SATELLITE BASED CROP MONITORING SYSTEM (SCMS)

Project Stakeholders:
Dr. Mahendra Bhandari (Mahendra.bhandari@ag.tamu.edu)
Dr. Ashutosh Tiwari (ashutosh.tiwari@ag.tamu.edu)

Project Advisor:
Dr. Lan Zhou (lzhou@tamu.edu)

Written By:
Saransh Agrawal
Ramana Heggadal Math
Phanender Chalasani

Date: November 20, 2024

1 Introduction

Farmers face significant challenges in accurately predicting crop yields and managing resources due to limited access to real-time, actionable insights. Traditional methods, such as relying on historical data or basic field observations, fail to account for dynamic factors like weather variability, soil health changes, and the spread of crop diseases.

Advanced data-driven systems are essential for addressing these issues. While UAV data provides high-resolution, localized insights, it is constrained by high costs, limited coverage, and operational complexity. In contrast, satellite data is scalable, cost-effective, and offers consistent, large-scale monitoring over time, making it more suitable for widespread agricultural applications. However, leveraging satellite data comes with its own challenges, such as handling large file sizes, API constraints, and the need for localized cloud cover checks to ensure data relevance to specific Areas of Interest (AOIs).

To bridge this gap, we provide a web based tool for a system that automates data sourcing, incorporates AOI-specific cloud filtering, and analyzes crop health using vegetation indices (VIs) and time-series data. We also create predictive models for predicting crop yield by analyzing NDVI and GCI values. We also explore the use of weather data which could further enhance yield predictions.

Beyond empowering farmers with real-time crop health monitoring and accurate yield forecasts, this system will also serve as a robust tool for researchers. By automating data acquisition and processing, it will enable researchers to conveniently source and analyze satellite data without worrying about the complexities of data filtering and pre-processing, facilitating deeper studies into agricultural patterns and technologies.

2 Data

Year	Seeding Date	Defoliation Date	Harvest Date
2020	2020-02-29	2020-07-13	2020-08-03
2021	2021-02-27	2021-07-27	2021-08-13
2022	2022-03-15	2022-07-19	2022-08-04
2023	2023-03-05	2023-07-18	2023-08-01

Table 1: Agricultural Dates for 2020-2023 cotton crop. Seeding date, also known as the planting date, marks the beginning of the cotton growing season. This is when cotton seeds are planted in the field. Defoliation is a process applied to cotton plants before harvest to remove leaves and prepare the cotton for picking. The harvest date is when the mature cotton is collected from the field.

The Satellite-based Crop Monitoring System (SCMS) relies on satellite imagery as its primary data source to monitor and analyze agricultural activities. The image is sourced from PlanetScope, a high-resolution imaging platform by Planet Labs PBC (2024), offering a 3-meter spatial resolution across four spectral bands: Red, Green, Blue (RGB), and Near-Infrared (NIR). These characteristics make PlanetScope data ideal for vegetation analysis and yield prediction.

The Normalized Difference Vegetation Index (NDVI) and Green Chlorophyll Index (GCI) are commonly derived from satellite imagery to assess vegetation health and chlorophyll content, respectively. For our analysis, we utilize 16-bit integer satellite images with a spatial resolution of 3m by 3m. NDVI is calculated using the near-infrared (NIR) and red bands according to the formula:

$$\mathrm{NDVI} = \frac{\mathrm{NIR} - \mathrm{Red}}{\mathrm{NIR} + \mathrm{Red}}$$

GCI, on the other hand, uses the NIR and green bands and is computed as:

$$\mathrm{GCI} = \frac{\mathrm{NIR}}{\mathrm{Green}} - 1$$

These indices provide valuable information about vegetation density, health, and chlorophyll content across the landscape.

For our modeling approach, we first extract NDVI and GCI values for each pixel in the satellite image. We then average these values over a 3x3 pixel area, where each pixel represents a 3m x 3m area on the ground. This averaging process corresponds to a single Field Identification Number (FID), representing a 9m x 9m area on the ground. We collect labeled dataset $D_1 \subseteq \{(x, y)\}$ where $x = \{\mathrm{NDVI}, \mathrm{GCI}, t\}$ where $t \in \mathbb{Z}$ represent number of days after seeding and $y \in \mathbb{R}$ is the yield value in kgs. The dataset D contains as many examples as there are FIDs across years 2020-23.

