Project : Develop Pipeline for High-Throughput Visualisation on Google Earth Engine

Past experience with this community

I have no prior experience with this community but I find the project quite interesting and also matches my skill set and would love to take up this project throughout the internship period and make further contributions even afterward .

Over the past month, I've had the privilege of collaborating with the FLINT, GCBM, and Moja Global community. Exploring these projects for the first time filled me with excitement, knowing that my contributions could make a real impact on analytics in the field. The thought of working on solutions utilised at such a massive scale was both exhilarating and humbling.

Initially, the projects seemed daunting, and it took some time for me to grasp their intricacies fully. However, once I gained a deep understanding, I made it a point to document every insight that crossed my mind. My documentation serves as a resource for others facing similar challenges, providing clarity and guidance. You can explore my documentation of
FLINT
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/FLINT.md ,
GCBM
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/The%20Generic%20Carbon%20Budget%20Model%20(GCBM)%20.pdf

Each weekly task presented an opportunity for me to pour my heart and soul into my contributions. For instance, during Week 1, I tackled the challenge of implementing an algorithm to summarise geospatial time series. After proposing my solution to the community and discussing it with my mentor, @simple shell, I meticulously documented the entire process, from pseudocode to analysing time and memory complexities. You can explore my Week 1 task 1

https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/WEEK1-Task1.md

During Week 2, while implementing another algorithm, I encountered gaps in my generated time series data. However, I saw this as an opportunity to optimize my algorithm further. Through research and experimentation, I devised innovative solutions like moving window smoothing and gap-filling interpolation. The images I obtained along the way are documented
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/Regular%20NDVI%20chart.png ,
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/ee-chart.png , and
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/gap%20filled%20NDVI%20time%20series.png
And my complete record can be viewed here
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/Geospatial%20Time%20series%20processing%20using%20GEE.pdf

Excitedly, I shared my progress with the Moja Global team and eagerly awaited their feedback. Progressing to the next level of optimization, I delved deeper into various aspects, exploring techniques such as image collection filtering, chart generation, NDVI generation, cloud masking, parallel processing, data aggregation, batch processing, and many more. These endeavors were meticulously documented to capture the breadth of my exploration and learning journey which is available here
https://github.com/NIKITA-M-VERMA/Outreachy-contribution-phase-2024-/blob/main/WEEK2-Task%202%20Performance%20Optimization%20%20(1).pdf

During my time in the community, I regularly reached out to my mentor when I had questions, either on Slack or through email. I also talked with other applicants in my team, sharing what I learned and asking for their thoughts on my work. Whenever I made progress or learned something new, I made sure to write it down and share it with everyone so they could benefit too. Engaging in discussions within community channels and personal direct messages with applicants and mentors alike, I relished the experience of being an active participant in this vibrant community.

Past experience with other communities

During my early time at university, I have taken part in an open-source contributing event organised by Kubernetes and CNCF , that's where I got to know about open source, how open source works, how communities work, etc. I have taken part in Girl script summer of Code, an open-source community event working on different

project ideas. It starts with a very little code base and eventually builds the codebase based on contributions from participants.
But during the contribution period only, I became an open-source contributor contributing to an organisation like Moja Global .

Relevant Projects

Over the past 1.5 years, I've honed my skills in C++, C, JavaScript, and Python, laying a robust groundwork for tackling diverse projects. My familiarity with Java facilitated a smooth transition into Google Earth Engine coding.

During my academic tenure, I actively contributed to open-source initiatives such as the CNCF Glossary project, where I played a pivotal role in enhancing documentation standards. Additionally, I developed a Hospital Management System, leveraging PHP, MySQL database, Bootstrap, Modal, HTML, CSS, and Ajax technologies. This project provided me with hands-on experience in database management, SQL querying, and relational database management systems (RDBMS). You can explore the Hospital Management System project
https://github.com/NIKITA-M-VERMA/The-Hospital-Management-System

On a personal level, I initiated the "First its Me" project to promote self-health awareness among college peers, significantly reducing stress and anxiety levels. You can view the "First its Me" project
https://github.com/NIKITA-M-VERMA/First-it-s-Me-
.
During my first year of college, I embarked on the development of a CCTV security project, enhancing my proficiency in Python programming. You can view the CCTV security project
https://github.com/NIKITA-M-VERMA/CCTV_Camera

Currently, I am as per curriculum engrossed in designing an Arduino-based 2-way Communication system using Bluetooth modules, delving into Arduino basics and prototype development.

Please describe which Google Summer of Code communities and projects you are applying for, and provide mentor contact information
I am not taking part in Google Summer of code. I preferred Outreachy because of its timeline that is more suitable for me so that I can give my best in it.

Community-specific Questions
I have most of my questions clarified throughout the contribution period doing research on Moja. Also being a member of the slack community, I understand how the community works and how things are being done and managed inside the community.

