Poop Tracker

Predictive modeling of dog movements through sensors

Derek Dong, Hunter Janetos, Flynn O'Sullivan, Mark Rittgers, Anthony Rose , Liam Warren

Introduction

Dogs poop a lot, and leaving poop on the ground poses environmental and public health concerns and may result in fines in certain municipalities. Our project aims to create a system consisting of an app connected via Bluetooth to a collar equipped with an Arduino microcontroller. This will allow a user to get notified on their dogs' defecation even when not physically present.

Requirements

Functional Requirements

- Bluetooth connection between the collar and the application capable of transferring data in real-time with at least 99% reliability within 40 feet
- The 3D printed Arduino collar attachment securely holds a power bank and the Arduino
- The application exports data to other locations such as the centralized cloud database
- The machine learning algorithm detects defecation events with at least 80%precision and 90% sensitivity

Non-Functional Requirements

- The machine learning algorithm runs efficiently on the Arduino without increasing latency
- The application never crashes during standard use
- Connection between devices in 10 seconds
- The application displays defecation locations accurate to 3 feet on a map

Testing

- We tested our collar on real dogs during actual excursions for live data collection in real-world conditions.
- Before live testing, we generated mock data to verify system behavior and database connectivity under controlled conditions.
- External testers also gathered additional data, helping us evaluate reliability of data collection across different users and environments.
- We used simulators and desktop testing tools to run and validate the app across multiple platforms, including iOS, Android, and Windows.
- Testing of Bluetooth connection was conducted initially at close ranges between devices connected to a power source, then later ran on its own battery

Evaluation

Results

•Connection: Reliable Bluetooth connectivity was maintained within 40 feet.

•Poop Detection Accuracy: Still under evaluation through continued testing.

•Application: The app successfully displays detected events and sends notifications to users.

Project Outcome

•The main project goals were achieved despite challenges such as account restrictions and gathering enough Arduinos for data collection.

•Future improvements could include user accounts, expanded testing, improved detection accuracy, and additional app-side features.

Design

Full System

- The Arduino connects through Bluetooth to the Device and Flutter app.
- Firebase hosts all data collection for ML training
- The user has access to the app

Physical Component

- 3D printed to fit the Arduino
- Battery soldered and attached through a gap
- Velcro holds them together to a collar

Application

- MVC pattern used
- Model holds the data
- App displays data and map
- Flutter app uses Bluetooth to communicate with the Arduino

Diagrams

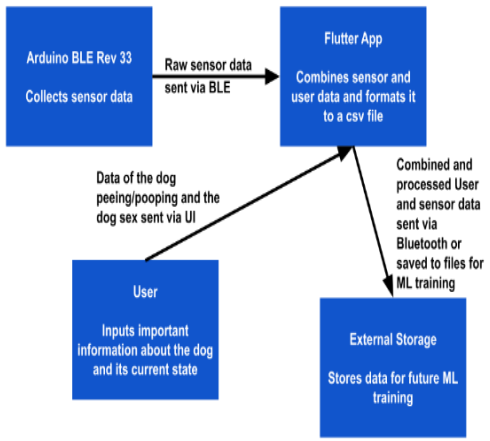

Figure 1: Data flow diagram of full system

```
INFO: Created TensorFlow Lite XNNPACK delegate for CPU.
— Model tensor info ————————————————
  Input  shape:       [  1 100   6]
  Input  dtype:       <class 'numpy.int8'>
  Input  scale:       0.061259
  Input  zero_point: -15
  Output shape:       [1 1]
  Output scale:       0.003906
  Output zero_point: -128

— Arduino sketch values ————————————
  // Paste these into your .ino file:
  const float  INPUT_SCALE      = 0.06125890f;
  const int8_t INPUT_ZERO_POINT = -15;
  const float  OUTPUT_SCALE     = 0.00390625f;
  const int8_t OUTPUT_ZERO_POINT= -128;

— Loading test data ————————————————
Loaded 2,907 rows from 9 file(s).
Windows: 113  |  normal=108  pooping=5  pos_rate=4.4%
  Test samples: 23  (normal=22  pooping=1)

— Classification report ————————————
              precision    recall  f1-score   support

      Normal      1.000     0.727     0.842        22
     Pooping      0.143     1.000     0.250         1

    accuracy                          0.739        23
   macro avg      0.571     0.864     0.546        23
weighted avg      0.963     0.739     0.816        23

— Sample predictions ———————————————
  Actual     Predicted    Confidence
  Normal     Normal       0.473  ✓
  Normal     Pooping      0.516  ✗
  Normal     Normal       0.465  ✓
  Normal     Normal       0.473  ✓
  Normal     Normal       0.465  ✓
  Normal     Normal       0.465  ✓
  Normal     Normal       0.496  ✓
  Normal     Normal       0.477  ✓
  Normal     Normal       0.496  ✓
```

Figure 2: Completed Test

Acknowledgements

We want to profoundly thank our sponsor Daniel Shepard and our advisor Craig Smith, alongside the University of New Hampshire Computer Science Department.