

BrainEx: Visual Exploration and Discovery in Time Series Data

Advisors:

Professor Rodica Neamtu and Professor Erin Solovey

By:

Michael Clements

Andrew Nolan

Jackson Powell



WPI

A Major Qualifying Project
WORCESTER POLYTECHNIC INSTITUTE

Submitted to the Faculty of the Worcester Polytechnic Institute in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science.

August 2020 - March 2021

Abstract

BrainEx is a tool for exploring and comparing sequences in time series data sets with a focus of applications on brain data. BrainEx is the successor of the tools ONEX and GENEX, adding multiprocessing support to improve the tool's performance. Our project seeks to extend BrainEx's graphical user interface with a cluster explorer, general usability improvements, and an integrated exploration pipeline between the cluster explorer, similarity search, and feature exploration tools. These improvements provide researchers with a powerful way to explore time series data. A user study and an expert review confirmed that BrainEx provides time series exploration in ways existing tools do not. The success of this project allowed us to submit a paper to the IEEE VIS conference.

Acknowledgements

The authors wish to thank many people for making this project possible:

Thank you Professor Neamtu and Professor Solovey for your weekly feedback and advice, and all the work you did to help us with the conference submission. You made this MQP experience very rewarding.

Thank you to everyone involved in the initial design and development of the BrainEx user interface, especially Ellery Buntel, Eric Schmid, and James Plante; this project wouldn't be possible without you.

Thanks to Alicia (Ally) Howell-Munson for taking time to talk to us about her use cases for BrainEx.

We would also like to thank everyone who worked on ONEX, GENEX, and Ziheng (Leo) Li for their fundamental work on the BrainEx engine that this tool is built upon. Without the power of BrainEx we would not be able to make such an effective exploration tool. Thanks so much.

This material is based upon work supported by the National Science Foundation under Grant No. 1835307.

Table of Contents

1	Introduction	1
2	Background	1
2.1	Time Series	1
2.2	Dynamic Time Warping	1
2.3	Synchrony Between Time Series	3
2.4	Clustering of Time Series	4
2.5	User Tasks with Clustering Data	4
2.6	BrainEx	4
2.7	fNIRS	6
2.8	Brain Computer Interfaces and Applications	7
3	Methods	7
3.1	Implementation Strategy	7
3.2	Project Approach	7
3.3	Code Structure	8
3.4	Iterative Design Approach	8
3.4.1	Initial medium-fidelity cluster explorer wireframes	8
3.4.2	Second iteration of medium-fidelity cluster explorer wireframes	9
3.5	Target Users	11
3.6	Testing Dataset	11
3.7	Testing Procedure	11
4	Implementation	12
4.1	The Django Server	12
4.2	React	13
5	Initial Evaluation	15
5.1	User Feedback	16
5.1.1	User Characteristics and Outline of the Study	16
5.1.2	Selecting Clusters	16
5.1.3	Exploring Clusters	16
5.1.4	Aesthetic Impressions	17
5.1.5	Guidance and Unclear Elements	17
5.1.6	Usage	18
5.1.7	Useful Information	18
5.1.8	Additional features	19
5.2	Changes to the Design	19
5.2.1	Aesthetic Changes	19
5.2.2	Usability Changes	19
5.2.3	New Features	20
6	Expert Review	21
6.1	Functional Analysis	21
6.1.1	Usage Scenario	21
6.1.2	Dataset Creation	22
6.1.3	Functional Requirements	22
6.2	Expert Evaluation	23

6.2.1	Study Design	23
6.2.2	Questionnaire	23
6.2.3	Participants	24
6.2.4	Limitations of the Study	24
6.3	Results	24
7	Conclusion	27
7.1	IEEE VIS Submission	27
7.2	Future Work	28
7.2.1	User Feedback	28
7.2.2	Synchrony	28
A	Informed Consent Script	32
B	First User Study Interview Script	33
C	Expert Review Questionnaire	35
D	BrainEx User Guide	39
D.1	Preparing Data	39
D.2	Selecting Clusters	39
D.3	Exploring a Selected Cluster	40
D.4	Exploring with Similarity Search	42
D.5	Exploring Similarity Search Results	43
E	Minimum Server Requirement Recommendations	44
F	BrainEx Server Description	45

List of Figures

1	An overview of the BrainEx pipeline. First the time series in the dataset are broken into all possible sequences and then clustered before allowing interactive querying over the representatives of those clusters.	x
2	An example of the pattern matching provided by Dynamic Time Warping where the black lines between the blue and red sequence indicate which point of one was matched to what point of the other. While the right side appears to be a traditional point-wise measure, this is because the patterns match in the non-warped time [1].	2
3	An example of a warped path through the cost matrix created when running DTW over the sequences shown on the axes [2].	2
4	Two time series with in-phase synchrony. Their oscillations are lined up.	3
5	Two time series with anti-phase synchrony. The phase of the time series is shifted half an oscillation from each other.	3
6	Two time series with an arbitrary phase shift. The oscillations follow the same pattern but are shifted slightly.	3
7	An overview of the BrainEx pipeline. First the time series in the dataset are broken into all possible sequences and then clustered before allowing interactive querying over the representatives of those clusters.	5
8	A screenshot of the home page of the BrainEx-Frontend application containing a table of pre-processed datasets. The "Explore Clusters" button takes the user to the cluster visualizations that our project created.	5
9	A screenshot of the page for loading in a dataset to preprocess in the BrainEx-Frontend.	6
10	A screenshot of the page to query similar sequences in the BrainEx-Frontend	6
11	A person wearing an fNIRS cap.	7
12	Initial splash page wireframe which would show overall cluster statistics in the table to advise the user for setting size and length filters on clusters seen on the next page.	9
13	Initial cluster explorer page wireframe which shows the representatives of the clusters matching the size and length filters which can be changed on this page.	9
14	Initial Selected Cluster Explorer page wireframe which shows the sequences of the selected clusters.	9
15	The revised wireframe of the cluster explorer which shows a heatmap table of the feature distributions over the clusters meeting the size and length filter criteria.	10

16	Wireframe of the Selected Cluster Explorer page when the single feature of Channel Name is selected, showing that a bar chart will be shown to indicate the makeup of the cluster with regards to this feature.	10
17	Wireframe of the Selected Cluster Explorer page when two features of Subject Name and Event Name are selected, showing that a heatmap will be shown to indicate the joint distribution of these features within the cluster.	10
18	Wireframe of the Selected Cluster Explorer page when more than 2 features are selected, showing that a parallel axes graph will be shown to indicate the distribution of cluster sequences for and between each feature.	10
19	The BrainEx Engine System Architecture.	12
20	The file structure of the ClusterExplorer and its subcomponents.	13
21	The file structure of the SelectedClusterExplorer and its subcomponents.	13
22	A screenshot of the current ClusterExplorer page with subcomponents labeled by the red text and boxes	14
23	A screenshot of the current SelectedClusterExplorer page with subcomponents labeled by the red text and boxes	15
24	A screenshot of the current SelectedClusterExplorer page with the CSVPopup component open. It can be seen that the workload feature is selected in both the FeaturesToShowTab and the CSVPopup	15
25	A screenshot of the current SelectedClusterExplorer page with the QueryWarningPopup component open. It can be seen that the representative is listed as the selected sequence with a warning about starting a similarity search.	15
26	On the left is the return button on the Cluster Explorer page, note that it is close to the edges. To be consistent with the rest of the application it's padding should more closely resemble the Selected Cluster Explorer's page seen on the right.	17
27	In dark mode, the shadow behind the axes labels obscures the text.	17
28	Despite the values ranging from about 10% to about 30% the colors in this heat map are relatively similar. By scaling the colors from, in this case, from 0 to 30% instead of 0 to 100 the differentiations will be more clear to the user.	18
29	In the initial prototype this black screen was what users were greeted with when they began exploring a specific cluster. The page is fully loaded at this point.	18
30	An example of the user tip and heatmap legend provided when the More Info button is clicked when viewing the heatmap visualization. A more organized tooltip on the heatmap itself can also be seen in the background.	20

31	One added feature is the ability to hover over a sequence in the series view to highlight it and have a tooltip appear in the upper-right of the visualization that describes that sequence's features and start index. The sequences are colored based on the first selected feature User. A legend is provided in the popup opened by the "More Info" button.	20
32	One added feature of the Cluster Explorer Page is a visualization of the distribution of all clusters in the dataset based on the number and length of sequences that they contain.	21
33	When a cluster is selected, the cluster distribution scatter plot visualization changes to a line plot of the selected cluster's representative to describe the general shape of sequences in that cluster.	21
34	The results of the user study.	25
35	An example of what the Cluster Explorer page will look like upon loading.	39
36	The Cluster Explorer page with the workload feature checked.	40
37	An example of what the Selected Cluster Explorer page will look like upon loading.	40
38	An example of the series plot view.	41
39	An example of the heatmap, here colors closer to white have lower values, while darker blue sections have more prevalence.	41
40	An example of the parallel axes graph between three features.	41
41	Here a small rectangle is drawn over the subsequences going through the Workload Easy. Those lines are highlighted in orange, the other lines become grey.	42
42	Similarity Search page opened from the "Similarity Search" button. Opening this page from the Selected Cluster Explorer page automatically populates the Target Sequence, Start Index, and End Index fields based on the selected sequence.	43
43	Explore Results page opened from the "Search" button on the Similarity Search page after the search completes.	43
44	Explore Data page opened from the "Explore Data" button on the Results. This page uses the same components and visualizations as the Selected Cluster Explorer, but operates on feature distributions for the results of a similarity search rather than a cluster.	43

Executive Summary

Introduction

Functional Near-Infrared Spectroscopy (fNIRS) is a neuroimaging technique that allows researchers to track the activity of the human brain. The interest in the data fNIRS collect is massive – by the end of the year 2020, it is expected that there will be nearly 500 papers published on the subject matter [3]. This research has been fueled in recent years by improvements in the equipment available for fNIRS. However, with this massive increase in the quantity of data collected comes new challenges in data analysis.

BrainEx is a general exploration tool which was built to address these new problems and facilitate future analysis of fNIRS data. Built on the foundation of ONEX and GENEX (general purpose tools for exploring time series data [4]), it allows for the real-time search of large datasets by similarity to a target time series. It does this by using a preprocessing phase where time series are sliced into smaller sequences, and then clustered by similarity [5]. This allows for querying that is many orders of magnitude faster than a naive calculation of the similarity of every other sequence [6].

Brain data analysts could benefit from more user-friendly tools to explore time series data. To address this challenge, various teams of students in WPI’s Human-Computer Interaction (HCI) lab have been collaborating on creating a user interface to make BrainEx more usable to analysts who may be unfamiliar with using a command line tool or who would prefer a graphical interface. A team of undergraduates at the WPI HCI lab has already created a prototype tool which would allow the real-time retrieval of the most similar sequences to a time series of interest through a graphical user interface. [7]. The functionality this interface presents, however, is limited.

The goal of our project is to increase the usability and functionality of the BrainEx tool for analyzing time series data. To achieve this goal, we will implement a cluster explorer that empowers analysts to better understand their datasets by exploring their composition and the distribution of features within them. We will also implement an integrated explo-

ration pipeline between the cluster explorer, similarity search, and feature exploration tools. We will then refine these implementations iteratively before conducting a final expert review in hopes of publishing our results to the IEEE VIS conference.

Background

Time Series, Similarity, and Clustering

Time series are a collection of data points, measured over time and indexed in the order they are observed [8]. This type of data is of particular interest due to its ubiquity in health sciences, economics, engineering, and many other fields [9]. It is useful to compute the similarity between two or more series, as this allows analysts to make meaningful comparisons between different time series [10].

Dynamic Time Warping (DTW) is a measure of similarity between two time series. Similarity can be useful for comparing and matching patterns between times series. DTW provides more flexibility than point-wise comparisons since it can match patterns that have a similar shape but may occur over a different amount of time [11]. However, the complexity of the algorithm implementing optimal DTW can be prohibitive as it is $O(n^2)$ for time and space to calculate the matrix for the optimal path [12]. There are numerous versions of DTW that have been designed to approximate the optimal warped path at lower complexities. Some strategies that these versions use include constraining the amount that the path can warp, running DTW over an abstraction of the data that has fewer data points, and using lower bounds to reduce the number of times DTW is run when using it for classification and clustering [12].

Additionally, it can be useful to cluster time series in order to better understand and visualize the data. Clustering based on distance provides a structure to unlabeled data, by sorting it into groups where the similarity between members of the same group is maximized, and the similarity between members of the different groups is minimized [13]. An effective distance based clustering technique can accelerate the computation of best-matching series by doing up-front preprocessing [5].

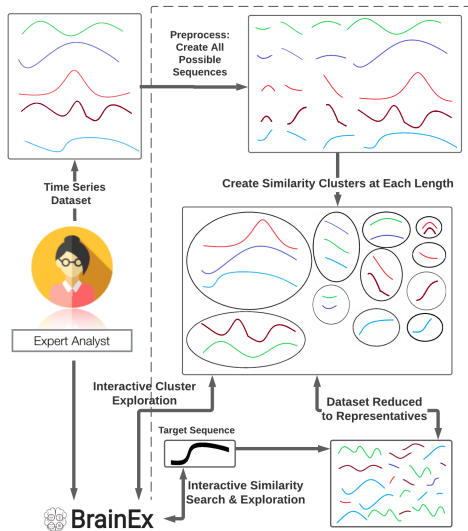


Figure 1: An overview of the BrainEx pipeline. First the time series in the dataset are broken into all possible sequences and then clustered before allowing interactive querying over the representatives of those clusters.

BrainEx

BrainEx is a tool for exploring and comparing sequences in time series datasets with a focus of application on brain data such as that produced using fNIRS [14]. In discussion with Professor Rodica Neamtu, she describes the BrainEx pipeline, which is summarized in Figure 1, as consisting of two main phases, offline preprocessing and interactive querying. In the preprocessing phase, BrainEx loads the time series data, calculates all possible length sequences of each time series, and clusters these sequences into groups of equal length. The clusters group equal length sequences that are similar according to the selected distance measure of Euclidean, Manhattan, or Minkowski distance. In the querying phase, BrainEx finds the k most similar sequences to a target sequence provided by the user analyzing the data. This tool is a culmination of two previous iterations of the tool, ONEX and GENEX, that adds multiprocessing through support for Apache Spark

and Python native multiprocessing [5, 4].

Previous work has created a graphical user interface for the BrainEx tool by implementing an Electron app that uses React JS for displaying the pages that interface with a Django API to access BrainEx[7]. The BrainEx-Frontend currently implements finding k similar sequences to a target sequence, but does not allow exploration of the clusters that BrainEx creates when preprocessing the time series dataset.

Brain Computer Interfaces and their Applications

Functional Near Infrared Spectroscopy (fNIRS) is a noninvasive form of neuroimaging that can provide time series data about the function of a person's brain. The user's brain activity is tracked using infrared light to measure hemodynamics, movement of blood, in the cortex. A relatively new technology, the first fNIRS experiments on humans occurred almost 30 years ago in 1991 [15]. Now, by the end of 2020 scientists expect nearly 500 papers to be published related to fNIRS this year alone [3]. An fNIRS cap is a headpiece with multiple fNIRS instruments on it. The fNIRS cap is accurate, non-invasive, and portable making it useful for researchers [16]. It's growth in popularity can be attributed to these reasons, as well as researchers discovering the effectiveness of fNIRS for understanding the human brain [15].

BrainEx specifically seeks to analyze time series data collected from fNIRS devices. However, fNIRS is just one example of a new type of technology known as brain computer interfaces (BCI). It is important to discuss these technologies in general to better understand where fNIRS and BrainEx fit in. BCI are a new technology within the field of human computer interaction that collects and analyzes cognitive data of a user [17]. Electroencephalography (EEG) and functional magnetic resonance imaging (fMRI) are two other popular examples of BCI technology used for collecting data. Some newer BCI also allow users to directly control computers. In recent research conducted by Ohio State, a BCI and machine learning was used to allow a man paralyzed by a spinal cord

injury to move his hand muscles again [18].

Methods

We define three areas of the code:

- The front end
 - The React and D3 JavaScript code used to display the application.
- The middle end
 - The Django python code used to connect the front end and back end through APIs.
- The back end
 - Python code for the BrainEx engine. Works independently of the front and middle end and can be used via a command line.

The development work done during the course of this project will impact all three parts of the code. New pages will need to be added to the React code on the front end. Any data visualizations will be written using D3 code. The middle end uses Django, any code added there will be in python and serve as API calls to the back end. For the second objective we will be working in the back end code. The back end code will be written in python.

To determine what we should implement for each objective we will follow an iterative design approach. To create visualizations each team member will first create different medium-fidelity wireframes that demonstrate possible workflows and features that the visualization can contain.

For testing the cluster explorer we will have participants complete the tasks of finding disproportionate clusters and identifying trends in the data. We will consider the students of the WPI HCI Lab as our target users for our testing as they work with the current BrainEx tool and would be using our contributions for further research. A team member will note comments about difficulties the test user experiences while completing the set tasks. Through these tests and noting any additional comments, we will propose changes and ask for clarification such that

the next iteration may better meet the users' expectations and needs.

Implementation

To support the functionality of the BrainEx front end, a Django server provides an interface to the BrainEx back end. Both the Django server and back end are implemented in Python3, where Django is a web framework that provides a web request API [19]. As the Django server accepts web requests, then it may be deployed either locally on localhost or on a separate server that the front end can access by changing its IP address variable used in the request URLs. By deploying the Django and back end on a remote server, then one can make use of more memory and machine cores than one's local machine has available. For the ClusterExplorer addition to the BrainEx front end, the following three API methods were added: ListPreprocessedDatasetClusterStatistics, ListPreprocessedDatasetClusterRepresentatives, and SubmitForCluster. These API calls make POST requests to receive the data needed for the cluster explorer.

The BrainEx front end is developed as an Electron app using React. React is a component based JavaScript library for developing user interfaces [20]. The BrainEx user interface and the new cluster explorer are built using this framework. Two new pages were created for our MQP, the Cluster Explorer page and the Selected Cluster Explorer. Within each of these pages we created several additional subcomponents to break the interfaces into logical parts.

Evaluation

Our first round of user testing revealed that the tool was usable at a high-level, but also showed several places where more work was needed. The team used this feedback to target their development work. This targeted development work can be loosely divided into aesthetic issues, general usability limitations, lack of guidance systems, and creation of desired features.

Users, for the most part, considered the aesthetic experience pleasant. Several small changes, such as

inconsistent button padding, were noted and fixed.

There were places where the usability of the application was sub-par, and users struggled to accomplish basic tasks. One such place was in the filtering of clusters in the Cluster Explorer page. Users expressed frustration at the fact that this process was constantly performing calculations, and therefore slow. The team remedied this by requiring users' to confirm their input before the application performs a filter operation on the cluster set. Another place, where users were actually uncertain as to whether the application was working properly, was in the Selected Cluster Explorer, where the blank portion of the screen for the visualization caused users to believe the application was still loading. To remedy this, the team had the application show a visualization by default, so users would know everything was loaded.

