Privacy-Preserving Visual Exploration of Healthcare Data with a Language Model Integration

Michael Grabner[1,†], Peter Waldert[1,*,†], Benedikt Kantz[1] and Tobias Schreck[1]

[1]*Institute of Visual Computing, Graz University of Technology, Austria*

Abstract

Healthcare records, measurements and datasets can quickly become highly complex in terms of storage, anonymisation and interpretation. With Neurodegen-Vis, we present a practical toolkit to anonymise, visualise and help interpret medical data in a Visual Analytics (VA) dashboard, with seamless integration of a Large Language Model (LLM). Our tool primarily aims to facilitate the discovery of relationships within the data while lowering usage entry barriers for users, such as medical doctors or researchers. The LLM integration can help to make the dashboard more accessible, by clarifying and explaining the meaning of features, giving an overview of the dashboard components and its interactions, and can help exploring the data by suggesting features to analyse and provide example prompts. Moreover, the language model integration can help the user interpret correlations in this specific dataset based on the meaning of certain features provided. The anonymisation is carefully performed using techniques from Differential Privacy, as we demonstrate this tool on Parkinson's Disease (PD) patient data.

Keywords

Visual Analytics, Healthcare Data, Differential Privacy, Language Models

1. Introduction

Medical datasets can be highly complex, ranging from measurements to social demographics. VA approaches help users to get an understanding of the data, clusters, and relationships among variables.

However, users face various challenges when interacting with a dashboard on complex data. Firstly, the user may have questions about the dataset itself, different conventions about feature names, or because they might not be a domain expert. Secondly, the user may have problems understanding the dashboard itself. Perhaps they do not understand the individual visualisations, how to select the views of interest and how the different parts of the dashboard interact. Thirdly, during data exploration, it can be hard to identify interesting aspects or where to start.

Motivated by these challenges, we implemented an LLM into a Visual Analytics dashboard for healthcare data to assist the user with these challenges. The dashboard itself is designed to identify correlations, dependencies, and groups within a dataset about Parkinson's disease patients. Figure 1 shows a screenshot of the primary view after data anonymisation.

The LLM is provided with information about the dashboard and the dataset. Combined with the general world-knowledge of LLM's it can help the user with exploration and explanation (see Example 1). The user has a classic, familiar chatbot interface to interact with. Additionally, the LLM is used to highlight and explain different views of the dashboard (cf. Examples 2 and 4).

Especially for critical data domains such as healthcare, the interpretation and conclusions should still be made by the user dependent on the data, not the LLM of course. Therefore, our focus is on explaining and exploring the dashboard and the data, rather than giving interpretations, and results only if the user can verify them on the dashboard. For these applications VA and LLM's compliment
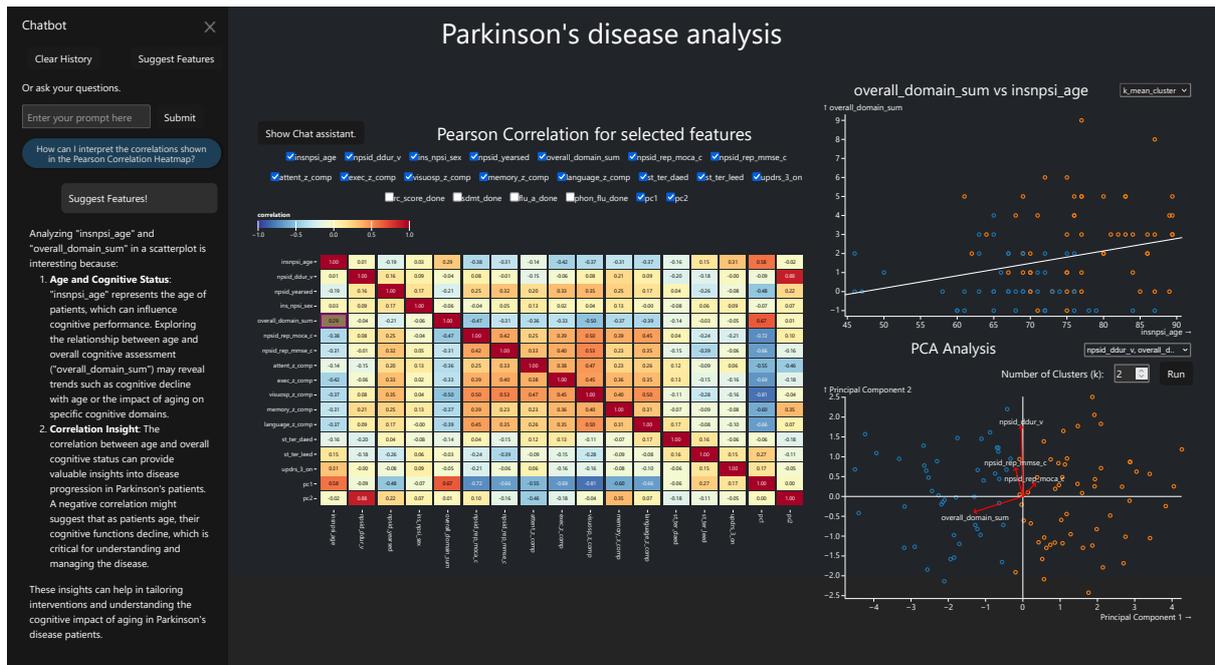
Figure 1: The dashboard shows a correlation heatmap on the left, and a scatter plot and biplot on the right. The chatbot interface is on the far left, where the *suggest features* function was used. Suggested features are highlighted in the heatmap and shown in the scatter plot. In the chat window, the LLM explains why this feature combination is interesting to analyse. The blue bubble displays an auto-generated follow-up prompt which can be clicked on to quickly and intuitively guide the user towards additional relevant information.