Still, some questions that I have is
What are the intended goals or objectives once the complete pipeline for high-throughput visualisation on Google Earth Engine is developed and implemented on a regular basis?
Also, I have seen a very good culture at Moja, the time mentors and community people space to make beginners understand how everything works is really appreciated. How did this culture maintain throughout the years?

……

Project Timeline:

Throughout this timeline, although it is mentioned repeatedly that work will be submitted to mentor(s) and community for feedback and review after a certain number of days, **code will be pushed to repo on regular basis so that daily progress could be tracked easily. Similarly, regular feedback or comments (if any) are welcomed from mentor(s) and community on daily code check-ins** so as to stay on track.

Phase 1: Research and Planning

Week 1 (May 27 - June 2): Algorithm Research and Understanding

- Conduct research on algorithms suitable for database aggregation and geospatial operations.
- Review existing Google Earth Engine scripts to understand their structure and functionality.
- Begin exploring potential approaches to extend existing scripts for prototype development.
- Establish a clear understanding of project goals and objectives.

  [During this week, the primary focus will be on building a solid foundation of understanding regarding the project objectives, existing workflows, and technical aspects of the pipeline. By delving into project documentation and engaging in preliminary discussions, we aim to establish a clear roadmap for future tasks and development activities.]

Week 2 (June 3 - June 9): Script Extension and Prototyping

- Extend existing Google Earth Engine scripts to create prototype solutions based on researched algorithms.
- Implement initial versions of pipeline components to test functionality and feasibility.
- Evaluate different approaches for database aggregation and geospatial operations within Google Earth Engine.
- Start integrating prototype solutions into a cohesive pipeline framework.

[During this week, the focus will be on creating detailed documentation of the project's workflow, similar to the example provided. By meticulously documenting the code flow, we aim to gain insights into potential areas for improvement and optimization, laying the groundwork for further development activities in the following weeks.]

Phase 2: Solution Development

Week 3 (June 10 - June 16): Performance Evaluation Planning

- Define metrics and criteria for evaluating the performance of the pipeline.
- Plan methodologies for calculating and assessing the expected performance of the proposed solution.
- Establish benchmarks and criteria for scaling the solution from smaller subsets to larger datasets.
- Outline strategies for profiling and benchmarking the pipeline's performance.

[During this week, the primary focus will be on profiling and optimising the existing pipeline to handle the massive simulation results efficiently. By analysing the performance characteristics and identifying areas for improvement, we aim to develop innovative approaches to post-processing that enhance the performance and utility of the pipeline while ensuring compliance with IPCC reporting guidelines.]

Week 4 (June 17 - June 23): Profiling and Analysing Pipeline Components

- Initiate profiling of the pipeline components specific to the high-throughput visualisation on Google Earth Engine project.

- Conduct detailed analysis of each component within the pipeline to identify performance bottlenecks and areas for optimization.
- Develop a comprehensive control flow diagram of the pipeline components, illustrating the sequence of operations and interactions between different modules.
- Benchmark each component of the pipeline to measure its performance and efficiency in processing spatially explicit outputs.
- Compare the performance metrics of individual components to identify disparities and analyse the implications for overall pipeline efficiency.

[During this week, the focus will be on profiling and analysing the components of the pipeline specific to the project's objectives. By conducting detailed analysis and benchmarking, we aim to gain insights into the performance characteristics of the pipeline and identify opportunities for optimization. The development of a control flow diagram will provide a visual representation of the pipeline's workflow, facilitating further analysis and optimization efforts.]

Phase 3: Optimization and Scaling

Week 5 (June 24 - June 30): Scaling Assessment

- Extend profiling efforts to include FLINT.Example models relevant to the high-throughput visualisation on Google Earth Engine project.
- Develop control flow diagrams for each component within the example models, capturing the sequence of operations and interactions.
- Ensure that the control flow diagrams maintain consistency with the overall pipeline's control flow diagram, allowing for seamless integration and comparison.
- Determine strategies for scaling the solution from example subsets to larger benchmark datasets.

Comparing the performance of different example models to gain insights into their relative efficiency and effectiveness in processing spatially explicit outputs.Document the profiling results, control flow diagrams, and comparative analysis, providing clear documentation of the example models' performance characteristics and areas for enhancement.

Week 6 (July 1 - July 7): Analysis and Optimization Implementation

- Analyse the control flow diagram developed for the pipeline for high-throughput visualisation on Google Earth Engine, examining the sequence of operations and interactions between components.
- Compare the control flow diagrams of different pipeline components to identify similarities and differences, aiming to ensure consistency and efficiency throughout the workflow.
- Conduct benchmarking of each component within the pipeline to measure its performance and efficiency in handling spatially explicit outputs.
- Analyse the benchmarking results to identify components that exhibit suboptimal performance or potential bottlenecks, prioritising them for optimization.
- Begin implementing optimizations identified during the analysis phase, leveraging best practices in algorithm design and software engineering to enhance pipeline efficiency.

Discussing the analysis findings and gather insights into potential optimization strategies, considering factors such as algorithmic complexity and resource utilization.Conduct initial testing of implemented optimizations to assess their impact on pipeline performance and validate their effectiveness in improving throughput.