Users expressed throughout the trials that they would like more guidance. In general, they expressed difficulty understanding what the visualizations were trying to show, and uncertainty on how to reach certain parts of the application. The team addressed this by creating better tooltips, which allow a user to mouse over certain fields and learn more information about them. Further, the team ensured that naming was consistent throughout the application – especially when referring to BrainEx specific terms like 'length of sequence' or 'size of cluster.' Finally, the team concluded that the creation of documentation would allow many of the areas of concern to be better navigated, since the target audience of this application is not novice users, but specialists.

Finally, users expressed interest in a better way of interacting with the time series view of the application, in order to better parse out the nature of the cluster they were viewing. The team was able to add mouse over compatibility, with a tooltip that gave the feature information for a hovered-over time series. They were also able to highlight the representative sequence and color the other sequences by the first selected feature.

Expert Review

After implementing the changes proposed by our initial user study, we decided to do a more in depth expert evaluation of our system in order to make a publishable paper.

Based on our use cases, we are able to develop *five* major functional requirements for the BrainEx system.

Requirement 1: Similarity Search

- a) Support retrieving and ranking any number of sequences similar to another sequence of an analyst's choice.
- b) Support exploration of search results and attributes of the sequences in the result set.

Requirement 2: Feature Distribution Exploration

- a) Support exploring the distribution of a single feature in a cluster or result set.
- b) Support exploring the joint distribution of two features in a cluster or result set.
- c) Support comparing the relationships between three or more features in a cluster or result set.
- d) Support identifying a sequence shape that well-represents a cluster or result set.

Requirement 3: Cluster Exploration

- a) Support exploring the range of cluster sizes and sequence lengths within clusters.
- b) Support finding clusters of similar sequences with skewed feature distributions (i.e. clusters that mostly contain a particular User, Channel, or Event).

Requirement 4: Integration

- a) Support searching and finding sequences of interest to explore further based on the results of cluster exploration.
- b) Support exporting sequences of interest and other findings to explore further in other tools.

Requirement 5: Accessible to all researchers

- a) Support fast computations, regardless of analyst's computer.
- b) Support researchers at all levels, from novice to expert.
- c) Support diverse experiments to be explored, and remain agnostic to the particular metadata (e.g. participant, events, channels, etc.) that are associated with the dataset.

Our second study was sent to 40 neuroscience researchers and HCI experts who represent our target user base. Of these, 10 responded and participated in our user study. The recruited participants represented a diverse group of target users. The education level and self reported expertise of the participants can be seen in Table 1. Six of the participants have published fNIRS or neuroscience research, and five have published HCI papers. Three of the participants had used BrainEx before. All participants used Chrome or Firefox to complete the study.

Study participants were asked to rank their agreement with statements matching the sub-requirements discussed earlier in the chapter. The Likert scale covered the range of strongly disagree to strongly agree.

The study determined that BrainEx effectively met its functional requirements and, based on user feedback, provided many improvements and new features that existing fNIRS analysis tools do not currently include. A majority of experts agreed that our system fulfills all the individual functional requirements. In most areas, eight or more out of ten participants agreed that our system met the requirements.

Conclusion and Future Work

This project was able to provide significant expansion to the BrainEx front end system. Even as the project comes to a conclusion student researchers have begun using the tool and a paper has been submitted to the IEEE VIS conference. These successes show that BrainEx is a promising tool for time series exploration and analysis.

However, there is always room for improvement. In general, the broad exploration of the datasets' clusters was the weakest functional requirement. The table-driven presentation of the set of all similarity clusters did provide some useful ways for users to gather insights, but was overwhelming at times. Future work on the platform could expand it with additional filters, paging options and multiple options in color encodings to lessen the cognitive workload of exploring a large set of clusters. In general, the manual exploration of a huge number of machine generated clusters, with arbitrarily-many attributes warrants further research. Users were particularly interested in ways to save clusters of interest, or useful filters. They suggested they would like to use these saved values both for use with external tools, and for revisiting parts of their data for deeper understanding.

In the future, BrainEx would also benefit from a feature that allows a user to select two time series sequences and directly compare them to receive a synchrony value. This would function similarly to the similarity search but instead of finding the most similar sequences, it would reveal the underlying similarity value between two sequences. This could be added to the React app through the addition of a new page. The computation to compare just two time series should not require the full power of BrainEx and can be written as functions in the JavaScript code. It could use a sliding window and similarity measurements to calculate the synchrony.

1 Introduction

Functional Near-Infrared Spectroscopy (fNIRS) is a neuroimaging technique that allows researchers to track the activity of the human brain. The interest in the data fNIRS collect is massive – by the end of the year 2020, it is expected that there will be nearly 500 papers published on the subject matter [3]. This research has been fueled in recent years by improvements in the equipment available for fNIRS. Less than 30 years ago, analysts conducted studies where they infrequently recorded measurements of brain activity on a single channel. Today, researchers leverage equipment which can measure 10 or more channels of brain activity, with increasing levels of frequency [15], and associate this data with a variety of other features of the experiment. However, with this massive increase in the quality and quantity of the data collected comes new challenges in data analysis.

BrainEx is a general exploration tool which was built to address these new problems and facilitate future analysis of fNIRS data. Built on the foundation of ONEX and GENEX (general purpose tools for exploring time series data [4]), it allows for the real-time search of large datasets by similarity to a target time series. It does this by using a preprocessing phase where time series are sliced into smaller sequences, and then clustered by similarity [5]. This allows for querying that is many orders of magnitude faster than a naive calculation of the similarity of every other sequence [6]. Before this project the only way to use BrainEx was through a Python API stored in a private Github repository [21].

Brain data analysts could benefit from more user-friendly tools to explore time series data. To address this challenge, various teams of students in WPI’s Human-Computer Interaction (HCI) lab have been collaborating on creating a user interface to make BrainEx more usable to analysts who may be unfamiliar with using a command line tool or who would prefer a graphical interface. A team of undergraduates at the WPI HCI lab has already created a prototype tool which would allow the real-time retrieval of the most similar sequences to a time series of interest through a graphical user interface. [7]. The function-

ality this interface presented, however, is limited. For example, it provided no presentation of the distribution of features over groups of similar sequences.

The goal of our project is to increase the usability and functionality of the BrainEx tool for analyzing time series data. To achieve this goal, we will implement a cluster explorer that empowers analysts to better understand their datasets by exploring their composition and the distribution of features within them. We will also implement an integrated exploration pipeline between the cluster explorer, similarity search, and feature exploration tools. We will then refine these implementations iteratively before conducting a final expert review in hopes of publishing our results to the IEEE VIS conference.

2 Background

2.1 Time Series

Time series are a collection of data points, measured over time and indexed in the order they are observed [8]. This type of data is of particular interest due to its ubiquity in health sciences, economics, engineering, and many other fields [9]. In all these fields, it is useful to compute the similarity between two or more series, as this allows analysts to make meaningful comparisons between different time series [10].

There are many well-cited similarity measures for time series. These range from simple point-wise measures (such as the Euclidean or Manhattan Distance) to more complicated elastic measures (such as Edit Distance with Real Penalty and Longest Common Subsequences [10].) Euclidean Distance can be evaluated in linear time, making it attractive for saving on computation. On the other hand, since the mapping of two time series is fixed, a misalignment or phase shift of one of the two sequence can make them appear dissimilar.

2.2 Dynamic Time Warping

Dynamic Time Warping (DTW) is a measure of similarity between two time series. Similarity can be useful for comparing and matching patterns between

times series. DTW provides more flexibility than point-wise comparisons since it can match patterns that have a similar shape but may occur over a different amount of time [11]. This flexibility is provided by stretching and shrinking parts of one sequence to minimize some distance measure, such as Euclidean Distance, between different points [11]. An example of this stretching and shrinking can be seen in the left half of Figure 2, the first segment of the red sequence is shrunk to match the blue sequence before the first bump. The red sequence is then stretched to match the blue sequence between the second and third bumps. Through these adjustments, the flat and curved parts of the sequences may be matched to their counterpart in the other sequence, minimizing the Euclidean distance between the matched points.

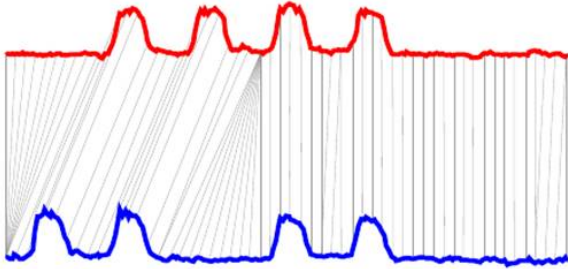


Figure 2: An example of the pattern matching provided by Dynamic Time Warping where the black lines between the blue and red sequence indicate which point of one was matched to what point of the other. While the right side appears to be a traditional point-wise measure, this is because the patterns match in the non-warped time [1].

One way to implement DTW is through a dynamic programming algorithm that creates a warped path through a matrix, indicating which points should be matched between the sequences to minimize the distance [11]. An example of such a matrix is shown in Figure 3 which plots the warped path for the sequences on the axes of the matrix. The shape of this path describes that Sequence B was stretched for its first and last points which were each matched respectively to the first and last three points of Sequence A

and shrunk in the middle of its sequence for 4 of its points to match one of Sequence A's points.

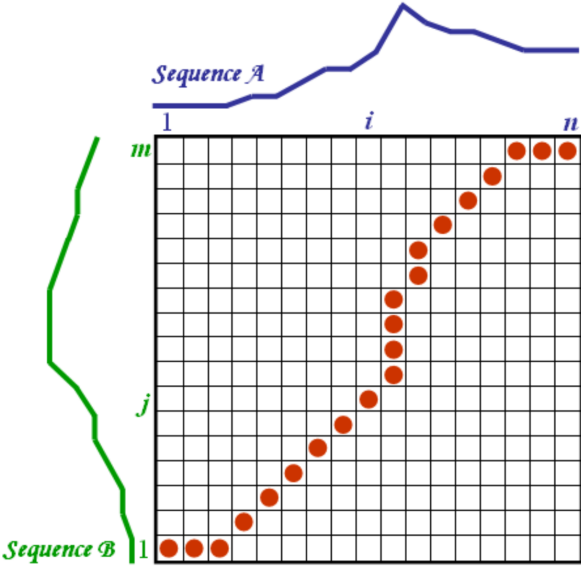


Figure 3: An example of a warped path through the cost matrix created when running DTW over the sequences shown on the axes [2].

However, the complexity of the algorithm implementing optimal DTW can be prohibitive as it is $O(n^2)$ for time and space to calculate the matrix for the optimal path [12]. There are numerous versions of DTW that have been designed to approximate the optimal warped path at lower complexities. Some strategies that these versions use include constraining the amount that the path can warp, running DTW over an abstraction of the data that has fewer data points, and using lower bounds to reduce the number of times DTW is run when using it for classification and clustering [12].

Another problem with DTW is that it only uses Euclidean Distance as a distance measure for the warped path. While this is a standard distance measure, different domains may be better suited to other measures such as Manhattan or Minkowski Distance [22]. GDTW is a generalization of the distance measure used for DTW developed by modeling the dis-

tance cost as being able to use a recurrence function to calculate the distance, which can be done for both Manhattan and Minkowski distances [23].

2.3 Synchrony Between Time Series

Synchronization describes the existence of a relationship between the states of two systems. There are many different ways to measure synchrony between time series. Generally these measurements produce a numeric value in a range, usually between 0 to 1. With larger values implying more synchronization and lower values insinuating increased independence [24]. Synchrony can be measured through both linear and non linear algorithms. The most popular, and simplest, synchrony measurement is linear cross correlation [25]. Other examples include mutual information, transfer entropy, and nonlinear interdependence [25].

There are four basic ways to categorize synchronous behavior: in-phase synchrony, anti-phase synchrony, synchrony with a phase shift, and no synchrony. In-phase synchrony is the easiest to notice, this synchrony shows time series behavior with no phase shifts. As seen in Figure 4, when two time series are in-phase synchronous, they directly overlap each other [24]. Conversely, when two time series have anti-phase synchrony, they reach the maximum and minimums of their oscillations at opposite times as shown in Figure 5 [24]. Anti-phase and in-phase synchrony are the clearest examples, however any phase shift can still be synchronous as shown in Figure 6 [24]. Synchrony provides an interesting measurement because phase shifted time series will likely have lower measures of similarity, but since synchrony can account for these shifts, they will still measure high values of synchrony.

Understanding synchrony between time series can provide many insights that similarity cannot. In particular for brain researchers who may be using BrainEx, we are specifically interested with in-phase synchrony between time series in several scenarios. We want to explore and measure synchronous brain data during collaborative tasks. The end goal is to use this synchrony to improve collaboration during human robot interaction. Based on conversation

with Professor Solovey, there are four experiments researchers using the BrainEx tool will want to conduct: measuring the synchrony between two humans collaborating to play a game. Second, we want to measure the brain synchrony between two humans working side by side, but independently. Thirdly, we want to measure the synchrony between two humans collaborating with a robot. And finally, the synchrony of one human collaborating with a robot. In the first two cases we expect to see different levels of synchrony. Synchrony should be higher when working collaboratively.

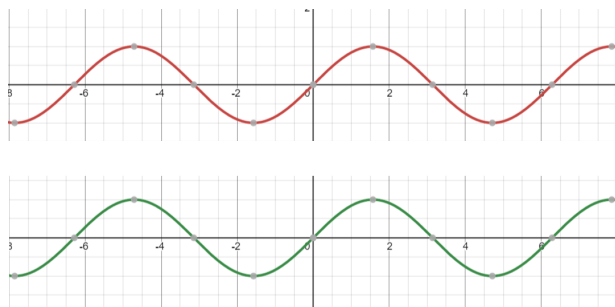


Figure 4: Two time series with in-phase synchrony. Their oscillations are lined up.

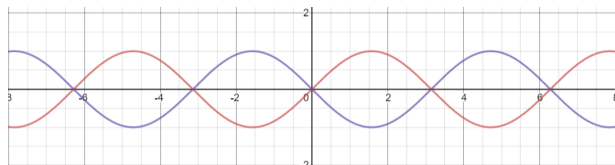


Figure 5: Two time series with anti-phase synchrony. The phase of the time series is shifted half an oscillation from each other.

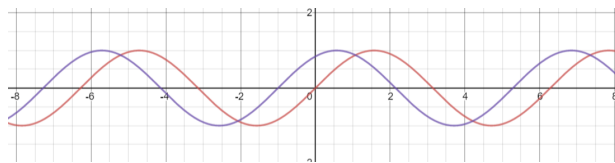


Figure 6: Two time series with an arbitrary phase shift. The oscillations follow the same pattern but are shifted slightly.

2.4 Clustering of Time Series

It can be useful to cluster time series in order to better understand and visualize the data by grouping similar time series together. Clustering based on distance provides a structure to unlabeled data, by sorting it into groups where the similarity between members of the same group is maximized, and the similarity between members of the different groups is minimized [13]. An effective distance based clustering technique can accelerate the computation of best-matching series by doing up-front preprocessing [5].

One of the most well-known clustering methods is k-means clustering. K-means clustering is an unsupervised clustering algorithm used to partition data into k clusters. To accomplish this k-means clustering assigns data to the cluster whose mean (the center of the cluster) is most similar [26]. Alternatively, an algorithm can specify the maximum allowable distance between an element of the cluster and its representative. This is desirable in domains where the researcher might know the general level of similarity they are interested in, rather than the structure of the specific dataset. Whereas k-means clustering, holding k constant, will produce clusters of varying densities depending on the underlying data, techniques like DBSCAN, holding the similarity threshold constant, will produce many clusters for sparse data, and few clusters for densely packed data [13].

2.5 User Tasks with Clustering Data

Visualizations are tools designed to help make data easier for a user to understand. Different types of visualizations are used to accomplish different tasks. Understanding these tasks is crucial to design, as they are what align the motivations of the users with the motivations of the developers. Doctor Tamara Muzner, in her book *Visualization Analysis and Design* classifies these tasks as analysis, search, production, and querying [27].

Analysis-based user tasks can be well-understood as being either discovery or presentations based. A user might take an existing hypothesis, or model, and try to explore the model to find evidence either sup-

porting or refuting their claim. Or, by serendipity, they may discover the suggestion of a new hypothesis that further exploration can confirm or deny. On the other hand, users may already know what their data is saying, and they may seek to leverage the power of their visualization tool to communicate their knowledge to a new audience.

Data visualization tasks can also be acts of production. A user might use a software system to annotate unlabeled data. Or, they might use a visual space to explore possible derivations of existing data. A user might want to see what the structure of the data is like after doing various transformations. When they are done with these transformations, they have produced a novel representation from existing data.

There is also the broad space of user-tasks which fall under types of search. These include 'lookup,' where a user is seeking to find a piece of information they already know the location of and 'browsing' where a user wants to look through a subset of the data they've specified. Finally, the lowest level of user goal tends to fall under the category of querying. A good visualization is designed to allow the user to identify information about a single datum. It may also allow them to quickly compare information about two pieces of data, or even summarize the whole dataset.

2.6 BrainEx

BrainEx is a tool for exploring and comparing sequences in time series datasets with a focus of application on brain data such as that produced using fNIRS [14]. In discussion with Professor Rodica Neamtu, she describes the BrainEx pipeline, which is summarized in Figure 7, as consisting of two main phases, offline preprocessing and interactive querying. In the preprocessing phase, BrainEx loads the time series data, calculates all possible length sequences of each time series, and clusters these sequences into groups of equal length. The clusters group equal length sequences that are similar according to the selected distance measure of Euclidean, Manhattan, or Minkowski distance. In the querying phase, BrainEx finds the k most similar sequences to a target sequence provided by the user analyzing the

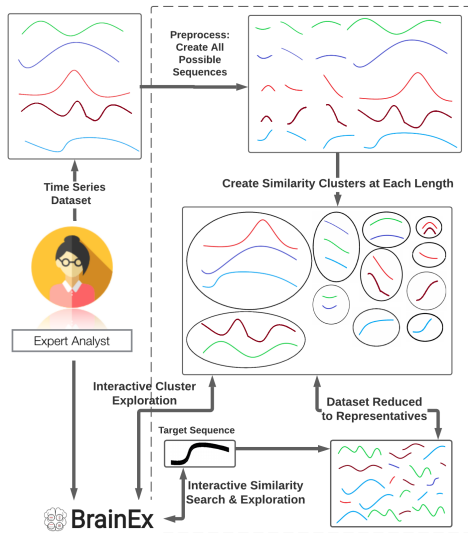


Figure 7: An overview of the BrainEx pipeline. First the time series in the dataset are broken into all possible sequences and then clustered before allowing interactive querying over the representatives of those clusters.

data. This tool is a culmination of two previous iterations of the tool, ONEX and GENEX, that adds multiprocessing through support for Apache Spark and Python native multiprocessing [5, 4].