> *What is the correlation between age and language z test and what does this suggest?*

> The correlation between age (insnpsi_age) and the language z test (language_z_comp) is -0.413. This negative correlation suggests that as age increases, scores on the language z test tend to decrease. In other words, older patients may generally perform worse on language-related cognitive tasks compared to younger patients.

Example 1: Chatbot integration: the user can ask questions to interpret correlations in the real data, without the LLM ever seeing any of the data per se, only differentially private analysis results from it. The according cell showing the correlation is additionally highlighted on the heatmap (cf. Figure 2). This functionality is intended to facilitate result verification and promote user interaction.

each other well, as the user can always verify everything the LLM outputs and the LLM can enhance the dashboard by helping the user in understanding and exploration.

1.1. Dataset

The dataset contains information on patients with Parkinson's Disease. It comprises approximately 100 features for 50 patients. It includes demographic data such as patient age and duration of PD diagnosis, alongside results from various cognitive tests and treatment details. Several columns contain redundant information, with some measurements represented both numerically and categorically. Additionally, cognitive tests were conducted using multiple methodologies, leading to potential domain shifts. Binary indicators in the dataset denote the use of specific test methods. After accounting

for duplicate information and test method markers, the dataset includes approximately 15 unique measurements.
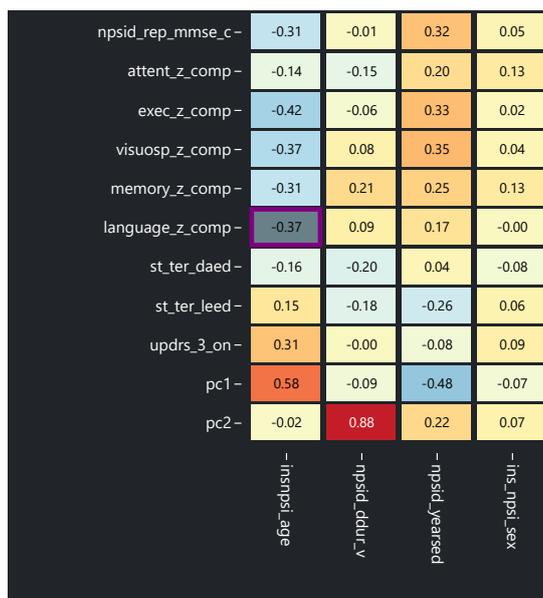
Figure 2: Part of the heatmap, where the cell with the correlation mentioned in the Example 1 interaction is highlighted upon interactino with the chat interface.

