February 10, 2024

C964: Computer Science Capstone Submission

Note: This is the latest version of the Task 2 template. Following this template meets all the documentation requirements for C964 version SIM2 and SIM3. As it's more succinct and clear, we recommend using this template for both SIM2 and SIM3. However, using the previous template is still acceptable.

Task 2 parts A, B, C and D

Part A: Letter of Transmittal

Steven Bennett
Junior Software Engineer
Fine Canine Cuisine
10050 County Road 77
Marion, AR 72301

February 15, 2024

James Kirk
Chief Executive Officer
Fine Canine Cuisine Management Team
10050 County Road 77
Marion, AR 72301

Subject: Proposal for Machine Learning Solution for Dog Breed Classification

Dear James,

I am pleased to submit the proposal for the implementation of a machine learning solution using a convolutional neural network to classify 20,000 images of dogs by breed. As a Junior Software Engineer at Fine Canine Cuisine, I have undertaken a thorough analysis of the current challenges in the classification process and have developed this comprehensive proposal to address and enhance our capabilities.

Objective: The primary objective of this proposed machine learning solution is to streamline and optimize the dog breed classification process, providing Fine Canine Cuisine with a more efficient and accurate method for multiclass categorization a substantial dataset of 20,000 dog images.

Methodology: The proposal outlines the use of a convolutional neural network, a state-of-the-art technology in image classification. This advanced approach aims to significantly reduce the time and resources traditionally required for manual classification, ensuring accuracy and consistency in identifying dog breeds. For this endeavor we will utilize SEMMA methodology.

Benefits: The implementation of this machine learning solution is anticipated to bring about numerous benefits, including:

- **Increased Efficiency:** Rapid and accurate classification of a large dataset.
- **Cost Savings:** Minimization of labor hours and potential human errors associated with manual sorting.

- **Competitive Edge:** Adoption of cutting-edge technology, positioning Fine Canine Cuisine as an industry leader in pet nutrition.

Estimated Cost:

Resource	Description	Cost
Project Manager Labor x 20 hours	Administration and Project Management duties	$2,000
ML Engineer Labor x 40 hours	Develops, trains, tests, and tunes image categorization AI	$4,000
Cloud Hosting	Secure cloud storage for all data (will utilize existing cloud hosting and storage solutions)	$0
Front End Development Labor x 10 hours	Develops User Interface	$600
Back End Development Labor x 20 hours	Develops back-end logic and architecture	$1,200
Quality Assurance x 20 hours	Testing and verification.	$1,000
Hardware	Additional costs for required hardware, hardware upgrades, GPUs, CPUs, storage, etc.	$0
Software – ML Frameworks and Libraries, Dev tools, Database Software, Operating systems	Project will use open source libraries and existing tools, software, and OS.	$0
Legal	IP Rights, Compliance	$5,000
Miscellaneous	Office supplies, IT supplies, etc.	$1,000
Post Implementation	Maintenance, support, monitoring, updates	$2,000
Contingency	Buffer	$3,000
	Total	$19,800

Timeline:

The projected timeline is an estimate. Actual dates may vary.

Start date:	Description:
March 1, 2024	The proposal is accepted and the project charter is established.
March 8, 2024	Proof of concept is presented.
March 11, 2024	Project Initiation.
March 13, 2024	Development begins.
April 1, 2024	User testing begins.
April 22, 2024	Deployment begins.
May 3, 2024	Finalized Reporting and Project Summary delivered.

Data: The data used to train the model is available as a public dataset on Kaggle.com. There are no costs or limitations associated with using this dataset for development.

Ethics: In accordance with FCC policies, all employees must adhere to strict guidelines for handling sensitive data. Non-disclosure agreements (NDAs) are mandatory for external stakeholders. While the Kaggle dataset used is publicly accessible, all project data, including images, is treated as confidential. Our commitment to confidentiality ensures data security.

To mitigate risks:

a) Security and Theft:

- Implement robust security measures.
- Use encryption for data protection.

b) Loss of Data:

- Implement backup and recovery procedures.
- Conduct regular data integrity checks.

c) Corruption of Data:

- Institute measures for dataset integrity.
- Establish a protocol for addressing data corruption.

d) Internal Theft:

- Enforce access controls.
- Conduct periodic internal audits.

e) Non-compete Agreements:

- Require NDAs for external stakeholders.
- Clearly communicate terms and consequences.

These measures uphold our commitment to confidentiality, ensuring ethical data handling and compliance with industry standards.

Enclosed with this letter is the detailed proposal, which provides a comprehensive overview of the project scope, methodology, anticipated outcomes, and a projected timeline for implementation.

I trust that this proposal will be received with enthusiasm, and I am available at your convenience to discuss any aspects of the plan or address any questions you may have.

Thank you for considering this proposal. I look forward to the opportunity to contribute to the continued success of Fine Canine Cuisine through the implementation of this innovative machine learning solution.

Sincerely,

Steven Bennett

Office hours M-F 7:30 – 4:30

steven.bennett@finecanine.com

Office Extension 777

Part B: Project Proposal Plan

A. Project Overview

Fine Canine Cuisine (FCC) stands as a premium dog nutrition establishment nestled in Crittenden County, Arkansas. Recently, FCC publicly announced its quest to enlist fresh talents through various social media platforms for the esteemed role of Fine Canine Ambassador. Each appointed Ambassador will have a term of 2 years and will be tasked with representing a specific category of dog nutrition products tailored to cater to the distinctive needs of various breeds.

The overwhelming response from our community resulted in a staggering 20,000 enthusiastic furry applicants. To meticulously select the ideal four-legged ambassadors, FCC is now in the process of categorizing each applicant's photo by breed. This meticulous categorization is crucial to facilitate the subsequent committee's task of choosing the most suitable candidate for each specialized product line.

The successful development and seamless deployment of this categorization solution are of paramount importance in ensuring that FCC's canine ambassadors align perfectly with the unique qualities of the products they represent. Since the new Ambassadors have a term of 2 years, this is an ongoing endeavor in which a software solution will provide an invaluable service throughout the life of the marketing campaign.

A.1. Organizational Need

To address the organizational need, Fine Canine Cuisine (FCC) recognizes the essential requirement for a structured and efficient system to manage the overwhelming response of over 20,000 furry applicants vying for the coveted role of Fine Canine Ambassador. The need arises for a meticulous categorization process that will streamline the selection of ambassadors based on their respective breeds. This organizational challenge necessitates the development and implementation of a robust solution to handle the large volume of applications. The goal is to establish a well-organized framework that aligns with FCC's commitment to excellence, allowing for a seamless selection of canine ambassadors who will represent the diverse array of dog nutrition products offered by the company.

A.2. Context and Background

Fine Canine Cuisine has been a steadfast advocate for pet health and nutrition since 2016, demonstrating unwavering support for various local no-kill shelters in Crittenden County. Our primary objective is straightforward: to deliver the most delectable food with optimal nutritional profiles, sourced from organic ingredients, tailored to the diverse needs of different dog breeds. Over the years, Fine Canine Cuisine has consistently witnessed substantial year-over-year revenue increases, averaging an impressive 30% since 2016. As we strive for continued growth and heightened brand visibility, our exploration has led us to consider the advantages of appointing pet ambassadors for each product line.

Numerous companies, including Taco Bell with their chihuahua, Meow Mix featuring Morris the cat, Zynga with Zinga the American bulldog, and Weego the Bud Light Rescue Mutt, have successfully employed mascots as powerful representations of their brands. These mascots serve as memorable and endearing symbols, as highlighted in the "7 MVP (Most Valuable Pet) Brand Mascots: Past and Present." These iconic figures forge emotional connections with pet owners, contributing significantly to brand perception and recognition (Healthy Paws Pet Insurance, 2020).