ONEX implemented the preprocessing pipeline that creates clusters of sequences that can then be queried quicker for sequences most similar to a target sequence using DTW [5]. This preprocessing allows for one-time time-intensive work of the clustering to be done offline and many-use fast queries of the data by users exploring the similarities of sequences within the preprocessed dataset. Such interactive querying is possible as there exists a triangular inequality for the representatives of the clusters, allowing high accuracy while reducing the time to find similar sequences by only using the representatives for comparisons to the target sequence [5]. By allowing for quicker, accurate querying, users of the tool may explore the similarity of time series more easily and interactively. GENEX expanded on ONEX by implementing generalized dynamic time warping, allowing

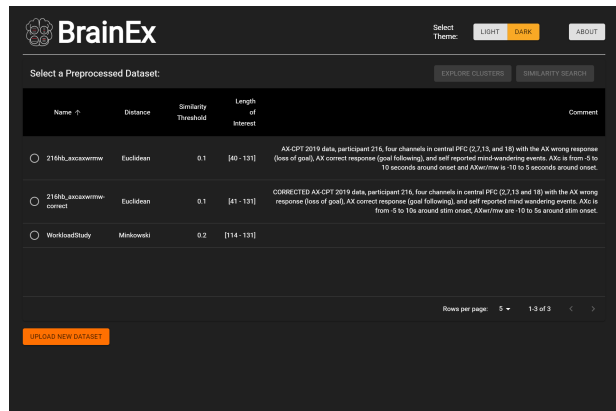


Figure 8: A screenshot of the home page of the BrainEx-Frontend application containing a table of preprocessed datasets. The "Explore Clusters" button takes the user to the cluster visualizations that our project created.

for Manhattan and Minkowski distances to be used in addition to Euclidean distance, while still retaining the clustering pipeline with the triangular inequality properties of ONEX [4].

Previous work has created a graphical user interface for the BrainEx tool by implementing an Electron app that uses React JS for displaying the pages that interface with a Django API to access BrainEx [7]. Before this project, the BrainEx-Frontend could find k similar sequences to a target sequence, but did not allow for exploration of the clusters that BrainEx creates when preprocessing the time series dataset. The main page of this interface is the home page seen in Figure 8 from which the user can preprocess a dataset, query similar sequences, or use our project's additions to explore the clusters of already preprocessed datasets listed in the table.

The first step in a user's interaction with BrainEx would be to preprocess a dataset so that they can query and explore it later more interactively. To preprocess a dataset, the user would click on the "Upload New Dataset" button, bringing them to the page seen in Figure 9 where they would load a .csv or .tsv file and set the parameters for the preprocessing phase.

Once the dataset is preprocessed, then it may be

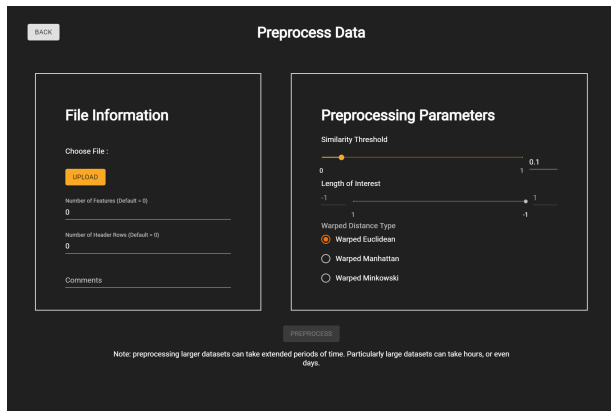


Figure 9: A screenshot of the page for loading in a dataset to preprocess in the BrainEx-Frontend.

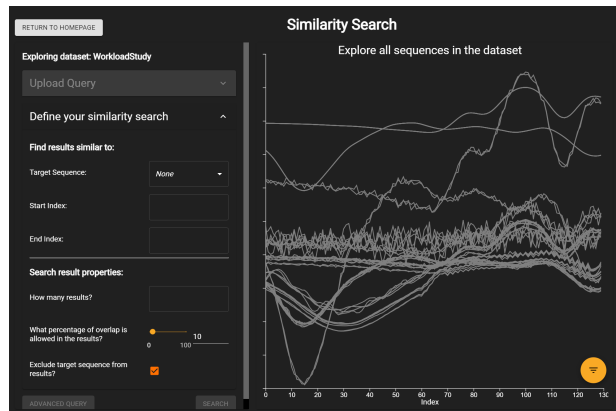


Figure 10: A screenshot of the page to query similar sequences in the BrainEx-Frontend

selected on the home page and queried by clicking the "Similarity Search" button, which becomes enabled when there is a dataset selected. This button brings the user to the page seen in Figure 10 where the user selects the sequence to query and other parameters to query some number of most similar sequences.

This previous work provides a baseline for our project to add to, as well as a reference for how to design our additions to keep the interface consistent for the user. For example, we used the same style for table pagination and sliders for numeric inputs that are used on the home and preprocessing pages respectively. Additionally, we matched the layout of the query page of using a left side panel for input and a larger right panel to show the results of that input where we also used colors consistent with the chosen overall color scheme of the application.

2.7 fNIRS

Functional Near Infrared Spectroscopy (fNIRS) is a noninvasive form of neuroimaging that can provide time series data about the function of a person's brain. The user's brain activity is tracked using infrared light to measure hemodynamics, movement of blood, in the cortex. A relatively new technology, the first fNIRS experiments on humans occurred almost 30 years ago in 1991 [15]. Now, by the end of

2020 scientists expect nearly 500 papers to be published related to fNIRS this year alone [3]. An fNIRS cap can be seen in Figure 11 as a headpiece with multiple fNIRS instruments on it. The fNIRS cap is accurate, non-invasive, and portable making it useful for researchers [16]. Its growth in popularity can be attributed to these reasons, as well as researchers discovering the effectiveness of fNIRS for understanding the human brain [15].

fNIRS relies on the fact that infrared light can penetrate human skin and is absorbed in different amounts by blood vessels dependent on the oxygenation of the blood. By placing an fNIRS scanner on the users head, preferably unobstructed by hair, the researcher can collect time series data about activity in varying portions of the brain. As fNIRS technology developed further, data could be collected from multiple parts of the brain simultaneously, each as its own time series. These measurements allow researchers to compare activity in different spots in the brain at the same time [15]. BrainEx was originally developed to focus on comparing these time series but has been expanded to be a more general-purpose time series exploration tool [6].

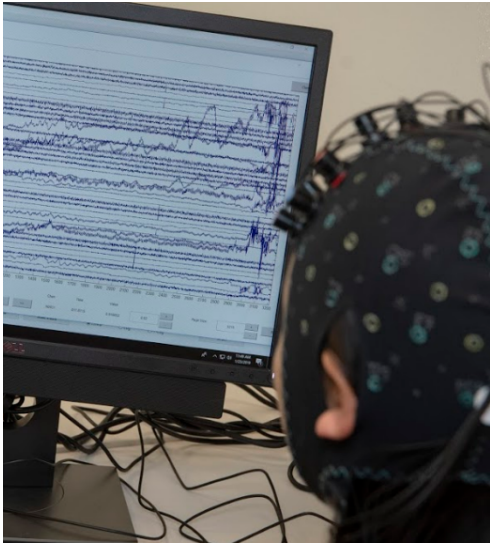


Figure 11: A person wearing an fNIRS cap.

2.8 Brain Computer Interfaces and Applications

BrainEx specifically seeks to analyze time series data collected from fNIRS devices. However, fNIRS is just one example of a new type of technology known as brain computer interfaces (BCI). It is important to discuss these technologies in general to better understand where fNIRS and BrainEx fit in. BCI are a new technology within the field of human computer interaction that collects and analyzes cognitive data of a user [17]. Electroencephalography (EEG) and functional magnetic resonance imaging (fMRI) are two other popular examples of BCI technology used for collecting data. Some newer BCI also allow users to directly control computers. In recent research conducted by Ohio State, a BCI and machine learning was used to allow a man paralyzed by a spinal cord injury to move his hand muscles again [18].

The data collected from BCI can be analyzed for many different applications. Medical technologies as we saw above but also military training, safety simulation and training for airline pilots and operators, and education all benefit from research and use of BCI technology [16, 28]. As previously mentioned,

BrainEx was developed as a tool to analyze data from fNIRS BCI. Based on conversations with Professor Rodica Neamtu, our work with BrainEx will be to encourage future work with human robot interactions.

3 Methods

3.1 Implementation Strategy

The goal of our project is to increase the usability and functionality of the BrainEx tool for analyzing time series data. To achieve this goal, we will implement a cluster explorer that empowers analysts to better understand their datasets by exploring their composition and the distribution of features within them. We will also implement an integrated exploration pipeline between the cluster explorer, similarity search, and feature exploration tools. We will then refine these implementations iteratively before conducting a final expert review in hopes of publishing our results to the IEEE VIS conference. The methods for the final expert review are described in Section 6 and the questionnaire used to conduct the review can be found in Appendix C.

3.2 Project Approach

The team has several objectives to complete in order to expand the BrainEx system. These include developing the cluster explorer and the integration pipeline. In order to avoid stretching development efforts too wide, the team will be addressing the objectives one at a time before progressing to the next phase. This approach will allow us to provide a more accurate timeline for completing each phase. It will also allow progress to be better tracked and continued in the event we run into a bottle neck and cannot complete all objectives. If the cluster explorer took too long we would not be left with a partially complete integration pipeline.

This phased approach will be accomplished in an iterative way. Each objective will be treated as a phase. Thus, we plan to have two main phases, implementing the cluster explorer lasting two terms, and implementing the integrated exploration pipeline and

their refinement taking one term. These phases will be broken down in an iterative way. Although it will be called one phase, the cluster explorer implementation will undergo at least three iterations of development and user review, allowing for it to be designed and improved several times. These phases will be capped on either side by a planning meeting and a retrospective. At the end of each iteration we should have a new deliverable that we are able to present to the advisors. This approach will allow us to focus our efforts on each objective and accomplish them in a reasonable order.

Beyond the structure of the phases the team will have regularly scheduled meetings to ensure things are progressing and to prevent blocking issues. Every Tuesday the team will meet with the advisors to update them on project progress and to ask questions. Immediately afterwards the team will meet to recap the advisor meeting and establish goals for the week. On Fridays, the team will meet to check in on weekly tasks and complete group work. On Mondays the team will meet to recap the week and prepare for the advisor meeting. Finally, on days the team does not meet, team members will check in via text to discuss daily progress and blocking issues.

3.3 Code Structure

We define three areas of the code:

- The front end
 - The React and D3 JavaScript code used to display the application.
- The middle end
 - The Django python code used to connect the front end and back end through APIs.
- The back end
 - Python code for the BrainEx engine. Works independently of the front and middle end and can be used via a command line.

The middle and front end code exists in the same git repository [7]. The back end exists in its own location [21].

The development work done during the course of this project will impact all three parts of the code. Both phases are about extending the BrainEx front end. This development will involve coding in the middle and front ends. New pages will need to be added to the React code. Any data visualizations will be written using D3 code. The middle end uses Django, any code added there will be in python and serve as API calls to the back end. For phase two we will also be working in the back end code. The back end code will be written in python.

3.4 Iterative Design Approach

To determine what we should implement we will follow an iterative design approach. To create visualizations, each team member will first create different medium-fidelity wireframes that demonstrate possible workflows and features that the visualization can contain. To advise these designs, we will informally discuss the user tasks with our advisors so that we can understand what information would be useful and how that information should be organized and displayed.

3.4.1 Initial medium-fidelity cluster explorer wireframes

We created our initial wireframe design based partially on the previous work wireframing the cluster explorer in [29], but with three screens instead of one. The first of these pages is a splash page (Figure 12) that users would arrive at after selecting the preprocessed dataset that they wanted to explore the clusters of. On this page users could view the overall statistics for the min, max, and average size of clusters containing sequences of a given length. Using this information, they could set the filters on the size and length of clusters they want to explore further before clicking a button to do so.

Once the button was clicked on the splash page, the user would be brought to a cluster explorer page (Figure 13) for exploring the range of clusters matching the size and length criteria set on the previous splash page. Users would explore these clusters by viewing the shapes of the representatives on the graph and

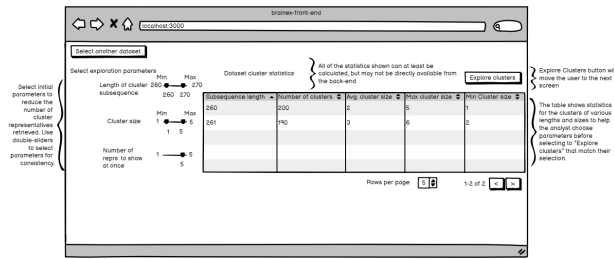


Figure 12: Initial splash page wireframe which would show overall cluster statistics in the table to advise the user for setting size and length filters on clusters seen on the next page.

matching them with the corresponding rows in the table which they can select to investigate individual clusters further. This page also provides the same size and length filters as the splash page, allowing users to further narrow their search without needing to change between pages as often.

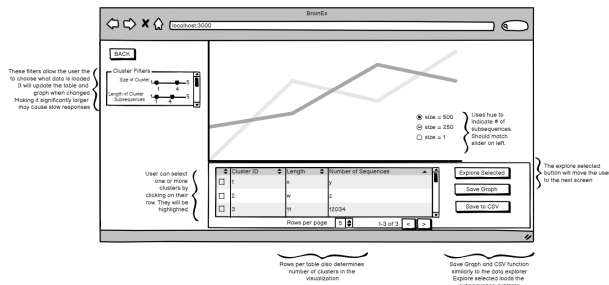


Figure 13: Initial cluster explorer page wireframe which shows the representatives of the clusters matching the size and length filters which can be changed on this page.

After selecting what clusters they wanted to explore and clicking the button to explore them, the user would be shown the Selected Cluster Explorer page (Figure 14) which would show the plot of the sequences within the cluster as well as the table showing the features of those sequences that can be sorted to show the distribution of the features.

After reviewing these wireframes with our advisors, they provided feedback about the lack of usefulness of the splash page and the current methods of visualiza-

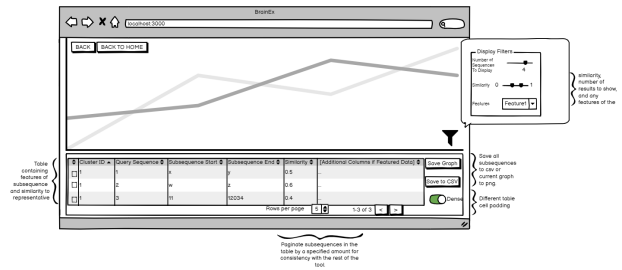


Figure 14: Initial Selected Cluster Explorer page wireframe which shows the sequences of the selected clusters.

tion due to their heavy overlap with the functionality of the Results page of the existing data explorer. Using this feedback, we created a second iteration of the wireframes which we then based our development on.

3.4.2 Second iteration of medium-fidelity cluster explorer wireframes

Our next iteration of the wireframes condensed the splash page with the cluster explorer and focused more on providing visualizations of the feature data rather than duplicating the functionality of the data explorer by only showing plots of the time series. This design consisted of two pages, a cluster explorer to find which cluster to investigate further and a selected cluster explorer to view different visualizations for various sets of features of the data.

The cluster explorer is shown in figure 15 and retains the length and size filters on the clusters shown in the table. The table shows the columns specified by the checkboxes in the lower-left of the figure where size and length are integers and the values for the feature columns are the salience of that feature type for the particular feature. In addition to showing the percents, the table cells for the features will be filled with a color hue that corresponds to the salience of that cell. By using these colors we can provide a quicker way for the user to discover patterns in the feature distributions. Once the user selects a cluster from the table and clicks "Explore Cluster" they are directed to the selected cluster explorer page.

On the Selected Cluster Explorer page, the user

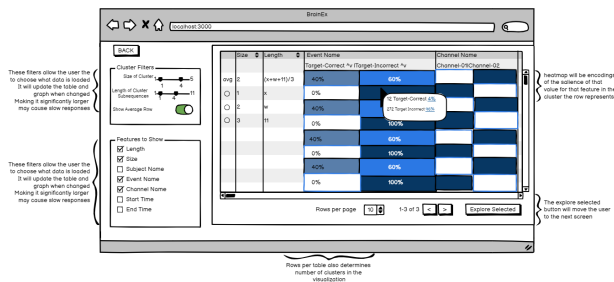


Figure 15: The revised wireframe of the cluster explorer which shows a heatmap table of the feature distributions over the clusters meeting the size and length filter criteria.

is provided with a summary of the representative of their cluster, allowing them to query for similar time series in the data explorer, and with checkboxes to choose which features they want visualized once they click the Update Visualization button. If one feature is selected (Figure 16), then a bar chart will be rendered on the page showing the relative salience of the different feature values.

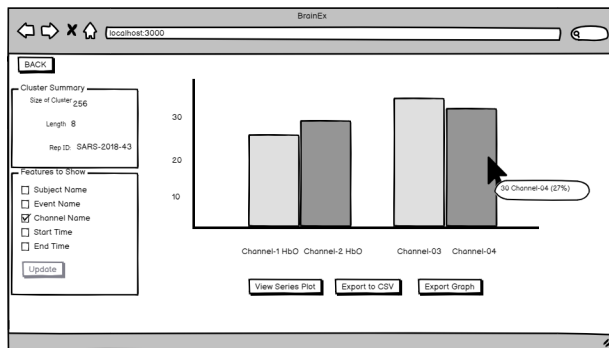


Figure 16: Wireframe of the Selected Cluster Explorer page when the single feature of Channel Name is selected, showing that a bar chart will be shown to indicate the makeup of the cluster with regards to this feature.

If two features are selected (Figure 17), then the visualization is changed to a heatmap showing the joint distribution of the two features selected.

If more than 2 features are selected (Figure 18),

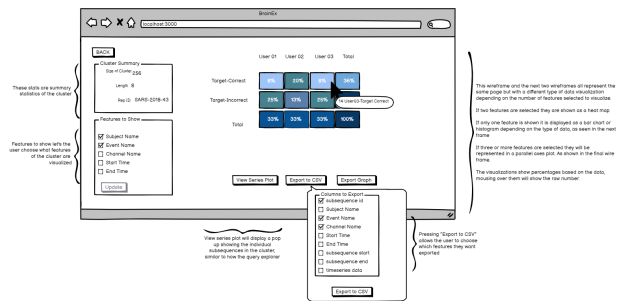


Figure 17: Wireframe of the Selected Cluster Explorer page when two features of Subject Name and Event Name are selected, showing that a heatmap will be shown to indicate the joint distribution of these features within the cluster.

then the visualization is changed to a parallel axes graph showing the trend of the sequence features among the selected features. As features may be categorical, the visualization will contain bar charts in the background to describe the relative densities of sequences for that feature. To further describe the trends, we will also allow for group selection of sequences to highlight the path they take through the graph.

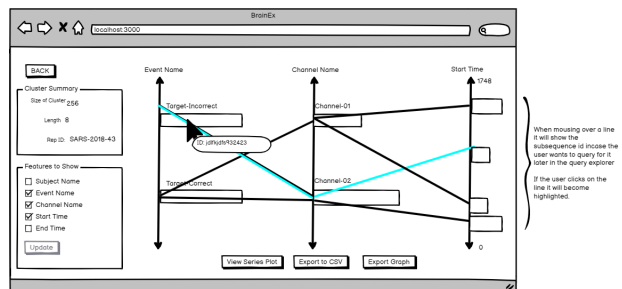


Figure 18: Wireframe of the Selected Cluster Explorer page when more than 2 features are selected, showing that a parallel axes graph will be shown to indicate the distribution of cluster sequences for and between each feature.

The Selected Cluster Explorer page will also contain an option to view the plot of the cluster sequences and export features of the graph as well as

the underlying data if analysts wish to investigate individual cluster data using other tools.

