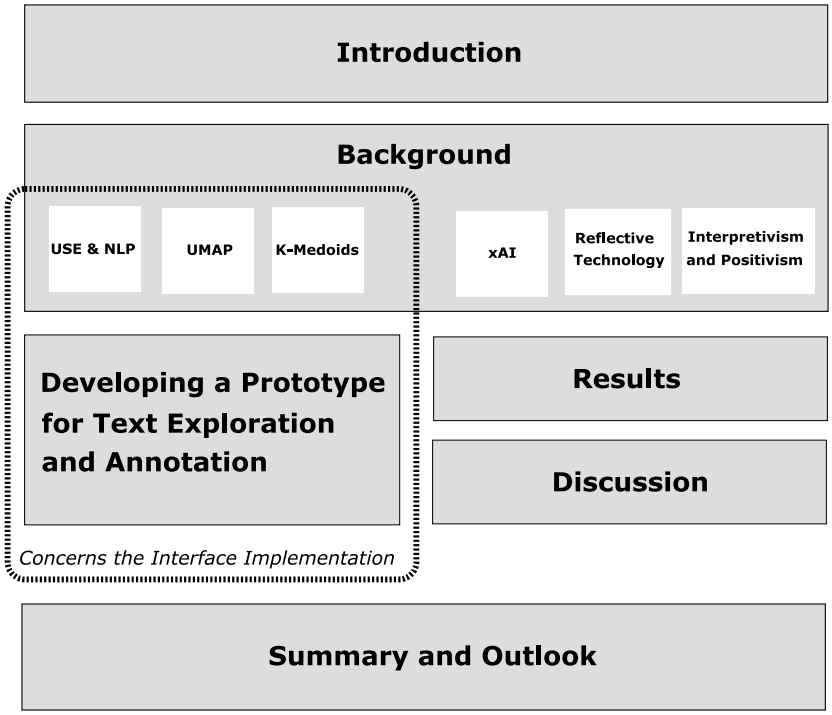


Figure 1.1: Structure of this work.

2 Background

2.1 Universal Sentence Encoder and NLP

The text processing pipeline developed at the HCC lab analyzes YouTube comments. Natural Language Processing (NLP) is the branch of research concerned with text analysis. According to Young et el., "Natural language processing (NLP) is a theory-motivated range of computational techniques for the automatic analysis and representation of human language" [YHPC18]

There are different ways to represent human language. One classical approach is the vector space model, wherein its simplest form, each word represents one dimension in the vector space [YHPC18]. This approach suffers from the *curse of dimensionality* [YHPC18]. This led to the embedding of words into low-dimensional space where individual words share dimensions [YHPC18].

It is also possible to embed sentences [CYyK+18]. In this case, two sentences can be compared against each other for their semantic similarity. One such model is the Universal Sentence Encoder published by Cer et al. [CYyK+18]. The Universal Sentence Encoder is made public as a pre-trained model. It embeds sentences into 512 dimensional output vectors [CYyK+18].

2.2 UMAP

Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP) is a dimensionality reduction algorithm [MHM20]. Dimensionality reduction is used to produce visualizations and as a pre-processing step to mitigate the effects of the curse of dimensionality [MHM20]. The text processing pipeline uses UMAP, first to decrease the 512 output dimensions of the Universal Sentence Encoder to 200 dimensions, then to produce the visualization.

UMAP is a k-neighbor based graph algorithm [MHM20]. The working of UMAP follows two steps. In the first step, a graph is constructed which resembles the data [MHM20]. In the second phase, "a low dimensional layout of this graph is computed" [MHM20] which fits the layout to the graph constructed in the first step. UMAP constructs the graph such that all data points must be connected to at least one other data point of the data set [MHM20]. The maximum number of neighbors is set as a hyperparamter [MHM20]. As the

graph constructed connects any point to at maximum the number of neighbors set as parameter, low numbers set the focus of UMAP towards the local structure of the data, whereas higher values set the focus on the global structure [MHM20].

UMAP is relatively fast which would allow an interactive change of data [MHM20]. UMAP also has partial GPU-support [NLR+21] which can increase the run-time even further.

2.3 K-Medoids

Clustering is the process of grouping similar data points together [PJ09]. The text processing pipeline supports multiple clustering algorithms. The final prototype of the interface for text exploration and annotation only used k-medoids. K-medoids has the advantage of calculating a centroid per cluster, which is the data point lying most centrally in the cluster [PJ09]. The centroid can be interpreted as the representative data point of the cluster [PJ09].

K-medoids works very similar to k-means [JD88] with the difference that the cluster center is chosen from the existing data points rather than computed. Short pseudocode based on the description of Park and Jun is shown in listing 2.1 [PJ09].

The algorithm creates initial clusters in the first step. For that, the relative distances of one point to all other points are summed. The relative distance is the distance of two data points a and b, divided by the sum of all distances of b to all other points (line 12). Then, the k points with the lowest summed relative distances are chosen as initial centroids. Each point is assigned to the closest centroid. As long as the overall dissimilarity decreases, 1. the points of the cluster are updated, and 2. the medoids are updated.

For better readability, a class Clusters was assumed in the listing, consisting of a list of Cluster objects that have a medoid attribute and a points list.

```
1  def dist(a, b): # distance metric of a to b
2  def dist(a, B): # sum of distance metric of a to all of B
3  def update_points_of_cluster(clusters):
4      # assign each point to nearest medoid
5  def update_medoids(clusters):
6      # set medoid to point of cluster which minimizes distance to all
           other points of cluster
7
8  def intial_clusters(X, k):
9      for x_1 in X:
10         cost = 0
11         for x_2 in X:
12             cost += dist(x_1, x_2) / sum(dist(x_2, x_3) for x_3 in X)
```

```
        X[x_1].cost = cost
    clusters = Clusters(medoids=k_with_smallest_cost(X, k))
    update_points_of_cluster(clusters)

def k_medoids(X, k):
    clusters = intial_clusters(X, k)
    clusters = update_medoids(clusters)

    prev_dissimilarity = INFINITY
    dissimilarity = sum(dist(c.medoid, c.points) for c in clusters)
    while dissimilarity < prev_dissimilarity:
        clusters = update_points_of_cluster(clusters)
        clusters = update_medoids(clusters)

        prev_dissimilarity = dissimilarity
        dissimilarity = sum(dist(c.medoid, c.points) for c in clusters)

    return clusters
```

Listing 2.1: Pseudo-implementation of k-medoids in Python

2.4 Explainable Artificial Intelligence

Explainable artificial intelligence (xAI) is research branch concerned with the explaining of model's predictions [MZR21]. Explainable AI supports "user understanding of complex models by providing explanations for predictions" [MZR21]. Not all models need explanations.

Interpretable models are "inherently human-interpretable models due to their low complexity of machine learning algorithms" [MZR21]. Examples are decision tree or binary vectors [RSG16].

Research into xAI includes many aspects. Mohseni et al. conducted a research review on xAI publications and categorized the research of xAI into social science, human-computer-interaction (HCI), visual analytics, and machine learning aspects [MZR21].

Visual Analytics and Machine Learning cover the technical aspects of xAI. It covers the research of methods that facilitate the interpretation of ML models [MZR21]. One example for such a method is Local Interpretable Model-agnostic Explanations (LIME) [RSG16]. LIME works by using a weighted sampling of inputs [RSG16]. The weight is set according to the proximity to the input to be explained [RSG16]. An interpretable model is then trained on the weighted samples and the predictions for those samples [RSG16]. This gives a good approximation of the decision boundaries of the local neighborhood, but is computationally expensive [SM19].