We create another dataset D_2, where each input variable x is augmented with additional weather-related features collected from the nearest weather station. Specifically, each $x_i = \{NDVI, GCI, t, max\_temp_t, min\_temp_t, dew_t, humidity_t, precipitation_t, solarradiation_t, moonphase_t\}$. This enhancement involves appending the relevant weather data for each day to every Field Identification Number (FID).

3 Methodology

We developed a user-friendly web application designed for crop yield prediction, specifically tailored for cotton producers. This platform serves as a comprehensive tool, enabling farmers to efficiently manage and analyze their farms. By providing insights into vegetation indices and expected yields, the application facilitates timely interventions to prevent crop diseases throughout the growth cycle. Additionally, it supports the management of multiple farms, empowering users to make informed economic decisions based on predictive yield analyses.

3.1 System Design

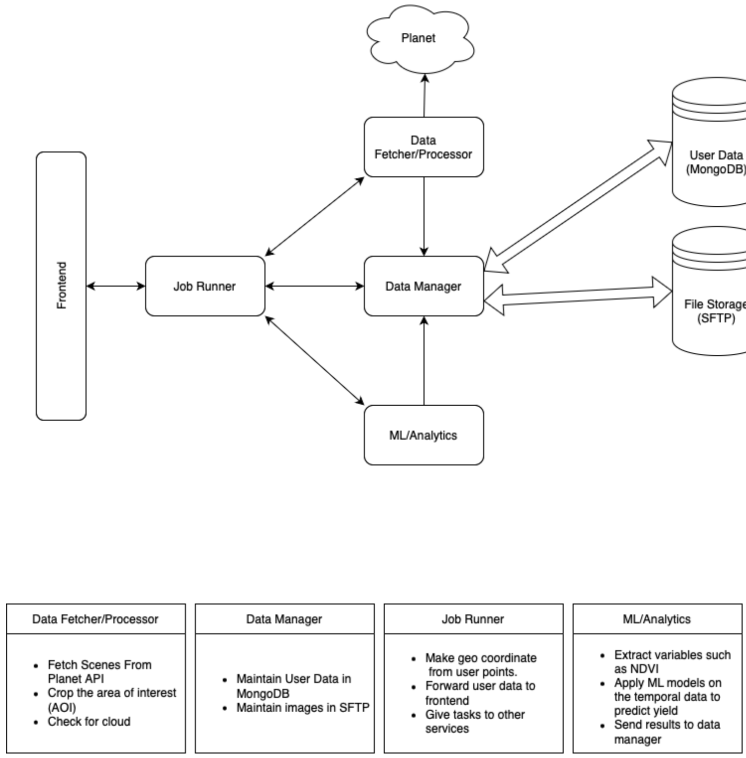

Figure 1: System design showcasing each micro-service and flow of requests between them.

The Satellite-based Crop Monitoring System (SCMS) employs a micro-service architecture, with each micro-service designed to perform a specific task. Daya et al. (2016) shows how this modularity ensures scalability, adaptability to changes in usage demands, and efficient separation of responsibilities. By isolating functionalities into distinct modules, SCMS allows for seamless code iteration and deployment cycles. Each module is containerized using Docker Merkel (2014), enabling flexible deployment configurations across machines. Communication between containers is facilitated through REST API Masse (2011), ensuring smooth interoperability.

Users will access the platform through a login page hosted by the frontend, which is accessible via a public IP address and domain name. This design ensures a seamless user experience while maintaining a robust, scalable backend infrastructure.

Our architecture provides several advantages:

- **Security:** Since the internal APIs are not publicly exposed, they inherently reduce the risk of attacks, ensuring data remains secure and protected from external threats.
- **Flexibility:** With Docker containerization, the system can be deployed on various cloud platforms such as AWS, Azure, or GCP without significant code alterations, making the solution highly portable across different environments.
- **Future Scalability:** The architecture is designed with future expansion in mind. Additional capabilities can easily be integrated by attaching new containers to the existing structure, eliminating the need to rewrite the foundational system.

