COMP 4560 Industrial Project Proposal

April 23, 2023

Daniel Mai
Dane Wanke
Dharmit Anghan
Jay Dodhiawala
Joseffus Santos
Raman Bhandari

Abstract:

Ergot is a plant disease that infects the developing grains of cereals and grasses. When ergot bodies instead of kernels emerge during kernel formation, ergot symptoms become visible. To find a strategy to prevent it sooner, it is crucial to identify the factors that encourage it to grow in grain. Our project aims to build a tool that supports this research and provides tools for data analysis for the Canadian Grain Commission. It will focus on data collection, feature engineering, and model exploration/validation to deliver a well-documented process and validation results to facilitate further data analysis.

Background

Courses completed

- COMP 3010: design concurrent systems and how to handle the associated challenges
- COMP 3020: design interfaces with respect to design principles and human ability
- COMP 3350: build large scale collaborative projects whilst incorporating best practices
- COMP 3380: design, build and query databases
- COMP 3190: introduction to artificial intelligence
- COMP 3820: introduction to bioinformatics
- COMP 4710: data mining

Daniel Mai

- Experience in research, industry projects, agile development cycle and project management.
- Experience in data cleaning, data visualization, data analysis and data mining techniques.
- Experience with MongoDB, Postgresql.
- Experience in building machine learning/deep learning models and its associated python libraries, especially with time series data, some of frequently used DL models: ESN, LSTM, auto-encoder.
- Experience in full-stack development including React, Angular, AWS lambdas.

Dane Wanke

- Empathy: Affective Computing model which balances numerous human input to detect emotion using machine learning.
- M.O.B: containerized web application built in react that uses machine learning for facial recognition. Databases are scripted and the user interface displays the most relevant points of data per query offering insight into the underlying algorithm.
- GroceryBot: web application built in vue that scrapes data from surrounding stores. Its interface uses OpenStreetMapsAPI to map information to their respective locations.
- Digi-Menu: web application built in react that focuses on experimental design
- Extensive experience managing projects as well as teaching myself, and others, the required project knowledge

Dharmit Anghan

- Experience with working on databases and website projects as a team, planning the timeline and workload of the project.
- Experience with designing the interfaces under usability principles.
- Overall, ability to work in groups and get the data through datasets and interest in gaining knowledge in a collaborative environment.

Raman Bhandari

- Experience working in a backend API team.
- Experience working full stack with React and Redux for frontend and ServiceNow for backend with a different flavor of JavaScript.
- Experience with Research with the Mathematics department, designing algorithms in C++ and MATLAB.
- Experience working with Docker images and Azure.

Jay Dodhiawala

- Experience with front-end development with react, typescript,etc.
- Experience using databases and a piece of good knowledge about their architecture.
- Worked with machine learning algorithms and understanding of how the basic algorithm works.
- Experienced with different testing procedures.
- Experience with DL libraries like Pytorch, Tensorflow, Keras.
- Experience with data cleaning and visualization with pandas, matplotlib, seaborn etc.
- Worked on projects as a team with an agile driven approach.
- All in all, I am always excited to learn new things, meet new people and learn from them.

Joseffus Santos

- Extensive experience in research, data analysis, and project management
- Extensive Python experience in industry and project work
- Experience in development in multiple environments: Mac, Linux, Windows
- Experience in creating cross platform UI: Qt, React, Angular, TypeScript, VBA
- Experience in ML tech stack: Jupyter, Tensorflow 2, scikit, pandas
- Experience in visualization of GIS data: GeoPandas, QGIS, React, Angular
- Experience in DevOps deployments in AWS: AWS cli, Terraform
- Experience in containerization: creation and modification of Docker images
- Experience in Backend Frameworks: Python Chalice/Flask, ASP NET
- Experience with MSSQL, PostGres, sqlite

Problem Statement:

Since the 1900s, the Canadian Grain Commission has been collecting harvest data quality data of crops grown in Canada. To date, this data has yet to be leveraged to predict quality outcomes. Predicting outcomes or having a recommendation system for planting based on environmental and genetic factors could save the industry millions in losses. For our project we will focus on modeling Fusarium and Ergot, incidence and severity in Canada's various crop regions as functions of weather, and geographic location.

Methodology and Timeline:

Our group will use a private GitHub repository as a collaboration medium, where we will host our deliverables and works in progress. Tasks and subtasks will be added to the repository as issues. The issues will reflect the deliverables outlined below. Since our datasets will be large we will host at least one copy of our database in a shared environment either on the cloud or a location physically accessible to our group.

There are 3 major stages in our approach to solve the problem. Data collection, Feature Engineering, and Model Exploration/Validation. These stages can be run in parallel. For example while scraping work is being completed. Some group members can start researching current attempts at using weather to predict our selected grain outcomes.

The first segment of work involves scraping data sources and storing the raw unprocessed data in a database. The raw tables will have the same schema as the raw incoming data. The scripts will then be validated with tests. The scraping classes will be written such that they can be rerun in the future to add more data to the raw data databases. Due to the sparse locations of weather stations and differences in precision, we plan to use other sources as well including Canada's open data portal to collect weather station data, and a European Space Agency (ESA) data source for high resolution satellite data.