HCI aspects involve research that proposes theoretical or design principles for a better human-explanation fit [MZR21]. Similarly, social science aspects try to improve explanations by integrating insights from social science on the nature of explanations [MZR21]. For example, Miller reviewed explanations from a social science perspective. A key finding of him was that explanations are contrastive, i.e., only sought in response to another event ("Why did this happen instead of this?") [Mil18].

This work focuses on the HCI and social science aspects of xAI.

2.5 Reflective Technology

With the omnipresence of computers in our daily lives, using them has become natural and intuitive for many people. This behavior has the risk of being unreflective, i.e., users do not question interfaces they use.

The lack of reflection can be abused in multiple ways using so-called dark patterns [GKB+18]. Gray et al. define dark patterns as "instances where designers use their knowledge of human behavior (e.g., psychology) and the desires of end-users to implement deceptive functionality that is not in the user's best interest" [GKB+18]. An example dark pattern is *preselection*, where some options are already selected in interfaces, e.g., the automatic subscription renewal leading unfocused users to an auto-renewing subscription [GKB+18].

Even without the conscious use of dark patterns, unreflective behavior may pose other risks, such when interfaces predict or recommend actions that may be biased because an underlying model itself was trained on biased data [DeB18]. An example highlighted by Debrusk is COMPAS, a model recommending criminal sentences, which used the offender's race as an input parameter but not the past arrests [DeB18].

Reflective technology is defined as "technology that both invites reflection and at the same time is reflective in its expression." [HR01]. Baumer further conceptualized reflection [Bau15]. He found that reflection instances require a "breakdown" moment, that they are followed by an inquiry and lead to a transformation [Bau15].

Reflective technology and xAI share a common goal, which is to improve the user understanding of a system ([SM19], [FF10]). Most xAI research assumes the need for explanations (e.g. [MZR21], [WYAL19], [LYAW19]). Research into reflective technology places explanations into a broader conceptual model of human technology usage as a whole and further asks how reflective behavior can be achieved.

Fleck and Fitzpatrick introduced a framework that assists in evaluating the level of reflection on the user side and how technologies can support those [FF10]. A summary of the reflection levels can be seen in table 2.1.

2.6 Interpretivism and Positivism

The philosophical foundation of social science can be broadly split into interpretivism and positivism [AP20]. Positivist researchers in the context of social science try to "ascertain objective facts about social reality" [BSM+20]. This thinking is in line with the empirical research of natural science [AP20].

Interpretivist research "seeks to understand the means by which social groups mutually co-construct and interpret their reality"; differently put, from an interpretivist viewpoint, social reality depends on the perspective of individuals [BSM+20].

Interpretivist research differs from positivist research in that interpretivist research tries to understand the motives of individuals – why they act the way they do – whereas positivist research tries to find unbiased and generalizable "laws dictating human behavior" [AP20]. Interpretivist research usually involves qualitative methods, positivist usually quantitative methods [AP20].

	R?	Description	Example Techniques
Level 0	✗	**Description: Revisiting** Description or statement about events without further elaboration or explanation.	Stating information
Level 1	✗	**Reflective Description: Revisiting with Explanation** Description including justification or reasons for action or interpretation, but in a reportive or descriptive way. No alternate explanations explored, limited analysis and no change of perspective.	Prompting user to explain, e.g. through the use of reflective questioning like "Why did you name the cluster like that?".
Level 2	✗	**Dialogic Reflection: Exploring Relationships** A different level of thinking about. Looking for relationships between pieces of experience or knowledge, evidence of cycles of interpreting and questioning, consideration of different explanations, hypothesis and other points of view.	Offering multiple alternative causes to trigger perspective change. Offering information about past user interaction, such as information considered or time spent on specific actions.
Level 3	✓	**Transformative Reflection: Fundamental Change** Revisiting an event or knowledge with intent to re-organise and/or do something differently. Asking of fundamental questions and challenging personal assumptions leading to a change in practice or understanding.	Such reflection happens by means of deep interaction with levels above.
Level 4	✓	**Critical Reflection: Wider Implications** Where social and ethical issues are taken into consideration. Generally considering the (much wider) picture.	Such reflection happens by means of deep interaction with levels above.

Table 2.1: Levels of reflection. Higher levels build on lower levels. R? indicates whether the level leads to conscious reflection on the user side. The summary is based on [FF10]. All descriptions are directly quoted from [FF10].

3 Developing an Interface for Text Exploration and Annotation

This chapter summarizes the development of the interface for text exploration and annotation. The resulting interface is referred to as *prototype* to distinguish it from the other interfaces.

3.1 The Text Processing Pipeline

The text processing pipeline was developed at the HCC lab to allow a faster exploration of 25914 YouTube comments concerning climate change videos. The pipeline is written in Python in a Jupyter Notebook. The simplified processing flow can be seen in figure 3.1. The *Clustering* step supports k-means, k-medoids, and agglomerative clustering.

The parameters at each step were sensibly chosen with respect to the clustering task at hand. The first UMAP step reduces the 512 output dimensions of the Universal Sentence Encoder to 200 dimensions. This is followed by the *Clustering* step. In the first stage of development of the prototype, k-means and k-medoids were considered. When it was decided to show the centroids of k-medoids as cluster representatives, k-means was removed for the lack of representatives. To produce a visualization, the text processing pipeline reduces the 200 dimensions to 2 dimensions. As UMAP's dimensionality reduction works on both unsupervised and (semi-)supervised data [MHM20], the pipeline can consider the labels created in the *Clustering* step or skip the labels. Skipping the labels doesn't risk reinforcing wrong clustering results, but for text corpora effectively gives a visualization where all points are close to each other, and no structure is visible.

The text-processing pipeline was run with sensible parameters as were preset on April 5, 2021. The visualization step was run multiple times on the labeled data with different parameters. Users of the interface may change the visualization parameters from a discrete number of values. The resulting data was put into JSON files.

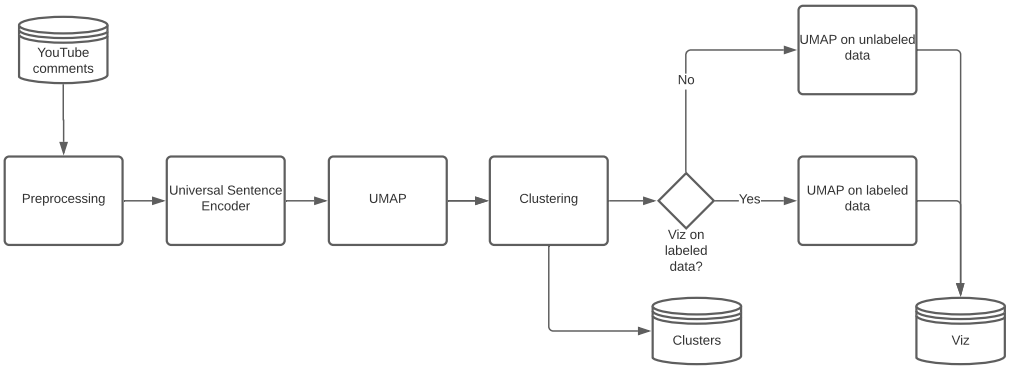


Figure 3.1: Simplified Text Processing Pipeline developed at HCC lab at FU Berlin.