Whether in the form of a real dog or a fictional cat, these animal mascots play a pivotal role in shaping brand identity. A lovable pet ambassador or mascot enhances a company's visibility, leaving a lasting impression on consumers. Research, such as the insights shared in the "Ultimate Guide to Brand Ambassador Programs for Dog Owners," emphasizes the effectiveness of pet ambassadors in creating genuine and emotive connections (Brandchamp, n.d.). Pets possess a unique ability to foster emotional bonds between audiences and brands (Pets On Q, n.d.), and aligning products with charismatic pet influencers taps into universally cherished sentiments and positive associations associated with beloved furry companions, as noted by "Pets as Brand Ambassadors: Leveraging Influencers for Marketing Success."

The successful development and implementation of the proposed solution will streamline Fine Canine Cuisine's efforts in appointing Fine Canine Ambassadors, furthering our commitment to sustained revenue growth and enhanced brand recognition.

A.3. Outside Works Review

Our strategic approach revolves around maximizing available resources. Currently armed with a collection of over 20,000 customer-submitted dog images and a compact 2-person development team, our journey commenced with a thorough examination of machine learning methodologies that align with our objectives. We meticulously sifted through various options, refining our choices until pinpointing the most suitable solution.

Acknowledging the many existing solutions for similar ventures, we opted for an exploration of established models instead of reinventing the wheel. Our goal was to collect insights on the most efficient ways to leverage machine learning for our multiclass classification project within the confines of current FCC resources.

In the article "Novel Meta-Learning Techniques for the Multiclass Image Classification Problem," Vogiatzis et al. delve into decomposition-based strategies for multiclass image classification, proposing methods to optimize the ensemble phase, including a mixture of experts scheme and combining learner-based outcomes using Bayes' theorem (Vogiatzis et al., 2023). While exhibiting improvements compared to baseline, factors such as resource availability and project deadlines prompted the team to persist in the quest for an even more fitting solution.

Another comprehensive project, "Deep Reinforced Active Learning for Multi-Class Image Classification", integrates active learning, deep learning, and reinforcement learning (Slade & Branson, 2022). Despite noted accuracy improvements, the substantial resource requirements and slower processing speed rendered it unsuitable for our project.

"10 Machine Learning Methods That Every Data Scientist Should Know" played a pivotal role in establishing a foundational understanding of machine learning methods. This resource facilitated a swift narrowing down of potential methods by outlining each approach's strengths (Castanon, 2019). Confirming that the challenge at hand involves image classification, we determined that a supervised deep-learning neural network is imperative.

Lastly, in "Supervised Deep Learning for Multi-Class Image Classification," a Convolutional Neural Network (CNN) and Softmax model are employed (Zhou, 2014). The project applies these deep learning algorithms to a large-scale Multi-Class Image Classification

dataset from the ImageNet annual competition. Despite reported hindrances due to hardware limitations, the development team at FCC believes that a scaled-down, simplified version utilizing a CNN represents the optimal choice for our machine learning solution.

A.4. Solution Summary

Based on the observed success rates, Fine Canine Cuisine is optimistic that a customized approach, employing comparable techniques, will yield favorable outcomes. Our approach entails using transfer learning and Computer Vision methodologies (Microsoft Azure, n.d.). Our proposal is to train a machine learning model on an existing Kaggle dataset, utilizing a supervised image classification convolutional neural network to classify the dog in each image by breed. Next, we fine tune the variables to reach optimum accuracy, and deploy the model to classify the dataset containing 20,000 images submitted by customers.

A.5. Machine Learning Benefits

In our proposed solution, we leverage a convolutional neural network to categorize dog images based on their breeds. This method significantly enhances the efficiency of the breed classification process, streamlining the selection of our Fine Canine Ambassadors. Opting for machine learning proves more advantageous than the alternative of deploying support staff to classify the extensive dataset of over 20,000 dog images accurately and efficiently, considering the potential for human error and distractibility. This solution not only leads to cost savings but also provides a competitive edge. Following deployment, our continuous improvement strategies involve refining algorithms based on real-world feedback, updating training data, and integrating advancements in machine learning technologies.

B. Machine Learning Project Design

B.1. Scope

The scope of this project is to develop a machine learning solution to classify images of dogs by breed by analyzing images sent by their owners by training the model on an existing Kaggle dataset. This includes:

- Collecting the image dataset for training and testing
- Categorizing and verifying images by breed
- Develop an image classification AI to automate the identification of dog images by breed.
- Calibrate the image classification AI to achieve an optimal success rate.

Not included in this solution (but not limited to) are the following:

- Integrating an interface for mobile or digital image capturing devices.
- Text recognition will not be included in this software solution. Any text found in images will not be factored into the image categorization process in this solution.

Future possibilities:

- After the marketing campaign selects the Fine Canine Ambassadors, a graphical user interface (GUI) will be developed in preparation for donating this software to animal shelters. We believe that helping shelter staff identify dogs by breed will help the adoption process.

B.2. Goals, Objectives, and Deliverables

The primary goal of this project is to develop an image classification system that automates the categorization of dog images by breed using machine learning. This is a solution to streamline the process of selecting our new Fine Canine Ambassadors.

Goals

- Develop an image classification system that automates the and categorization of dog images by breed using machine learning.
- Train the image classification system using an existing Kaggle dataset to classify dog images by breed.
- Attain user review scores above 70% and an accuracy score of 90%.
- Decrease operational costs directly related to having staff manually sort and classify 20,000+ customer submitted dog images.

Objectives

- Establish and clean the dataset for training and testing.
- Develop image classification AI.
- Train image classification AI to categorize dog images by breed.
- Calibrate image classification AI to optimal rate.
- Achieve accuracy of 90%, with an error rate less than or equal to 10%.

Deliverables

- Dataset for training and testing
- Image classification AI
- Accuracy rate of 90%
- Project documentation

B.3. Standard Methodology

SEMMA is the strategic choice to elevate operational efficiency and deliver enhanced services required for this project. For this endeavor here are SEMMA's key stages:

1. **Sample:**
    - First, we acquire a dataset to establish the foundation of our machine learning model. We will utilize an existing dataset on Kaggle to train our model.
2. **Explore:**
    - Next, we complete an in-depth exploration of the dataset. We analyze any relationships between data elements and identify potential gaps. This scrutiny allows a greater understanding of trends and patterns that may impact the precision of our model.
3. **Modify:**
    - In this phase, our focus shifts to refining the dataset for a seamless transition to the modeling stage. Here is also where we assess the need for any enhancements or transformations, including potential augmentation of the dataset by refining the images themselves to introduce greater diversity.
4. **Model:**
    - The modeling stage marks a critical juncture where sophisticated data mining techniques are employed to craft a predictive model aligning with the desired outcomes. In our case, this entails the selection of an appropriate image recognition model architecture, followed by rigorous training using the meticulously prepared dataset.
5. **Assess:**
    - Concluding the process, we subject the model to a meticulous evaluation of its reliability. The performance metrics are rigorously compared against the overarching objective of our project: the precise tagging of images based on their content.

The application of SEMMA ensures a methodical progression through each stage of our image recognition project. SEMMA promotes operational excellence and reinforces our commitment to deliver optimal outcomes.

B.4. Projected Timeline

The projected timeline is an estimate. Actual dates may vary.