After reviewing this second iteration with our advisors we decided to base our initial development on this set of wireframes which we implemented by adding the D3 visualizations and React components in the BrainEx-Frontend git repository. We also added calculations of statistics and feature distributions to the middle end and back end to decrease the front end computation and thereby increase the responsiveness of the front end visualizations. We made iterative changes to these designs by performing user tests for each iteration and collecting feedback about the design.

3.5 Target Users

BrainEx is a tool that allows for exploration of time series with a focus on brain data. Therefore, the target users of this application are primarily neuroscience researchers, but could also be any researcher interested in performing exploratory analysis of a time series dataset.

Based on discussion with our advisors, we will consider the students of the WPI HCI Lab as our target users for our initial user testing as they work with the current BrainEx tool and would be using our contributions for further research. The WPI HCI Lab is directed by Professor Erin Solovey and consists of 6 graduate and 10 undergraduate students researching various areas of human-computer interaction including a focus on the relevant areas of brain-computer interaction and human-robot collaboration [30]. Due to the lab's focus on these areas, it has been used as a pool of target users for previous work including that which created initial designs of the BrainEx interface which we are extending [29]. By recruiting test users from this lab we are able to test the actual users that will be using our contributions and so will be able to provide the most relevant feedback due to their knowledge and use of the existing tool, its purpose, and the domain.

As we may not always have access to users from the HCI lab for each stage of testing, we will also consult with our project advisors for evaluations of our design as they have detailed knowledge about the

implementation and use of the BrainEx tool.

3.6 Testing Dataset

We will be using a half dataset during development to decrease time spent when loading the data for debugging purposes. However, a full dataset will be used during user testing. For our initial user study, these datasets were created from previous fNIRS experiments and are stored in the back end git repository [21]. Once we decide on a particular set of designs to test with users from the WPI HCI lab, we will use the full dataset so that we account for the effect of large amounts of data on the use of the visualizations.

For the initial user tests, two datasets will be used. The first dataset will be a small dataset created from larger fNIRS data. This dataset is labelled SART2018_ADJUSTED and contains 100 rows and 96 columns. The second dataset will be called 003-SART-August2017-CZA-DC12. This dataset contains 1 row and 16379 data points. This second dataset is the raw feed of one channel during a previous fNIRS experiment. For our second round of user tests, the expert review, we have developed different datasets which we describe in Section 6.1.2.

3.7 Testing Procedure

For testing the cluster explorer we will have participants complete the tasks of finding disproportionate clusters and identifying trends in the data. These tasks are outlined in our interview script which can be found in Appendix B. We will ask the participant to talk about what they are thinking as they complete the task. A team member will note comments about difficulties the test user experiences while completing the set tasks. Using this data, we will be able to identify usability issues from where they mentioned uncertainty in what step to take next. Through these tests and noting any additional comments, we will propose changes and ask for clarification such that the next iteration may better meet the users expectations and needs.

4 Implementation

This chapter provides a more technical and detailed description of the software we developed. It will look at the two main areas programmed, the Django server and the React front end. A general overview of the relationship between the React, Django, and BrainEx engine can be seen in Figure 19. The discussion of the design found in this chapter will focus on how the interface works technically. For an understanding of the design process see the Methods and Evaluation chapters. For an overview of how to use the tool as a user, please see Appendix D, the BrainEx user guide.

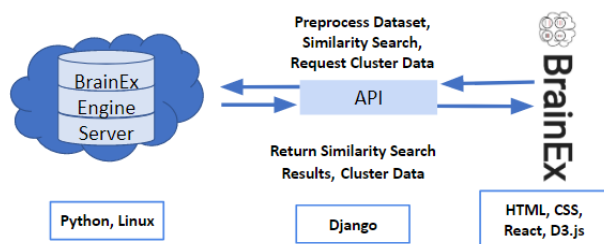


Figure 19: The BrainEx Engine System Architecture.

4.1 The Django Server

To support the functionality of the BrainEx front end, a Django server provides an interface to the BrainEx back end. Both the Django server and back end of BrainEx are implemented in Python 3. Django is a web framework that provides a web request API [19]. Since the Django server accepts web requests, it may be deployed either locally on localhost or on a separate server that the front end can access by changing its IP address variable used in the request URLs. By deploying the Django and back end on a remote server one can leverage the cloud and make use of more memory and machine cores than one’s local machine has available. These additional resources can make more computationally intensive operations, particularly preprocessing datasets, more efficient by utilizing the multiprocessing capabilities of BrainEx that were introduced in GENEX [6]. While more resources will make the application faster, during our

testing we found the number of processors was the primary source of performance improvements. We recommended that a server hosting BrainEx have at least the specifications described in Appendix E for reasonable efficiency and other required attributes.

For the cluster explorer addition to the BrainEx front end, the following three API methods were added to the program: `ListPreprocessedDatasetClusterStatistics`, `ListPreprocessedDatasetClusterRepresentatives`, and `SubmitForCluster`.

`ListPreprocessedDatasetClusterStatistics` receives a POST request with a body containing a hash of the preprocessed dataset to get the statistics for. The statistics returned in the response include the maximum and minimum size of clusters for each possible length of sequences in the cluster. This information is used to provide the bounds on the `DoubleSlider` size and length input filters on the Cluster Explorer page. This method also responds with the average feature distribution over all clusters of the dataset, which the Cluster Explorer page shows in the average row if the show average switch is on, to provide a point of comparison with the distribution of individual clusters.

`ListPreprocessedDatasetClusterRepresentatives` receives a POST request with body parameters containing a hash of the preprocessed dataset to get the representatives for. The response from this method contains a list of all the clusters in the dataset. Each cluster contains its size, sequence length, single feature distribution, and representative’s id, start index, end index, and timeseries data. This information is used to populate the `TableArea` on the Cluster Explorer page.

`SubmitForCluster` receives a POST request with body parameters containing the dataset hash and the representative’s id, start index, and end index. If the request is successful, then the response contains the sequences, single feature distribution, and joint feature distribution for the cluster with the given representative. The sequences list the id, start index, end index, and timeseries data of each sequence and is used to create the series view and parallel axes visualizations on the Selected Cluster Explorer page. The single feature distribution contains the number of sequences, percent of the cluster, and the difference of

this percent from the average percent of the dataset for each option within each feature and for the start index of the sequences. This single feature distribution is used to create the bar chart and histogram visualizations on the Selected Cluster Explorer page. The joint feature distribution contains the number of sequences and distribution of the cluster for each pair of feature values and is used to create the heat map visualization on the Selected Cluster Explorer page.

For the integration pipeline addition to the BrainEx front end, the API method `SubmitForQuery` was extended. This extension adds the single and joint feature distributions of the result set sequences found for the query. These distributions allow for the same visualizations to be provided for a set of similarity search results as for a cluster.

4.2 React

The BrainEx front end is developed as an electron app using React. React is a component based JavaScript library for developing user interfaces [20]. The BrainEx user interface and the new cluster explorer are built using this framework. Two new pages were created for this project, the Cluster Explorer page and the Selected Cluster Explorer. Within each of these pages we created several additional sub-components to break the interfaces into logical parts. These component pages and sub-components can be seen in figures 20 and 21.

The first page in the cluster explorer workflow is the `ClusterExplorer` component. A screenshot of this page can be seen in figure 22 with annotated sub-components. When the page is opened, it first loads the features of the dataset in order to populate the `FeaturesToShowTab` component. The sizes and sequence lengths of the clusters are loaded in to populate the `DoubleSlider` component. The `DoubleSlider` component is rendered inside of the `ClusterFilterTab` component which also takes the average feature distribution of the dataset in order to enable the show average row toggle. Finally, the data for the representatives and features of the clusters are used to populate the `TableArea` and `ClusterExplorerVisualization` components. This data all comes from API calls to the server as discussed in the previous section.

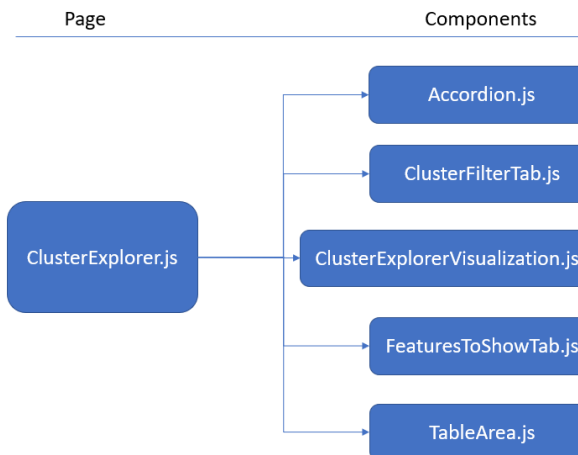


Figure 20: The file structure of the `ClusterExplorer` and its sub-components.

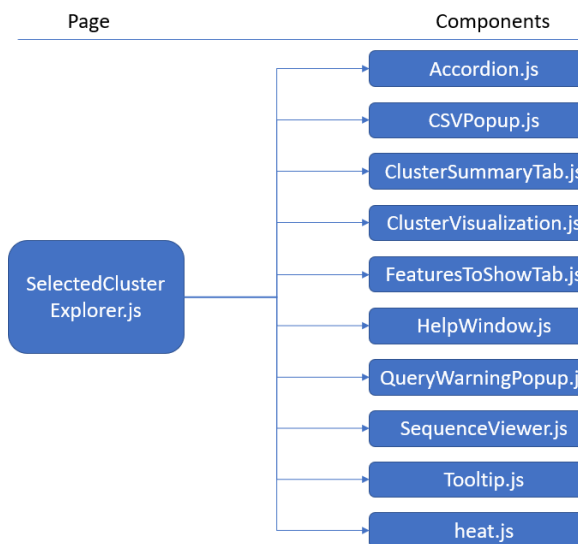


Figure 21: The file structure of the `SelectedCluster Explorer` and its sub-components.

Once the data is loaded, the operations on this page happen solely on the client side. The content in the TableArea component can change in several ways: columns can be sorted, the "Apply Filters" button in the ClusterFilterTab can be pressed to filter the rows of the table, the number of rows to be displayed can be set to 5, 10, or 20, and finally feature checkboxes can be checked to add or remove columns in the table. All of these operations happen on the client side. The rows in the table are selectable. When a row is selected it changes the visualization from a scatter plot of the number of length of sequences in all the dataset's clusters to a line plot of the representative of the selected cluster and enables the "Explore Cluster" button. This button navigates the user to the Selected Cluster Explorer page for the selected row.



Figure 22: A screenshot of the current Cluster Explorer page with subcomponents labeled by the red text and boxes

The second page in the workflow is the Selected-Cluster Explorer component. This page can be seen in figure 23. When a user selects a row to explore from the table in the previous page, the SubmitForCluster API request is made to grab the specific sequence data and statistics for the cluster to be explored. The SelectedCluster Explorer will not load until this data is received. So once the SelectedCluster Explorer is loaded it has all of the data it needs to create the different visualizations. The FeaturesToShowTab in the lower left side of the page is very similar to the

FeaturesToShowTab in the Cluster Explorer. It loads the features as defined by the dataset, plus a feature called Start Index. The Start Index reflects where the sequence starts within the larger time series. The ClusterSummaryTab displays the size and sequence length statistics of the cluster, as well as a representative id. Both of these components rely on data that was loaded and used in the Cluster Explorer.

On the right side of the page is a panel that displays a different component depending on a variety of factors. By default when the page is loaded, the first feature will be selected and a bar chart (or histogram if this feature is numeric) will be displayed. Features that JavaScript detects to have numeric data types will have a "Numeric?" toggle on them. By default this option is enabled. This toggle allows users to specify if the feature should be treated as categorical or numeric (and potentially continuous). If the numeric option is turned on the data will be placed into buckets and displayed as a histogram when it is the only feature selected.

When two features are selected the display will change to be a heatmap displaying the joint distribution of the two features selected. When three or more features are selected a parallel axes plot will be displayed to show the relationships between the features. The bar chart, heatmap, and parallel axes are all handled by a component named ClusterVisualization.

Alternatively, the visualization area can show the Series Plot. When "Plot sequences" is pressed, the cluster visualization is replaced with an interactive plot of the time series sequence data of the cluster. This component also uses D3. For instructions on how to use this tool and the other visualizations, please see Appendix D.

The other buttons on the bottom of the page are rendered on the page itself, not in a subcomponent. The "More Info" button brings up the GenericDialog component as a pop up window with instructions that vary depending on which visualization is displayed. The "Export Graph" button allows the user to save the current visualization to disk. The "Export To CSV" button opens the CSVPopout component, as seen in Figure 24. It allows the user to export the cluster's data to a .csv file for further exploration.

They can select a subset or all of the features available in the dataset. By default the features currently checked in the FeaturesToShowTab are selected in the CSVPopup window as well. The "Similarity Search" button opens the QueryWarningPopup component, as seen in Figure 25. It provides integration from the cluster explorer to similarity search for the user, asking if they would like to perform a similarity search on a listed selected sequence. The sequence is the representative of the cluster by default, but can be changed by clicking on a sequence in the Series Plot. The integration pipeline works in both directions. On the results page of similarity search, the "Explore Data" button brings the user to the SelectedClusterExplorer page, but instead of cluster data it shows the feature distributions and sequences of the result set for the target sequence loaded on the API call to SubmitForQuery.

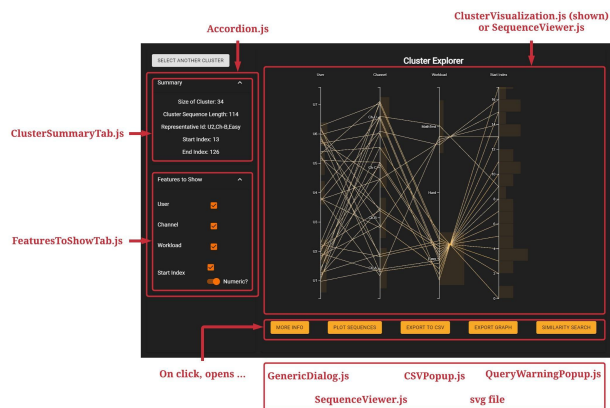


Figure 23: A screenshot of the current SelectedClusterExplorer page with subcomponents labeled by the red text and boxes

5 Initial Evaluation

This chapter examines the outcomes of our first user study. First we recap the design of the study and then we discuss the results of the user study through the lens of our seven followup questions.

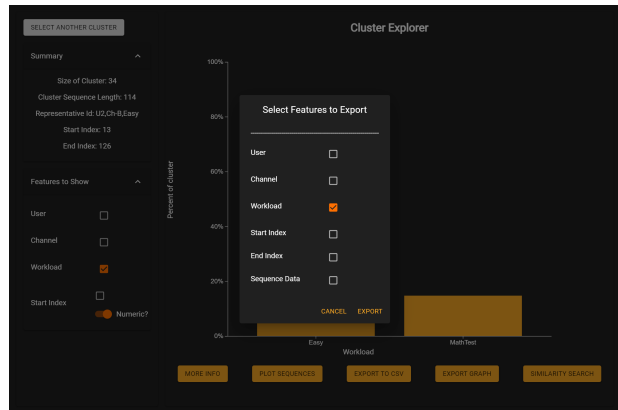


Figure 24: A screenshot of the current SelectedClusterExplorer page with the CSVPopup component open. It can be seen that the workload feature is selected in both the FeaturesToShowTab and the CSVPopup

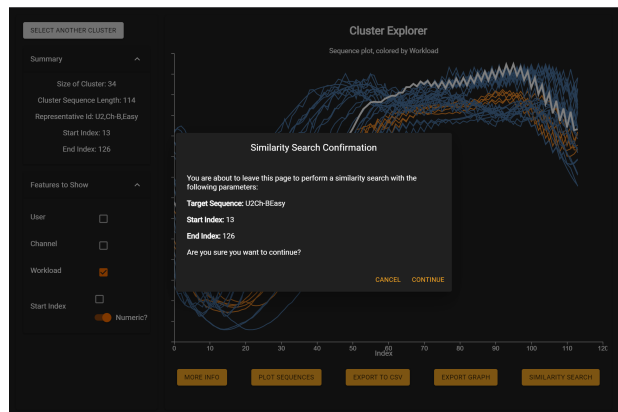


Figure 25: A screenshot of the current SelectedClusterExplorer page with the QueryWarningPopup component open. It can be seen that the representative is listed as the selected sequence with a warning about starting a similarity search.

5.1 User Feedback

5.1.1 User Characteristics and Outline of the Study

As described in chapter 3 of the paper, the team conducted a user study on the prototype cluster explorer UI. The study consisted of users completing four background questions, accomplishing two tasks within the UI, and responding to seven follow up questions reflecting on their experience in the UI. The full script and questions can be found in appendix B. The description of our methodology for these tests can be found in chapter 3.7.

Our initial round of user testing included five participants from the WPI HCI lab. All of the users have experience working with time series and clustering data, but not necessarily visualizations of clusters. The users were all familiar with the concept of BrainEx. Additionally, all of the users worked with BrainEx before to some extent. However, only one user had seen the cluster explorer before this experiment.

5.1.2 Selecting Clusters

Each user rated their satisfaction with the process of identifying a cluster with disproportionate feature representation between a 7 and 8 to give an average score of 7.6 out of 10. Users generally noted that it took a few moments to understand the workflow, but it was easy to use once they understood the relationship between the input sliders and the table. Several users noted that having an update button to apply the slider filters would help clarify their use for finding a cluster to explore. Adding this feature would also help address the other main issue of slow loading times in the table and unexpected behavior when trying to input values. At the time of the user study, as the values of either input sliders are changed, the table updated immediately, causing the page to lag and become unresponsive as the table is repeatedly filtered on intermediate filter ranges. A similar issue also occurs with the text input fields of the filters because the filters immediately apply error checking on their values to ensure that the minimum and maximum selected are valid for the possible range of val-

ues. This error-checking then prevents users from easily changing the value because the filter seems not to respond if the user is trying to take an action that would create an intermediate invalid value, such as an empty field when trying to change all digits of the value.

In addition to the difficulties that users had with the input sliders, they also expressed a desire for a visualization of the overall cluster distribution with regards to size and length as well as visualizations of the shape of the representative for each cluster in the table. During the user study, the distribution of clusters could only be approximated by changing the filter range for the size and length and watching the number of clusters in the range change. However, only using the filters would be quite slow as it would require waiting for all clusters to be filtered repeatedly for each change to the filter. Instead, users suggested that either more overall statistics, such as median in addition to the existing average, be added to describe the distribution or a visualization, such as a scatter plot, be added to show the relationship between the length and size of the dataset's clusters.

5.1.3 Exploring Clusters

Users as a whole rated their satisfaction with the process of identifying relationships within a selected cluster as better than average. However, all users expressed some difficulty understanding what precisely the system was trying to convey. Users identified that the axes labels were not as helpful as they wanted – for example, the bar chart visualization displayed a decimal, rather than percent value. They also noted inconsistencies such as overlapping axes, or long numbers which would be cut off. Further, in visualizations like the heatmap, users did not immediately know what values the color scales represented, and suggested they would like an option to view a legend.

With the parallel axes visualization, users spent a large chunk of time trying to figure out exactly what information was being displayed. While this does indicate that the display may not be perfectly clear, several users expressed that they understood part or all of the display. Given that this was possible in a short session, it is reasonable to assume that

this visualization style will be usable for analysts who spend more time with it. Two users did mention that the parallel axes display showed connections between features, however, rather than distribution between features pairwise. This suggests that only some parts of the information were effectively conveyed.

