Optimizing EEG Visualization Through Remote Data Retrieval

N. Capp, C. Campbell, T. Elseify I. Obeid and J. Picone
The Neural Engineering Data Consortium, Temple University
{noah.capp, tuf46810, tug35668, iobeid, picone}@temple.edu

A common practice in analyzing brain activity is capturing electroencephalograms (EEG), taken by place electrodes on the patient's scalp which measure electrical activity via voltage differences. These multi-channel voltage signals are commonly stored in the European Data Format (EDF). An EDF file of an unpruned EEG can become quite large – 1MB of an EDF file translates to roughly 1 minute of an EEG recording session with a sampling rate of 250Hz. Additionally, full datasets from the TUH EEG Corpus [1] can reach sizes of over 800GB. Because of the size of the EEG recordings, there is a need for efficient visualization and analysis tools. In this work, we focus on enhancing the user experience by streamlining the process of retrieving EDF data from our server.

An EEG visualization tool [2], written in Python, was developed to facilitate annotation of EEG events in the TUH EEG Corpus [1] and supports many other visualization techniques. Its annotation capabilities are fully integrated within the tool, overlaying the EEG signals; an improvement to software with similar capabilities (e.g. EDF Browser [3] and EEGLab [4]). This software also has spectrogram and signal energy visualizations that are also implemented within the viewing of the raw or filtered EEG signals.

There is cohort retrieval system [5] integrated within the visualization tool that allows search queries to be sent to an EEG dataset that return EDF files relating to the given search terms. This system uses both keyword terms and natural language processing to process the input search query. The medical reports found in the TUH EEG Corpus are then used, along with automatically extracted EEG signal events to rank the most relevant EEG sessions. The tool then allows users to easily visualize and/or annotate EEGs returned by cohort retrieval system.

Previously, visualizing and/or annotating an EEG recording required the EDF file to be locally available on the user's machine. This often meant our annotators would have roughly 75 GBs of disk space consumed by EDF files. This is less than desirable for a number of reasons involving data integrity and security. An application programming interface (API) has been developed to provide a means of streaming EDF data from our server and into the EEG visualization software. Popular music streaming services provide APIs with similar functionality (e.g. Spotify [6] and SoundCloud [7]).

Through this API, a user can send an HTTP GET Request [8], which contains a file path on the server. The API returns either the paths to the available files in a given directory on the server, or the file itself. This API is protected through a private key, which prevents any unauthorized users from accessing this data [9]. This functionality ensures that users do not need to manually download large batches of EEG data, clearing space on the users' hard drives and relieving CPU usage on the server. The API is also utilized by the cohort retrieval system [5]. This system returns relative file paths to the most relevant EEG data to the search queries. In previously implementations, this meant the entire TUH EEG Corpus must be locally available on an external hard drive to be able to visualize the data. Instead, the EEG data can simply be streamed into the user's system using the developed API.

A live demonstration of this integrated visualization and cohort retrieval tool will be provided at the conference. The software is publicly available at *https://www.isip.piconepress.com/projects/tuh_eeg/ downloads/nedc_demo*.

REFERENCES

[1]   I. Obeid and J. Picone, "The Temple University Hospital EEG Data Corpus," *Front. Neurosci. Sect. Neural Technol.*, vol. 10, p. 00196, 2016.

[2]   N. Capp, E. Krome, I. Obeid, and J. Picone, "Rapid Annotation of Seizure Events Using an Extensible Visualization Tool," presented at the IEEE Signal Processing in Medicine and Biology Symposium, 2017, p. 1.

[3]   T. van Beelen, "EDFbrowser," *Teuniz*, 2013. *http://www.teuniz.net/edfbrowser/*.

[4]   A. Delorme and S. Makeig, "EEGLab," Swartz Center for Computational Neuroscience, 2002. *https://sccn.ucsd.edu/wiki/EEGLAB*.

[5]   J. Picone, I. Obeid, and S. Harabagiu, "Automated Cohort Retrieval from EEG Medical Records," presented at the 26th Conference on Intelligent Systems for Molecular Biology, 2018, pp. 1–7.

[6]   "Web Playback SDK," *Spotify for Developers*, 2018. *https://developer.spotify.com/documentation/web-playback-sdk/*.