Work is required to determine an appropriate processing procedure to make the data consistent. The schema for the tables to hold the processed data will be determined once the group completes scraping. Research will be required to select an appropriate procedure to handle null or inconsistent input from raw data. If time permits we can encode these options as parameters when the processing scripts are run to provide options for future users.

When working with datasets, we typically try to find patterns or similarities between features. To understand which features are essential, we must explore and visualize our data to understand data distribution and anomalies.

From research conducted earlier in the process, we will explore the best models that have used weather as predictors and either try to improve upon those models or use the models to guide us in developing a novel model to explore.

Lastly, we aim to create a meaningful user interface that inspires research and provides insight into the most relevant data points per the corresponding machine learning algorithm. This solution may be developed from scratch or by using an existing user friendly framework, e.g. Tableau.

Ultimately meeting as a team with Sean, Tiffany and Olivier (as per their availabilities) every 2 weeks.The following is the detail of the timeline:

Combined Project Tasks	
Task	Due
Infrastructure Setup - Discuss architecture. - Code integration. - Release Process. - Setup individual environments. - Setup tasks on repository - Create setup documentation and scripts - Split up tasks - Setup common database	May 13
Check in with CGC	**May 17**
Scraping - Refactor scripts into reusable object oriented chunks - Turn scripts into importable python modules - Scrape raw weather station data - Collect Land Moisture Satellite data - Collect Land Temp Satellite data Literature Review of Current techniques must be done, and findings ready to be shared with the rest of the team - Determine current models using weather as input - Determine processing techniques with weather as input - Determine what was determined as useful weather features - Determine current models prediction metrics Update documentation	May 26
Check in with CGC	**May 31**
Data Processing - Lint the data to handle missing, inconsistent and unexpected data based on techniques that came up in research - Pre-process the data - Conduct exploratory data analysis - Visualizing data and starting research on the relationship between features using data mining techniques. - Engineer new features from raw data. - Finalized the data based on our research. Update documentation	June 31
Check in with CGC	**June 14**
Explore base models: - Create priority queue based on data for what models we should use - Building machine learning/deep learning base models for prediction - Fine tune models	July 5

Update documentation	
Check in with CGC	**July 5**
Experimental Analysis: - Testing models on different kinds of data (i.e., hourly, daily, …) - Run the base models on the same dataset and compare different results we have with different models and be able to answer why the best model is the best model? - Tune the hyper-parameter of current best base models and re-run it on multiple datasets. - Make visualization for each of the cases along with statistical analysis. - Model Validation performance metrics Update documentation	July 25
Check in with CGC	**July 19**
Create UI - Map overlay Finalize documentation	Aug 2
Check in with CGC	**Aug 2**
Prepare final presentation - Building slides for the presentation - include how this project can be continued or improved in the future work - A final document with all the tasks done in the project - Decide who will present which parts of the projects	Aug 14

Infrastructure, facilities and expert personnel requirements:

To successfully begin this project, we will first need access to the data collected and held by the Canadian Grain Commission. In addition to data, insight and support from qualified bioinformaticians and bio-statisticans will be vital to our success. Such information will prove useful when additional context outside of our own areas of expertise may be required for guidance in our decision making processes. Insight and support will be provided by Olivier Tremblay-Savard from the University of Manitoba as well as Tiffany from the Canadian Grain Commission. We have already made these arrangements with Olivier and Tiffany and plan to communicate with both of them over email.

Outcome and Deliverables:

Shrink goals

- A normalized database populated with raw data of factors.
- Importable Python modules for scraping weather station data and if possible satellite data.
- Scripts and modules to clean the data of inconsistencies.
- A short report on current models and methods using weather data compiled.
- An ML/AI model which uses linear modeling. The model should be able to be recreated and validated from scratch using our pipelines.
- Basic documentation

Expected goals

- Additional ML/AI models, exported that can be queried with new data, the model should be able to be recreated and validated from scratch using our pipelines. The models we currently plan to use are:
    - Decision Tree
    - Random Forest
    - Support Vector Machine
    - Deep Neural Network
- Further model validation, optimization, performance metrics and a summary comparing performance based on the additional models above.
- Data visualizations using seaborn, matplotlib.
- More detailed documentation including tutorials of the system's use case scenarios.
- A report outlining how the system could be improved further as per our stretch goals
- A user interface that simplifies interactions with the system and its components

Stretch goals

- Further model research, experimentation and optimization for higher levels of system accuracy (including those mentioned above)
- Database quality of life and optimization features including indexes, triggers, system partitioning and automated backups
- Automated deployment using scripted docker and kubernetes
- Genomic sequence retrieval and comparisons given selected samples
- Upgrade user interface to a responsive website using React for enhanced data visualization including but not limited to simplifying user interaction with the system and the use of maps (OpenStreetMaps) to drive further research.
    - View for tracking the spreading of diseases
    - View for searching up expected crop yields given pre-defined growing conditions