5.1.4 Aesthetic Impressions

Our third follow up question explored the users' thoughts related to the design of the user interface in terms of aesthetics. The overall impression was positive. The original BrainEx user interface that this project builds upon underwent design and user testing as well. By remaining consistent with the previously developed design the new additions stayed aesthetically pleasing.

Nevertheless, we were made aware of several minor visual issues in the cluster explorer. The buttons within the cluster explorer needed additional padding in several locations (Figure 26). On the Selected Cluster Explorer page the visualizations had several visual issues as well. As previously mentioned, the axes for the graphs sometimes got cut off. Additionally, the color gradient in the heat map was not distinct enough to differentiate between values (Figure 28). In dark mode the shadow of the labels on the parallel axes graph also introduced visual problems (Figure 27). These issues did not affect usability and were easy to address.



Figure 26: On the left is the return button on the Cluster Explorer page, note that it is close to the edges. To be consistent with the rest of the application it's padding should more closely resemble the Selected Cluster Explorer's page seen on the right.

5.1.5 Guidance and Unclear Elements

For most of the users, this was their first time seeing the new cluster explorer interface. For all of the users this user test was their first experience exploring data with the tool. So, some of the unclear elements

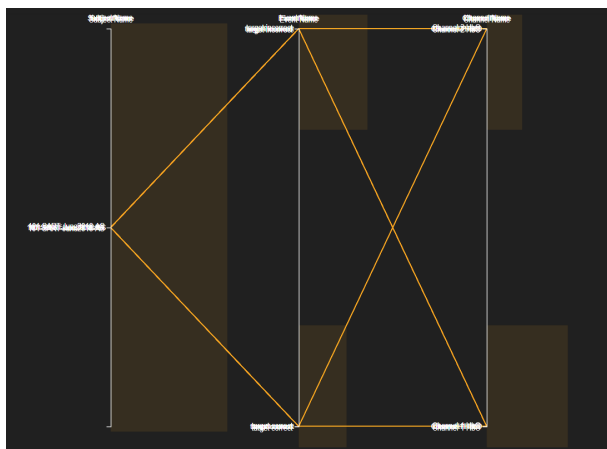


Figure 27: In dark mode, the shadow behind the axes labels obscures the text.

will be addressed in the official release of this tool as part of the final product is a user manual with tutorials and descriptions of the tool. Users of this tool are not expected to be casual users, so some training is expected. Nevertheless, there was confusion over many features in the tool. Even with a guide and training, the tool should be easily understandable and approachable so we will try to address these issues.

One of the first things users noticed and wished for more guidance on was when to begin interacting with the tool. During the user study, the cluster explorer did not include indicators of when the page is loading, nor did it include positive affirmations when the loading is complete. Users exhibited confusion over why nothing was happening (waiting for a load to complete) and over whether they should do anything (load complete). A React component exists for notifications that was used in the previous data explorer section of the BrainEx front end. The team was able to re-purpose this element for the cluster explorer as well.

Despite the concerns discussed earlier, users were able to understand how to search and select specific clusters easily. However, upon selecting a specific cluster to explore, users required more guidance. Related to the loading issue, users were unsure if the initial black screen on the Selected Cluster Explorer

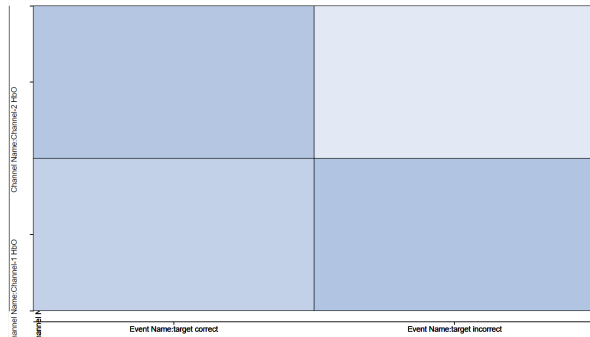


Figure 28: Despite the values ranging from about 10% to about 30% the colors in this heat map are relatively similar. By scaling the colors from, in this case, from 0 to 30% instead of 0 to 100 the differentiations will be more clear to the user.

page was a loading screen (Figure 29). Once the users realized the page was loaded, they began exploring but were often unsure how to interpret the different visualizations. Several users did not initially understand how to interpret the heat maps, and most users had difficulty using and interpreting the parallel axes graph. Even more unclear was how to find the parallel axes graph. Several users required prompting to select more than two features in the Selected Cluster Explorer. Afterwards, the users seemed to like the parallel axes graph but remarked they would not have discovered it on purpose. Finally, there was significant confusion regarding the "Numeric?" switch. Users were not able to understand the use or meaning of this switch. Once it was explained to them, they saw the value of the switch and requested it be turned on by default.

Most of these issues would be resolved if the users received the prior training or tutorials mentioned in the start of this section. The users in this study agreed that having a guide would be beneficial and remedy most of the confusion. However, an additional solution would be to include tooltips and help buttons, a change that would be consistent with the data explorer part of the BrainEx front end. Mousing over different components, such as the "Numeric?" switch, could reveal its use and alleviate confusion for the users. Additionally, help buttons on each

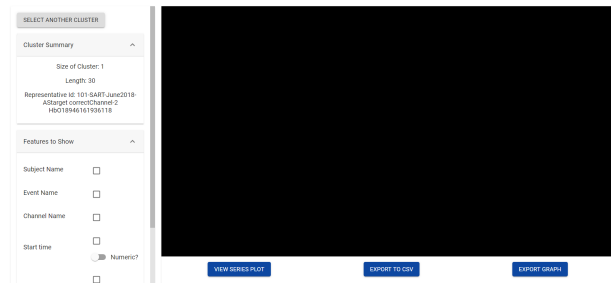


Figure 29: In the initial prototype this black screen was what users were greeted with when they began exploring a specific cluster. The page is fully loaded at this point.

page that recapped sections of the tutorial would allow users to better understand the page and different possibilities. This change could remind users of the parallel axes graph if it's not discoverable on its own. Finally, as mentioned in previous sections, clarifying the axes labels and adding legends to the visualizations will also solve some of the confusion regarding how to interpret the visualizations.

5.1.6 Usage

The team was interested in which features in the cluster explorer would be most useful for BrainEx users exploring clusters in brain data. We were excited to learn users interested in exploring clusters believe that all parts of the app will be useful to their research. One user specifically was interested in using the tool as soon as possible to explore the shapes of clusters in their dataset.

5.1.7 Useful Information

The team was also interested in what information BrainEx users would want access to when exploring clusters. Several users expressed a desire for both the length and size of the clusters to be shown by default in the table for selecting a cluster as both are important for selecting a meaningful cluster and adding features can be a slow operation when waiting for the table to update. One user also expressed interest in seeing the shape of the representative for each cluster when viewing the table to select a cluster so that they

could relate the shape of sequences viewed in the data explorer to possible clusters to explore. Interest was also expressed in the subset of the brain data features of Event Name, Subject Name, and Channel Name, however, due to the agnosticism of BrainEx with regards to the type of data being explored and its features, there may be limited possibilities for adding information about these features.

5.1.8 Additional features

Even before they were prompted to do so, respondents mentioned future features for the application. Several users, when viewing the time series data within a selected cluster, were disappointed that they could not create associations between features and individual plots of brain data. Two users suggested that this would help them tell if these were all the same data, with small changes in start index. Another user stated that they assumed such a feature was in development.

Users also expressed interest in a way to integrate the cluster exploration and the dataset query tool. Particularly, one user said they wished there was a way to copy the necessary parameters from a representative sequence identified in the selected cluster view, so they could retrieve the most similar sequences from the dataset exploration tool.

5.2 Changes to the Design

After completing the initial round of user tests, the team implemented code changes to address the concerns found in the study described in the previous section. These changes can be classified into three categories: aesthetic changes, usability changes, and new features. The following subsections discuss the specific changes implemented to address these concerns.

5.2.1 Aesthetic Changes

While users found the application to be mostly aesthetically appealing, there were a few details they singled out. One such issue was inconsistency padding throughout the application. The team was able to

quickly ensure that all buttons received the exact same size padding. Another such issue was the color scale in the heatmap. Users noticed that, for certain, high-spread data configurations, the differences in color for two heatmap cells could become imperceptible. To remedy this, the team used the local maximum, rather than 100 percent, as the most saturated part of the color scale. Finally, we replaced the odd-looking drop shadow on the axes and adding margins so they were not cut off, ensuring the axes' text is readable.

5.2.2 Usability Changes

The user study participants found the program to be very usable overall once it was explained to them. However, there were concerns expressed over the clarity of several features and users were often confused about how to use parts of the tool. To address this the team implemented several changes to the original design to improve the usability of the system. Tooltips were added to several confusing parts of the cluster explorer. On the selectable features found on both the Cluster Explorer and Selected Cluster Explorer pages tooltips were added to describe the features. A tooltip was added to the average row toggle in the Cluster Explorer to explain what the row did. Finally, tooltips were added to the "Numeric?" toggle on the Selected Cluster Explorer page to describe what the feature did, addressing a common cause of confusion in the study. More tips for the users were added in the form of a "More Info" button on the Selected Cluster Explorer page. This button brings up a pop up window containing a description and legend, if applicable, pertaining to the visualization the user is currently viewing. An example of such a tooltip can be seen in Figure 30 for a heatmap in which a legend is applicable because the range of colors used for the heatmap varies based on the data in the cluster.

During the study, users found several features and options particularly helpful. To help make the system easier to use, these features have been enabled by default. They include the length of sequence feature on the Cluster Explorer page and the "Numeric?" toggle on the Selected Cluster Explorer. Addition-

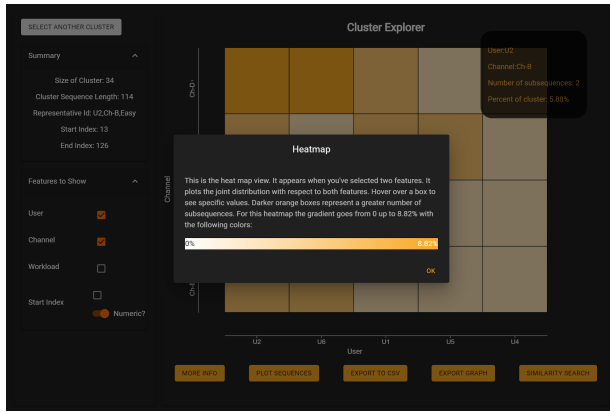


Figure 30: An example of the user tip and heatmap legend provided when the More Info button is clicked when viewing the heatmap visualization. A more organized tooltip on the heatmap itself can also be seen in the background.

ally, to address users’ confusion over the black screen when arriving on the Selected Cluster Explorer Page, the first available feature is enabled by default. This allows the page to display a bar chart when loaded, making the user feel more comfortable to begin exploring.

On the Cluster Explorer page two other changes were made to improve the usability and clarity of the system for. The team updated several labels, which now consistently refer to ‘size of cluster’ and ‘sequence length.’ This will hopefully remove any confusion users may have had about these names. Additionally, the length of sequence and size of cluster sliders have been modified to be more responsive. They now do not update the table until the user presses a button. This improves the responsiveness of the cluster explorer by allowing users to modify the ranges of these features they wish to view without waiting for the table to reload each time they change one number.

5.2.3 New Features

The user study participants noted that they would use all of the current functionality. However, they also sought additional features to integrate the clus-

ter and data explorers and to provide a more general overview of the clusters. At the time of the user study, the primary integration was the presence of the series view which showed the plot of each sequence in the cluster. Participants found this plot somewhat useful to describe the shape of the cluster, but thought that it was still in development and wanted additional information to clarify the distribution of features over the sequences shown. The team added a tooltip and a coloring of the sequences based on the first selected feature to this screen to provide feature information when the user hovers over a sequence. These added features to the series view can be seen in Figure 31.

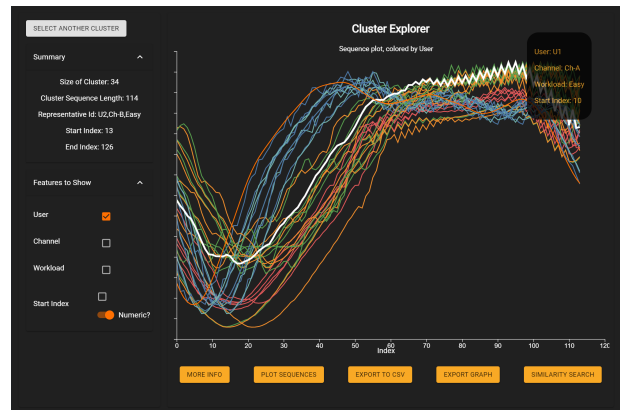


Figure 31: One added feature is the ability to hover over a sequence in the series view to highlight it and have a tooltip appear in the upper-right of the visualization that describes that sequence’s features and start index. The sequences are colored based on the first selected feature User. A legend is provided in the popup opened by the “More Info” button.

The team also added plots of the cluster representatives for each cluster in the table of the Cluster Explorer page and a visualization comparing the size and length of the clusters in a dataset on the Cluster Explorer page. These added visualizations to the Cluster Explorer page can be seen in Figures 32 and 33.

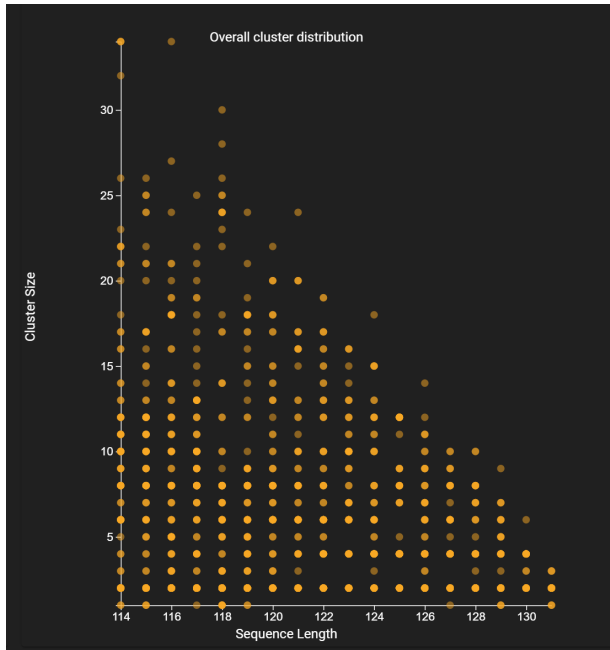


Figure 32: One added feature of the Cluster Explorer Page is a visualization of the distribution of all clusters in the dataset based on the number and length of sequences that they contain.

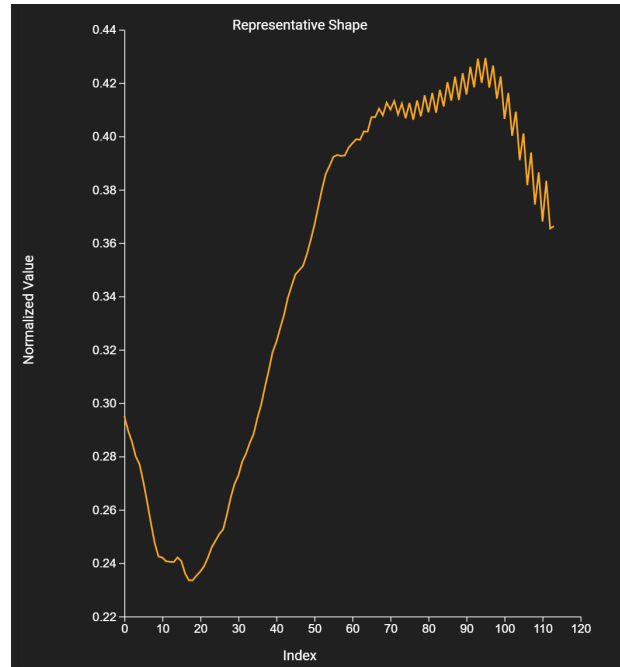


Figure 33: When a cluster is selected, the cluster distribution scatter plot visualization changes to a line plot of the selected cluster’s representative to describe the general shape of sequences in that cluster.

6 Expert Review

After implementing the changes proposed by our initial user study, we decided to do a more in depth expert evaluation of our system in order to make a publishable paper. This chapter examines the outcomes of our second user study. First, we discuss the design of the study and our functional requirements. Then we analyze the results of the user study through the lens of our five functional requirements.

6.1 Functional Analysis

With BrainEx, we set out to bring the powerful advances in time series data mining to fNIRS analytics. However, we first had to identify requirements specific to fNIRS analysis. First, we generated a usage scenario covering a range of potential use cases for an fNIRS researcher using BrainEx. From this, we cre-

ated a list of functional requirements necessary for a system to support this usage scenario.

6.1.1 Usage Scenario

Data generated from fNIRS studies often consists of multivariate time series of brain signals coming from multiple channels. It also contains metadata documenting the participant, the sensor channel, and any events occurring during the experiment session. A common goal of such research would be to better understand the impact of one or more of these attributes on the corresponding time series data. For example, researchers may want to answer questions such as: *What parts of the dataset look the most similar to this particular instance of user distraction? Are there particular patterns that are frequent in the dataset, in general? Are there particular patterns that are frequent for a particular study participant? Or sensor*

location? Or event? Or a combination of these factors? To answer these questions, researchers would not want to assume that all sequences in the dataset are the same length, since real-world tasks can vary. However, the search results should still find the most similar sequence.

We developed the following usage scenario to motivate the design of the BrainEx interface. A neuroscientist has performed a fNIRS study, collecting data from 8 participants and 4 channels. In this study, participants were asked to complete the following tasks: a perceptual task known to be of a high cognitive burden, one known to be of a low cognitive burden, and one task where users were asked to complete a math test of unknown burden. The analyst is interested in better understanding this math test, and would like to use the data produced during the other two calibrated tasks to aid.

Since this dataset contains data of unknown properties, an analyst would need to get a sense of the distribution of the data by understanding which sequences are similar to each other in the dataset. In this case, they would not have any singular sequences of interest in mind. Instead, they would need to explore the entirety of the data.

In order to understand the impact that their observations, such as channel and user task, have on the underlying brain activity, an analyst would need to view the distribution of these data. They would need to understand the distribution of feature labels both on the set of all natural groupings of data, and within any specific grouping which is of interest for them. An analyst may also want to get a sense of the 'shape' of a time series which represents some grouping.

An analyst may also be able to identify a singular sequence whose properties are of interest to them based solely upon their data collection. If they had a specific user who took the math test, they could select the data from one of that user's channels, and see what other sequences behaved similarly to theirs.

In the process of exploring, an analyst might come across a sequence which is of particular interest. It is important to be able to search for any number of sequences similar to the one discovered. Once these are retrieved, they will need to explore the distribution

of the related metadata.

When an analyst has identified any subset of the data, whether by exploration or searching, it is important that they are able to explore the distributions of one or more features of this data.