[7]   "HTTP API Guide," *SoundCloud*, 2018. *https://developers.soundcloud.com/docs/api/guide*.

[8]   "Hypertext Transfer Protocol – HTTP/1.1," *Network Working Group*, 2007. *https://www.w3.org/Protocols/HTTP/1.1/rfc2616bis/draft-lafon-rfc2616bis-03.html*.

[9]   "Using API Keys," *Google Inc.*, 2018. *https://cloud.google.com/docs/authentication/api-keys*.

Optimizing EEG Visualization Through Remote Data Retrieval

N. Capp, C. Campbell, T. Elseify, I. Obeid and J. Picone

The Neural Engineering Data Consortium, Temple University

NEURAL ENGINEERING DATA CONSORTIUM
www.nedcdata.org

**College of Engineering
Temple University**

Abstract

- An electroencephalogram (EEG) is a multi-channel signal which describes the electrical activity in the brain through electrodes placed on the scalp.
- A visualization tool was developed to allow users to create annotations directly overlaying the EEG signals. This tool also includes filtering options, as well as multiple alternative visualization methods (e.g. spectrogram or energy views).
- The tool integrates an NIH-funded cohort retrieval system, which allows users to seamlessly find EEG events in the TUH EEG Corpus that are relevant to any given search query.
- The features within the tool are customizable through a preferences window, allowing users to save settings based on their annotation and visualization needs.
- The tool is capable of processing EEG signals stored only on local disk. The software should be able to retrieve data not locally available, to avoid data storage and integrity issues.
- This visualization tool is written in PyQt, which allows for Graphical User Interface programming that is supported across multiple platforms.

Overview of the Visualization Software

- The signal viewer loads EEG signals in a European Data Format (EDF), and provides a user interface to view and manipulate the signal.

Clinical EEG Data

- EEG signals are stored as sampled data signals in an EDF format. They are typically sampled at 250 Hz using 16 bits/sample and stored in a PCM format.
- A montage is used to redefine channels as the difference of channel voltages. This kind of differential analysis reduces noise.
- A TCP montage is commonly used to accentuate important events such as seizures. However, the tool offers a user interface to create custom montages.
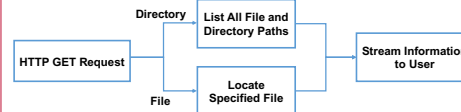

Data Requirements

- Storing EDF files is not a trivial task. The quantity and size of files is a major problem in managing datasets such as the TUH EEG Corpus.
- Each sample of the EEG signal is stored as two bytes per sample. The number of electrodes (channels) in these signals is often between 20 and 30. The aggregate data rates start at approximately 36 Mb/hour and increase linearly with the sample frequency, length of the recording, and number of channels.
- It is not feasible for our researchers to each store copies of the corpora on local disk space. Instead, researchers store around 50 GB of EEG data on their personal computers at any given time.

Release	Length (hrs.)	Est. Size (GB)
TUH EEG	15,757	567.2
TUH EEG Abnormal	1,142	41.1
TUH EEG Epilepsy	833	30.0
TUH EEG Seizure	425	15.3
TUH EEG Slowing	27	1.0

- It is therefore undesirable to have multiple copies of the same data between researchers. Instead, a single central location should be used so that all researchers are using the same data.

Retrieval of Specific EDF Files

- The visualization tool previously assumed that EDF files were locally available. The software can now also request the EDF files from the server.
- When attempting to open an EDF file, the software can look at the user's local file system for available EDF files. All required data is stored on local disk.
- Retrieving an EDF file from the server utilizes the EDF Retrieval API. Providing a file path on the server allows the software to retrieve that EDF file. Only a single EDF is stored on local disk.
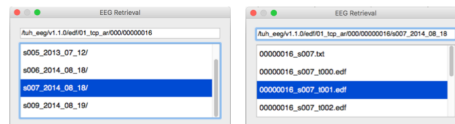
- To facilitate the annotation process, the software also checks the location of the requested EDF file for an associated annotation file. If it exists, this annotation file will also be sent to local disk, so that users can also view annotations.
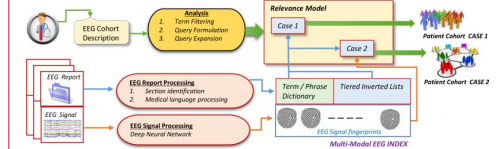

Design of EDF Retrieval API

- An API was developed to efficiently interface with the server that stores the EDF files.
- This API is protected by means of a private key. Anyone without a private key will not have access to the API. This helps to protect the data from potential malicious attacks.
- The API is accessed through an HTTP GET Request. This request will contain the private key of the user, and a file path on the server.