2. Related Work

The visualisation and analysis of high-dimensional data is a long-standing challenge in data science, and to date many approaches for interactive exploration of this data have been proposed. The Rank by Feature framework [1] was among the first approaches to identify and rank data features by their correlation, hence supporting the user in selecting interesting and relevant data. It used a heatmap to show the feature importance. In general, tabular data can be interactively visualised by general tools like spreadsheet software, and specialized development suites like Tableau or Microsoft PowerBI. Lineup [2] is a tabular data visualisation specifically for comparing and ranking the rows of a multivariate table, having the user interactively find appropriate weights for the different features. Correlation matrices and scatter plots are widely used tools to show an overview of pairwise correlations, amenable for overviewing and searching [3]. Recently, Large Language Model technology has drastically advanced the capability of Natural Language Processing, and existing implementations like Llama or ChatGPT allow to integrate LLM-based Natural Language Interfaces (NLIs) into interactive visualisation applications. There are many ways in which LLMs can help users to navigate and understand visualisations and data, as presented in the framework in [4].

In our work, we design a dashboard for exploring tabular data of numerical and categorical values. We make use of existing visualisations like correlation heatmaps, scatter plots and principal component plots. While these techniques are not novel, they can greatly aid user understanding and serve as an expert tool for data exploration. Our main contribution is the integration of an LLM interface to assist the user and to apply it to an anonymized real-world health data set. Our use case shows the dashboard design and LLM integration are helpful and hence can allow for explaining and guiding users.

3. Anonymisation using Differential Privacy

In order to protect patients' private data, our tool anonymises the incoming dataset before displaying it to a (potentially unauthorised) user. In a first step, names and directly personally identifiable information is stripped away, which only requires users to specify the affected columns.
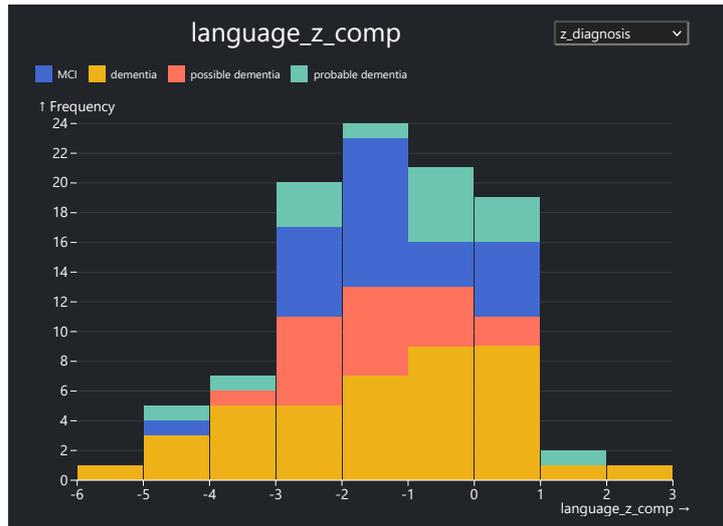
Figure 3: The feature *language z-component* in the histogram view. The colour encodes the categorical feature `z-diagnosis`, upon selection in the dropdown on the top right.

We then generate (up to an arbitrary amount of) synthetic tabular data based on the distribution of features across the input dataset. As such tabular data is often a mix of ordinal, categorical and continuous features, we have to treat them separately in the anonymisation phase. The categorical data is synthetically generated based on the Multiplicative Weights Exponential Mechanism (MWEM) [5], with an implementation in the third-party package `smartnoise-sdk`, developed by OpenDP[1] (and based on their formally verified `opendp` library). Next up, the continuous columns are then only sampled from their respective input distributions. For simplicity of this demonstrator, we assume a normal distribution in all numerical columns (also taking bounds and, for example, integer constraints into account). This still faces our generation procedure with the challenge of preserving correlations. Correlation can be preserved using randomised-order, correlation-integrated sampling from the respective feature distributions. One starts by obtaining a (Pearson-) correlation matrix $C \in \mathbb{R}^{n \times n}$ of the respective input features (in our case, a roughly 60x60 matrix) and passing it through the MWEM implementation to skew it with noise and making it differentially private with respect to the input data. During the sampling procedure, in random order, the columns are sampled from their respective (normal) distributions parametrised earlier. Each sample value is then skewed according to the correlation between all previously sampled values and the feature correlation contained in C. The output dataset then approximately follows the input distribution and keeps correlations consistent. The dashboard and corresponding LLM interaction (and, of course, the figures included in this work) now only rely on the synthetically generated data.

