Return to Bali   version 1.1   March 2021

Towards AI in Ethnobotany

Introduction

Return to Bali's goal is to facilitate the creation of under-represented knowledge in machine learning. Return to Bali applies various machine learning procedures to the study of Ethnobotany on the island of Bali where Mead and Bateson, of second order cybernetics fame, collected field data for the book
Balinese Character: A Photographic Analysis (1942).

While Bali has been subject to numerous exploitative practices, this project aims to bend machine learning to collect data responsibly and to assess if machine learning might help maintain the body of local ecological knowledge that has been documented as declining specifically amongst Balinese youth.

Return to Bali includes a dataset, bali-26, produced from field videos documenting 26 ethnobotanically significant plants of South East Asia, several of which are indigenous to the island of Bali. The videos show the plants in multiple stages of growth and include fruits, leaves, branches and bark where applicable, from multiple locations within the field study site of Central Bali and under varying lighting conditions. It also includes several instances of some of the plants (such as snakefruit) at local markets.

The high definition mobile phone video files shot by data collectors in the field were converted to labeled images with the Catch & Release software package: https://github.com/realtechsupport/c-plus-r . Catch & Release also allows one to parse text from video interviews.

The bali-26 data set is the first (minor) collection of ethnobotanically significant plants of South East Asia made amendable to neural network based image classification. As such, it expands the domain of machine learning to include forms of knowledge that have not yet been represented. And on a purely technical level, the project demonstrates that even the best machine learning algorithms struggle to understand the visual complexity contained in the rich flora of Bali captured in the wild.

Additional Documentation: http://www.realtechsupport.org/new_works/return2bali.html

License

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)
Cite this software project as follows: 'Return to Bali version1.1'

Context

Return to Bali runs on Linux and Mac OS under Python3 and Flask with Chromium or Firefox.

The software uses the PyTorch framework to train and test image classifiers and connects to the Google Speech API (free) for speech processing. Library versions and dependencies are given in the requirements file.
The software has been tested on a desktop (i7-4770 CPU with 16GB of memory) and a laptop (i7-3667 CPU with 8GB of memory) under Ubuntu (20. 04 LTS and 18.04 TLS under kernels 5.2.8 and 5.3.0) and under Mac

OS (Catalina) with images sourced from .mp4 and .webm video (HD [1920 x 1080] at 30f/s; .mp4 H.264 encoded) from multiple (android OS) mobile phones and GoPro Hero 6 action cameras.

Browser Installation

Recommended browser: Chromium.

Install Chromium on Ubuntu:
sudo apt install -y chromium-browser

Install Chromium on Mac OS:
https://apple.stackexchange.com/questions/78805/chromium-builds-for-mac-os-x/215426#215426
(Currently recommended method)

Install the free Classic Cache Killer:
https://chrome.google.com/webstore/detail/classic-cache-killer/kkmknnnjliniefekpicbaaobdnjjikfp?hl=en

Software Installation

Clone the Return to Bali repository on GitHub

Open a terminal window and type:

>   git clone *https://github.com/realtechsupport/return-to-bali.git*

Cd to the project directory and run the following commands to update the python environment:

>   *chmod +x basics.sh*
>   *sudo sh basics.sh*

(This script just updates your Ubuntu installation and requires sudo to do so.)

Create a virtual environment:

>   *python3 -m venv env*

Activate the environment:

>   *source ./env/bin/activate*

Cd to to the project directory again. Install Requirements and Dependencies.
(This may take about 30 minutes.)

>   *pip3 install -r requirements.txt*

Enable the virtual environment

Activate the virtual environment:

source ./env/bin/activate

Start

(in the project directory):

python3 main.py ubuntu firefox debug   or
python3 main.py mac chromium no-debug

Specify all three items: OS, browser and debug mode. Supported OS: Ubuntu and Mac OS. Supported browsers: Chromium and Firefox (less stable). To run in debug mode replace' no-debug' with 'debug'.

The terminal window will display comments. You should see the launch screen in a browser window:
Return to Bali

crtl + / ctrl – increase / decrease zoom factor.

Stop

Stop the app from the terminal:

ctrl-c

Errors

If you see browser errors .. clear the browsing history:

c*trl-H - clear browsing data - clear data*

Description of the individual modules

The bali-26 dataset

This module describes the plants contained in the bali-26 dataset. The descriptions include typical uses as well as the locations in which field videos of the plants were collected.

Weather and satellite views

This module depicts rainfall, temperature data, a satellite image view and the current weather conditions of the field study site. The weather data was compiled by the Bali Botanical Gardens, and the view here (top diagram) shows the data for 2018 to 2019. Current temperature and rainfall is superimposed upon this map (red and green dots).

Climate, flora and rituals

This module aligns the weather patterns mentioned above with Balinese rituals of significance to Ethnobotany as well as the flowering and fruiting seasons of the plants contained in the Bali26 dataset.

Informal integrated agriculture

This module contains a short video with Made Darmajan, one of our data collectors and a rich source of informal knowledge on local flora.

Train image classifiers on a subset of the bali-26 dataset

This module allows one to create a fully functional - albeit low dimensional - convolutional neural network from one of several sample image sets. This module offers some insights into the impact of parameter settings on network training and performance.

Altering the parameters will impact training time and classifier performance. For example, reduced epochs and fewer images per category will shorten the training time but may compromise accuracy.

Two networks available: a simple (vanilla) CNN and the Alexnet architecture. The ore-trained option is available for the Alexnet architecture and usually improves performance – but does not allow control over pre-training and thus the logics with which your system is primed. This may be undesirable when networks are pre-trained on ImageNet or other biased image collections. You can select the normalization. A set of image normalization parameters have been calculated from the bali-26image collection (support local image logic!) and work well with the samples provided, which include:

bali-3   *Cacao, Passiflora and Aroid/Suweg*
bali-3C   *Durian, Nilam and Snakefruit*
bali-3D   *Coffee Arabica, Papaya, Sugarpalm*

Unless you have a GPU enabled computer, training on your home device is a slow process.(55 min for 3 categories with 1200 images trained for 20 epochs with Alexnet model on an Intel i7-4770 CPU @3.4GHz with 16GB RAM).Click 'train-network' once you make your choices and go for a walk. Training on generic computers will take an hour or more. When all is done, the training results and Top-1 errors per category.

Evaluate image classifiers on the bali-26 dataset

This module loads one of several deep neural net classifiers trained on the bali-26 dataset and allows you to select an image from the validation set to test the selected classifier performance. The best prediction and the confidence level are listed as are the top 3 possibilities (two alternates to the first choice). These results are calculated in real time on your local computer. Left click to select a single image from the image gallery, then click 'classify image' to obtain the results. The first time you run AIE, the selected trained model will be downloaded from pCloud automatically. The gallery will only load once. To inspect additional images, left click to select a new image, then click 'classify image' again, and repeat.

Video enhanced classification

This module demonstrates a novel classification approach that takes video specific information into account to improve the performance of a set of imperfect classifiers.

Details and demonstration are here: https://github.com/realtechsupport/video-enhanced-classification