WebGlobe - A cloud-based geospatial analysis framework for interacting with climate data

Arun Sharma, Syed Mohammed Arshad Zaidi,
Varun Chandola
University at Buffalo, State University of New York
Buffalo, New York
arunshar,szaidi2,chandola@buffalo.edu

Melissa R. Allen, Budhendra L. Bhaduri
Oak Ridge National Laboratory
Oak Ridge, Tennessee
allenmr,bhaduribl@ornl.gov

ABSTRACT

While climate models have evolved over time to produce high fidelity and high resolution climate forecasts, visualization and analysis of the output of the model simulations has been limited, typically constrained to single dimensional charts for visualization and basic aggregate statistics for analytics. Same is true for the large troves of observational data available from meteorological stations all over the world. For richer understanding of climate and the impact of climate change, one needs computational tools that allow researchers, policymakers, and general public, to interact with the climate data. In this paper, we describe, webGlobe, a browser based GIS framework for interacting with climate data, and other datasets available in similar format. webGlobe is a unique resource that allows unprecedented access to climate data through a browser. The framework also allows for deploying machine learning based analytical applications on the climate data without putting computational burden on the client. Instead, webGlobe uses a client-server framework, where the server, deployed on a cloud infrastructure, allows for dynamic allocation of resources for running compute-intensive applications. The capabilities of the framework will be discussed in context of a use case: identifying extreme events from real and simulated climate data using a Gaussian process based change detection algorithm.

KEYWORDS

Visualization, Analysis, Climate Data, Geospatial Analysis

ACM Reference format:

Arun Sharma, Syed Mohammed Arshad Zaidi, Varun Chandola and Melissa R. Allen, Budhendra L. Bhaduri. 2018. WebGlobe - A cloud-based geospatial analysis framework for interacting with climate data. In *Proceedings of 7th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data, Seattle, WA, USA, November 6, 2018 (BigSpatial 2018)*, 5 pages. DOI: 10.1145/3282834.3282835

ACM acknowledges that this contribution was authored or co-authored by an employee, or contractor of the national government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only. Permission to make digital or hard copies for personal or classroom use is granted. Copies must bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. To copy otherwise, distribute, republish, or post, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
BigSpatial 2018, Seattle, WA, USA
© 2018 ACM. 978-1-4503-6041-8/18/11...\$15.00
DOI: 10.1145/3282834.3282835

1 INTRODUCTION

Climate scientists and meteorologists agree that the climate has been changing and the rate of climate change, in particular, global warming, has been alarmingly rapid in the past few decades (see Figure 1), which has led to increased extreme events [12]. Such events impact the lives of the human population in many ways [16]. The reasons for climate change are hotly debated, though there is overwhelming support to the claim that humans are primary contributors to climate change [13]. Regardless of the reason, it is universally accepted, that the rising temperatures will indeed impact our lives in many ways and it is vital to understand these impacts and near and distant future. One way to understand the impact of current and future climate change is from data. The data could be current and historical measurements of various climate parameters or outputs of climate model simulations for near and distant future.

The goal of this paper is to introduce an integrated analysis and visualization framework, called *webGlobe*, that enables a broad community of stakeholders, including researchers, policy makers, educators, and general public, to better understand climate change and its global and regional impact, from data. Current solutions are either limited to a small set of predefined visuals, such as the one shown in Figure 1, or require considerable technical expertise to operate the available tools. The reason for these limitations is that the underlying climate data, both historical and future forecasts obtained by running climate model simulations, are difficult to access, massive to handle, and are available in formats that are difficult to handle within the more accessible GIS environments, such as Google Earth and Quantum GIS. In the absence of such integrated solutions, users typically string together series of tools to perform different aspects of the pipeline. Such solutions are inefficient and often exclude the users who do not possess necessary technical expertise and/or infrastructure to execute the different tools within the pipeline.

1.1 Use Case: Identifying interesting changes in local climate.

In this use case, the user is interested in identifying when the climate for a given location on earth exhibits a significant change from the past. This question is typically formulated for a particular climate parameter, e.g., *surface air temperature*. For instance, the output of climate model simulations produce graphics such as Figure 1, which indicate that the global temperatures are rising. But for a policymaker or an urban planner, the more interesting question is – when will the climate for a given location change

Figure 1: Surface temperature anomalies for land and ocean surface since 1880. Both raw and smoothed (five-year lowess smoothing) values are shown. Src: <https://climate.nasa.gov/scientific-consensus/>. Current understanding of climate change is limited to such global plots.

significantly? Statistical and machine learning driven methods have been proposed to answer such questions [8, 9].