3.2 Requirements

Functional requirements were elicited by researching cluster exploration tools found in the literature and comparing their functionality. A key set of requirements were given beforehand. The interface was required to:

- Allow users to explore shared topics of text corpora
- Allow users to explore contained documents
- Allow users to see relevant feature terms
- Show quality measures for clusters

A variety of interfaces were taken into account to validate and extend the requirements and have the it resemble currently used tools for text exploration and annotation.

Interfaces developed by Legg et al. [LSD19] and Ruppert et al. [RSB+17] each offered similar functionality as required. Furthermore, LIT [TWB+20] and UMAP Explorer were considered. In the following, the reviews for all tools are listed.

ActiVAte

Legg et al. developed an interface for active learning, i.e., the data can be incrementally labeled by the user [LSD19]. *ActiVAte* could not be run locally because the source code was not publicly available. Thus, all analysis was based on the publication. Legg et al. demonstrated the interface on the MNIST dataset [LC10].

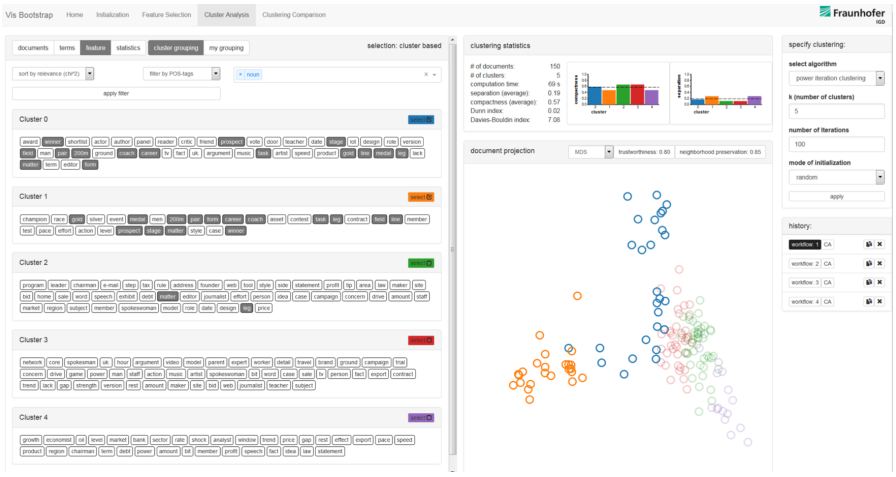
Figure 3.2: Interface of ActiVAte, annotated by its components. Taken from [LSD19].

ActiVAte provides a visualization of MNIST with the help of dimensionality reduction algorithms (*Sample Pool View* in Figure 3.2). Manually classified points in the visualization are highlighted in the label the user annotated it with. The *Classifier View* is a drag-and-drop field, where users can drop new digit images onto the corresponding label region. Both color highlighting and drag-and-drop on the corresponding label region take advantage of the small overall label size. This approach is infeasible for data sets with many more labels because they would lead to a loss of overview in the *Classifier View* and the colors in the *Sample Pool View* would become increasingly harder to differentiate. There is also a *Configuration View* to change the parameters of algorithms used in the clustering pipeline. The *Test Accuracy View* gives metrics on how well the model based on the user input so far performed on test data sets. This only works if there already exists test data. Lastly, the *Confusion Matrix View* shows how often user-set labels varied from the model's prediction. Overall, the interfaces leverage the fact that there are only ten possible labels in the data set. This makes it difficult to generalize the interface to other clustering tasks.

Visualization System by Ruppert et al.

Ruppert et al. developed an interface that gives users granular control of a text processing pipeline based on classical NLP without the use of machine learning [RSB+17]. This interface is not publicly available, so all analysis was based on the publication. The intended users of this interface are data analysts who have a high need of controlling all aspects of the text processing pipeline [RSB+17]. The interface spans multiple views.

(a)



(b)

Figure 3.3: Interface of the visualization system developed by Ruppert et al. **(a)** shows the *Feature Selection* view. **(b)** shows the *Cluster Analysis* view. Taken from [RSB+17].

.

The *Feature Selection* view (figure 3.3 (a)) gives the users control over the extracted features. The pipeline is based on a token-based vector space model. As such, each word has a clear representation in the vector space or isn't considered for further processing. Alternatively, users may choose bi-grams or tri-grams. Users are provided with multiple quality metrics such as *tf-idf*.

The *Cluster Analysis* view (figure 3.3 (b)) allows users to explore the text corpus, and to explore and evaluate clusters. On the left side of the view, terms of all clusters can be filtered and listed, for example, by the most frequent terms per cluster. A document projection is shown on the right side, cluster metrics like their compactness are given, and options to change used algorithms and

their parameters.

LIT

Tenney et al. released the Language Interpretiblity Tool (LIT), an open-source project containing an interface that visualizes and helps to analyze NLP models [TWB+20]. This interface utilizes much more sophisticated features to examine the underlying NLP model than the other interfaces. Thus, it is aimed at users familiar with NLP and machine learning. It shows a 3d visualization of the dataset embedded into three dimensions by either PCA or UMAP (left side in figure 3.4). It is disputed whether the additional third dimension outweighs the inherent problems of three-dimensional graphics on screens and paper [Sza18].

Confusion matrices, salience maps, and datapoint generators explain the model. LIT facilitates understanding of the underlying model. It does not offer tools to annotate clusters or comments.

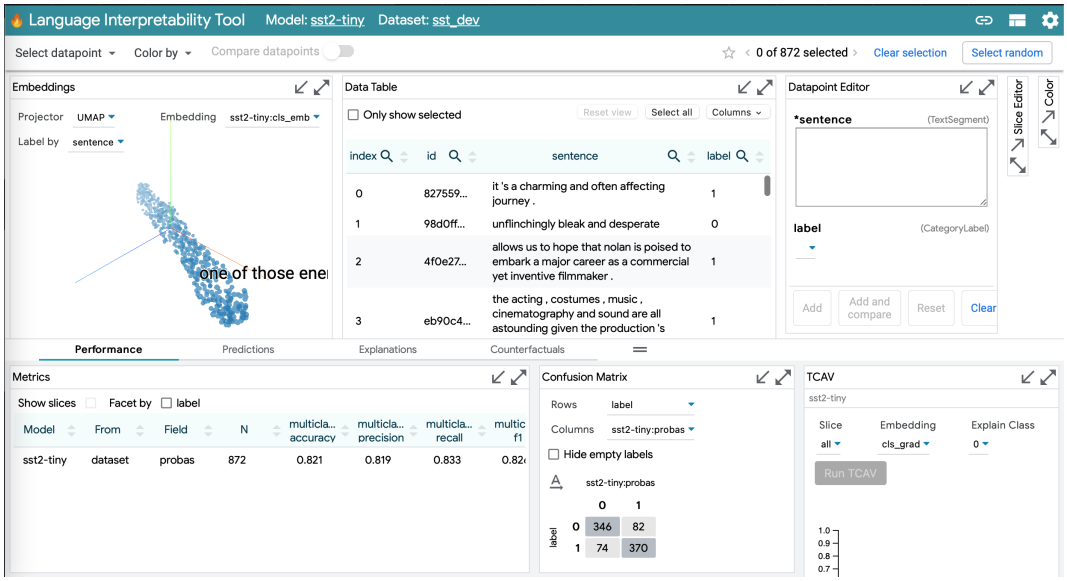


Figure 3.4: Interface of LIT.