3.2 Frontend

The Frontend serves as the user-facing component of SCMS, providing an intuitive and interactive interface for farmers and researchers. It is built using React.js Fedosejev (2015) and interacts only with the Job Runner module. The key functionalities of the frontend include:

- **User Registration and Authentication:** Securely handle user accounts.
- **AOI Marking:** Enable users to define and edit Areas of Interest (AOIs) on the map.
- **Project Creation:** Allow users to initiate projects by inputting key details, such as AOIs and timelines.
- **Invoking Requests:** Facilitate requests to fetch satellite imagery or analytics for specific timeframes.
- **Analytics Display:** Present date-specific and cumulative analytics derived from the backend.

The frontend ensures a seamless experience for users while offloading computational tasks to the backend modules.

3.3 Job Runner

The Job Runner acts as the orchestrator for all backend operations. It handles API requests from the frontend, delegates tasks to the appropriate modules, and consolidates results from various sources. This module is the only backend module with publicly exposed APIs, this ensures that sensitive user data is always protected via private APIs. Its responsibilities include:

- **User Authentication:** User sign-ups, verifying login credentials and maintaining authentication tokens for each login session.
- **Request Coordination:** Frame and route API requests from the frontend to corresponding micro-services, such as the Data Fetcher, Data Manager, or ML/Analytics modules.
- **Response Management:** Aggregate data and analytics results from backend modules into structured responses for the frontend to display.
- **Workflow Orchestration:** Manage dependencies between tasks to ensure smooth execution across modules.

3.4 Data Fetcher/Processor

The Data Fetcher/Processor module handles interactions with the Planet API for retrieving satellite images. It is responsible for automating data acquisition and performing essential pre-processing tasks. Key functionalities include:

- **API Interaction:** Formulate and execute API calls to fetch imagery for specific AOIs and timeframes.
- **Preprocessing:** Perform AOI cropping and cloud filtering to ensure the data is clean and relevant for analysis.

3.5 Data Manager

The Data Manager module is the core component for managing SCMS's diverse data storage needs. It abstracts the complexities of data storage and retrieval, providing a unified interface for other modules. The data is categorized as follows:

- **MongoDB Storage:** Handles metadata such as user profiles, AOI definitions, satellite image metadata, and analytics results.
- **SFTP Storage:** Stores large files, including raw satellite imagery (TIFF format) and other relevant plots.

Key functionalities of Data Manager include:

- **CRUD operations:** Manages users and project details, such as AOI, farm location and type of crop and analytics data.
- **Data Storage:** Receives satellite image data from Data Fetcher, links with appropriate user and project and store the file in SFTP server.

3.6 ML/Analytics:

The ML/Analytics module is responsible for applying advanced analytics and machine learning techniques to enable SCMS's predictive capabilities. It supports tasks such as:

- **Crop Yield Prediction:** Use time-series data, vegetation indices, and weather data to predict yields.
- **Variable Extraction:** Derive metrics like NDVI and GCI from satellite imagery for downstream modeling.

By isolating analytical workloads into its own module, SCMS ensures flexibility in scaling and upgrading resources as needed.

3.7 Yield prediction model

Random Forest Regressor Breiman (2001) is an ensemble method that combines the predictions of multiple decision trees to improve predictive accuracy and control overfitting. For our analysis we train two random forest models, one with data D_1 and another with utilizing weather data in data D_2. We provide the training hyperparameters of our models below. **Modeling:** $M_1(D_1)$ and **Modeling:** $M_2(D_2)$

- Number of Estimators for Random Forest: 100
- Full length tree (to capture subtle differences in variances)

The training results for both models are as follows:

- M_1 MAE = 3.66, RMSE = 5.55, R2 = 0.81
- M_2 MAE = 1.30, RMSE = 1.99, R2 = 0.97

4 Results

4.1 Crop Monitoring Tool

The user signs in and is directed to the Project Pane, an intuitive interface designed to present interactive UI elements for exploring and analyzing project data. The Project Pane displays all active projects. For each project, users can select a specific date range for analysis. Based on the selection, satellite images within the specified range are either dynamically downloaded from the PlanetAPI or served directly if previously downloaded. Cropped Area of Interest (AOI) images (as shown in figure2) are downloaded to the SCMS web app.