2 CHALLENGES AND LIMITATIONS OF STATE OF ART SOLUTIONS

In order to develop solutions to address questions posed by the above use case, and other similar ones, three broad challenges need to be addressed:

- (1) *Availability of data.* This challenge has been alleviated in recent times with many public enterprises making climate data (remote sensing measurements and simulation outputs) publicly available through efforts such as Earth System Grid Federation [10], NASA Earth Data [3], and NOAA Climate Data Online [2]. However, most of these data repositories operate under different access protocols, which are challenging to navigate.
- (2) *Handling scale of data and analysis.* Global outputs of climate simulation runs are typically contained in large files stored in the Network Common Document Format or NetCDF [14]. For example, daily output for a single climate parameter (e.g., temperature) at a $1^\circ \times 1^\circ$ spatial resolution requires approximately 76 MB of storage for one year. Moreover, machine learning models have computational complexity that ranges from $O(N)$ to $O(N^3)$, where N denotes the size of the data, which is proportional to the length of time span being analyzed. Typical stakeholders do not possess adequate storage and computing infrastructure to support such analyses.
- (3) *Visualization of data and analysis outputs.* While several sophisticated tools for visualization of climate and weather data exist, most of these require a user to install a software on their personal computers, e.g., Ultrascale Visualization and Climate Data Analysis Tools (UVC DAT) [17], Panoply [15], Integrated Data Viewer [18]. On the other hand web browser based tools are more accessible to the general user community. However, existing browser based solutions (ClimateWizard [11], NASA WorldView [4]) provide a limited access to data sets and typically do not allow users to analyze the climate data.

3 SYSTEM ARCHITECTURE

The architecture of the proposed webGlobe system is outlined in Figure 2. The system follows a traditional client-server architecture, where a cloud-based server (webGlobe server) is responsible for the analytics and data serving needs. The current deployment of webGlobe¹ is hosted in a federated cloud environment, called *Aristotle*². The Aristotle federation is a National Science Foundation funded system consisting of data infrastructure building blocks, new allocations and accounting models, and a cloud metrics tool to support scientists with flexible workflows and analysis tools for large-scale data sets.

Figure 2: Architectural Overview of webGlobe

The general workflow is as follows. The users access the system through a browser based interface. In the current deployment, we use the GLOBUS identity management system [7] to manage the user authentication. The user can upload new climate data sets or access pre-loaded data sets. Three types of interactions are supported. First, is the visualization of the data sets as animated spatiotemporal data cubes or as plots (See Figure 3), second is the analysis of the data using a set of analytical tools, and third is by loading the data set into a *Jupyter Notebook*³ environment in which the user can analyze the data using various Python libraries. The output of analyses are also stored within the system and are globally available. A usage tracking system is implemented to ensure that the cloud resources are being optimally utilized.

3.1 WebGlobe Server

The webGlobe server has three roles. The first role is that of a *staging area* for the climate and weather data that is pulled from various data repositories across the world. When a user points to a desired data set (or a list of data sets), the server pulls the data into the cloud storage. This is done to allow efficient visualization and analysis of the data, without relying on the movement of large amounts of data directly between the client and the remote repositories over the Internet. Moreover, as described below, the data is converted from

¹<http://ewn-worldwind.ornl.gov:8080/ewn-webglobe/>

²<https://federatedcloud.org/>

³<http://jupyter.org/>

Figure 3: Screenshot of the browser based webGlobe client.

the native NetCDF into a format which can be stored in a distributed storage and thus allows for distributed analytics. The second role is to support big data analytics on the data set using distributed analysis methods. The third role is to allow users to interact with data within a python environment (Jupyter notebooks).

Data Storage. The widely used NetCDF format allows for storing scientific data sets in a single self-describing, machine independent, file format which is optimized for array oriented data. However, the binary format is not directly ingestible into most analytical and visualization frameworks. Tools provided by the Hierarchical Data Format (HDF) community, such as the NASA HDF products⁴ allow loading such data sets directly into analysis environments such as Apache Spark [20]. However, they do not support distributed analysis which is essential when the size of the data is large. Recently, tools such as SciSpark [19] have been developed to provide a bridge between NetCDF formats and Spark. We have developed a light-weight version of SciSpark in which the NetCDF data (or a list of data sets⁵) is loaded and converted to a *Resilient Distributed Data* (RDD) object, which refers to an immutable distributed collection of object. The data is stored as a collection of <key, value> pairs, where each key encodes the spatial attributes (latitude and longitude) of a given location, and the corresponding value consists of a time-series of measurements for the climate parameter. This distributed strategy is optimal for the type of analyses that are supported in the framework. One can add spatial indices to the keys to support efficient spatial queries. A typical <key, value> pair appears as: <-17.70, 89.04>, <23.04, 23.98, 20.14, 19.24, . . . >. The RDD objects are stored in a Hadoop Distributed File System (HDFS). The meta data associated with the data set is extracted and stored into an auxiliary MySQL database.