UMAP Explorer

Grant Custer developed the UMAP Explorer[1] , which allows users to explore the MNIST data [LC10] set on a 2d UMAP embedding. The tool is open-source and based on React. As can be seen in figure 3.5, it is kept simple and

[1]https://github.com/GrantCuster/umap-explorer (last accessed on 13.09.2021).

only has one user-configurable parameter allowing to choose between PCA and different UMAP runs with different hyperparameters. Similar to *ActiVAte*, the interface takes advantage of the small number of labels in total and colors them differently in the visualization.

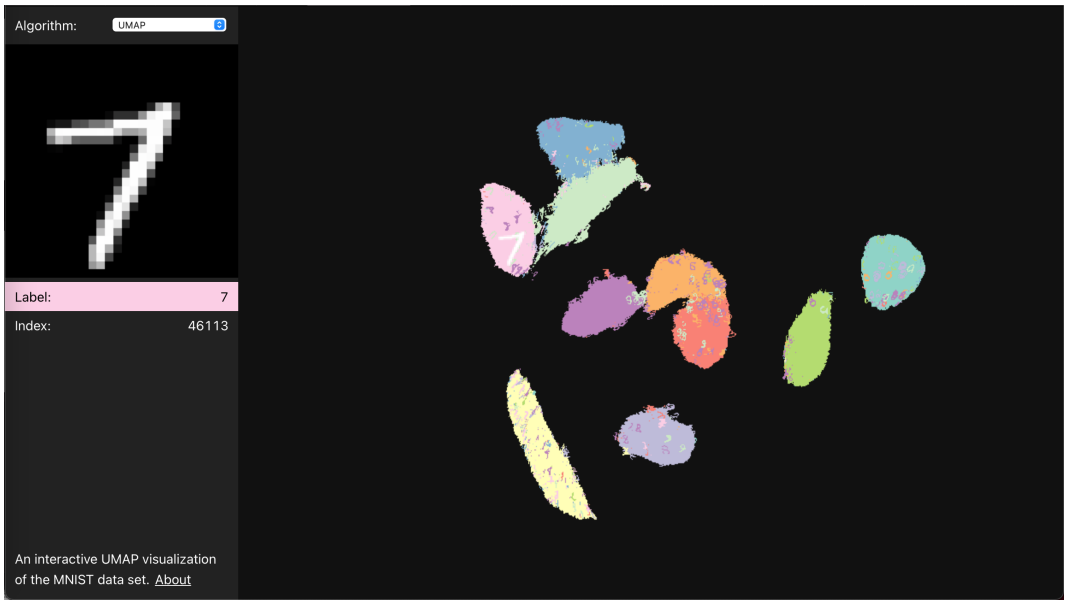


Figure 3.5: UMAP Explorer.

Comparison of Requirements

All interfaces were considered as base projects for this work to reduce development time. Although *ActiVAte* and *Visual System* offered insights into how to design the interface and what it should do, they could not be considered because they were not open source.

The analysis of the interfaces shows that the interfaces vary greatly in their offered features and, thus, the extent of technical expertise required to use them. Table 3.1 shows an overview of the requirements of each interface. *Visual System* and *LIT* both require a great technical expertise. This makes both interfaces infeasible to use, as the target user group of this work are non-technical experts. Although *LIT* in particular satisfied nearly all requirements, they already offered many explanations. As the goal of this work was to find out what explanation needs users have, showing explanations could bias users in what questions they ask such an interface. As the remaining interface was the *UMAP Explorer*, it was taken as a starting point for the development of this prototype and successively adapted to the requirements.

Requirements	*ActiVAte*	*Visual System*	*LIT*	*UMAP Explorer*	*Prototype*
User may explore shared topics of text corpora	✗	✓	✓	✗	✓
User may explore contained documents	✗	✗	✓	✗	✓
User may see relevant feature terms	✗	✓	✓	✗	✓
User may see relevant feature terms	✓	✓	✓	✗	✓
Suited for higher dimensional input space	✗	✓	✓	✗	✓
Input can be annotated	✓	✗	✗	✗	✓
Supports customization of hyperparameters	✓	✓	✓	✓	✓
Published as open source	✗	✗	✓	✓	✓
Quality Metrics are shown	✓	✓	✓	✓	✓
Target user	NTE	TE	TE	NTE	NTE

Table 3.1: Comparison of each interface analysed. NTE are non-technical experts, TE are technical experts

3.3 HCI Design Process

There could be no usability tests conducted. As the whole data set was tailored to the need of one social scientist's research, including them in the design process would have defeated the primary purpose of the prototype: exploring their explanation needs. This restriction required involving other means of designing the user interface such as expert reviews.

Informal evaluation is a feasible substitute for more formal approaches like heuristic evaluations [Bar20]. For the fast-paced prototype development, informal evaluation was considered beneficial. During the prototype development, three informal evaluations were conducted with two members of the HCC lab at FU, both of whom having experience in the fields of UI/UX design and Human-Computer-Interaction.

3.4 Result

The project is available online on Github.[2]. The source code of the interface is MIT licensed. A hosted demo is available via Github Pages. A screenshot of the main screen can be seen in figure 3.6. It is a fork of the UMAP Explorer. The following design decisions were made during the development:

- As the UMAP Explorer was build using React, it was decided to use Material-UI[3] as a component library to increase the development time. Material-UI adheres to Google's Material Design[4]. D3[5] was used to visualize the results of UMAP.
- There is only one primary color. Only the currently selected cluster is highlighted in the visualization. *ActiVAte* and *UMAP Explorer* each used distinct colors per cluster. This is infeasible for text clustering because the number of clusters may be large. For reference, the text processing pipeline initially has 40 clusters.
- As k-medoids was used as a clustering algorithm, each cluster has one medoid, which was interpreted as the overview comment (as can be seen at the right side of figure 3.6). On each cluster, k-medoids was run again with $k = 4$. This way, each cluster has four sub-clusters with four respective representatives. Those are interpreted as representative comments (as can be seen at the right side of figure 3.6).

[2] `https://github.com/defo10/interface-for-text-exploration-and-annotation`
[3] `https://mui.com`
[4] `https://material.io/design`
[5] `https://d3js.org`

- Users can rename each cluster and move each comment to another cluster. New clusters can be created.

- The left side of the interface represents the global view and allows interaction with all clusters. The user can explore the clusters and see metrics about them (labels, size, density). A squared euclidean distance was used as a density metric.

- The center part of the interface shows a visualization of a sample of the clustered data. At the bottom, there are sliders to change the visualization. The two most relevant parameters of UMAP [MHM20] can be changed: The number of neighbors and the minimum distance between points.

- The interface is a prototype. As such, no backend was developed. Instead, the output of the text processing pipeline was parsed to json files. These are available in the project repository, too.

- The interface only supports desktop screens and was tested on Safari, Chrome, and Firefox.

- A class diagram of all components added can be seen in fig. 3.7. Overall, the project spans 18 TypeScript files with 2079 lines (on average, 115.5 lines of code per file). This compares to 1082 lines of code of the *UMAP Explorer*. Much of the codebase regarding the projection of the *UMAP Explorer* could not be reused, however.