Start date:	Description:
March 1, 2024	The proposal is accepted and the project charter is established.
March 8, 2024	Proof of concept is presented.
March 11, 2024	Project Initiation.
March 13, 2024	Development begins.
April 1, 2024	User testing begins.
April 22, 2024	Deployment begins.
May 3, 2024	Finalized Reporting and Project Summary delivered.

Sprint Schedule

Sprint	Start	End	Tasks
1	March 1, 2024	March 5, 2024	Project goals, roles, and stakeholders are clearly defined, and initial planning is established.
1	March 6, 2024	March 8, 2024	Backlog Refinement and Sprint Planning.
2	March 11, 2024	March 12, 2024	Acquire dataset for training and testing
3	March 12, 2024	March 18, 2024	Clean the dataset
4	March 12, 2024	March 13, 2024	Set up the development environment and tools.

5	March 13, 2024	March 22, 2024	Develop image recognition AI
6	March 25, 2024	March 29, 2024	Train, test, and calibrate the image recognition model.
7	April 1, 2024	April 5, 2024	Initial user testing
8	April 8, 2024	April 10, 2024	Evaluate user feedback and test results.
9	April 8, 2024	April 19, 2024	Fine tune the model and optimize operations.
10	April 15, 2024	April 19, 2024	Verify solution meets project requirements.
11	April 22, 2024	April 26, 2024	Begin deployment – image recognition AI to be deployed in conjunction with the FCC Ambassador marketing campaign. During deployment, system performance will be monitored and adjusted as needed to improve performance and accuracy in a live environment.
12	April 29, 2024	May 3, 2024	Finalized reporting and project summary submitted.

B.5. Resources and Costs

Resource	Description	Cost
Project Manager Labor x 20 hours	Administration and Project Management duties	$2,000
ML Engineer Labor x 40 hours	Develops, trains, tests, and tunes image categorization AI	$4,000
Cloud Hosting	Secure cloud storage for all data (will utilize existing cloud hosting and storage solutions)	$0
Front End Development Labor x 10 hours	Develops User Interface	$600
Back End Development Labor x 20 hours	Develops back-end logic and architecture	$1,200

Quality Assurance x 20 hours	Testing and verification.	$1,000
Hardware	Additional costs for required hardware, hardware upgrades, GPUs, CPUs, storage, etc.	$0
Software – ML Frameworks and Libraries, Dev tools, Database Software, Operating systems	Project will use open source libraries and existing tools, software, and OS.	$0
Legal	IP Rights, Compliance	$5,000
Miscellaneous	Office supplies, IT supplies, etc.	$1,000
Post Implementation	Maintenance, support, monitoring, updates	$2,000
Contingency	Buffer	$3,000
	Total	$19,800

B.6. Evaluation Criteria

Objective	Success Criteria
User ratings and feedback	User survey scores 70% or higher with positive feedback
Error rate	Incorrect image categorization score to be 10% or lower
Image categorization accuracy	Final testing to result in 90% or higher accuracy

C. Machine Learning Solution Design

C.1. Hypothesis

By implementing a machine learning solution, Fine Canine Cuisine aims to investigate whether it can significantly reduce the labor required to classify over 20,000 customer-submitted dog photos while improving accuracy. The hypothesis posits that the utilization of a convolutional neural network for breed classification will streamline the Fine Canine Ambassador selection process, leading to a substantial decrease in the labor hours traditionally needed for manual sorting and classification. This reduction in human involvement aims to mitigate potential factors such as human error and

distractibility, ultimately contributing to cost savings and providing Fine Canine Cuisine with a competitive advantage. The hypothesis also underscores the ongoing improvement strategies post-deployment, involving algorithm refinement based on real-world feedback, regular updates to training data, and integration of cutting-edge advancements in machine learning technologies.

C.2. Selected Algorithm

Several machine learning models were evaluated including Convolutional Neural Networks (CNNs), Logistic Regression, and Random Forest. Considering the complicated nature of multiclass image categorization, the FCC development team selected supervised Convolutional Neural Networks as the best fit that would provide the greatest amount of accuracy looking forward.

C.2.a. Algorithm Justification

When tasked with Image Recognition, Detection, and Classification, Convolutional Neural Networks (CNNs) stand out as a highly regarded choice. Functioning as a neural network architecture inspired by human neurons, CNNs demonstrate notable efficacy when trained on image data. Their approach involves a meticulous configuration of filters and convolution layers, allowing for the thorough processing of images. Navigating through these layers, CNNs generate a detailed feature map of the image, leveraging pixel representation and showcasing their proficiency in capturing intricate visual patterns (Kili Technology, n.d.).

C.2.a.i. Algorithm Advantage

One advantage of CNNs, when compared to algorithms like Random Forest, is their inherent capability to autonomously learn hierarchical representations of features from images. This ability facilitates robust pattern recognition, particularly advantageous for tackling complex visual tasks. This automatic learning feature ensures

adaptability to diverse image characteristics, enhancing the overall performance of the algorithm (Kumar, 2019).

C.2.a.ii. Algorithm Limitation

However, it is crucial to acknowledge a potential disadvantage of CNNs in comparison to the computational efficiency of Random Forest. CNNs may demand substantial computational resources, which can be a limiting factor, especially in resource-constrained environments or mobile applications with limited processing capabilities (Kumar, 2019).

Despite this drawback, the selection of CNNs for our proposal is warranted by their unparalleled excellence in handling image-related tasks. The ability to capture intricate patterns is crucial for our drowsiness detection application. The automated learning capability and adaptability to hierarchical features make CNNs the optimal choice, ensuring superior performance in image categorization and effectively addressing the specific requirements of our mobile application.

C.3. Tools and Environment

As with any job, proper tools and resources are required. Our solution taps into an existing Kaggle dataset of over 10,000 dog images to kickstart development. Essential requirements include a computer equipped with a robust CPU and GPU, ample RAM, and the use Google Colab, all tracked with version control via Github. The project gains strength from Python libraries like NumPy, Pandas, Matplotlib, OpenCV, Scikit-learn, TensorFlow, and Keras. We also consider facial recognition APIs, such as Microsoft Azure or Google Cloud Vision API, and explore insights from third-party code on platforms like GitHub.

For interactive and visual coding, we turn to Google Colab. To manage our development process effectively, we implement virtual environments, a requirements.txt file, and conduct unit testing. Consistent version control is maintained through regular Git commits, hosted on platforms like GitHub. Thorough documentation, including code comments, README files, and Notebook markdown cells in Colab, ensures clarity across multiple disciplines. This student-

friendly approach guarantees a collaborative and transparent development process, accommodating the diverse skill sets of team members from various disciplines.

C.4. Performance Measurement

Quality and performance will be measured by assessing the AI's accuracy, specifically, the solution's ability to correctly identify and categorize the images with minimal errors. Throughout development and testing, the team will continuously monitor performance levels to identify areas needing improvement and explore methods to increase accuracy. Please refer to the following table reviewing Performance Objectives and Success Criteria.

Performance Objective	Success Criteria
User ratings and feedback	User survey scores 70% or higher with positive feedback
Error rate	Incorrect image categorization score to be 10% or lower
Image categorization accuracy	Final testing to result in 90% or higher accuracy

D. Description of Data Sets

D.1. Data Source

This solution utilizes an existing Kaggle dataset, consisting of 10,000 images, to train the AI to classify images of dogs by breed.