6.1.2 Dataset Creation

To help us further understand the usage scenario described above, we created a sample dataset. We randomly selected two sequences from a pre-existing fNIRS dataset. We then retrieved similar sequences to each of these, and labelled these clusters as Hard and Easy, with uniformly distributed channel and user labels. We then applied Gaussian noise, and warped the sequences from the Easy category, and labelled the resulting sequences as MathTest. This test dataset would capture a key potential application of BrainEx – the discovery and exploration of a trend in a dataset which is best captured using warped measures of similarity.

6.1.3 Functional Requirements

Based on our use cases, we are able to develop *five* major functional requirements for the BrainEx system. The first requirement is the ability to compare one sequence to the rest of the dataset by finding which sequences it is most similar to.

Requirement 1: Similarity Search

- a) Support retrieving and ranking any number of sequences similar to another sequence of an analyst's choice.
- b) Support exploration of search results and attributes of the sequences in the result set.

The second requirement is the ability to explore the feature distribution in a set of sequences that are naturally grouped together.

Requirement 2: Feature Distribution Exploration

- a) Support exploring the distribution of a single feature in a cluster or result set.
- b) Support exploring the joint distribution of two features in a cluster or result set.
- c) Support comparing the relationships between three or more features in a cluster or result set.
- d) Support identifying a sequence shape that well-represents a cluster or result set.

Similarly, BrainEx should support the ability to explore the distribution of all such natural groupings in the dataset.

Requirement 3: Cluster Exploration

- a) Support exploring the range of cluster sizes and sequence lengths within clusters.
- b) Support finding clusters of similar sequences with skewed feature distributions (i.e. clusters that mostly contain a particular User, Channel, or Event).

From the insights gained by exploration into clusters of sequences and feature distributions, BrainEx should support using these insights to start new explorations and searches on sequences found to be of interest.

Requirement 4: Integration

- a) Support searching and finding sequences of interest to explore further based on the results of cluster exploration.
- b) Support exporting sequences of interest and other findings to explore further in other tools.

In addition, to the functional requirements above, the final requirement was that the tool is not limited to a small group of highly trained researchers with access to high performance computing.

Requirement 5: Accessible to all researchers

- a) Support fast computations, regardless of analyst's computer.
- b) Support researchers at all levels, from novice to expert.
- c) Support diverse experiments to be explored, and remain agnostic to the particular metadata (e.g. participant, events, channels, etc.) that are associated with the dataset.

6.2 Expert Evaluation

6.2.1 Study Design

For this study, participants were invited to use an instance of BrainEx that included a preprocessed dataset containing 8 users with activity in 4 channels to respond to a questionnaire. The questionnaire can be seen in Appendix C. The tool can be found at <http://brainex.wpi.edu:5000/>. The dataset was generated to represent the usage scenario described in Section 6.1.1 using the process described in Section 6.1.2. To ensure the study delivered meaningful insights, we sought to draw on the experience of experts in the fields of fNIRS, data visualization, and/or human computer interaction research. The participants were encouraged to spend time exploring the preprocessed dataset in BrainEx before answering the questionnaire. The participants were also encouraged to refer back to the application while taking the survey. The study was designed to take approximately 45 minutes to an hour to complete.

6.2.2 Questionnaire

The questionnaire was an online survey developed in Qualtrics. The form consisted of three main sections: one for general demographic questions, the largest section focused on the functional requirements defined in Section 6.1.3, and one section for general BrainEx usability and usage questions. The general demographic questions collected background information about the participants' experiences with fNIRS, other brain activity tools, data visualization, and HCI. The functional requirements section of the

Table 1: Participant Demographics

This table provides demographics details for the 10 user study participants. It includes their academic positions and their self-identified expertise in several fields.

Participant	Position	fNIRS	Neural Data Analysis	HCI	Data Visualization
P1	Other	Expert	Knowledgeable	Expert	Knowledgeable
P2	PhD Candidate	Knowledgeable	Passing Knowledge	Knowledgeable	Knowledgeable
P3	Bachelor	Passing Knowledge	No Knowledge	Knowledgeable	Passing Knowledge
P4	PhD Candidate	Knowledgeable	Knowledgeable	Knowledgeable	Knowledgeable
P5	Bachelor	Knowledgeable	Knowledgeable	Passing Knowledge	Knowledgeable
P6	Post Doc	Knowledgeable	Expert	Expert	Knowledgeable
P7	Master	Passing Knowledge	Expert	Expert	Expert
P8	PhD Candidate	Knowledgeable	Passing Knowledge	Knowledgeable	Passing Knowledge
P9	PhD Candidate	Passing Knowledge	Passing Knowledge	Passing Knowledge	Passing Knowledge
P10	PhD Candidate	Knowledgeable	Passing Knowledge	Knowledgeable	Knowledgeable

survey asked study participants to explore a preprocessed dataset through the lens of the usage scenario described in Section 6.1.1. For each of the bullet points within the first four functional requirements, participants were asked to rank how much they agreed with the statement on a 5-point Likert scale. They were also asked to provide any insights they made about the dataset, and any positive or negative comments about their experience completing the requirement. The final section asked the study participants to rate the general usability of BrainEx as well as share other comments about BrainEx.

6.2.3 Participants

Our study was sent to 40 neuroscience researchers and HCI experts who represent our target user base. Of these, 10 responded and participated in our user study. The recruited participants represented a diverse group of target users. The education level and self reported expertise of the participants can be seen in Table 1. Six of the participants have published fNIRS or neuroscience research, and five have published HCI papers. Three of the participants had used BrainEx before. All participants used Chrome or Firefox to complete the study.

6.2.4 Limitations of the Study

The results of this study rely on the subjective responses of the survey participants. By limiting the

study participants to data visualization and HCI experts as well as fNIRS researchers who may actually use this tool, we hope to minimize bias in the responses. The participants in this study cover the range of possible user types for BrainEx. It is also important to note that since the fNIRS research community is small, and well-connected, the Likert scale results may experience a positive skew due to familiarity with the research team.

The dataset used for this study is smaller than most fNIRS datasets and may not be reflective of all possible brain datasets. Thus, we assume some use scenarios may result in future users interacting with BrainEx in ways that the study participants did not. The functional requirements defined for this paper can be abstracted from our usage scenario to cover possible use cases. In addition, the questionnaire was designed to encourage participants to explore the tool and all of its features.

6.3 Results

Study participants were asked to rank their agreement with statements matching the sub-requirements discussed in Section 6.1.3. The Likert scale covered the range of strongly disagree to strongly agree, these were mapped to the range 1 to 5 for visualization purposes (Fig. 34).

The study determined that BrainEx effectively met its functional requirements, and based on user feedback provided many improvements and new features

User Study Results

Users were asked to rank their agreement with the following statements BrainEx supports...

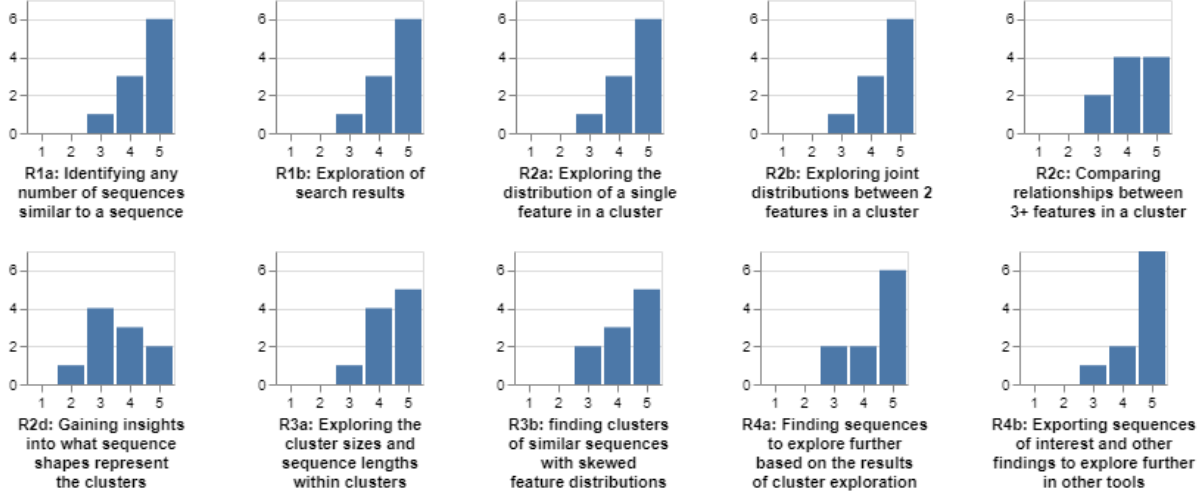


Figure 34: The results of the user study.

that existing fNIRS analysis tools do not currently include. A majority of experts agreed that our system fulfills all the individual functional requirements. In most areas, eight or more out of ten participants agreed that our system met the requirements.

Requirement 1: Similarity Search

Participants agreed that BrainEx successfully met our functional requirements for similarity search. Participants ranked both sub-requirements very high as seen in Figure 34. Nine out of ten participants said they agreed or strongly agreed with the statements, six of the participants strongly agreed with both statements. Six of the participants (P1, P2, P4, P5, P6, P10) provided additional positive feedback in the optional text field informing us the task was very easy to perform and provided results that looked correct and useful. P10 summarized their experience: “I loved it - very comprehensive. I liked that I could query the most important aspects of the data, and have a fine-grained level of control.” Despite the positive feedback, P4 and P10 expressed issues with the filter feature of the similarity search and P6 provided general feedback about the UI effects. These are minor fixes and do not affect the overall usability of the

tool.

Requirement 2: Feature Distribution Exploration

Participants were overall satisfied that the system allowed them to explore feature data through BrainEx. In general, the more features users attempted to explore, the less strong their agreement. In the case of exploring a single feature, all but one participant (P9) at least somewhat agreed that BrainEx supported them, with six participants believing this strongly. The results were similar when considering the visualization of joint distributions. Only one participant (P9) expressed neutrality, and six participants strongly agreeing.

The sub-requirements of exploring the distributions of three or more features had weaker, but still favorable results. Two participants (P7, P9) were neutral as to whether the system supported this use case. The rest agreed, but only four participants strongly agreed.

In particular, users expressed an appreciation for the options presented in BrainEx. P10 listed the fine level of control as a positive experience when interacting with the system. P5 found the tasks related to

feature distribution more challenging than the other tasks, citing the amount of work they had to do in manually examining the data. P6 expressed confusion about the dataset presented in the trial. Despite this, they were able to use the feature-wise visualization to explore the dataset, and make statements about the different user attributes. They felt the software was “very adaptable,” as they were not bound to specific filters. This suggests BrainEx may be useful to analysts who still need to learn more about their dataset of interest. However, additional guidance and support would improve the user experience.

Requirement 3: Cluster Exploration

Both sub-requirements of the cluster exploration requirement were found to be well supported. Figure 34 shows that eight out of ten participants agreed or strongly agreed with requirements 3a and 3b. P5, P6, and P10 liked the fine control provided over what characteristics of the clusters they could view at a time when selecting the cluster. Despite the positive feedback, P2, P4, P5, P6, and P10 found the task of finding an interesting cluster overwhelming due to the number of clusters and desired more features in the cluster exploration tool.

Some of this feedback pertained to features that currently exist in the tool that participants may not have been aware of. For example, the ability to sort and filter the table to more manageable sizes exist as described in Section ???. Additionally, a toggle is available to show the average feature distribution for the dataset as a baseline for comparison of feature skewness in individual clusters as desired by P10.

Additional features were also suggested in the feedback to improve the efficiency of scanning the cluster list and the ability for cluster comparison. P4 and P10 suggested adding a mechanism for paging through the table quicker and to change the color gradient used for different feature groups to make the cluster table quicker to sift through and easier to separate different feature sets. To improve the ability to compare clusters, P2 suggested being able to view multiple cluster representatives at once so that they could be directly compared.

Future work can make the task of finding an in-

teresting cluster less overwhelming and more informative. Filtering, sorting, and baseline comparison features can be made more exposed to the users and features for more powerful sorting, fine-grained filtering, and navigation through the table can be added. Additionally, the suggested features for comparing multiple clusters and cluster representatives as well as differentiating between feature groups in the table can also be added. However, the results of these survey results still indicate that BrainEx successfully supports the exploration of clusters within a dataset.

Requirement 4: Integration

The integration requirement has two sub-requirements. BrainEx supports searching and finding sequences of interest to explore further based on the results of cluster exploration. BrainEx supports exporting sequences of interest and other findings to explore in other tools. Both of the integration requirements scored very highly. Eight out of ten participants agreed or strongly agree with requirement 4a and nine out of ten participants agreed or strongly agreed with requirement 4b. Requirement 4b was rated as strongly agree by seven participants implying they found BrainEx’s ability to export findings into other tools to be very strong. The feedback for this section matched the positivity of the responses. In particular, four participants (P2, P5, P6, P10) filled out an optional text feedback section to share that they thought the integration of the cluster explorer and similarity search was very easy to use and allowed for “exploring potentially interesting aspects of the data.” (P10) P6 rated this requirement the lowest (neither agreeing or disagreeing with the first sub-requirement), and found the user task somewhat overwhelming and hard to keep track of the patterns and data they were comparing. However, they found the concept of the integration pipeline very promising stating “The pipeline of clustering and search has a lot of potential to explore the data” (P6).

Future work could be done to polish the workflow pipeline and make it easier for analysts to remember the clusters they began the similarity search from. However, the results of this user study show

that BrainEx successfully completes functional requirement 4 and shows that the cluster exploration and similarity search pipeline provides a novel and powerful exploration workflow for time series exploration.

Requirement 5: Accessible to all researchers

The accessibility requirement was measured by the ability of participants to use BrainEx successfully during the study. Based on feedback of only two users (P2, P8) having bandwidth issues when searching for over a 1000 similar sequences, BrainEx supports fast computations. Additionally, based on the positive feedback for the other requirements and the varied expertise and experience of the participants, BrainEx can support researchers of all levels. However, improvements can be made to reduce the overwhelming nature of the presented information to further support analysts.

Usability and Additional Feedback

While not surveyed directly, based on the text feedback for each of the functional requirements, participants found BrainEx generally usable. Study participants particularly enjoyed the amount of detailed control they had for exploring the data as well as the ease of using the visuals. For example, P1 commented that “very clear presentation of the results and enable users to explore different attributes” with regards to the Similarity Search requirement. However, participants found the amount of information to be overwhelming at times. For example, P6 noted after the cluster exploration task that “The number of clusters is often overwhelming” and that “It is not easy to identify which ones are most important to look at or to compare a specific small selection of clusters.” BrainEx does provide filtering on the table of clusters, however, this feedback indicates that this functionality should be better exposed and expanded to allow the user more control over managing the data. We conclude that study participants did not find severe usability issues with the system, but there is room for improvement to expose hidden features, distinguish and clarify elements of the user

interface, and allow for better management of large amounts of information.

Participants expressed interest in the many small changes to BrainEx, as well as the addition of some other potential features. P2 was uncertain as to why the tool allowed them to select only one cluster for visualization at a time, and suggested a multi-select tool to expand their understanding of the layout of the dataset. P7 believed that it was difficult to understand the sequence shapes inside of a cluster, and thought it might be worthwhile to add a visualization which displays an ‘envelope’ – a 2D space which captures all the shapes gathered within a cluster. They shared several images of these they’ve created during their research.

7 Conclusion

This project was able to provide significant expansion to the BrainEx frontend system. Evidence shows the work was successful. Even as the project comes to a close, student researchers outside the MQP team have begun using the tool. And a paper has been submitted to the IEEE VIS conference about our work. These successes show that BrainEx is a promising tool for time series exploration and analysis. However, there is always room for improvement. In this chapter we discuss our successes implementing this tool and additional work that can be implemented by future developers.

7.1 IEEE VIS Submission

Based on a thorough literature review and comparison of related systems, we determined that no existing system met the functional requirements we established in Section 6.1.3. Based on the success of our expert review, we provided sufficient evidence that BrainEx meets these requirements and thus provides a novel exploratory framework for time series data. The MQP team, along with Jim Plante, Eric Schmid, and our advisors (Professor Neamtu and Professor Solovey) were able to write a paper and submit it to the 2021 IEEE VIS conference. We eagerly await the results of the paper review.

7.2 Future Work

7.2.1 User Feedback

Our second study suggested that BrainEx meets the functional requirements set out in Section 6.1.3. Participants with diverse backgrounds agreed that BrainEx supports basic exploration operations on the data. Participants were particularly enthusiastic about the interface for similarity search, and the compatibility with diverse labels on the time series data. They generally agreed that the system allowed for broad exploration of a dataset as a whole. Several expressed enthusiasm about the potential utility of BrainEx as an exploration pipeline, where each operation performed enhanced their understanding of the patterns within a dataset.

In general, the broad exploration of the dataset's clusters was the weakest functional requirement. The table-driven presentation of the set of all similarity clusters did provide some useful ways for users to gather insights, but was overwhelming at times. Future work on the platform could expand it with additional filters, paging options and multiple options in color encodings to lessen the cognitive workload of exploring a large set of clusters. In general, the manual exploration of a huge number of machine generated clusters, with arbitrarily-many attributes warrants further research. Users were particularly interested in ways to save clusters of interest, or useful filters. They suggested they would like to use these saved values both for use with external tools, and for revisiting parts of their data for deeper understanding.

Additional feedback provided by the users included the ability to analyze multiple clusters at once. Currently, the cluster explorer allows you to select one cluster to explore its features. Analysts say that the ability to explore several clusters at once could reveal useful patterns.

On the similarity search tool, the users also had several suggestions. Restrictions should be added on the upper limit for end index and make it clear what the maximum value is. Currently, there are checks to make sure the minimum indices are valid but nothing on the upper end. Additionally, one user requested

an easy way to input the maximum and minimum indices without typing them. Finally, the similarity search sequence view only depicts the target sequence. When a user specifies a target range with start and end indices which do not highlight the entire sequence, it should show the user the range they have selected, possibly by highlighting that section a different color.

After the completion of the expert review, a working version of BrainEx was released for WPI researchers to use. It is available at <http://brainex.wpi.edu:5000/>. One user who began using the tool alerted the team to two issues that can also be implemented as future work. First, the front end does not support deleting preprocessed datasets. Currently, the server administrator has to manually remove bad datasets from the system. Additionally, the system needs to check dataset validity before preprocessing. Currently, if the dataset labels include white space characters the preprocess will fail but the dataset will remain on the server, in an unusable state. Checking for valid feature names and being able to delete bad datasets was not tested in the scope of our user studies but would be an important feature for users of this system.

7.2.2 Synchrony

When this project was originally proposed, the team was going to add a synchrony tool to BrainEx. This tool would take two time series as arguments and output a measure of synchrony. The development and testing effort needed to create the cluster explorer and integration pipeline was more significant than what we initially assumed. Therefore, the timeline was adjusted to focus on polishing these tools, pushing synchrony down the road to a future team. In the future, BrainEx would benefit from a feature that allows a user to select two time series sequences and directly compare them to receive a synchrony value. This would function similarly to the similarity search but instead of finding the most similar sequences, it would reveal the underlying similarity value between two sequences. This could be added to the React app through the addition of a new page. The computation to compare just two time series should not

require the full power of BrainEx and can be written as functions in the JavaScript code. It could use a sliding window and similarity measurements to calculate the synchrony.