- For performance reasons, the number of comments shown was limited to 20 comments.

3.5 Explainability Scenarios

Wolf proposed the usage of explainability scenarios during the development of AI systems to better understand possible explanation needs of users [Wol19]. In the context of this work, they help in two ways. Firstly, they act as usage scenarios, i. e. they show how the interface could be used. Secondly, they prove that, in theory, the interface developed raises explanation needs. The scenarios are shown after the result section because they require knowledge about the interface. They were created during the development, however, as suggested by Wolf [Wol19].

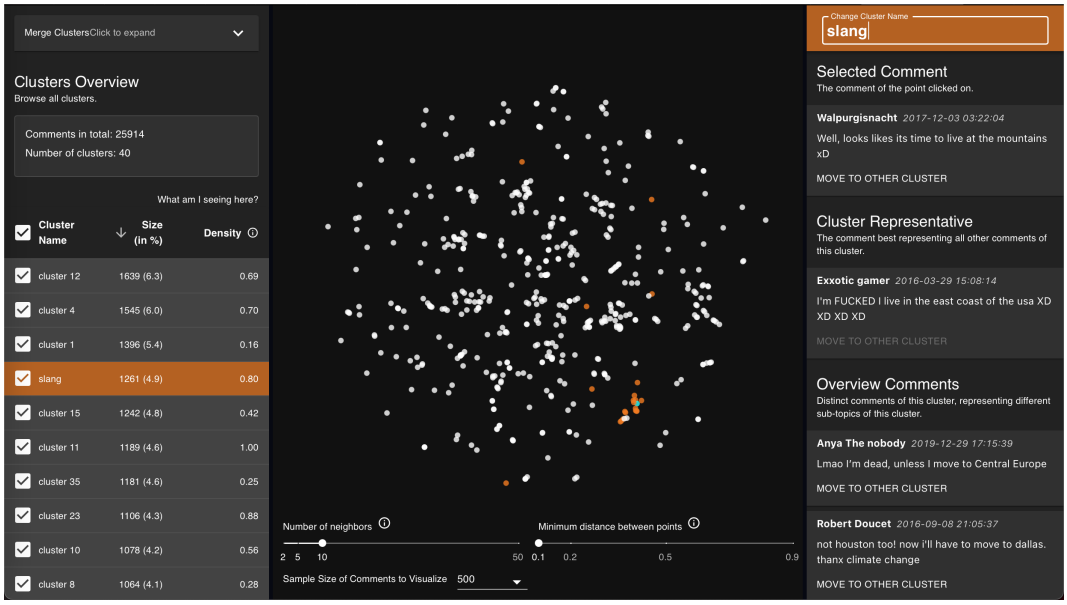


Figure 3.6: The interface for text exploration and annotation.

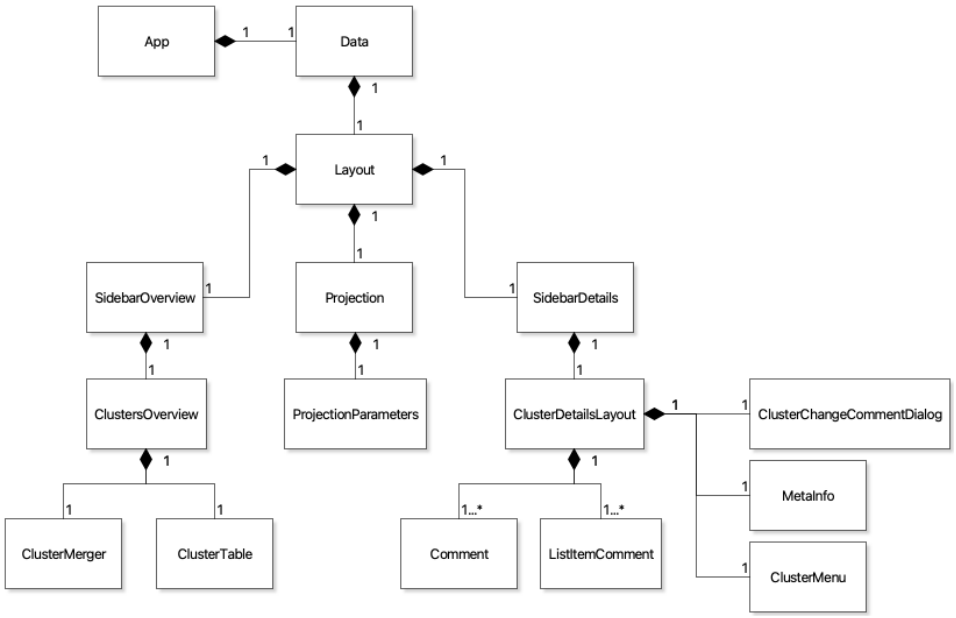


Figure 3.7: Class diagram showing all React components created.

Scenario 1: Getting an Overview of what?

Anna, a social scientist, is looking for ways to explore large quantities of textual data. The comments under the Youtube video "How Earth Would Look If All The Ice Melted" by Science Insider caught her attention. She wonders if the

comments have a common theme.

Anna uses the interface to analyze the comments. Upon opening, Anna is given a dashboard containing all clusters pre-computed by the program and a sample visualization of those comments. She immediately notices the visualization and wonders what the points shown mean and why some points are spread around.

Her attention is drawn to the clusters table. The first row, for example, contains the cluster name, 12, the size, 1639, and density, 0.84. Simone is surprised by the large size of numbers she is seeing and what they mean. Why are all cluster names numbers, she is wondering? And what does density mean? Is a high density good or bad? Trying to find out the interaction of the clusters with the visualization, Anna unchecks one of the rows. She notices that some points disappear, and some appear in the visualization. She concludes that these must be the points of that cluster. If these are the points of the cluster, why are there only so few points, though, Anna wonders?

Let's inspect one of those clusters, Anna thinks. She clicks on the first row. Some points in the visualization change their color. In the appearing panel, Anna notices a comment below "Cluster Representative." Anna thinks this might give her a good overview of the cluster. The comment is captioned with "the most centrally located point in the cluster". Anna isn't sure what that implies. Is the comment best describing all other comments of the cluster or the average of all comments? If the location, "centrally located", is important, does the position in the visualization imply meaning, that is, are closer points more similar, even if they don't share the same cluster?

Upon further scrolling, Anna sees the section "Overview Comments", which is described as giving an overview of the cluster. However, for the selected cluster, none of the comments provide a clear shared theme. Anna remains unsure why the program gave these comments as overview comments and why these comments are in the same cluster.

Scenario 2: Change is necessary

David is a joint political and data scientist. For his Ph.D. on conspiracy theories, he wants to see how many viewers of YouTube videos believe in conspiracy theories. Climate Change, David knows, has many deniers. He finds the Youtube video "How Earth Would Look If All The Ice Melted" by Science Insider and wants to use the interface to see how many comments cover conspiracy theories.

Due to his prior studies, David is already familiar with the basic ideas of NLP and Clustering. David knows that texts are difficult to cluster, so he immediately recognizes the visualization depicting the clusters. David knows

that clustering often involves parameters. He notices the sliders but doesn't know what the individual sliders do. Curious as David is, he decides to move the "Number of Neighbors" slider. It seems to change the denseness of clusters, he concludes. Next, he experiments with the "Minimum distance between points" slider. To his confusion, the changes seem similar to the one of the other slider. David remains unsure about the sliders and decides to move on for now.