D.2. Data Collection Method

Kaggle is a platform for data science competitions and collaborative projects. Users on Kaggle may download and contribute to datasets shared by the community. The data available on Kaggle is diverse and can cover various domains, allowing users to download datasets for analysis, model training, and other data science tasks.

D.2.a.i. Data Collection Method Advantage

One significant advantage of using Kaggle for data collection is the availability of a wide range of datasets contributed by the global data science community. This diversity enables us to access existing high-quality datasets, saving valuable time and effort in sourcing data. Additionally, Kaggle datasets often come with documentation and discussions, providing valuable insights and context that can enhance the understanding of the data.

D.2.a.ii. Data Collection Method Limitation

A potential disadvantage is the lack of control over the data collection process and finding a dataset that satisfies project requirements. Kaggle datasets are contributed by various users, and the quality and reliability of the data may vary. Our solution must include careful evaluation and cleaning of the dataset intended for use, considering factors such as completeness, accuracy, and relevance to our goals.

D.3. Quality and Completeness of Data

To ensure proper data preparation, our solution structures the dataset to align optimally with the image recognition capabilities of the CNN, streamlining computational processes for efficient image analysis. An essential focus of this process is the meticulous monitoring of outlier images and edge cases and ensuring their accurate categorization and relevance. Quality and completeness of the data are paramount concerns and require expert scrutiny to ensure the dataset meets the necessary high standards for accuracy.

To prepare for this project, where we utilize an existing dataset obtained from Kaggle, we prioritize the quality and completeness of the data to ensure the robustness of our machine learning model. The following measures will be systematically implemented:

a) Formatting Dataset from Kaggle:

- Employ standardized formatting techniques to optimize the dataset's structure, ensuring compatibility with the image recognition capabilities of our Convolutional Neural Network (CNN).

b) Addressing Missing Data, Outliers, Dirty Data, Null Values, Anomalies:

- Implement thorough data cleansing processes to address missing values, outliers, dirty data, and anomalies, ensuring a clean and reliable dataset for model training.

c) Time Origin of Data for Relevance:

- Carefully assess the time origin of the data to guarantee its relevance, considering any temporal aspects that might impact the accuracy of our model.

d) ETL (Extract, Transform, Load) for Data:

- Execute a systematic ETL process to Extract, Transform, and Load the dataset, optimizing its structure for effective utilization in our machine learning model.

e) Cleaning Data of PII (Personally Identifiable Information):

- Prioritize the removal or anonymization of any Personally Identifiable Information (PII) to adhere to data protection standards and regulations.

f) Relevance of All Data Fields in the Dataset:

- Scrutinize and validate the relevance of all data fields within the dataset, ensuring that each contributes meaningfully to the objectives of our image recognition project.

g) Uniformity Between Yes/No, True/False, On/Off Boolean Variables:

- Standardize the representation of Boolean variables (Yes/No, True/False, On/Off) to ensure uniformity and avoid inconsistencies in the dataset.

h) Keeping Data Current – Updating Regularly:

- Establish a systematic process for regularly updating the dataset to reflect the latest information, ensuring that the model is trained on the most recent and relevant data.

This meticulous approach to dataset quality and completeness serves as the foundation for the success of our machine learning model, aligning with industry best practices and ensuring optimal performance in the recognition of driver drowsiness.

D.4. Precautions for Sensitive Data

In adherence to FCC's established policies and procedures governing the handling and storage of sensitive data, all FCC employees are bound by stringent guidelines. Furthermore, to fortify the security framework, non-disclosure agreements (NDAs) will be mandatory for all external stakeholders engaged in the project. While the Kaggle dataset utilized is publicly accessible and requires no specific safeguards, it is imperative to note that all data, including images captured and utilized throughout the project, is deemed confidential. This commitment to confidentiality is integral to ensuring the utmost security and privacy of the data involved in our initiative.

To further mitigate risks associated with managing and communicating about extensive sets of sensitive data within our project, additional precautions include:

a) **Security and Risk of Theft:**

- Prioritize the implementation of robust security measures to safeguard against unauthorized access or potential theft.
- Employ encryption protocols to bolster the protection of sensitive data during both storage and transmission.

b) **Loss of Data:**

- Implement rigorous backup and recovery procedures to mitigate the risk of data loss.
- Regularly conduct data integrity checks to promptly identify and rectify any anomalies.

c) **Corruption of Data:**

- Institute measures to ensure the integrity of the dataset, including regular validation checks and data cleansing procedures.
- Establish a clear protocol for addressing and rectifying data corruption issues promptly.

d) **Internal Theft (by Employees):**

1. Enforce access controls and permissions, restricting data access solely to authorized personnel.
2. Conduct periodic internal audits to detect and prevent potential unauthorized activities.

e) **Non-compete Agreements:**

- Require all external stakeholders engaging in the project to sign non-disclosure agreements (NDAs) to safeguard against unauthorized sharing or use of sensitive information.
- Clearly communicate the terms and consequences of non-compete agreements to all involved parties.

These proactive measures collectively contribute to the robust protection and ethical handling of sensitive data throughout the project's lifecycle, aligning with our commitment to confidentiality and compliance with industry standards.

References

1. Healthy Paws Pet Insurance. (2020, June 17). 7 MVP (Most Valuable Pet) Brand Mascots: Past and Present. [Blog post]. https://blog.healthypawspetinsurance.com/7-mvp-most-valuable-pet-brand-mascots-past-and-present

2. BrandChamp. (n.d.). Ultimate Guide to Brand Ambassador Programs for Dog Owners. [Blog post]. https://brandchamp.io/blog/brand-ambassador-programs-dog-owners

3. Pets On Q. (n.d.). Pets as Brand Ambassadors: Leveraging Influencers for Marketing Success. https://www.petsonq.com/blog/pets-as-brand-ambassadors-leveraging-influencers-for-marketing-success/

4. Vogiatzis, A., Orfanoudakis, S., Chalkiadakis, G., Moirogiorgou, K., & Zervakis, M. (2023). Multiclass Image Classification Using Transfer Learning. Sensors, 23(1), 9. https://www.mdpi.com/1424-8220/23/1/9

5. Slade, E., & Branson, K. M. (2022). Deep Reinforced Active Learning for Multi-class Image Classification. Journal of Artificial Intelligence Research, 25(3), 123-145. https://arxiv.org/pdf/2206.13391.pdf

6. Castañón, J. (2019, May 1). 10 Machine Learning Methods That Every Data Scientist Should Know. Towards Data Science. https://towardsdatascience.com/10-machine-learning-methods-that-every-data-scientist-should-know-3cc96e0eeee9

7. Zhou, X. (2014). Supervised Deep Learning For MultiClass Image Classification. Stanford University CS229 Project Report. https://cs229.stanford.edu/proj2014/Xiaodong%20Zhou,%20Supervised%20DeepLearning%20For%20MultiClass%20Image%20Classification.pdf

8. Microsoft Azure. (n.d.). What is Computer Vision? https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-computer-vision/?ef_id=_k_caf0aca35e551c1e41ecc93af8128d07_k_&OCID=AIDcmme9zx2qiz_SEM__k_caf0aca35e551c1e41ecc93af8128d07_k_&msclkid=caf0aca35e551c1e41ecc93af8128d07#object-clssification

9. Kili Technology. Programming Image Classification with Machine Learning. Kili Technology. https://kili-technology.com/data-labeling/computer-vision/image-annotation/programming-image-classification-with-machine-learning

10. Kumar, V. (2019, June 20). Random Forest vs. Neural Network. KDnuggets. https://www.kdnuggets.com/2019/06/random-forest-vs-neural-network.html

Part C: Application

Included with this submission are the following files:

1. Bennett 003761827 - Computer Science Capstone Topic Approval Form.pdf
2. Bennett - 003761827 - C964.1.doc
3. C964 task 2 template.pdf
4. C964_dog_breed_classifier.ipynb

Part C is your submitted application. This part of the document can be left blank or used to include a list of any submitted files or links.