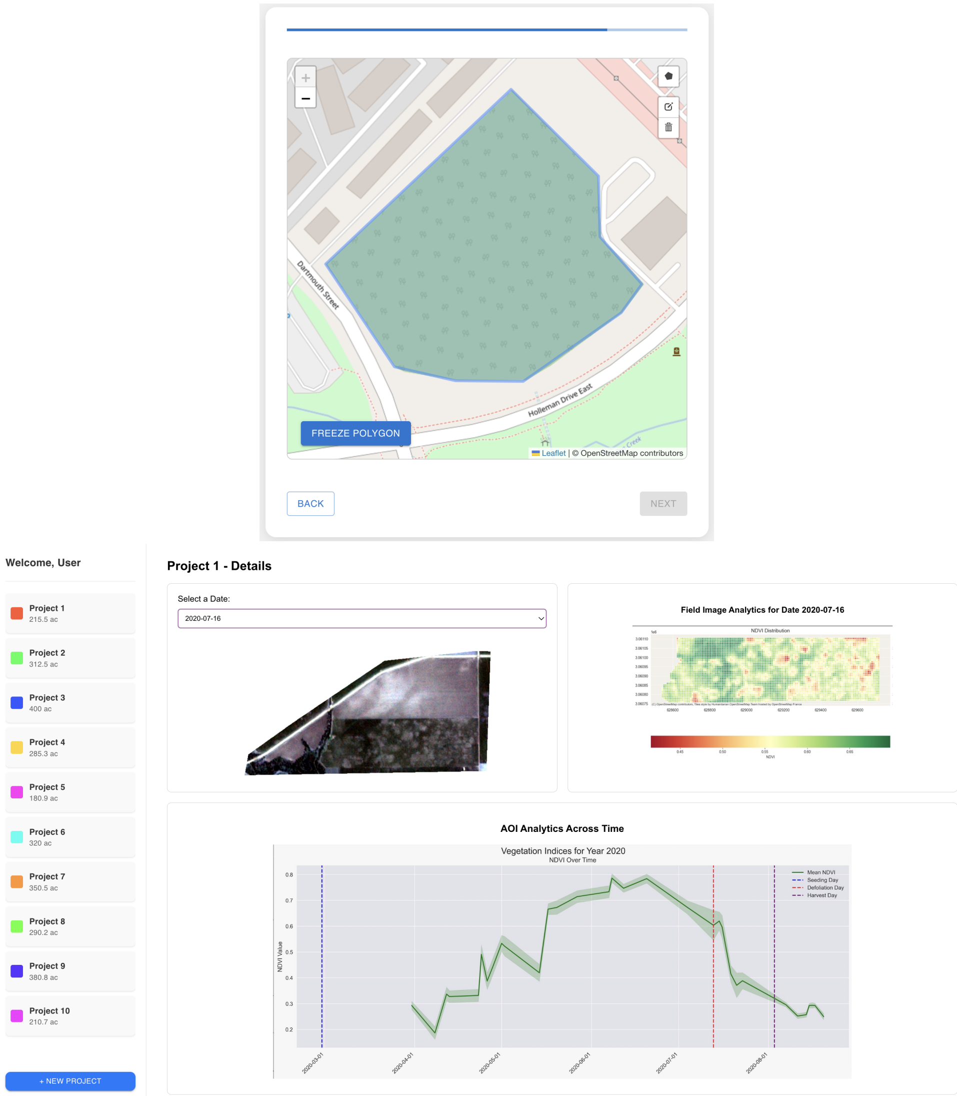

Figure 2: *top* Marking of AOI for a field while creating project. *bottom* Project pane showing procured satellite image, NDVI heat map and average NDVI across time.

The interface (figure2) includes detailed Visualization Panels offering insights derived from the image data. Metrics such as NDVI and GCI are displayed as field localizations and trends, enabling users to track vegetation health over time. These visualizations are located in the Analytics Pane, positioned at the bottom of the screen for easy access.

Additionally, users can select a specific timestamp from the timeline to view detailed results, including:

- Cropped Satellite Image corresponding to the AOI for the selected date.
- Grid Plots of NDVI and GCI, providing spatially resolved metrics for the AOI.
- Crop Yield Prediction, leveraging data collected up to the selected date to offer actionable insights.