Upon clicking on one of the points, the interfaces shows a list of text comments under different headlines and captions. Under the Overview Comments for the selected cluster, David couldn't find one about conspiracies. He decides to change the name of the cluster to "No CT". He clicks on another cluster and sees one comment which he believes might be about climate change denial. He follows the system dialog to move the comment to a new cluster. A highlighted comment appears, depicting the change. However, David notices, the visualization didn't change. He is unsure why that is. He is also uncertain if now only the selected comment will get moved during the next clustering iteration, or other similar ones as well, as he had wished.

Feeding the changes into the system and requesting a cluster update, David is unsure what influence his manual input has on the new cluster results. He wonders if clusters he visited and found to be well-clustered will now be split again.

4 Results

All themes concerning usability were removed. The resulting themes broadly covered two topics: How the user[1] thinks and how the user interacts with the interface. Both topics and their associated themes are discussed in the following.

4.1 How the User thinks

The user does not rely on the model. The user's thinking can be split into two themes: The user employed interpretivist thinking and was convinced of a low reliance on the model.

4.1.1 Interpretivist Thinking

Although not explicitly stated, the user showed strong interpretivist thinking. Stating their previous experiences with computer scientists, the user found that there were differences in their beliefs. They addressed the supposed belief of computer scientists on multiple occasions. When asked about the usefulness of the visualization for them, they explicitly said:

> But I also noticed that the focus of designers, including <> but also you and all others was way more on the visualization, that you care way more about the visualization and how the data is arranged and structured than is useful for my purposes. And maybe, maybe that's due to a different kind of scientific work.

On another occasion, the user tried to identify the cause for the different beliefs:

> The fundamental problem is that you assume that there's a reality in the data and that this reality is to be structured in a specific way, and that one of those structures resembles the actual structure and the others do not.

[1] The study person is named user for readability reasons.

The criticized belief is a classical positivist belief. In pointing out the difference in thinking, the user criticizes the notion of computer scientists to find ”the one true structure”. Rather than seeing the clusters as a model of reality, the user argued that

> the cluster isn't really anything. It's constructed. It's a constructed structure. This structure only exists in this model and not through reality, a social reality.

When computer scientists think of good clustering, they expect all data points of one cluster to have distinct features that separate them from the other clusters [PJ09]. Indeed, this was broadly given as the definition of clustering in the background chapter. The user, however, was much more interested in outliers; they found ”that the purpose of this tool is to surprise me and to offer me new perspectives on the data”. Due to their ”qualitative work”, it was not important to see how ”consequent, how coherent” clusters were.

4.1.2 Low Reliance on the Model

The statements of the user regarding their mental concept of the interface's model are multi-faceted. The user had an in-depth knowledge of the comments and the clusters. They would justify comments considered by the interviewer as misplaced. For example, when the user explored a cluster filled with comments containing the word ”Netherlands”, one comment was about Belgium. The user justified the prediction of the model, stating,

> It is there because Belgium is next to the Netherlands and this model finds both comparable or close to each other.

In a similar case, the user stated that the model had ”somewhat of a feeling for geography”. The clustering was not questioned. The user explained that they did not question the clustering because it worked well enough. This indicates that the user had some confidence in the performance of the system.

On the other hand, the user stated that they do not trust the model's prediction regarding the visualization or the clustering. Regarding the visualization, the user stated twice during the interview that they ”don't believe what can be seen there anyway”. They said that they don't find it essential how exactly clusters are shown in the visualization concerning the visualization parameters. Due to their former experience with dimensionality reduction of the text processing pipeline, they did not expect the visualization to ”show the exact distances between clusters”.

They also emphasized that understanding how the clustering works would be ”interesting from a theoretical perspective” but unnecessary for their work.

When asked if they would wish more information on how the model worked, they replied,

> Yes, so the question is if I need the information or if this information would be useful. Actually, I don't need it because I can conclude from the data what I want.

Indicating that they did not rely on the model's predictions due to not needing perfect results and not believing them.

4.2 How the User interacts with the Interface

The user's fundamental challenge while exploring the comments was managing the complexity of exploring the large text corpus. This influenced the user's interaction with the interface in general and the visualization in particular.

4.2.1 Superficial Exploration to tackle the Complexity of Large-Scale Text Corpora

The user did not go into much detail regarding any particular cluster. Instead, they tried a fast-paced procedure of checking overview comments, labeling the cluster, and continuing. At first, the user still checked around the first dozen comments plus the overview comments to validate their label. Later on, the user relied only on the overview comments alone.

The user summarized their process, stating

> But anyway, to gain control over the corpus, I first need to see all overview comments of all clusters once and I need to label them differently.

The user repeatedly mentioned that they would need this shallow process of exploration not to get "overwhelmed by what [they] see" and because they wanted to "work efficiently". They explained that they had no time at that stage "to take care of *[a deeper exploration]* now". In total, the user stated six times during the inquiry that they did not mind any individual comment of any given cluster at first, but that they would first need to get a big picture, such that "clusters can be labeled in my mental model of the corpus. [...]. Otherwise I would lose myself in the process. That's always very important".

Regarding the example mentioned above of the Belgium cluster considered misplaced by the interviewer, they replied

> No, it doesn't bother me at all because I'm in an exploration phase. I'm exploring the corpus, understanding the structure of the corpus, and hypothesize what it could mean.

The user wished for the model to assist them "in their workflow and not to prove anything or draw any conclusions for me".

The user was aware of their superficial approach. When confronted with other cluster comments seemingly not fitting, they mentioned that they would have noticed that comment eventually on their own, too. After the "exploration phase", the user said they

> would go into the cluster and look if the label fits the other comments, not only the overview comments, but more like 30 to 40 additional comments.

It can be seen that the user's primary concern was managing the complexity of large text exploration, rather than understanding or questioning the clustering.

4.2.2 Visualization considered as Reference Point

The user did not interact with the visualization or the visualization parameters. Towards the end of the inquiry, the user was asked to share their opinion about the visualization. They stated they "definitely need the visualization" but only as a "reference" to the clustering.

According to the user,

> Yes, and like I said, I need this spatial visualization but only as a reference. I think I wouldn't trust the clustering if I didn't see this mapping.

The user said they would not need ways to interact with the visualization, namely parameter tuning, as long as it provides an overview. They would need the visualization as "something to hold on to".

The user contributed the absence of using the visualization to the lack of "new perspectives" and "different interactions" offered by the visualization.

5 Discussion

Much of the literature indicated that the interface should have triggered explanation needs. The inherent complexity of exploring large-scale text corpora and interpretivist thinking often found in social science were identified as the root causes for this phenomenon. The superficial workflow of the user was also identified as problematic. Ideas as to how to prompt more thorough exploration are discussed.

5.1 Exploration of Large-Scale Text Corpora and Interpretivist Thinking Decrease the Need for Explanations

The interface did not trigger explanation needs during the interview.

There is only a little research as to what triggers explanation needs. According to Lim et al., a "deviation from expected behavior" triggers explanation needs [LYAW19]. According to Mittelstadt et al., "an anomaly or abnormal event" suggests explanation needs [MRW19]. The explainability scenarios proposed by Wolf [Wol19] showed that explanation needs existed in theory.