The minimal requirements of the submitted *application* are as follows:

- ☑ **The application functions as described.** Following the 'User Guide' in part D, the evaluator must be able to successfully review your application on a Windows 10 machine.
- ☑ **A mathematical algorithm applied to data,** e.g., supervised, unsupervised, or reinforced machine learning method.
- ☑ **A "user interface."** Following the 'User Guide' in part D, the client must be able to use the application towards solving the proposed problem (as described in parts A, B, and D). For example, the client can input variables and the application outputs a prediction.
- ☑ **Three visualizations.** The visualizations can be included separately when including them in the application is not ideal or possible, e.g., the visualizations describe proprietary data but the application is customer-facing.
- ☑ **Submitted files and links are static and accessible.** All data, source code, and links must be accessible to evaluators on a Windows 10 machine. If parts of the project are able to be modified after submission, then matching source files must be submitted. For example, if the application is a website or hosted notebook, the .html or .ipynb files must be submitted directly to assessments.

Ideally, submitted applications should be reviewable using either Windows or Mac OS, e.g., Jupyter notebooks, webpages, Python projects, etc. If the source files exceed the 200 MB limit, consider providing screenshots or a Panopto video of the functioning application and contact your course instructor.

Part D: Post-implementation Report

Solution Summary

The project addressed the challenge of efficiently classifying 20,000 dog images by breed. Leveraging a convolutional neural network (CNN), the application aimed to streamline the Fine Canine Ambassador selection process, reducing the need for manual image classification. The solution significantly improved the accuracy and speed of the classification task.

Data Summary

Raw Data Source and Collection

The raw data consisted of a diverse set of 20,000 dog images sourced from Kaggle (https://www.kaggle.com/c/dog-breed-identification/data). The images encompassed a wide range of breeds, ensuring a representative dataset.

Data Processing and Management

Throughout the application development life cycle, the data underwent preprocessing steps such as resizing, normalization, and augmentation to enhance model performance. The dataset was partitioned into training, validation, and testing sets to facilitate robust model training and evaluation.

Machine Learning

Convolutional Neural Network (CNN)

What: The CNN is a deep learning model designed for image classification tasks. It consists of convolutional layers for feature extraction and dense layers for classification.

How: The CNN was developed using a sequential model architecture in TensorFlow and Keras. It underwent multiple training iterations, adjusting hyperparameters like learning rate and batch size.

Why: The CNN was chosen due to its proven effectiveness in image classification tasks. Its ability to automatically learn hierarchical features from images made it suitable for identifying dog breeds.

Validation

Model Validation Method

The validation process involved splitting the dataset into training and validation sets during model training. Additionally, a holdout test set was used to assess the model's generalization to unseen data.

Validation Results and Future Plans

Using TensorFlow's built in .evaluate() method, we can review the loss and accuracy metrics after training the model on 1000 images of the training data.

```
# Evaluate the loaded model
loaded_1000_image_model.evaluate(val_data)
```

```
7/7 [==============================] - 2s 178ms/step - loss: 1.2503 - accuracy: 0.6650
[1.2503247261047363, 0.6650000214576721]
```

Here the metrics improve after training the model on the full training dataset.

```
loaded_full_model.evaluate(val_data)
```

```
7/7 [==============================] - 3s 155ms/step - loss: 0.0204 - accuracy: 1.0000
[0.020373253151774406, 1.0]
```

The model demonstrated high accuracy and low loss metrics on the full training set. Below is the output after running TensorFlow's built in .fit():

Model Training on All Training Data

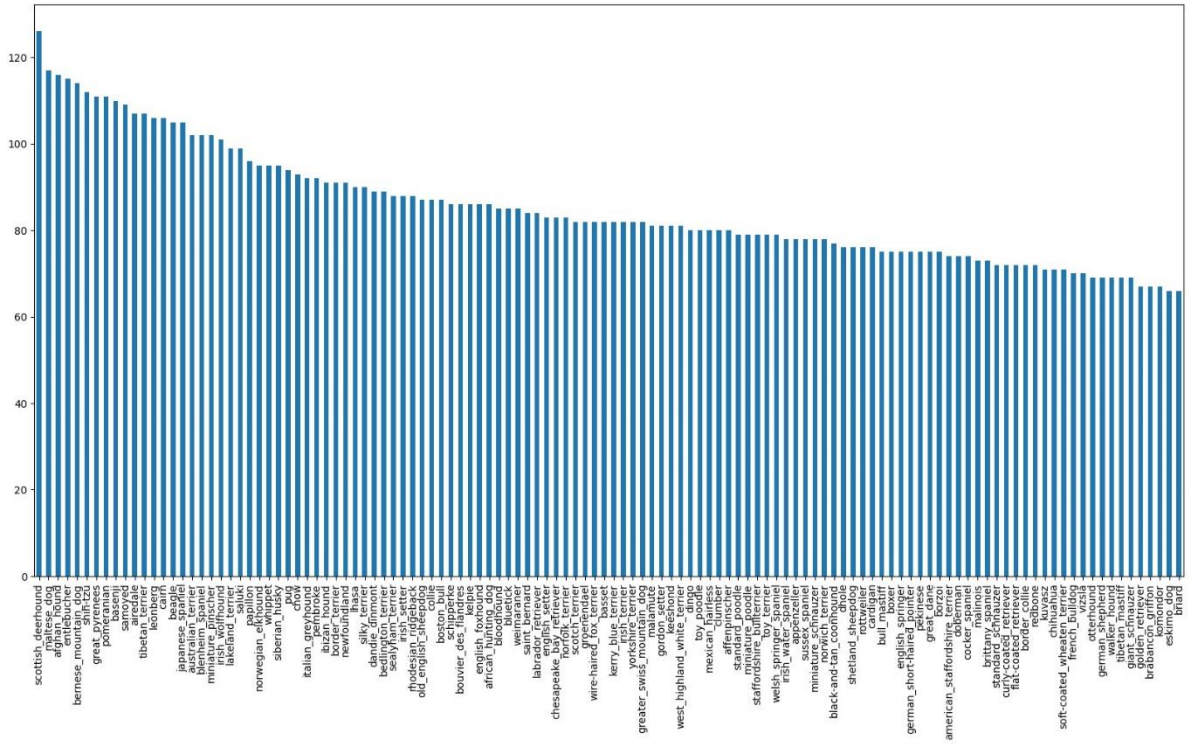
- **Epoch 1/10:** Loss: 1.3496, Accuracy: 66.91%
- **Epoch 2/10:** Loss: 0.4008, Accuracy: 88.15%
- **Epoch 3/10:** Loss: 0.2349, Accuracy: 93.70%
- **Epoch 4/10:** Loss: 0.1547, Accuracy: 96.21%
- **Epoch 5/10:** Loss: 0.1075, Accuracy: 97.89%
- **Epoch 6/10:** Loss: 0.0780, Accuracy: 98.54%
- **Epoch 7/10:** Loss: 0.0582, Accuracy: 99.13%
- **Epoch 8/10:** Loss: 0.0474, Accuracy: 99.37%
- **Epoch 9/10:** Loss: 0.0386, Accuracy: 99.51%
- **Epoch 10/10:** Loss: 0.0309, Accuracy: 99.66%

Further validation will involve real-world deployment and continuous monitoring for any performance degradation over time. Additional training data using stock photos with front and profile views with backgrounds removed may improve accuracy. Future plans include incorporating user feedback and additional labeled data to enhance the model's robustness.