[During this week, the focus will be on analysing the control flow diagram, benchmarking results, and implementing optimizations to enhance the performance of the pipeline for high-throughput visualisation on Google Earth Engine. By conducting thorough analysis and collaborating with stakeholders, we aim to identify and address areas for optimization, ultimately improving the efficiency and utility of the pipeline.]

Phase 4: Implementation and Evaluation

Week 7 (July 8 - July 14): GCBM Integration

- Set up GCBM (Generic Carbon Budget Model) on local machines for development.
- Dive into the codebase of small and medium GCBM models to understand their intricacies.
- Focus on generalising the FLINT Framework by incorporating GCBM.
- Pay close attention to generic parameters that impact code performance.
- Ensure thorough understanding of GCBM code flow to facilitate seamless integration with FLINT.
- Begin initial runs of GCBM on the local machine to test functionality and identify potential issues.

Collaborate with team members to discuss any challenges encountered during integration.

Week 8 (July 15 - July 21): GCBM Profiling and Control Flow Analysis

- Conduct profiling of small and medium GCBM models to gather performance data.
- Analyse profiling results to identify areas of improvement and optimization.
- Develop control flow diagrams for each component within the FLINT pipeline.
- Ensure control flow diagrams accurately depict the sequence of operations in GCBM models.
- Benchmark each component of FLINT in conjunction with small and medium GCBM models.

Compare benchmarking results to assess the impact of GCBM integration on pipeline performance.Document profiling, control flow analysis, and benchmarking results for reference and future optimization endeavours.

Phase 5: Documentation and Finalization

Week 9 (July 22 - July 28): Analysing Model Complexity

- Extract profiling and benchmarking results for FLINT, FLINT.Example models, and GCBM Models.
- Develop a code to compare and visualise the profiling results stored in JSON files.
- Analyse the profiling and benchmarking data to understand sources of complexity and overhead in each model.
- Identify components or processes within the pipeline that contribute significantly to runtime or resource consumption.

  Discuss findings with the team to prioritise optimization efforts and address identified issues.Document insights gained from the analysis for future reference and optimization strategies.

Week 10 (July 29 - Aug 4): Bottleneck Identification and Analysis

- Identify potential bottlenecks within the pipeline for high throughput visualisation on Google Earth Engine.
- Analyse the reasons behind these bottlenecks to understand their impact on performance.
- Explore potential solutions and optimization techniques to address identified bottlenecks.

- Prioritise optimization efforts based on the severity of bottlenecks and their impact on pipeline efficiency.
- Implement optimizations to improve the performance of the pipeline and reduce processing time.
- Test optimised code to ensure that the proposed solutions effectively address identified bottlenecks.

Document the optimization process, including the identification of bottlenecks, analysis of reasons, and implementation of solutions.Communicate findings and progress with stakeholders to keep them informed of optimization efforts.Review and refine optimization strategies as necessary to achieve desired performance improvements.

Week 11 (Aug 5 - Aug 11): Performance Evaluation and Optimization

- Compute the overall performance metrics of the pipeline for high throughput visualisation on Google Earth Engine.
- Analyse the results of performance evaluations to assess the efficiency and effectiveness of the pipeline.
- Identify areas where the pipeline may still have bottlenecks or areas for improvement.
- Implement additional optimizations to address any remaining bottlenecks and enhance overall performance.
- Test the optimised code to evaluate its impact on benchmarking and profiling results.

Monitor the performance of the pipeline under various scenarios to ensure consistent efficiency.Document the outcomes of performance evaluations and optimization efforts for future reference.Communicate the results of performance evaluations and optimization efforts with stakeholders.Iterate on optimization strategies as necessary to achieve optimal performance of the pipeline.

Week 12 (Aug 12 - Aug 18): Documentation and Optimization

- Optimise the code to improve results and overall performance of the pipeline for high throughput visualisation on Google Earth Engine.
- Create a template for profiling and benchmarking that can be used for future projects or models.

- Develop a systematic way to store and organise results from previous models or iterations of the pipeline.
- Create documentation outlining the pipeline's functionality, usage, and best practices.
- Prepare any additional resources or materials to support the use and maintenance of the pipeline.

  Document the optimization process, including the changes made and their impact on results.Ensure that all code changes are well-documented and easily understandable for future reference.Finalize documentation and optimization efforts to conclude the project successfully.

Week 13: August 19 - August 23
Backup days, for any backlogs, bugs or pending tasks.

- Working on feedback or additional work (if any) asked by mentors and rest of the community on the tasks done so far.
- Review and refine documentation to ensure clarity and completeness.
- Submitting all the remaining work to mentor.
- Writing user as well as developer documentation.

This timeline ensures a structured approach to developing the pipeline for high-throughput visualisation on Google Earth Engine, allowing for thorough understanding, prototyping, performance evaluation, optimization, and documentation within the internship duration.