Similar cases during the interview did not trigger them, though. In two instances, the user was confronted with comments of clusters that did not correspond to the label the user chose. The non-fitting comments seem to deviate from the "expected behavior" [LYAW19] and can also be considered an "abnormal event"[MRW19]. The user replied either that the model's clustering was working "well enough" or that there could be underlying reasons for the model's choice, however.

No explanation needs were triggered with regards to the visualization either. The user found the visualization useful to understand the model but did not interact with it during the inquiry. This supports the notion that the visualization is seen as an explanation of the model itself. This is in line with Mohseni et al.'s research, who identified model visualizations as design goals for data experts and AI novices [MZR21]. However, the user explicitly stated that they did not mind how the clusters are shown in the visualization and did not expect them to be shown correctly. This questions the usefulness of parameter tuning opportunities for visualizations which is particularly striking

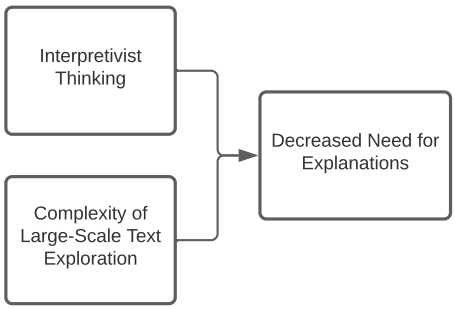


Figure 5.1: Factors contributing to a decreased need for explanations.

.

as the right choice of parameters is fundamental for the correct working of such algorithms, [MHM20].

The lack of explanation needs also raises questions on whether explanations would be used by non-technical experts at all, if presented to them. This supports the thesis of Górski and Ramakrishna, who found a lack of user studies in xAI research [GR21].

Two factors were found contributing to the lack of theoretically existing explanation needs.

5.1.1 Interpretivist Thinking Decreases the Need for Explanations

The user employed an interpretivist thinking, as was shown in the Results chapter of this work. The statements of the user were similar to those Baumer et al. gathered when designing a text exploration tool for interpretivist researchers, who were found to utilize text exploration tools as "curiosity-driven wandering" [BSM+20]. This aligns with the user asking to be "surprised" and "be offered new perspectives" by the interface.

If and if why interpretivist thinking contributes to lower explanation needs is not explored in the literature. Following the conceptual model proposed by Lim et al., explanation needs are triggered by a "deviation from expected behavior" [LYAW19]. Accordingly, absent explanation needs might be explained by the user's expected behavior of the model. The user mentioned that for their research, they don't expect the model to perform perfectly as they don't draw conclusions from the clusters; they were merely seen as means to explore texts and to find interesting types of comments. Due to the qualitative research of the user, they would not need to draw conclusions regarding the text corpora as a whole (as would be assisted by the clustering) but only regarding few

comments showing certain interaction phenomena. The expectation of the user, then, was not to have a "perfect" model but to have a model that's "good enough", thus not triggering explanation needs.

The user's wish to be offered "new perspectives" by the interface also counteracts a deviation from expected behavior. If misfitting comments of a cluster are considered a new perspective by the user, then those comments aren't deviations from the expected behavior; they are expected behavior.

5.1.2 Exploration of Large Text Corpora Decreases the Need for Explanations

A fundamental theme identified was the need of the user to tackle the complexity of exploring large-scale text corpora. To gain a conceptual model of the data, the user rapidly explored clusters and annotated them with a label of their choice. The user described this process as "gaining the control of the corpus". The perceived overwhelm of the clusters and comments is based on cognitive science. Humans usually can only hold between 3 to 15 information bits [VMP18]. Considering that there were 40 initial clusters in the model, humans cannot immediately create simple conceptual models of the data. The user would first need to explore the clusters. According to Miller, "people look for explanations to improve their understanding of someone or something so that they can derive a stable model that can be used for prediction and control" [Mil18]. This was not the goal of the user, however. In their first stage, the user was interested in gaining an overview of all clusters, not in understanding the model.

Samek and Müller found that the requirements for transparency, and thus the need for model explanations, depend on the overall performance of the model and the impact of wrong predictions [SM19]. Clustering algorithms do not make one prediction but for each data point decide what cluster it belongs to, so the number of predictions equals the number of data points. The impact of a single wrong prediction, a data point put in the subjectively wrong cluster, is low. If only a few predictions are subjectively found to be misplaced, most of the predictions would still seem fitting. How many subjectively considered misplaced comments there would need to be for the whole cluster to be "wrong" is not clear. The second variable described by Samek and Müller, the overall performance, was considered "good enough" by the user. This could have contributed to lower explanation needs as well.

5.2 Prompts for Self-Reflection could increase Explanation Needs

During the first stage of their exploration, the user would check the overview comments provided by the interface and the first dozen comments provided before labeling a cluster. In a later stage, the user proceeded to check the overview comments only. This rapid exploration was done to build a mental model of all clusters. The user found it inefficient to explore more comments before having seen all clusters. However, by not checking all comments, the user relies on the model's predictions, even if they did not think so themselves. The user assumed this to be unproblematic as they would still be the one concluding. This assumption could lead to biased conclusions. If the user only checked the overview comments, they would only check 40 $clusters \cdot 4\ comments = 160\ comments$ of 25914 comments in total (0.6 %). The user emphasized that they would check more comments later, citing around "30 to 40" comments, which would additionally be considered per cluster. Again, this would lead to 40 $clusters \cdot 44\ comments = 1760\ comments$ at maximum of 25914 comments in total (around 7 %). If only under 10 percent of all comments were read, the model's decision of what comments to show first could bias the user in their conclusions. More so, if over 90 % of comments are not read, any conclusion could and should be questioned.

It seems advisable to promote a more thorough analysis. This could be achieved by designing the interface with user reflection in mind. Increased reflection could, arguably, lead to more explanation needs because the user would inquire about their own and the model's behavior. Explanations could be considered reflection-prompts themselves. The exact nature of how reflective technology and xAI are connected is not researched yet.

The annotating of clusters was the predominant interaction of the user with the interface. In the following, some ideas are presented as to how to incorporate reflection prompts. The reflection levels described in table 2.1 hint at ways as to how to increase reflection.

Level 1 of reflection is the prompt for a reflective description. Users should be put in a position to reflect on why they made a decision. This could be promoted by prompting *reflective questions* (see [FF10]). For example, another text field could be integrated into the interface, asking the user to explain why they named a cluster as they did. This could prompt users to revisit and explain their decision.

Level 2, exploring relationships, seems promising to be promoted by xAI. Methods like feature attributions, providing counterfactuals, or data editors may be used to let users explore relationships. In one instance of a misplaced comment, the user did weigh possible causes for the clustering results. Providing explanations might help such exploration. Notably, the goal of reflective technology

would not be the transparency of the model, as is the case for xAI ([MZR21], [SM19]), but the transparency of the users choices to themselves.

Showing multiple causes to prompt reflections [FF10] contradicts the call for few selective explanations [Mil18]. Miller argued that good explanations are selective, i.e., they simplify the decision to a few variables. Obtaining user reflection, however, would call for the opposite, for showing all variables of a decision such that users obtain the full picture.

Apart from xAI methods, the interface could also indicate how many comments of a cluster users have seen. This could prompt users to reflect on how many (or few) comments they have seen.