Analytics. We have implemented several machine learning based and statistical methods in the webGlobe server which can be applied to the datasets stored in the HDFS as RDD objects. All methods are implemented in the Spark environment and leverage the inherent parallelism in the methods. For instance, the change detection algorithm [9], operates on each spatial location independently. The readers are referred to the original paper for more details. Figure 4

shows the excellent speedup achieved by using the webGlobe system to run the change detection code on NOAA climate data for 32 years (1979-2011).

Figure 4: Speedup obtained for the Gaussian Process based change detection algorithm implemented in webGlobe on NOAA climate data during 1979-2011.

Notebook Server. While the current analysis algorithms implemented in webGlobe are relevant to a broad segment of the user community, the technically savvy users often have their own analytical routines that they would like to apply to the data sets. To support this functionality, we allow users to fork Jupyter notebooks which provide a browser based analysis environment using Python. The notebook is dynamically created and already contains the implementations to load the selected data set as an RDD. After that, the user has flexibility to transform the data into any other data structure and analyze it. Since the notebook is deployed on the cloud, the computational burden on user's resources is minimal.

Dynamic Resource Allocation. One of the key strengths of cloud computing is its elasticity. The webGlobe framework exploits this to the fullest extent. When an analysis job is submitted, the system dynamically creates a Spark cluster with desired number of nodes on the cloud. The cluster is pre-configured to read data from the HDFS as a distributed RDD. After processing the data, the spark cluster is terminated and the resources are returned to the cloud system.

4 WEBGLOBE CLIENT

The browser-based client interface (See Figure 3) is built using the WebWorldWind library [1], which is a 3D virtual globe API for HTML5 and JavaScript. The interface is accessible across all major browsers and on all devices including desktops, tablets, and mobile phone screens. The features include: uploading climate data sets from remote locations, visualizing data as spatiotemporal animations and univariate plots, submitting analysis codes (including those corresponding to the one use case discussed earlier), and forking off Jupyter notebooks for further analysis.

A key advantage of building on top of the WebWorldWind library is that it allows integration of several other applications that have been built using the same library into webGlobe. For instance, the current webGlobe system also allows for ingesting other geospatial data sets available through traditional protocols such as the Web Map Service (WMS).

⁴<http://hdfEOS.org/examples/spark.php>

⁵In many cases, climate data is available as a sequence of files, typically one for each year.

Figure 5: Visualization capabilities within webGlobe. Left panel – spatiotemporal animations. Right panel – time series plotting using the Plot.ly JavaScript library.

5 RESULTS AND EVALUATION

We performed Gaussian process based change detection algorithm on NOAA climate [2] and NCEP/NCAR Reanalysis [5] datasets and compared with location-wise monthly events occurred over the years (1996-2011) which are available from NOAA Storm Events Database [6]. These events comprises of extreme hot events (such as droughts, excessive heat etc.) and extreme cold events (such as winter storm, blizzard etc). Figure 6 shows the overall evaluation of the algorithm on both datasets using accuracy as a performance metric.

Figure 6: Left panel – Accuracy values on NOAA and NCEP/NCAR data while detecting extreme cold events such as blizzard and winter storm. Right panel – Accuracy values on NOAA and NCEP/NCAR data while detecting hot events such as drought and extreme heat.

Figure 7 shows the indication of alarm raised by the change detection algorithm while detecting anomalies in temperature change using NOAA climate data [2] which demonstrates the extreme hot and cold weather events for a geographical location over the years. The extreme heat events comprises of droughts, excessive heat etc., while extreme cold events comprises of winter storm, blizzard etc.

6 CONCLUSIONS

The objective of this paper is to introduce webGlobe, a unique resource available to the community interested in understanding the impacts of climate change through spatiotemporal data. The framework is designed to address the various challenges that the stakeholders face in handling such data, including cross-platform support through a web browser based interface, a cloud computing based server support to handle compute and data infrastructure

Figure 7: Top panel – Alarms raised while detecting extreme hot events such as drought and extreme heat during 1998-2011. Bottom panel – Alarms raised while detecting extreme cold events such as blizzard and winter storm during 1996-2011.

requirements, and a flexible and scalable ecosystem to allow integration of different analytical methods.