4.2 Prediction analysis

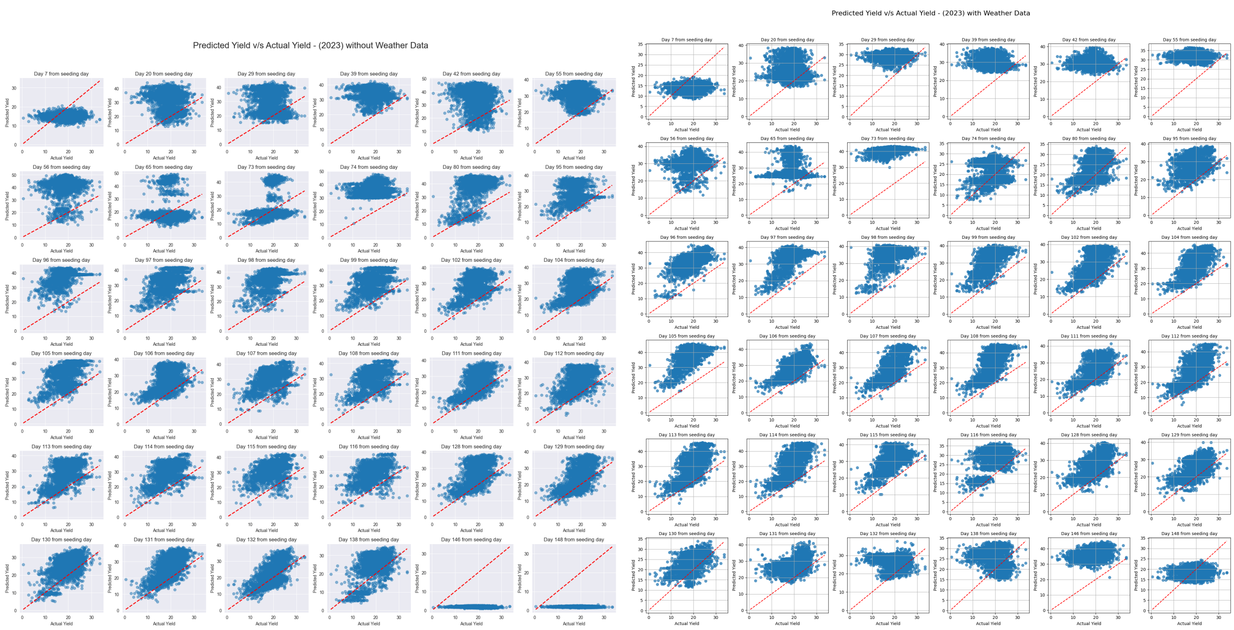

Figure 3: Prediction vs Actual graph (*left* model M_1 and *right* M_2) for satellite images taken across crop life, starting from 7 days after seeding (*top left*) to after defoliation (*bottom right*).

For the 2023 yer we have 36 images taken across the crop life cycle. We use both the models to predict the yield at each of the days when image is taken. Figure3 shows the predictions of model utilizing D_1 across 36 images. For the first image taken 7 days after planting, the model acts as a mean estimator and hence does not qualify for a good prediction. As we move along days we see the convergence of predictions along the actual value. The last two images were taken after defoliation date and hence predicts close to 0 yield.

In our attempt to utilize weather data with data D_2 the prediction results for year 2023 are shown in figure3. This method does not surpass the results observed with model M_1 showcasing a poor correlation of weather data with yield.