4. Visualisation Approach

Currently, there is a high amount of interest in the interaction of LLM and VA interfaces in the Visual Analytics community. The framework to describe our dashboard to the Generative Pre-trained Transformer (GPT) was inspired by [4]. Hutchinson briefly describes what challenges and opportunities Large Language Models provide for Visual Analytics integrations [6].

In Neurodegen-Vis, a correlation heatmap (cf. Figure 1), scatter plots (cf. Figures 4 and 5), and histograms (cf. Figure 3) are used to provide an overview of the data and enable the exploration of variable dependencies. For cognitive z-scores, which were obtained using different test methods, the visualisation colour-codes these methods and shows average values, to identify potential shifts due to methodological changes. Furthermore, a Principal Component Analysis (PCA) can be applied to detect

---

[1]https://opendp.org/

underlying group structures. The results are presented as biplots for interpretability. The different clusters are colour coded in the biplot, and can also be selected to be colour coded in the scatter plot, to further analyse the influence of the original features and how they could be grouped.

4.1. Natural Language Interface

The LLM should enhance the understanding and exploration of the dashboard knowledge, including information about the datasets, feature descriptions, details about the dashboard itself, and general world knowledge. The user can interact with the LLM on top of a familiar chatbot interface. The LLM implementation can also highlight cells of the correlation heatmap and change the scatter plot view.

> *Can you give a short overview of the data and the dashboard?*

> The data consists of various features from patients with Parkinson's Disease, including demographic information, cognitive assessments, and clinical visits. It assesses cognitive status, test results, disease duration, and other relevant factors. The Parkinson's Disease Visual Analytics Dashboard consists of three main views: a Pearson Correlation Heatmap to explore correlations between features, a Feature Scatter Plot for visualising selected feature relationships, and a PCA Biplot to display data clustering and feature loadings. Users can interact with the heatmap to filter and drill down into specific feature relationships.

Example 2: The initial prompt, shown and answered when the application loads, provides an overview of the dashboard and the dataset.

The chatbot can explain the different features (see Example 4). This is helpful for clarifying ambiguous terms to experts and can help non-domain experts to understand the data. It is not only convenient to have the feature descriptions within the dashboard itself (rather than in a separate document), but with the chatbot's world knowledge, the chatbot is also capable of providing more complex answers to follow-up questions and dashboard-specific queries. Moreover, it ensures that everyone who has access to the dashboard also has access to this knowledge base, reducing the chance of misunderstandings or inconsistencies. It also enables interactive explanation and exploration of the dashboard (see Example 2). It can explain individual views in detail (e.g. what a biplot is) as well as their interactions and the deeper meaning behind the visualisations on the dashboard.

> *How can I change the shown features in the scatter plot?*

> To change the shown features in the scatter plot, you need to interact with the Pearson Correlation Heatmap. Here's how:
>
> 1. Click on a non-diagonal cell in the heatmap to select the corresponding feature pair (Feature X and Feature Y). This will update the scatter plot to display the selected features.
> 2. If you click on a diagonal cell in the heatmap, it will change the histogram instead of the scatter plot.
>
> Additionally, you can select different colour encoding options for the scatter plot using the dropdown provided in the scatter plot view to visualise data based on different categories or clusters.

Example 3: Explaining the interaction between heatmap and scatter plot using the LLM integration.