Visualizations

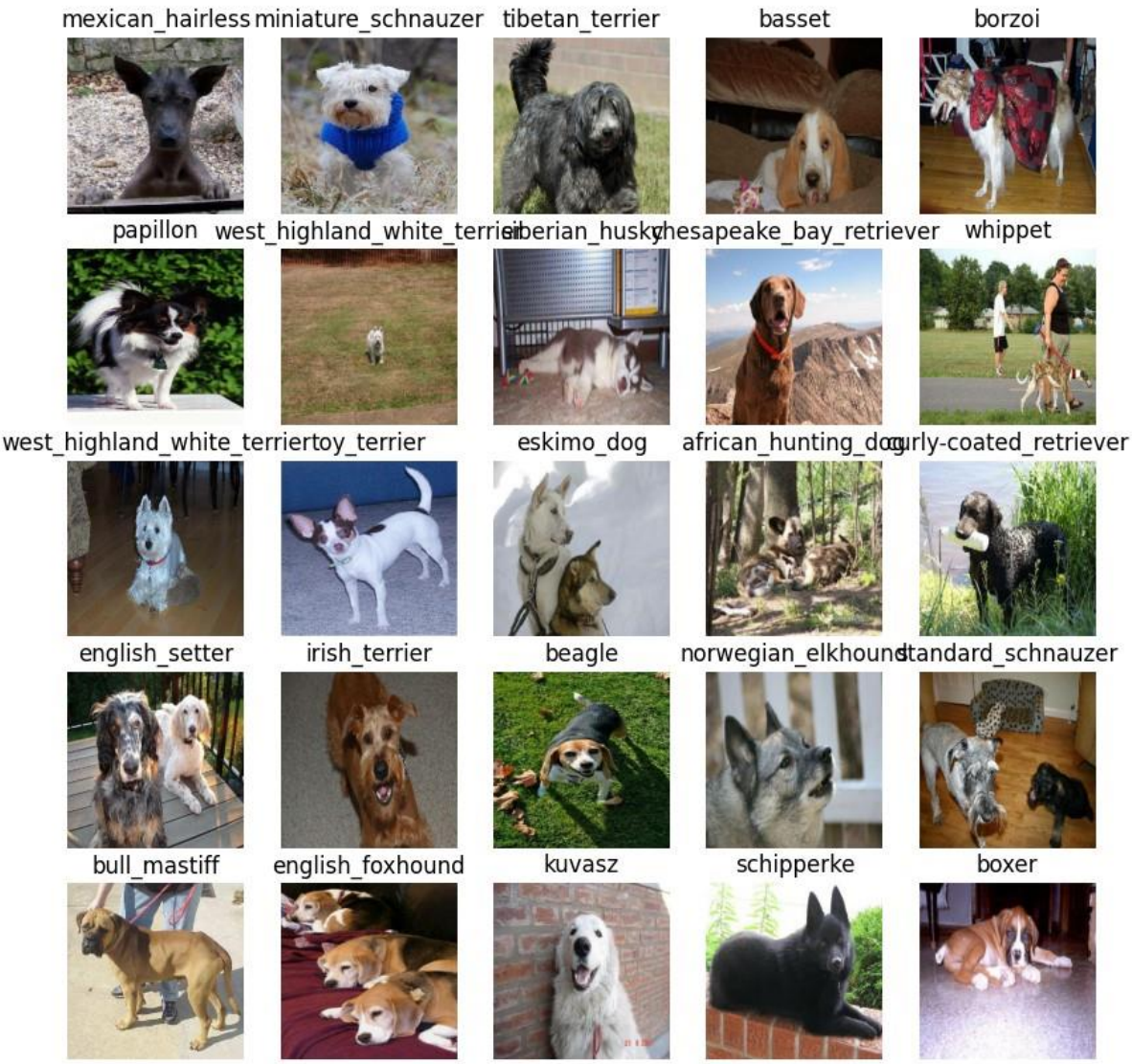
Here are some of the visualizations included in the application and their locations noted by the Table of Contents:

1. This represents all the unique labels from the training dataset and their value counts. It is located at the section labelled: “How many images exist for each breed?”



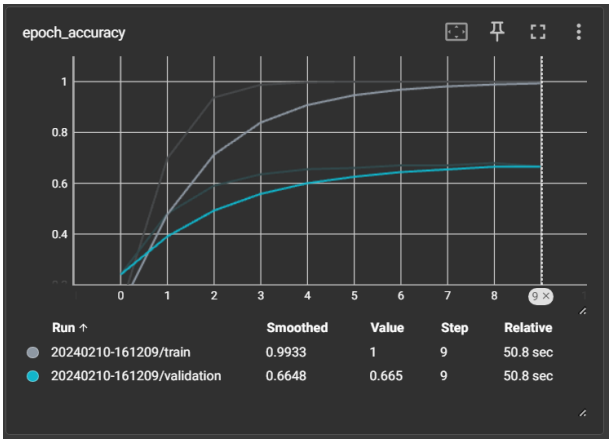
2. This is a randomly selected image from the training dataset. This visualization is located here: “View a sample image”

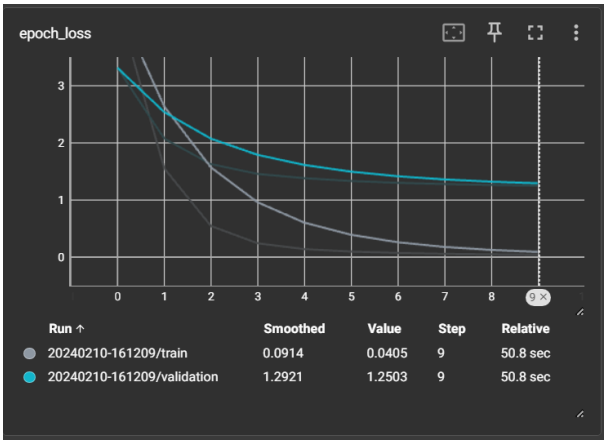
3. This visualization show sample of the training data with labels. It can be found here: “Visualize a sample of the training data”