4.3 Crop Disease Localization

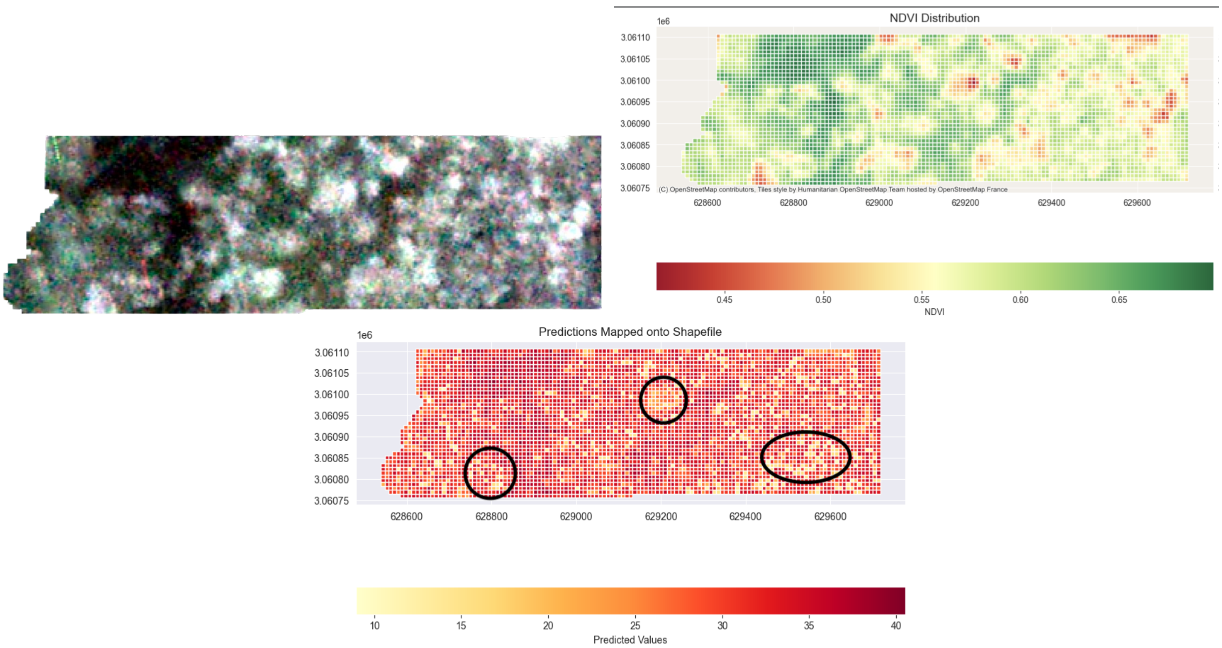

Figure 4: Image in the *top left* shows the satellite image of the farm taken a few days before defoliation. *top right* shows the heat map on NDVI values at the defoliation date, where red zones identify the disease ridden area. *bottom* shows the heat map of yield value predicted 10 days before defoliation and shows how yield prediction can help in early localization of crop disease.

With a good enough yield predictor we can also detect possible crop damaging diseases, which is useful for early intervention and saving the field. Possible interventions include spraying of chemicals or burning the affected area to save rest of the crop. In Figure 4 we showcase how a yield estimator heat map can be used to identify regions of possible infestation.

5 Conclusion

In conclusion, the Satellite-based Crop Monitoring System (SCMS) represents a significant advancement in agricultural technology, offering farmers and researchers a powerful tool for crop yield prediction and management. By leveraging high-resolution satellite imagery from PlanetScope, the system provides real-time insights into crop health through vegetation indices like NDVI and GCI. The user-friendly web application allows for efficient management of multiple farms, enabling timely interventions to prevent crop diseases. The system's architecture, built on a micro-service model with Docker containerization, ensures scalability, security, and flexibility. The random forest models developed for yield prediction, particularly the model utilizing NDVI and GCI data M_1, have shown promising results with an R2 score of 0.81, demonstrating the system's potential for accurate yield forecasting.

For future work, there is significant potential to enhance the predictive capabilities of SCMS. While the current models show good performance, integrating additional data sources and exploring more advanced machine learning techniques could further improve accuracy. The robust system design, with its modular architecture, is well-positioned to accommodate new crops and models. Researchers can easily integrate their independent models for various crops via Docker, expanding the system's applicability across different agricultural contexts. This extensibility allows for continuous improvement and adaptation to diverse farming needs. Additionally, further investigation into the correlation between weather data and yield predictions could provide valuable insights, potentially leading to more comprehensive and accurate forecasting models.

References

Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

Shahir Daya, Nguyen Van Duy, Kameswara Eati, Carlos M Ferreira, Dejan Glozic, Vasfi Gucer, Manav Gupta, Sunil Joshi, Valerie Lampkin, Marcelo Martins, et al. *Microservices from theory to practice: creating applications in IBM Bluemix using the microservices approach.* IBM Redbooks, 2016.

Artemij Fedosejev. *React. js essentials.* Packt Publishing Ltd, 2015.

Mark Masse. *REST API design rulebook.* " O'Reilly Media, Inc.", 2011.

Dirk Merkel. Docker: lightweight linux containers for consistent development and deployment. *Linux journal*, 2014(239):2, 2014.

Planet Labs PBC. Planet Application Program Interface: In Space for Life on Earth. `https://api.planet.com`, 2024. Accessed November 20, 2024.