As seen in Example 1, the chatbot also has access to results of the correlation analysis, enabling it to make more informed suggestions and assist in analysing and understanding the dashboard and data. There is an initial prompt that provides a brief overview of the data and the dashboard. Combined with the suggested follow-up prompts, this offers the user guidance on how to formulate effective queries and helps them craft meaningful prompts. The *suggest features* function supports users in exploring the dataset and identifying interesting aspects of the data. It not only proposes notable features but also highlights them visually. As shown in Figure 1, the corresponding cell in the correlation heatmap is highlighted, and the scatter plot updates to reflect the suggested features.
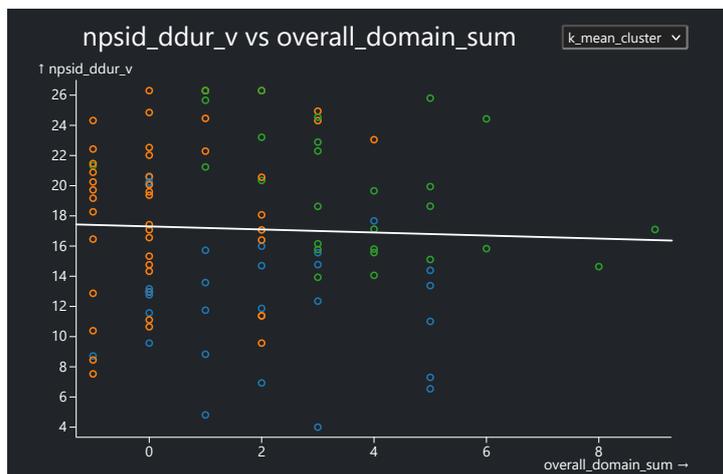
Figure 4: Scatter plot of two features in the anonymised dataset, right now coloured according to the k-means cluster results. The colouring can be changed to one of the categorical features through a dropdown.

Furthermore, when relevant to the chatbot's response, elements of the visualisation are dynamically highlighted. Currently, cells in the correlation heatmap can be emphasised, as illustrated in Example 1 and Figure 2. This aspect is especially important to us, as the user should continue focusing on the numerical and graphical information to draw insights, rather than relying solely on the LLM's analysis.

> *What is the feature overall_domain_sum and why could it be interesting to analyse?*

> The feature "overall_domain_sum" represents the total number of cognitive assessment tests in which a patient did not achieve adequate scores across various cognitive domains (such as attentional, executive, visuospatial, memory, and language). It is a cumulative measure reflecting the severity of cognitive impairment. Analysing this feature can be interesting because it helps to quantify the extent of cognitive deficits in patients with Parkinson's disease. A higher overall domain sum indicates more significant cognitive impairment, which could impact the patient's daily functioning and quality of life. It can also help in understanding correlations with other clinical measures, treatment responses, and overall disease progression, aiding in personalised medical decision-making.

**Example 4:** Feature explanation using the LLM.

5. Implementation

The dashboard was implemented as a *React*[2] application using *TypeScript*. Plots were created with the *ObservableHQ*[3] and *D3*[4] libraries to enhance interactivity. All calculations were performed in the TypeScript frontend. The dataset's small size (less than 100 KB) made this feasible. For the heatmap, the standard Pearson correlation was used, and in the scatter plot the k-means++ algorithm was applied. The OpenAI API was used to facilitate usage based on the model "gpt-4o-mini". Replacing the default Language Model integration with an alternative model, most notably a locally hosted one, requires close to no effort, the user only needs to supply a different API endpoint configuration. The chatbot was given a system prompt instructing it as a helpful chatbot for a Visual Analytics dashboard and that it should give short and precise answers.

Additionally, to gain domain knowledge, the chatbot was given the feature descriptions of all features in the data, the Pearson correlations and a JSON file which describes the components and interactions between the different views. The framework was adopted from Zhao et al [4].

Two dedicated functions enable the dynamic visual highlighting of specific cells in the correlation heatmap during chatbot conversations. The LLM is prompted to call these functions by passing relevant feature names as arguments when such highlighting would support answering a user's query. These functions trigger visual updates in the dashboard to guide the user's attention (cf. Figure 2). To ensure safety and stability, no code generated by the LLM is executed directly. Instead, only the two predefined functions can be called, and the output of the LLM is verified to ensure it contains only valid feature references. After each user query, the LLM is also prompted to generate a short, engaging follow-up question, which is then displayed to the user in the form of a 'bubble'.