References

- [1] Z. Li, “Brainex.” [Online]. Available: <https://apocalyvec.com/brainextutorial/>
- [2] “Gentxwarper: Mining of gene expression time series with dynamic time warping techniques.” [Online]. Available: <https://www.psb.ugent.be/cbd/papers/gentxwarper/DTWalgorithm.htm>
- [3] L. Hocke, I. Oni, C. Duszynski, A. Corrigan, B. Frederick, and J. Dunn, “Automated processing of fnirs data—a visual guide to the pitfalls and consequences. algorithms,” *Algorithms*, vol. 11, 2018.
- [4] R. Neamtu, R. Ahsan, C. D. T. Nguyen, C. Lovering, E. A. Rundensteiner, and G. Sarkozy, “A general approach for supporting time series matching using multiple-warped distances,” *IEEE Transactions on Knowledge and Data Engineering*, 2020.
- [5] R. Neamtu, “Interactive time series exploration powered by the marriage of similarity distances,” *Proceedings of the VLDB Endowment*, vol. 10, pp. 169–180, 11 2016.
- [6] —, “Brainex: an interactive and versatile similarity searchtool powered by distributed clustering,” n.d.
- [7] E. Schmid, J. Plante, and E. Buntel, “Brainex-frontend.” [Online]. Available: <https://github.com/WPIHCILab/BrainEX-Frontend>
- [8] R. H. Shumway and D. S. Stoffer, *Time Series Analysis and Its Applications with R Examples*, 4th ed. Springer, 2017.
- [9] *Extracting Optimal Performance from Dynamic Time Warping*. ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2016.
- [10] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, “Querying and mining of time series data: Experimental comparison of representations and distance measures,” *Proc. VLDB Endow.*, vol. 1, no. 2, p. 1542–1552, Aug. 2008. [Online]. Available: <https://doi.org/10.14778/1454159.1454226>
- [11] D. J. Berndt and J. Clifford, “Using dynamic time warping to find patterns in time series,” in *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, ser. AAAIWS’94. AAAI Press, 1994, p. 359–370.
- [12] S. Salvador and P. Chan, “Fastdtw: Toward accurate dynamic time warping in linear time and space,” 2004.
- [13] T. Warren Liao, “Clustering of time series data—a survey,” *Pattern recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.
- [14] J. Dubey, Y. Li, Z. Li*, R. Neamtu, E. Oktay, E. T. Solovey, and M. Sumaria, “Towards neuroadaptive technology using time warped distances for similarity exploration of brain data,” *WPI Undergraduate Research Publication Collection*, 1 2020, <https://wpiugpubs.pubpub.org/pub/cwyy0fec>. [Online]. Available: <https://wpiugpubs.pubpub.org/pub/cwyy0fec>
- [15] M. Ferrari and V. Quaresima, “A brief review on the history of human functional near-infrared spectroscopy (fnirs) development and fields of application,” *NeuroImage*, vol. 63, pp. 921–935, 2012.
- [16] (2020) Functional near-infrared spectroscopy (fnirs) cognitive brain monitor patent details. [Online]. Available: <https://technology.nasa.gov/patent/LEW-TOPS-84>
- [17] N. Naseer and K. Hong, “fnirs-based brain-computer interfaces: a review,” *Frontiers in Human Neuroscience*, vol. 9, 2015.
- [18] (2020) Brain-computer interface restores sense of touch with haptic signals. [Online]. Available: https://www.mathworks.com/company/mathworks-stories/battelle-neurolife-system-turns-thoughts-into-movement.html?source=15574|&s_eid=psm_15574

- [19] “Django introduction,” <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction>, accessed: 12-5-2020.
- [20] “Getting started - react,” <https://reactjs.org/docs/getting-started.html>, accessed: 12-5-2020.
- [21] L. Ziheng and E. Buntel, “Brainex.” [Online]. Available: <https://github.com/WPIHCILab/BrainEx>
- [22] S. Cha, “Comprehensive survey on distance/similarity measures between probability density functions,” 2007.
- [23] R. Neamtu, R. Ahsan, E. A. Rundensteiner, G. N. Sárközy, E. J. Keogh, H. Dau, C. Nguyen, and C. Lovering, “Generalized dynamic time warping: Unleashing the warping power hidden in point-wise distances,” *2018 IEEE 34th International Conference on Data Engineering (ICDE)*, pp. 521–532, 2018.
- [24] A. Pikovsky and M. Rosenblum, “Synchronization,” *Scholarpedia*, vol. 2, no. 12, p. 1459, 2007, revision #137076.
- [25] T. Kreuz, “Measures of neuronal signal synchrony,” *Scholarpedia*, vol. 6, no. 12, p. 11922, 2011, revision #183067.
- [26] Y. Li and H. Wu, “A clustering method based on k-means algorithm,” *Physics Procedia*, vol. 25, pp. 1104–1109, 2012.
- [27] “Visualization analysis & design (online access included),” vol. 2, no. 8, 2015.
- [28] G. Brockington, J. Balardin, G. Zimeo Morais, A. Malheiros, R. Lent, L. Moura, and J. Sato, “From the laboratory to the classroom the potential of functional near-infrared spectroscopy in educational neuroscience,” *Frontiers in Psychology*, vol. 9, 2018.
- [29] K. R. Bresnahan, M. A. Goodwin, and Y. Lin, “Applying hci design practices to the development of the brainex user-interface to facilitate fnirs research,” 2020.
- [30] “Wpi human-computer interaction research group.” [Online]. Available: <http://interaction.mystrikingly.com/>

A Informed Consent Script

Introduction: You are being asked to participate in a research study. Before you agree, however, you must be fully informed about the purpose of the study, the procedures to be followed, and any benefits, risks or discomfort that you may experience as a result of your participation. This form presents information about the study so that you may make a fully informed decision regarding your participation.

Purpose of the study: The purpose of this study is to better understand how a user might interact with a tool for exploring functional near-infrared scans of the human brain.

Procedures to be followed: The study will be conducted via Zoom video conference. First, we will ask you a couple of questions, including about your research interests and experience with software. Then, we will provide you with a link to the prototype software, and a list of tasks for you to perform, such as the identification of trends within a dataset or to retrieve a certain query from the prototype. We will observe you performing these tasks, via screen share and video conference, and ask you to describe your process for doing so. We will then conduct a post-interview where we ask you questions about your experience with the prototype software.

Risks to study participants: No personally identifiable information will be kept in association with this study. Non-personally identifiable data, such as experience with software, research qualifications, and age window will be kept in association with this data.

Benefits to research participants and others: Subjects have the opportunity to try out a prototype time series explorer before it is released to the general public. Their feedback will help us design a tool which may be helpful to their future projects.

Record keeping and confidentiality: Records of your participation in this study will be held confidential so far as permitted by law. However, the study investigators, the sponsor or it's designee and, under certain circumstances, the Worcester Polytechnic Institute Institutional Review Board (WPI IRB) will be able to inspect and have access to confidential data that identify you by name. Any publica-

tion or presentation of the data will not identify you. The undergraduate student team and their advisors will have access to any records produced during this study. They will immediately anonymize this data by replacing any names mentioned with pseudonyms, and any ages mentioned with 10 year age windows. These records will be kept in digital form.

Compensation or treatment in the event of injury: We believe the risk of injury in this study is minimal. You do not give up any of your legal rights by signing this statement.

Your participation in this research is voluntary. Your refusal to participate will not result in any penalty to you or any loss of benefits to which you may otherwise be entitled. You may decide to stop participating in the research at any time without penalty or loss of other benefits. The project investigators retain the right to cancel or postpone the experimental procedures at any time they see fit.

B First User Study Interview Script

*Consent form and a brief introduction will be given to the participants prior to the test. Therefore, this script does not include discussion of the consent form. The consent form can be found in Appendix A

**before the interview begins we will set up a server or a computer running the BrainEx software, the link to the app or directions on how to launch it will be shared with the user during the interview after background questions.

Introduction

Hello [test user's name], I'm [Interviewer's name] and I will be administering this test with you today. My teammates, [teammate 1's name] and [teammate 2's name] will be joining us in the Zoom call as note takers and observers. Just so I make sure I cover everything I am going to read to you from a script.

We are a group of undergraduate students working on our MQP to expand the software, BrainEx. As a member of the HCI lab you may be familiar with this software already. In case you are not here are the key points. BrainEx is a tool for exploring and comparing sequences in time series datasets with a focus on applications of brain data. Part of our work in extending this software includes adding a cluster explorer functionality to the front end of the application. These new pages allow researchers to study the clusters created by BrainEx based on their time series data. Potentially allowing them to discover patterns and similarities in cognitive function or other time series sources. Your participation today will allow us to make this cluster explorer as effective as possible.

This test will require no more than 60 minutes of your time. We will be taking notes during the process. Your participation is voluntary. You do not have to answer any questions and you may end the test at any time you wish.

As you go through the test, please talk out loud and share your thoughts as much as possible. It will provide us with valuable data to help us improve our design.

If you have questions at any time, please ask them.

We may not answer right away since we want to see if users can learn and understand the application. But if you have any unanswered questions after the test we will be happy to respond.

Ok, let's begin.

Background Questions

Do you have any questions so far? OK. Before we look at the site, I'd like to ask you just a few quick questions.

1. Are you familiar/have you worked with time series data before?
 - (a) If answer is no, briefly explain the concept of time series data:
 - Time series are a broad set of data whose values change with respect to time. Any data that can be collected on regular intervals of time can be used as a time series.
2. Have you worked with BrainEx or any of its predecessors, ONEX or GENEX, before?
3. Are you familiar with the concept of clustering data?
4. Have you used data visualization software/tools before?
 - (a) If yes, which software?
 - (b) If yes, did you do any work with clustering and can you describe it?

Now you can load the BrainEx front end. Please get to the page and share your screen. When you are ready we will give you a few simple tasks to test.

User Tasks

1. From the dataset SART2018ADJUSTED, find a large cluster with a majority of event names labeled as target correct. Look at the distribution of the subsequences across event-name, channel and user. Can you identify any relationships between these data?

2. From the cluster you selected in step 1, share any observations you may have about the cluster's contents.
3. From the 003-SART-August2017-CZ A-DC12 dataset, identify the cluster that contains the most samples. Make a comment about what else is true about this cluster.
4. From the cluster you selected in step 3, share any observations you may have about the cluster's contents.
5. Time permitting. Take some time to explore and become comfortable with the tool. Share any observations you have while exploring.

Follow Up Questions

Thank you for doing the test. Now we will wrap up with just a few quick questions about how the test went.

1. For the task of finding disproportionate clusters, rate your satisfaction with each prototype on a 1-10 scale and say why.
2. For the task of identifying relationships between the data, rate your satisfaction with each prototype on a 1-10 scale and say why.
3. For the aesthetic design, what is your overall impression of our interface and how it allows you to explore the clusters of the time series?
4. Where do you think you would need more guidance in terms of using the feature? What seems unclear?
5. What parts of the app do you think would be useful and would definitely like to use?
6. What do you think is the most important information you would need to see when exploring clusters?
7. Are there any views, visualizations, or other features that you would like to see added to the prototype? How would you use those features?

C Expert Review Questionnaire

We would like you to try out a new tool called BrainEx and then answer questions and provide feedback based on your experience. The questionnaire, combined with exploring the tool, should take between 30 and 45 minutes to complete, but feel free to take as much time as you need to explore BrainEx.

What is BrainEx?

BrainEx is a tool for efficient visual exploration and discovery of sequence similarity in brain signals. It can be used to gain a better understanding of a dataset, including what attributes of the dataset appear within similar sequences. This could inform work on brain-computer interfaces and human-computer interaction by enabling non-experts to familiarize themselves with data that was collected. Built on the foundation of ONEX and GENEX*, it allows for the real-time search of large datasets to discover similar time series and subsequences within the dataset, even those with different lengths or appearing in a longer time series.

What can I do with BrainEx?

Explore visualizations of sequences within a dataset, including filtering by attributes of the dataset (e.g. users, sensor channels, events) Select an individual time series, or a subsequence within the time series, and search for other sequences that are similar within the larger dataset, including subsequences within other time series and sequences of different lengths Explore the results of similarity search, including filtering by attributes of the dataset (e.g. users, sensor channels, events) Explore the underlying sequence clusters that were generated based on similarity to see distributions of attributes (e.g. users, sensor channels, events)

What should I do next?

We would like you to try out BrainEx (using a com-

puter/laptop, not a mobile device) and provide feedback in the questionnaire. For this exercise, we have already created a preprocessed dataset for you to explore. This dataset is called WorkloadStudy. In order to import this dataset into BrainEx, a researcher would upload a CSV containing any number of time series. Each time series could also have custom attributes such as user, channels, events.

WorkloadStudy is a dataset of brain signals collected with functional near-infrared spectroscopy (fNIRS). There are four channels (Ch-A,Ch-B,Ch-C,Ch-D) that would represent different sensor locations, collected from eight users (U1,U2,U3,etc). There are four event labels (Easy, Hard, MathTest). In the study, users performed calibrated tasks that were known to be Easy and Hard, as well as a Math Test, which was of an unknown difficulty. We are interested in using the labelled events (Easy and Hard) to better understand the MathTest.

The working version of the BrainEx, preloaded with datasets to explore, can be found at:

<http://brainex.wpi.edu:5000/>

By taking part in this survey you agree that your feedback can be used for research purposes and published anonymously. You are free to end your participation in this survey at any time. Additionally, if you don't feel confident answering a question you may leave it blank.

Contact:

Michael Clements (mjclements@wpi.edu)
Andrew Nolan (acnolan@wpi.edu)
Jackson Powell (jjpowell@wpi.edu)
Erin T. Solovey (esolovey@wpi.edu)
Rodica Neamtu (rneamtu@wpi.edu)

When you are ready, please navigate to <http://brainex.wpi.edu:5000/> using a computer or laptop (not a mobile device). There will be a preloaded dataset named WorkloadStudy. Take some time to explore the dataset using the Explore Clusters or

Similarity Search buttons. The Similarity Search button allows you to search for similar subsequences to a chosen time series. The Explore Clusters button allows you to visualize the clusters created by the underlying BrainEx engine. Tooltips on buttons and labels throughout the tool provide additional information for using the tool. Once you feel comfortable with the tool you can continue with the survey.

Check the box if you feel comfortable with the exploring you've done in BrainEx, you are prepared to start the survey

I am prepared to take the survey

General Information

Current Academic Position: ----

Please rate your expertise (No knowledge, Passing Knowledge, Knowledgable, or Expert) in the following areas: fNIRS data, Neural Data Analysis (other than fNIRS), Human Computer Interaction, Data Visualization.

Have you ever authored a paper in the following fields? (fNIRS data, Neural Data Analysis (other than fNIRS), Human Computer Interaction, Data Visualization).

Have you ever used BrainEx before?

The remainder of the survey will ask you to rate your level of agreement with statements about BrainEx including about its specific features and general usability. Feel free to use the BrainEx application at any time during your survey.

Similarity Search

BrainEx uses a variety of warped distance measures (e.g. Dynamic Time Warping (DTW), Warped Minkowski, Warped Manhattan, Warped Chebyshev) to calculate the similarity between different pieces of time series data. It is designed to use these measures of similarity to return relevant sequence data, including sequences of different lengths, when queried.

BrainEx computes similarity very quickly, so it is possible to quickly retrieve several hundred results from a single query.

In addition, BrainEx supports exploring what features similar sequences share. The search results include any additional metadata about the sequences that had been included in the original import of the dataset (e.g. User, Channel, Event) and includes filtering and sorting by features as well as sequence length. BrainEx also supports visualizing distributions, based on one or more of these attributes (e.g. how many sequences from User 2 are in the results? What about different combinations of users, channels and features?)

As an exercise, we would like you to try to search for sequences similar to any of the MathTest events in the WorkloadStudy dataset. Can you retrieve similar sequences? Can you make any statements about these similar sequences? Press NEXT when you are done.

Similarity Search How much do you agree or disagree with following statements?

Feel free to refer back to the BrainEx application (<http://brainex.wpi.edu:5000/>) at any time to answer the questions.

BrainEx supports identifying any number of sequences similar to a sequence of my choice. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

BrainEx supports exploration of search results and attributes of the sequences in the result set. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

Share any information you gathered about the dataset from completing this task

Provide any notes about positive experiences

Provide any notes about negative experiences

Cluster Exploration *When BrainEx preprocesses the data, it makes similarity-based clusters for all subsequence lengths. Within a cluster, all sequences will have the same length. The user is able to view all of these clusters, as well as which user-imported custom attributes are present in them. A user can select a specific cluster and view all of the data associated with all of the sequences within.*

As an exercise, we would like you to try to explore the entirety of the WorkloadStudy dataset. Can you find clusters that seem to contain mostly sequences collecting during the Hard task? What about clusters that seem to be balanced between Events? What does that tell you about a cluster? Can you find attributes (such as specific Users or Events) that are often clustered together? That are rarely clustered together? Can you explore other attributes and how they are distributed in the clusters? Can you explore patterns when you look at the sequence plot of your selected cluster? Can you look for multi-feature patterns in the data? Press NEXT when you are done.

Cluster Exploration

How much do you agree or disagree with the statements below?

Feel free to refer back to the BrainEx application (<http://brainex.wpi.edu:5000/>) at any time to answer the questions.

BrainEx supports exploring the range of cluster sizes and sequence lengths within clusters. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

BrainEx supports finding clusters of similar sequences with skewed feature distributions (i.e. clusters that mostly contain a particular User or Channel or Event). (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

BrainEx supports exploring the distribution of a single feature (e.g. a participant name) in a cluster. (Strongly disagree, somewhat disagree, neither agree

nor disagree, somewhat agree, strongly agree)

BrainEx supports exploring joint distributions between two features in a cluster (to see, for example, if there is a correlation between Channel and Event). (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

BrainEx supports comparing relationships between three or more features in a cluster. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

BrainEx supports gaining insights into what sequence shapes represent the largest and/or most interesting clusters. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

Share any information you gathered about the dataset from completing this task

Provide any notes about positive experiences.

Provide any notes about negative experiences.

Integration

The similarity search and clustering were designed to work together. Once you have found an interesting cluster, you could do additional similarity searches based on sequences in the cluster of interest. Keep in mind, clusters always contain sequences of the same length, while search results can contain results of different lengths. BrainEx also supports export of figures, similarity search results and clusters for integration with other tools in an analysis pipeline or for presentations.

As an exercise, we would like you to choose an interesting cluster, and then perform a similarity search based on its sequences. Explore the results of this similarity search. Do insights you found about the cluster extend to the similarity search result? Are there things that are true of the search results that are not true of the cluster, or vice versa? We would also like you to export any interesting findings you

have discovered in this task or in previous tasks. Press NEXT when you are done. like to share with us?

BrainEx supports searching and finding sequences of interest to explore further based on the results of cluster exploration. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

BrainEx supports exporting sequences of interest and other findings to explore further in other tools. (Strongly disagree, somewhat disagree, neither agree nor disagree, somewhat agree, strongly agree)

Optional, feel free to share a screenshot or description of any patterns you found

Provide any notes about positive experiences.

Provide any notes about negative experiences.

General BrainEx Questions

Reflect on your experience using BrainEx. Were you able to find your way around the application? Was there enough context to understand what various features did? Press NEXT when you are done.

General BrainEx Questions

Would you use BrainEx in your future research? If so, please describe how you would use it.