- If the file path is a path to a directory, the API returns all files and directories in the given directory on the server. This is returned as a list of strings.
- If the file path is a path to a file, the API streams that file down to the user's local disk storage. In the context of the visualization software, these files will either be text reports or EDF files.

API Usage in NIH Cohort Retrieval

- The API was developed with the visualization tool in mind. It seamlessly integrates into the workflow of the already developed software.
- Assuming the entire TUH EEG Corpus is available locally, the system can work from local disk.

- This corpus is often not locally available, due to its size. Users with access to the server however, can utilize the EDF Retrieval API.
- Under this implementation no local EDF data is required. The software forwards all results from the Cohort Retrieval system directly to the EDF Retrieval API, providing access to the EEG events.
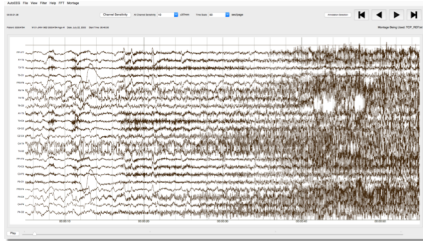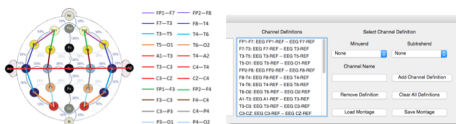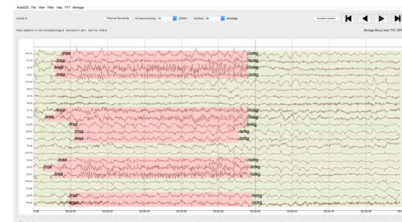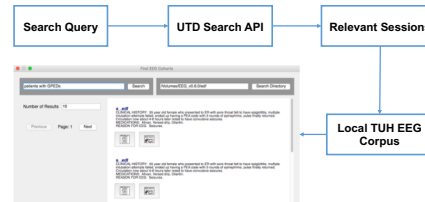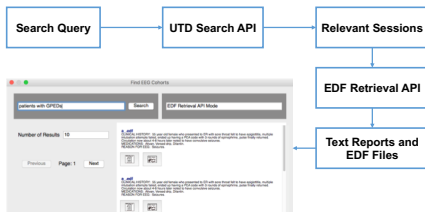

Cohort Retrieval

- The cohort retrieval system allows users to find EEGs most relevant to a search query.
- Keyword search and natural language interfaces are supported (e.g., "show me all the EEGs that have...").
- Using both unstructured text reports and automatically extracted EEG signal events, the cohort retrieval system finds the most relevant EEG sessions to the inputted search query.

- This GUI establishes the ability to view multiple windows of EEG signals, along with their respective medical reports. This effectively creates a search engine interfacing with the TUH EEG Corpus
- Each new plotting window contains all functionality provided in the framework that the original visualization tool utilizes.

Summary

- The visualization tool has been upgraded to allow for users to stream EDF files into the software directly from the server, removing local disk storage dependencies.
- An API was developed to streamline this process, so that little to no user intervention is required to retrieve EDF files from the server.
- Users can navigate the server's file system to find EDF files to visualize and/or annotate.
- The API is also utilized in the NIH-funded cohort retrieval system, creating a fully functioning search engine for the TUH EEG Corpus. This can support clinical work, education and research.

Future Work

- Currently, the software stores all data received from the API temporarily on local disk storage. This can be avoided by forcing the API to return chunks of EDF data that can instead be stored in memory.
- This software, and many of its features, lends itself to parallel processing. Retrieving query results or processing EEG channels could be done in parallel, instead of sequentially, to increase performance of cohort retrieval and EEG visualization.

Acknowledgements