Most of those reflection prompts discussed would make exploring large-scale data sets less efficient because they shift users' focus from labeling and exploring clusters to understanding the system. A trade-off between efficiency and thorough understanding seems inevitable and requires the involvement of users as much as designers.

6 Summary and Outlook

This work aimed to explore the explanation needs of a social scientist working in an interface for text exploration and annotation. The interface was developed as part of this work and is published as open-source under MIT license. It used the UMAP Explorer as a basis and allowed the annotation, merging and deletion of clusters, and the movement of data points. It relied on the results of a text processing pipeline built by the HCC lab at FU Berlin. The pipeline clustered 25914 YouTube comments of videos concerning climate change and provided a visualization using UMAP. The contextual inquiry conducted with the social scientist showed that no explanation needs were triggered, although the literature on explanation triggers would suggest otherwise ([LYAW19], [MRW19], [Wol19]). Causes for this phenomenon were researched. Interpretivist thinking, one large philosophical underpinning of social science, and the complexity of large-scale text exploration were found as contributing factors. Prompts for reflection could elevate the need for explanations.

Only one interview was conducted. As such, more research is necessary to validate the results of this work. The social scientist had prior experience with the data and clusters. They were aware of key algorithms used, but their understanding would only cover the broad ideas in a non-mathematical sense. This seems inevitable to prevent. To explore large-scale text corpora of tens of thousand text documents would almost always require some form of technical aid. If a social scientist were chosen who had no connection with the text processing pipeline, they would not be a domain expert of that data set.

K-medoid might not fit the actual data well, which could lead to worse performance of the clustering. A worse performance was acceptable in this work because it would, in theory, introduce more seemingly wrong comments in clusters, thus, more explanation triggers.

The interface only constitutes a high-level prototype. Feedback regarding cluster merging and moving comments to other clusters are considered in the local data but not in the visualization, which would require another run of UMAP. As the user did not use the visualization, this did not limit their user experience.

Possibly, explanation needs weren't triggered because the user was not aware of explanation types. Further research to see if they would use them if explanations were shown in the interface could yield answers.

Bibliography

[AP20] Husam Alharahsheh and Abraham Pius. A review of key paradigms: positivism vs interpretivism. *Global Academic Journal of Humanities and Social Sciences*, 2:1–2, 12 2020.

[Bar20] Carol Barnum. *Usability testing essentials : ready, set– test*. Morgan Kaufmann, Amsterdam, 2020.

[Bau15] Eric P.S. Baumer. Reflective informatics: Conceptual dimensions for designing technologies of reflection. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, page 585–594, New York, NY, USA, 2015. Association for Computing Machinery.

[BSM+20] Eric P. S. Baumer, Drew Siedel, Lena McDonnell, Jiayun Zhong, Patricia Sittikul, and Micki McGee. Topicalizer: reframing core concepts in machine learning visualization by co-designing for interpretivist scholarship. *Human–Computer Interaction*, 35(5-6):452–480, 2020.

[CYyK+18] Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder, 2018.

[DeB18] Chris DeBrusk. The risk of machine-learning bias (and how to prevent it). *MIT Sloan Management Review*, 2018.

[FF10] Rowanne Fleck and Geraldine Fitzpatrick. Reflecting on reflection: framing a design landscape. In *Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction*, pages 216–223, 2010.

[GKB+18] Colin M. Gray, Yubo Kou, Bryan Battles, Joseph Hoggatt, and Austin L. Toombs. *The Dark (Patterns) Side of UX Design*, page 1–14. Association for Computing Machinery, New York, NY, USA, 2018.

[GR21] Łukasz Górski and Shashishekar Ramakrishna. Explainable artificial intelligence, lawyer's perspective. ICAIL '21, page 60–68, New York, NY, USA, 2021. Association for Computing Machinery.

[HR01] Lars Hallnäs and Johan Redström. Slow technology – designing for reflection. *Personal and Ubiquitous Computing*, 5(3):201–212, August 2001.

[JD88] Anil Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice Hall, Englewood Cliffs, N.J, 1988.

[LC10] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[LSD19] Phil Legg, Jim Smith, and Alexander Downing. Visual analytics for collaborative human-machine confidence in human-centric active learning tasks. *Hum.-Centric Comput. Inf. Sci.*, 9(1), December 2019.

[LYAW19] Brian Y Lim, Qian Yang, Ashraf M Abdul, and Danding Wang. Why these explanations? selecting intelligibility types for explanation goals. In *IUI Workshops*, 2019.

[MHM20] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

[Mil18] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences, 2018.

[MRW19] Brent Mittelstadt, Chris Russell, and Sandra Wachter. Explaining explanations in ai. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 279–288, New York, NY, USA, 2019. Association for Computing Machinery.

[MZR21] Sina Mohseni, Niloofar Zarei, and Eric D. Ragan. A multidisciplinary survey and framework for design and evaluation of explainable ai systems. *ACM Trans. Interact. Intell. Syst.*, 11(3–4), August 2021.

[NLR+21] Corey J. Nolet, Victor Lafargue, Edward Raff, Thejaswi Nanditale, Tim Oates, John Zedlewski, and Joshua Patterson. Bringing umap closer to the speed of light with gpu acceleration, 2021.

[PJ09] Hae-Sang Park and Chi-Hyuck Jun. A simple and fast algorithm for k-medoids clustering. *Expert Systems with Applications*, 36(2, Part 2):3336–3341, 2009.

[RF96] Mary Elizabeth Raven and Alicia Flanders. Using contextual inquiry to learn about your audiences. *SIGDOC Asterisk J. Comput. Doc.*, 20(1):1–13, February 1996.

[RSB+17] Tobias Ruppert, Michael Staab, Andreas Bannach, Hendrik Lücke-Tieke, Jürgen Bernard, Arjan Kuijper, and Jörn Kohlhammer. Visual interactive creation and validation of text clustering workflows to explore document collections. *Electronic Imaging*, 2017(1):46–57, January 2017.

[RSG16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery.

[SM19] Wojciech Samek and Klaus-Robert Müller. *Towards Explainable Artificial Intelligence*, pages 5–22. Springer International Publishing, Cham, 2019.

[Sza18] Danielle Albers Szafir. The good, the bad, and the biased: Five ways visualizations can mislead (and how to fix them). *Interactions*, 25(4):26–33, June 2018.

[TWB+20] Ian Tenney, James Wexler, Jasmijn Bastings, Tolga Bolukbasi, Andy Coenen, Sebastian Gehrmann, Ellen Jiang, Mahima Pushkarna, Carey Radebaugh, Emily Reif, and Ann Yuan. The language interpretability tool: Extensible, interactive visualizations and analysis for nlp models, 2020.

[VMP18] Vanessa Volz, Kevin Majchrzak, and Mike Preuss. A social science-based approach to explanations for (game) ai. In *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–2, 2018.

[Wol19] Christine T. Wolf. Explainability scenarios: Towards scenario-based xai design. IUI '19, page 252–257, New York, NY, USA, 2019. Association for Computing Machinery.

[WYAL19] Danding Wang, Qian Yang, Ashraf Abdul, and Brian Y. Lim. *Designing Theory-Driven User-Centric Explainable AI*, page 1–15. Association for Computing Machinery, New York, NY, USA, 2019.

[YHPC18] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13(3):55–75, 2018.