The *suggest features* function changes the view of the scatter plot and highlights the features on the heatmap. The LLM output needs to be a list of two features. This works robustly with a standard prompt, which strictly defines the format and emphasises that no additional text should be used. Afterwards, the chatbot should explain why the suggested features are interesting, which is shown to the user in the chatbot window (cf. Figure 1).
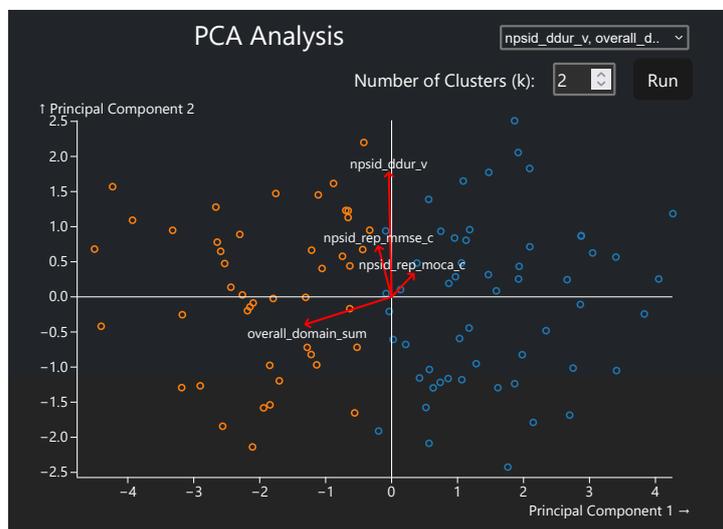
Figure 5: The PCA view allows for the dynamic selection of loadings to show their influence on the principal components. Users can also start the analysis dynamically with a chosen number of clusters.

---

6. Use Case

We demonstrate the applicability of our system on a use case involving the exploration of relationships between attributes of patient's information. This explorative approach is enabled through heatmaps and scatter plots, allowing the identification of linear dependencies. One such observation found through our system might be that cognitive performance declines with increasing age (*insnpsi_age*), while the duration of PD diagnosis has little effect on test scores. Figure 1 illustrates a scatter plot comparing the *overall_domain_sum* (number of cognitive z-tests where patients did not achieve an adequate score) with patient age. It reveals that patients who took the VOSP test exhibit different average scores, which could potentially skew the data. This should be accounted for in subsequent analyses.

In the PCA biplot, users can select features of interest to visualise their influence on the principal components. Scatter plots and histograms are dynamically generated by clicking on off-diagonal and diagonal cells of the heatmap, respectively, with the colour coding selectable via a dropdown menu.

Use of the LLM   We also present an example of how the LLM can assist users in understanding the data and navigating the dashboard. First, the user receives an overview of the dashboard through an initial prompt (see Example 2). With a follow-up prompt (Example 3), the user learns how to change the scatter plot view by interacting with the heatmap. While exploring the data, the chatbot can be used to clarify the meaning of specific features (see Example 4) and to help interpret correlation values. Additionally, through the suggested feature function and example prompts, the user is guided toward potentially interesting aspects of the data. Highlighted cells allow the user to verify insights visually and encourage further interaction with the visualisations.

7. Discussion

7.1. Risks of LLM Integrations

While chatbots and LLMs can be highly useful, their integration also presents certain risks and challenges [7]. These include potentially misleading or incorrect explanations, hallucinations and the risk of users becoming overly reliant on the chatbot's output rather than engaging with the visualisations directly.

One advantage of integrating an LLM into a Visual Analytics dashboard is that users can cross-check the chatbot's responses against the underlying data and visualisations. Additionally, interactions with the chatbot can dynamically influence the visualisations, for example by highlighting a relevant cell in the correlation heatmap. This can help users to verify results and encourage them to engage with the visual interface rather than relying solely on textual explanations.

Currently, the scope of visualisation manipulation through chatbot interaction is limited. Expanding the range of supported interactions could help mitigate these risks further. However, while such enhancements may reduce the impact of these issues, they cannot eliminate them entirely. Users must remain aware of the limitations and capabilities of LLMs when interpreting their outputs.

7.2. Limitations

The current LLM integration can help users gain a broad overview of the visualisations and interactions (see Example 2). In its current version, it does not have direct access to the raw data and therefore cannot provide detailed information about the visualisations. However we do provide statistical properties about the individual columns and correlations between the different features.

The dashboard is designed using standard visualisations to help identify correlations and offer an overview of the data. However, it lacks more advanced, application tailored tools such as zooming, filtering or brushing, which are necessary for more detailed analysis. Overall, the examples in this work show promising results, however their effectiveness was not explicitly validated in user studies.

Reliance on Good Prompting: The LLM integration is mainly focused on the chatbot interface, which requires users to write effective prompts. While we provided some prompt guidance by suggesting follow-up prompts and an initial prompt, the user is still responsible for writing good prompts.