Are there any other features you would like to see included in BrainEx in the future? If so, what are they?

Do you use any other tools for fNIRS or brain signal analysis? If so, please compare BrainEx with other analysis tools you are familiar with.

Did you have any bandwidth issues or latency in BrainEx? If so, please describe.

What browser did you use? (e.g. Safari, Chrome, Firefox, etc.)

Are there any other additional comments you would

We thank you for your time spent taking this survey.

Your response has been recorded.

D BrainEx User Guide

This guide explains the basics of how to use the BrainEx cluster explorer tool.

D.1 Preparing Data

BrainEx is a tool for examining time series data, therefore in order to explore clusters you must have time series data. The BrainEx front end accepts time series data in the form of .csv or .tsv files, and it is agnostic to the source of this data. Once you have collected your timeseries data and prepared your file you can start BrainEx and load your data. When you arrive on the home screen you can press "Upload New Dataset" to begin the process of loading your file. Select your file and specify the number of header rows and the number of features present in your data. In this step you will also specify the length of subsequences you are interested in from your data. This step is important because only lengths in that range will appear in the cluster exploration. Once the configurations are determined, you can start preprocessing. Depending on your machine and the size of the data this may take up to several hours. Once the data is preprocessed you are ready to go.

D.2 Selecting Clusters

After your data is preprocessed, you will see a new row for your data in the table on the homepage of BrainEx. Select this row and press the "Explore Clusters" button. You will be redirected to a page that looks like figure 35. This is the Cluster Explorer page, on this page you can see and filter a list of all of the clusters created by BrainEx when processing your data. You should see a table containing all of the clusters in the dataset. In the upper left hand corner of the page will be a pair of sliders. These allow you to see the upper and lower bounds for the size of the clusters and length of subsequences in the clusters you are viewing. By default all of the clusters are shown in the table, these sliders allow you to narrow the amount of clusters in the table. When you are satisfied with the values of the sliders, you

can press the "Apply Filters" button to view a table containing the clusters.

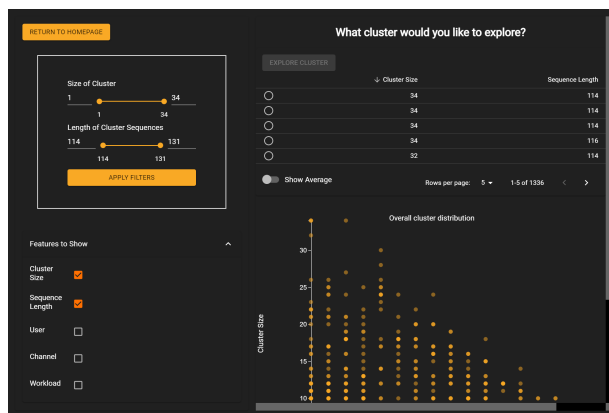


Figure 35: An example of what the Cluster Explorer page will look like upon loading.

A table will be visible on the right hand side of the screen. By default 5 rows are shown in the table, this can be changed to be 10 or 20. You may notice that the table contains columns labeled cluster size and sequence length. By default these features are turned on. In the bottom left corner of the page there is a panel containing all of the other features available in the dataset. Checking these boxes will display more data in the table. Cluster size and sequence length are numeric values. Enabling the other features will add columns for each possible value of that feature and the percentage of that feature present in each cluster. For example, see figure 36. In this example there are three possible workloads. Higher prevalence of a value in a cluster are colored darker. All features can be used to sort the table.

Below the table is either a scatter or line plot visualization. When the representatives are loaded, the visualization starts as a scatter plot of the cluster size on one axis and the sequence length on the other axis. This scatter plot shows the overall distribution of the filtered set of clusters in the table. When a cluster in the table is selected, the scatter plot changes to a line plot of the selected cluster's representative.

With the tools on this page you should be able to get a general sense of the shape and spread of clusters

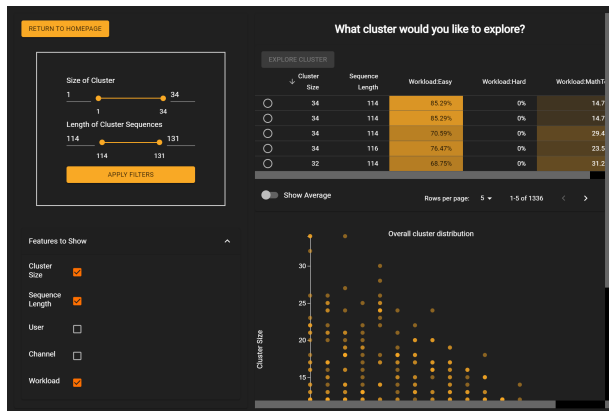


Figure 36: The Cluster Explorer page with the workload feature checked.



Figure 37: An example of what the Selected Cluster Explorer page will look like upon loading.

in your dataset. If you want to explore the breakdown and contents of a specific cluster, all rows in the table are selectable. If you choose one and press "Explore Cluster", you will be able to observe the contents of a specific cluster.

D.3 Exploring a Selected Cluster

Once a cluster is selected you can explore the breakdown of features within it using several different visualizations. By default a bar chart will be displayed for the first feature in the dataset when the page loads. The page should look something like Figure 37 once it is loaded. However, you can check and uncheck features to change the visualization. The Selected Cluster Explorer has three main types of visualizations for features: single feature bar charts/histograms, two feature heatmaps, and three or more feature parallel axes plots. A fourth visualization exists which displays a graph of all of the subsequences in the cluster.

To view the display of the subsequences you can press the button labelled "Plot Sequences". Sequences are colored based on the first feature selected. In this view you can mouse over individual series to see the features of time series that the sequence comes from. By looking at the start index you can determine if time series come from the same source but are shifted. You can also see only one sequence from

each time series by using a toggle in the "More Info" popup. Sequences can also be selected to change their color and the target sequence that the "Similarity Search" button will load. An example view of the series plot can be seen in Figure 38.

The first feature visualization is a bar chart. This will display when only one feature is selected. The bar chart shows the percentage of each each value of that feature present in the cluster. Mousing over the bar charts will show you the percentage, name of the feature, and the count of that feature.

When two features are selected the visualization becomes a heatmap. The heatmap shows the joint distribution of two features. Similar to the bar chart it shows the percentage of each feature in the cluster, but this time instead of one feature it show the percentage of a pair of features present in the cluster. Darker colors are more present while whiter colors are closer to zero. An example of a heat map in the Selected Cluster Explorer can be seen in Figure 39

When three or more features are selected you will see a parallel axes graph. This plot shows the pairwise relationship between different features. It can be used to tell if certain features connect to other features. An example of this can be seen in Figure 40 In this visualization the axes can be moved to potentially view other relationships. A user can create a selection rectangle that will highlight only certain

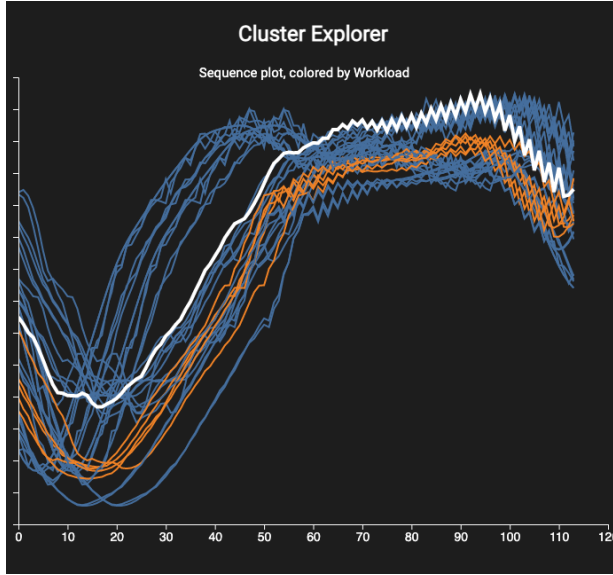


Figure 38: An example of the series plot view.

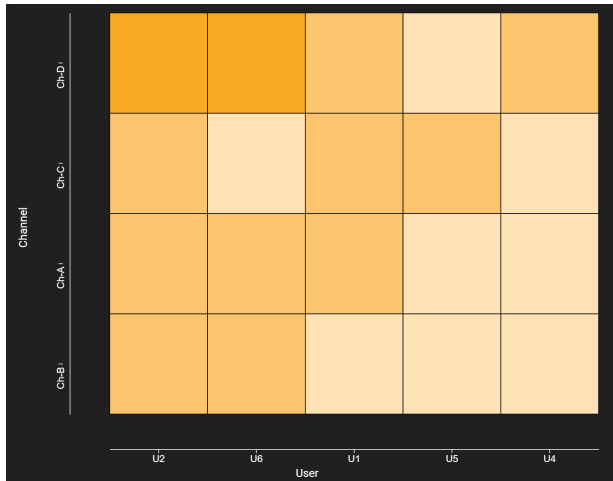


Figure 39: An example of the heatmap, here colors closer to white have lower values, while darker blue sections have more prevalence.

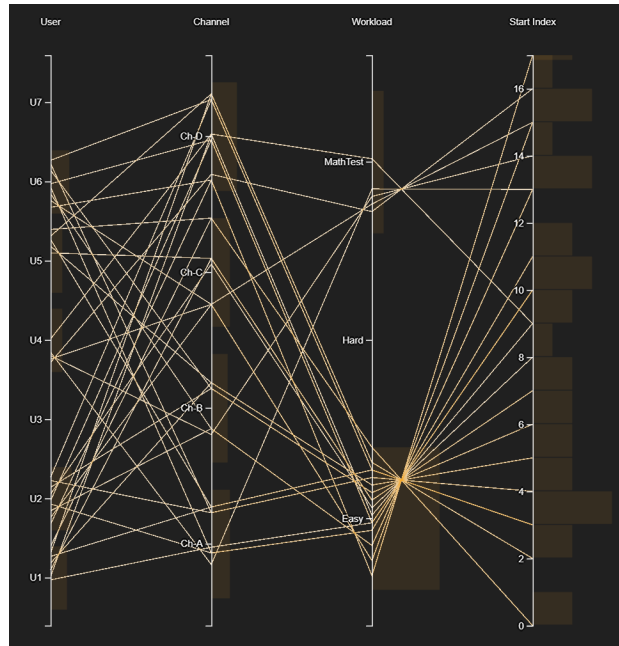


Figure 40: An example of the parallel axes graph between three features.

connections to see how they continue throughout the features. An example of this can be seen in Figure 41.

Another key feature of the Selected Cluster Explorer is the "Numeric?" toggle. By default this setting is turned on for features it applies to. For features whose data type is numeric this feature allows the data to be placed into buckets to make it more readable. If all of the numeric features, such as start time, are included then the feature may have many unique values and be hard to interpret. When bucketed it may be more clear if there are trends in the data. This will convert the the bar chart into a histogram when only a single numeric feature is selected to be visualized. The number of buckets can be adjusted with a slider that is visible in the "More Info" window.

Four additional buttons exist on this page. "Export To CSV" allows the user to select features and then export the time series data within the cluster to a .csv file to be used for other purposes. The "EXPORT GRAPH" button will save the current visual-

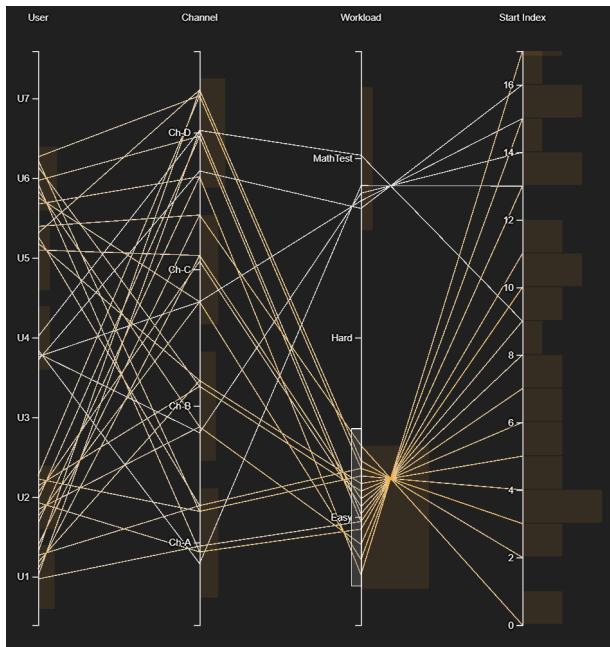


Figure 41: Here a small rectangle is drawn over the subsequences going through the Workload Easy. Those lines are highlighted in orange, the other lines become grey.

ization as an image. The "More Info" button brings up a popup describing the current visualization in case you ever need a reminder. Finally, the "Similarity Search" button will open a popup warning the user that they are about to load the Similarity Search page with the currently selected sequence as the target sequence. If no sequence was selected in the series view, then it will load the representative of the cluster. This button serves as one half of the integration between Similarity Search and Cluster Exploration. The other half is enabled by an "Explore Data" button on the Results page that loads the SelectedClusterExplorer page with the feature distributions of the sequences in the result set returned by the similarity search query.

With all of these tools you should be prepared to explore some clusters in time series with BrainEx.

D.4 Exploring with Similarity Search

Once the "Similarity Search" button is clicked on either the home page or the Selected Cluster Explorer page, then the user is directed to the Similarity Search page. If the page is accessed from the Selected Cluster Explorer page, then the Target Sequence, Start Index, and End Index fields are automatically populated using the selected sequence (Figure 42). Otherwise, the Target Sequence must be selected from a dropdown or the line plot and the indexes must be entered manually. The line plot and the filter button in the bottom-right of the page allows for exploration of all the sequences in the dataset by their shape and feature respectively.

After exploring the sequences in the dataset, the user may specify other properties for their similarity search. These properties include the number of similar sequences to return, the overlap allowed between returned sequences, and whether the time series with the target sequence should be excluded from the results. The overlap field allows for the user to handle cases where the resulting sequences may be all from the same time series, just slightly shifted by their start time. Once a user has finalized their search options, they press the "Search" button to begin their search. A generic dialog should pop up indicating that the search has begun.

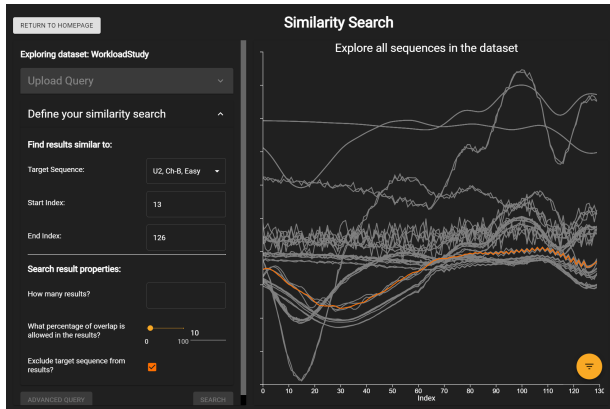


Figure 42: Similarity Search page opened from the "Similarity Search" button. Opening this page from the Selected Cluster Explorer page automatically populates the Target Sequence, Start Index, and End Index fields based on the selected sequence.

D.5 Exploring Similarity Search Results

Once the similarity search completes, the user should be presented by the Results page, such as can be seen in Figure 43.

This page allows the user to explore the ranking of similar sequences by providing filter capabilities, a linked line plot and table, export options, and access to feature distribution exploration of the results. In addition to providing filtering on the features of the sequences, the plot can also be filtered by the number of results shown and how close the sequences should be to the target sequence. The user can interact with the table and plot by hovering over a sequence in one tool to highlight it in the other. For outside exploration, the result set of sequences can be exported as an svg of the line plot or a .csv file of the sequence data using the "Save Graph" and "Save to CSV" buttons respectively.

For more in-depth exploration of the features in the resulting sequences, the user may click on the "Explore Data" button. This button brings the user to the Selected Cluster Explorer page, but loads the feature distributions of the result set, rather than a cluster. This page provides the same visualizations as the cluster version, but with slight textual changes to

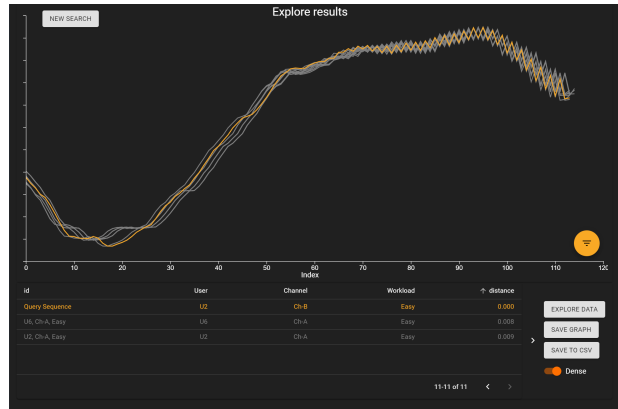


Figure 43: Explore Results page opened from the "Search" button on the Similarity Search page after the search completes.

reflect the use of similarity search results rather than a cluster. These differences change the terminology used, the representative to be the target sequence of the search, and the "Select Another Cluster" button to "Back to Results", bringing the user back to the Results page instead of the Cluster Explorer page. Some of these changes can be seen in Figure 44.

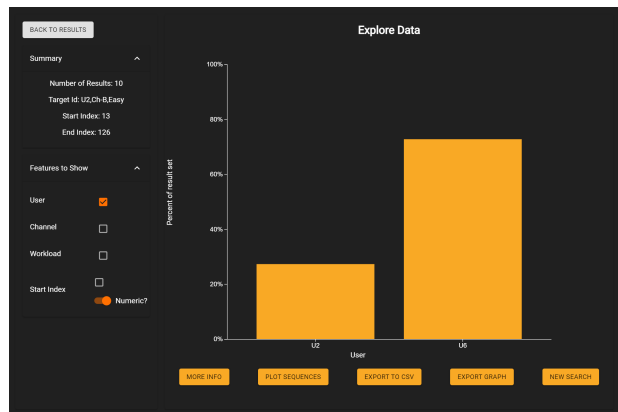


Figure 44: Explore Data page opened from the "Explore Data" button on the Results. This page uses the same components and visualizations as the Selected Cluster Explorer, but operates on feature distributions for the results of a similarity search rather than a cluster.

E Minimum Server Requirement Recommendations

Below is the list of minimum attributes recommended for supporting a remote server hosting BrainEx and a Django server for the BrainEx front end. The resource values are approximations as they were not determined using benchmark testing.

- OS: Linux, can be headless.
- Processors: 8
- RAM: 32 GB
- Disk Space: 40 GB
- JDK installation
- An open port to listen on as well as a domain or static ip that can reliably be connected to.
- Maximum request entity size might be slightly larger than is typical. We expect requests as large as 40MB in extreme cases.
- Slightly unusually large outgoing responses from this server. We have seen as big as 234MB in preliminary testing.

F BrainEx Server Description

In the interest of providing the HCI lab with a working, deployed version of BrainEx, we have setup the BrainEx back end to run constantly on a server, and created a simple web server to serve the BrainEx front end. To use brainex, one connects to BrainEx.wpi.edu on port 5000. The Django API, deployed as a hypercorn server, listens on port 8000.

To make modifications to the BrainEx instance, one need simply connect to BrainEx.wpi.edu via ssh. There, they will find the scripts required to restart BrainEx's front end, as well as its back end.