7 ACKNOWLEDGEMENTS

This manuscript has been co-authored by employees of UT-Battelle, under contract DE-AC05-00OR22725 with the US Department of Energy. The authors would also like to acknowledge the financial and intellectual support for this research by the Integrated Assessment Research Program of the US Department of Energy's Office of Science, Biological and Environmental Research (DOE BER). This work is supported in part by NSF ACI-1541215.

REFERENCES

- [1] 2015. Web World Wind. webworldwind.org. (2015).
- [2] 2016. Climate Data Online. NOAA National Centers for Environmental Information. (2016).
- [3] 2016. National Aeronautics and Space Administration (NASA), Global Change Master Directory (GCMD). (2016).
- [4] 2017. NASA Worldview. <https://github.com/nasa-gibs/worldview>. (2017).
- [5] 2018. NCEP/NCAR Reanalysis data. NOAA NOAA Earth System Research Laboratory. (2018).
- [6] 2018. Storm Events Database. NOAA National Centers for Environmental Information. (2018).
- [7] R. Ananthakrishnan, J. Bryan, K. Chard, I. Foster, T. Howe, M. Lidman, and S. Tuecke. 2013. Globus Nexus: An identity, profile, and group management platform for science gateways and other collaborative science applications. In *2013 IEEE International Conference on Cluster Computing (CLUSTER)*. 1–3. <https://doi.org/10.1109/CLUSTER.2013.6702693>
- [8] Varun Chandola and Ranga Raju Vatsavai. 2010. Scalable Time Series Change Detection for Biomass Monitoring Using Gaussian Process. In *Proceedings of NASA Conference on Intelligent Data Understanding*.
- [9] Varun Chandola and Ranga Raju Vatsavai. 2011. A Gaussian Process Based Online Change Detection Algorithm for Monitoring Periodic Time Series. In *Proceedings of SIAM Data Mining Conference*.
- [10] Earth System Grid Federation 2017. <https://esgf.llnl.gov/>. (2017).
- [11] Evan H. Girvetz, Chris Zganjar, George T. Raber, Edwin P. Maurer, Peter Kareiva, and Joshua J. Lawler. 2009. Applied Climate-Change Analysis: The Climate Wizard Tool. *PLOS ONE* 4, 12 (12 2009), 1–19.
- [12] Stephanie C. Herring, Andrew Hoell, Martin P. Hoerling, James P. Kossin, Carl J. Schreck III, and Peter A. Stott (Eds.). 2016. Explaining extreme events of 2015 from a climate perspective. *Bulletin of the American Meteorological Society* 97, 12 (2016).

- [13] IPCC. 2014. *Climate Change 2014: Synthesis Report. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press. 151 pages.
- [14] Harry L. Jenter and Richard P. Signell. 1992. NetCDF: A Public-Domain-Software Solution to Data-Access Problems for Numerical Modelers. (1992).
- [15] Panoply 2017. <https://www.giss.nasa.gov/tools/panoply/>. (2017).
- [16] Jonathan A. Patz, Diarmid Campbell-Lendrum, Tracey Holloway, and Jonathan A. Foley. 2005. Impact of regional climate change on human health. *Nature* 438, 7066 (11 2005), 310–317.
- [17] Emanuele Santos, Jorge Poco, Yaxing Wei, Shishi Liu, Bob Cook, Dean N. Williams, and Cludio T. Silva. 2013. UV-CDAT: Analyzing Climate Datasets from a User’s Perspective. *Computing in Science and Engineering* 15, 1 (2013), 94–103.
- [18] Unidata 2015. Integrated Data Viewer version 5.1. <http://www.unidata.ucar.edu/software/idv/index.html>. (2015).
- [19] Brian Wilson, Rahul Palamuttam, Kim Whitehall, Chris Mattmann, Alex Goodman, Maziyar Boustani, Sujen Shah, Paul Zimdars, and Paul M. Ramirez. 2016. SciSpark: Highly interactive in-memory science data analytics. In *2016 IEEE International Conference on Big Data, BigData 2016, Washington DC, USA, December 5-8, 2016*. 2964–2973.
- [20] Matei Zaharia, Reynold S. Xin, Patrick Wendell, Tathagata Das, Michael Armbrust, Ankur Dave, Xiangrui Meng, Josh Rosen, Shivaram Venkataraman, Michael J. Franklin, Ali Ghodsi, Joseph Gonzalez, Scott Shenker, and Ion Stoica. 2016. Apache Spark: A Unified Engine for Big Data Processing. *Commun. ACM* 59, 11 (2016), 56–65.