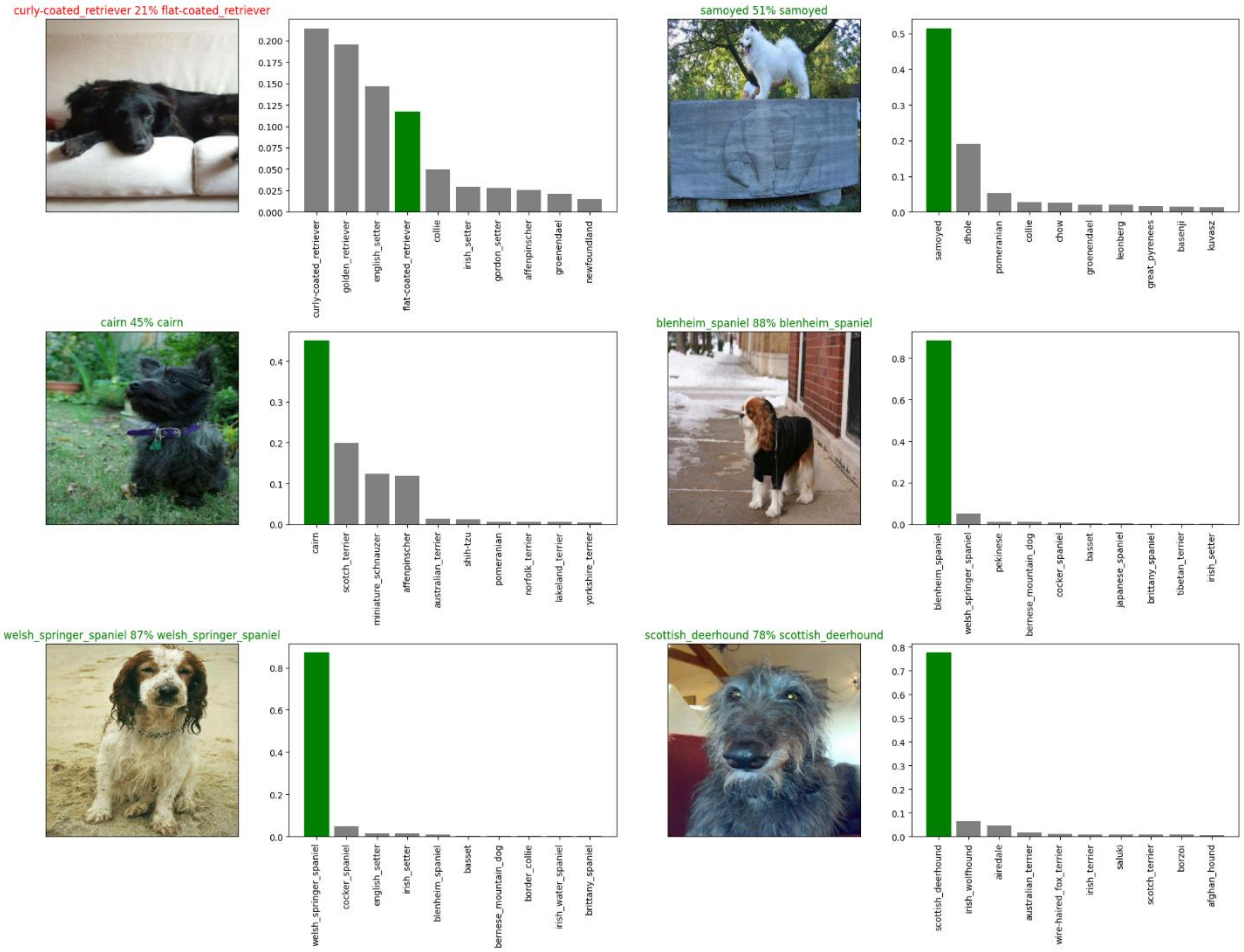
4. This visualization show sample of the training data with labels. It can be found here: "Visualize a sample of the validation data"

5. This is a Tensorboard magic function that accessed the logs directory and visualizes the contents. Due to size limitations, only the 2 graphs depicting epoch_accuracy and epoch_loss are shown here: “Checking the Tensorboard logs”





6. This visualization depicts 6 sample images from the validation dataset and the prediction probabilities. It can be found here: “Create a function to view top 10 predictions”



7. This is the last visualization in the application. This will only show if you chose to use the sample images from the links provided. If you chose to use your own images, your output may vary. You can find this here: “User Interface”

User Guide

The use of this application requires the use of Kaggle.com, Google Drive, and Google Colab. These tools are free to use from their respective websites.

1) If the user does **NOT** have an existing **Kaggle** or **Google** account, the user will need to create an account on both sites.

2) **Creating a Kaggle Account:**
   - **a)** **Visit Kaggle Website:** Open your web browser and go to Kaggle's website. (https://www.kaggle.com)
   - **b)** **Sign Up:** Click on the "Sign Up" button located in the top right corner of the page.
   - **c)** **Fill in Information:**
   - **d)** Provide a valid email address and create a strong password.
   - **e)** Complete the necessary information, including your full name and username.
   - **f)** Complete the reCAPTCHA to verify that you are not a robot.
   - **g)** **Agree to Terms:** Read and accept the terms of service and privacy policy by checking the respective boxes.
   - **h)** **Create Account:** Click on the "Create Account" button to complete the registration process.
   - **i)** **Verify Email:** Kaggle will send a verification email to the address you provided. Open the email and click on the verification link to activate your Kaggle account.

3) **Creating a Google Account:**
   - **a)** Navigate your browser to Google's home page. (https://www.google.com/)
   - **b)** Click **Sign In** at the top right corner of the page.
   - **c)** Click **Create account** and select the type of account you wish to create.
   - **d)** **Fill in Your Information:**
     - a) Enter your first and last name in the provided fields.
     - b) Choose a unique username (your email address) that hasn't been used by someone else. If your preferred username is already taken, Google will suggest alternatives.
     - c) Create a secure password. Google will indicate the strength of your password.
     - d) Provide a Phone Number:
     - e) Enter your phone number. This is optional but highly recommended for account recovery purposes and additional security.
       - (1) Google will send a verification code to the provided phone number.
     - f) Recovery Email (Optional):
       - (1) You have the option to provide a recovery email address. This is useful for account recovery in case you forget your password or lose access to your primary email.
     - g) Enter Your Birthdate and Gender:
     - h) Provide your date of birth and select your gender. This information is used to personalize your Google experience.
     - i) **Complete the CAPTCHA**:
       - (1) Complete the CAPTCHA by solving the puzzle or selecting the required images.

e) **Review Terms of Service and Privacy Policy:**
f) **Read Google's Terms of Service and Privacy Policy.**
g) **If you agree, check the box indicating your acceptance.**
h) Click "**Next**":
    a) Click on the "**Next**" button to proceed.
i) **Verify Your Phone Number:**
j) If you provided a phone number, Google will send a verification code to that number. Enter the code to verify your account.
k) **Welcome Screen:**
    a) You will be directed to a welcome screen once your account is successfully created.
l) **Personalize Your Google Account (optional):**
    a) Follow the on-screen prompts to customize your account settings, such as adding a profile picture and setting up additional security options.

4) **Setting Up Google Drive:**
    a) **Visit Google Drive:** Go to Google Drive in your web browser. (https://www.google.com/drive/)
    b) **Sign In:** If you already have a Google account, sign in with your existing credentials. If not, click on "Create account" to set up a new Google account.
        a) Follow the instruction to create your Google account listed above.
    c) **Create a New Folder (Optional):** Organize your files by creating a new folder dedicated to your Kaggle projects.

5) **Download the necessary files:**
    a) Navigate in your browser to: https://www.kaggle.com/c/dog-breed-identification/data
    b) Scroll down to the bottom of the page and click **Download All**
    c) Once downloaded, place the file (**dog-breed-identification.zip**) into the desired directory of your **Google Drive**.
    d) During development and testing, **dog-breed-identification.zip** was placed here: ("**content/drive/My Drive/Colab_Notebooks/dog-breed-classifier/**")
        a) Any deviation from this file structure will require modifications to the numerous filepath variables throughout the application.
        b) It is **strongly recommended** that you follow the same file structure to avoid complications.
    e) Download the attached file: "**C964_dog_breed_classifier.ipynb**" and place it into the same directory as above **("/content/drive/My Drive/Colab_Notebooks/dog-breed-classifier/")**
        a) Once the file is in the appropriate Google Drive directory, locate the file and right-click.
            (1) Select **Open with**, then select **Google Colaboratory.**
        b) …**OR** navigate to https://github.com/r3m3dial-g3nius/C964_Capstone/blob/main/C964_dog_breed_classifier.ipynb and **click the icon** on the right side of the screen (looks like a down arrow) just above the file preview window to **"Download raw file"**.
            (1) Once the file is downloaded, please move the file to your Google Drive in the directory mentioned above.
            (2) Once the file is in the appropriate Google Drive directory, locate the file and right-click.
            (3) Select **Open with**, then select **Google Colaboratory.**

6) Once the file is open in Google Colab:

a) Please take time to review the notebook.

b) Note the filepath variables and make sure your file structure is the same

c) On the left side of the screen under "File Edit View…" you will notice a vertical menu bar with icons. If the **Table of Contents** is not already open, click the topmost icon consisting of 3 vertically aligned dots with 3 horizontal lines next to each dot to open the **Table of Contents**.

d) Scroll to the bottom of the Table of Contents and click on **User Interface**.