Reproducibility of LLMs: In our paper, we evaluated only the model "gpt-4o-mini", and did not evaluate the reproducibility of prompts and changes of answers based on minor changes in the questions, which can be problematic [8, 9]. For more general conclusions about the applicability and usefulness of the tool, more analysis on these concerns would have to be done.

Although the generalisability of the paper is somewhat limited, and the effectiveness of the LLM has to be further evaluated, the paper still shows the potential of LLM integrations for Visual Analytics dashboards and applications.

8. Conclusion & Future Work

In this work, we presented a comprehensive data exploration dashboard integrated with a Large Language Model to make the dashboard more accessible. As our use case focuses on patient data, we also explained how to anonymise it beforehand, resulting in a synthetically generated dataset of 100 entries that matches the original dataset's characteristics, which we then visualise in the dashboard.

In the current state, there is only limited interaction between the chatbot and the dashboard, namely with the *suggest features* and *correlation heatmap highlighting* functions. The latter is done dynamically, where with every prompt it is checked whether highlighting or interacting with the dashboard provides more insight, and the tool then does so if needed.

Despite certain problems, much of the interaction with the tool happens via prompting. As a user guidance measure, we provide prompt suggestions within the dashboard which meaningfully accelerate the dataset exploration process.

The dashboard could be generalised for different datasets, and in a perfectly integrated data environment, the most interesting dataset or data view could be loaded using the LLM. Future work should also evaluate the effectiveness and precision provided by the various available LLMs.

Acknowledgments

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] J. Seo, B. Shneiderman, Knowledge discovery in high-dimensional data: Case studies and a user survey for the rank-by-feature framework, IEEE Trans. Vis. Comput. Graph. 12 (2006) 311–322. doi:10.1109/TVCG.2006.50.

[2] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, M. Streit, Lineup: Visual analysis of multi-attribute rankings, IEEE Trans. Vis. Comput. Graph. 19 (2013) 2277–2286. doi:10.1109/TVCG.2013.173.

[3] M. Chegini, L. Shao, R. Gregor, D. J. Lehmann, K. Andrews, T. Schreck, Interactive visual exploration of local patterns in large scatterplot spaces, Comput. Graph. Forum 37 (2018) 99–109. doi:10.1111/cgf.13404.

[4] Y. Zhao, Y. Zhang, Y. Zhang, X. Zhao, J. Wang, Z. Shao, C. Turkay, S. Chen, Leva: Using large language models to enhance visual analytics, IEEE Transactions on Visualization and Computer Graphics 31 (2025) 1830–1847. doi:10.1109/tvcg.2024.3368060.

[5] M. Hardt, K. Ligett, F. McSherry, A simple and practical algorithm for differentially private data release, in: Guide Proceedings, volume 2, Curran Associates Inc., 2012, pp. 2339–2347. doi:10.5555/2999325.2999396.

[6] M. Hutchinson, R. Jianu, A. Slingsby, P. Madhyastha, Llm-assisted visual analytics: Opportunities and challenges, arXiv preprint arXiv:2409.02691 (2024).

[7] M. R. Morris, Prompting Considered Harmful, Commun. ACM 67 (2024) 28–30. doi:10.1145/3673861.

[8] T. Kosch, S. Feger, Prompt-Hacking: The New p-Hacking?, arXiv (2025). doi:10.48550/arXiv.2504.14571. arXiv:2504.14571.

[9] T. Kosch, S. Feger, Risk or Chance? Large Language Models and Reproducibility in HCI Research, interactions 31 (2024) 44–49. doi:10.1145/3695765.

[10] T. Schreck, S. Lengauer, P. Waldert, B. Kantz, M. Grabner, L. Schilcher, D. Percic, C. van Leeuwen, M. Lissandrini, A. Romanovych, Deliverable 5.1: Visualization components for sequences, networks, text, and high- dimensional data, 2024. doi:10.5281/zenodo.14628086.

Acronyms

DP Differential Privacy. 1

GPT Generative Pre-trained Transformer. 4

LLM Large Language Model. 1

LM Language Model. 7

MWEM Multiplicative Weights Exponential Mechanism. 4

NLIs Natural Language Interfaces. 3

PCA Principal Component Analysis. 4

PD Parkinson's Disease. 1

VA Visual Analytics. 1

VOSP Visual Object and Space Perception. 8