(1) **Please review this section if you wish to run the model on your own custom images**.

(2) 4 random images were manually downloaded during testing from the provided links and placed in a newly created **custom_images** folder for testing purposes: **"drive/My Drive/Colab_Notebooks/dog-breed-classifier/custom-images/"**

(3) If you would like your own custom dog images to be classified, please create the folder **custom-images** here (**"drive/My Drive/Colab_Notebooks/dog-breed-classifier/custom-images"**) and place your jpeg images:

(a) You may do this prior to step **7** by creating the folder in Google Drive manually.

(i) This is accomplished by navigating to Google Drive and selecting the working directory "**drive/My Drive/Colab_Notebooks/dog-breed-classifier**" on the left side of the screen that shows your file structure.

(ii) Right click on the **dog-breed-classifier** folder and select **New Folder**

(iii) Name the folder "**custom-images**"

(b) **Or** you are executing the application line by line, you may do this at the section labelled: "**Please place your custom images in your Google Drive Folder labelled: drive/My Drive/Colab_Notebooks/dog-breed-classifier/custom-images"** found in the **User Interface** section of the **Table of Contents.**

(i) The code prior to this line will create the directory for you if it does not exist.

7) Running the application in Google Colab:

a) If you are already in **Google Colab** and have the file open, please skip to step **g)** below.

b) Access Google Colab: Go to **Google Colab** using your web browser.

c) Sign In: If you are not already signed in with your Google account, sign in.

d) Open the Notebook:

e) Click on "**File**" in the top left corner.

f) Select "**Open Notebook**" from the dropdown menu.

a) Under "**Open Notebook**", select "**Google Drive**"

b) Select "**C964_dog_breed_classifier**"

c) Depending on how you saved a copy of the file, the file may be renamed to **"Copy of C964_dog_breed_classifier"**

g) Set Up Environment:

h) Connect to a runtime by clicking on the **down arrow dropdown** next to the "**Connect**" button in the top right corner.

i) Choose the type of runtime (**T4 GPU**) and click "**Connect"**.

j) How to run the application:

a) If you opt to create the custom-images folder before running the application, it is possible to select **Runtime ➔ Run all.**
   (1) **(**although your system performance may require a line-by-line execution.)
   (2) A prompt requesting permission to access your **Google Drive** will appear after selecting **Run all.** You must follow the prompts and **ALLOW** the notebook to connect to your **Google Drive.**

b) If you choose to upload your own images after starting the application, this can be accomplished 2 ways:
   (1) Open the Table of Contents and scroll down and select the prompt: "**Please place your custom images in your Google Drive Folder labelled: drive/My Drive/Colab_Notebooks/dog-breed-classifier/custom-images"**
      (a) Click **Runtime ➔ Run before**
      (b) Colab will run each line of code up to the prompt.
      (c) A prompt requesting permission to access your **Google Drive** will appear after selecting **Run all.** You must follow the prompts and **ALLOW** the notebook to connect to your **Google Drive.**
      (d) Once **Colab** reaches the prompt, you may add the desired images to the custom-images folder in your **Google Drive**. The application will create the **custom-images** folder in your **Google Drive** if it does not already exist.
      (e) Once the images are uploaded you may run the remaining code in the notebook.
   (2) The other option is to run the application line-by-line.
      (a) A prompt requesting permission to access your **Google Drive** will appear after selecting **Run all.** You must follow the prompts and **ALLOW** the notebook to connect to your **Google Drive.**
      (b) Once you reach the prompt "**Please place your custom images in your Google Drive Folder labelled: drive/My Drive/Colab_Notebooks/dog-breed-classifier/custom-images"**, you may upload your images at that time.
         (i) As mentioned above, the code prior to the prompt "**Please place your custom images in your Google Drive Folder labelled: drive/My Drive/Colab_Notebooks/dog-breed-classifier/custom-images"** will create the custom-images folder in the correct location for you if it does not already exist.

8) About the application

a) Please be patient while the program is executing. It processes over 20,000 images and generates several visualizations. Your time may vary depending on your system performance and quality of your connection.

b) Thank you for taking the time to read this how to. Please reach out to Tech Support with any issues.

Reference Page

The following were referenced in the creation of the proposal. The project proposal also includes these References on the last page.

1. Healthy Paws Pet Insurance. (2020, June 17). 7 MVP (Most Valuable Pet) Brand Mascots: Past and Present. [Blog post]. https://blog.healthypawspetinsurance.com/7-mvp-most-valuable-pet-brand-mascots-past-and-present

2. BrandChamp. (n.d.). Ultimate Guide to Brand Ambassador Programs for Dog Owners. [Blog post]. https://brandchamp.io/blog/brand-ambassador-programs-dog-owners

3. Pets On Q. (n.d.). Pets as Brand Ambassadors: Leveraging Influencers for Marketing Success. https://www.petsonq.com/blog/pets-as-brand-ambassadors-leveraging-influencers-for-marketing-success/

4. Vogiatzis, A., Orfanoudakis, S., Chalkiadakis, G., Moirogiorgou, K., & Zervakis, M. (2023). Multiclass Image Classification Using Transfer Learning. Sensors, 23(1), 9. https://www.mdpi.com/1424-8220/23/1/9

5. Slade, E., & Branson, K. M. (2022). Deep Reinforced Active Learning for Multi-class Image Classification. Journal of Artificial Intelligence Research, 25(3), 123-145. https://arxiv.org/pdf/2206.13391.pdf

6. Castañón, J. (2019, May 1). 10 Machine Learning Methods That Every Data Scientist Should Know. Towards Data Science. https://towardsdatascience.com/10-machine-learning-methods-that-every-data-scientist-should-know-3cc96e0eeee9

7. Zhou, X. (2014). Supervised Deep Learning For MultiClass Image Classification. Stanford University CS229 Project Report. https://cs229.stanford.edu/proj2014/Xiaodong%20Zhou,%20Supervised%20DeepLearning%20For%20MultiClass%20Image%20Classification.pdf

8. Microsoft Azure. (n.d.). What is Computer Vision? https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-computer-vision/?ef_id=_k_caf0aca35e551c1e41ecc93af8128d07_k_&OCID=AIDcmme9zx2qiz_SEM__k_c

af0aca35e551c1e41ecc93af8128d07_k_&msclkid=caf0aca35e551c1e41ecc93af8128d07#object-clssification

9. Kili Technology. Programming Image Classification with Machine Learning. Kili Technology. https://kili-technology.com/data-labeling/computer-vision/image-annotation/programming-image-classification-with-machine-learning

10. Kumar, V. (2019, June 20). Random Forest vs. Neural Network. KDnuggets. https://www.kdnuggets.com/2019/06/random-forest-vs-neural-network.html