Dog Breed Identification

Presented by
Yue Bi
Outline

• Dataset
• Model
• Experiment Results
Outline

- Dataset
- Model
- Experiment Results
Figure. Stanford Dogs Dataset
Figure. Stanford Dogs Dataset
<table>
<thead>
<tr>
<th>Image id</th>
<th>Breed</th>
</tr>
</thead>
<tbody>
<tr>
<td>000bec180eb18c7604dcecc8fe0dba07</td>
<td>boston_bull</td>
</tr>
<tr>
<td>001513dfcb2ffafc82ccc4d8bbaba97</td>
<td>dingo</td>
</tr>
<tr>
<td>001cdf01b096e06d78e9e5112d419397</td>
<td>pekinese</td>
</tr>
<tr>
<td>00214f311d5d2247d5dfe4fe24b2303d</td>
<td>bluetick</td>
</tr>
<tr>
<td>0021f9ceb3235effd7fcede7f7538ed62</td>
<td>golden_retriever</td>
</tr>
<tr>
<td>002211c81b498ef88e1b40b9abf84e1d</td>
<td>bedlington_terrier</td>
</tr>
<tr>
<td>00290d3e1fdd27226ba27a8ce248ce85</td>
<td>bedlington_terrier</td>
</tr>
<tr>
<td>002a283a315af96eaea0e28e7163b21b</td>
<td>borzoi</td>
</tr>
<tr>
<td>003df8b8a8b05244b1d920bb6cf451f9</td>
<td>basenji</td>
</tr>
<tr>
<td>0042188c895a2f14ef64a918ed9c7b64</td>
<td>scottish_deerhound</td>
</tr>
</tbody>
</table>

*Table. Stanford Dogs Dataset (labels)*
Dataset Overview

• Dataset: 10222 images

• Breed: 120

• Pick top 25 frequent breed to classify (2668 images)

• Randomly pick 80% of the images as training set and other 20% as validation set
## Table. Top 25 frequent breed (labels)

<table>
<thead>
<tr>
<th>Order</th>
<th>Breed</th>
<th>Order</th>
<th>Breed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>scottish_deerhound</td>
<td>14</td>
<td>cairn</td>
</tr>
<tr>
<td>2</td>
<td>maltese_dog</td>
<td>15</td>
<td>beagle</td>
</tr>
<tr>
<td>3</td>
<td>afghan_hound</td>
<td>16</td>
<td>japanese_spaniel</td>
</tr>
<tr>
<td>4</td>
<td>entlebuchar</td>
<td>17</td>
<td>australian_terrier</td>
</tr>
<tr>
<td>5</td>
<td>bernese_mountain_dog</td>
<td>18</td>
<td>blenheim_spaniel</td>
</tr>
<tr>
<td>6</td>
<td>shih-tzu</td>
<td>19</td>
<td>miniature_pinscher</td>
</tr>
<tr>
<td>7</td>
<td>great_pyrenee</td>
<td>20</td>
<td>irish_wolfhound</td>
</tr>
<tr>
<td>8</td>
<td>pomeranian</td>
<td>21</td>
<td>lakeland_terrier</td>
</tr>
<tr>
<td>9</td>
<td>basenji</td>
<td>22</td>
<td>saluki</td>
</tr>
<tr>
<td>10</td>
<td>samoyed</td>
<td>23</td>
<td>papillon</td>
</tr>
<tr>
<td>11</td>
<td>airedale</td>
<td>24</td>
<td>whippet</td>
</tr>
<tr>
<td>12</td>
<td>tibetan_terrier</td>
<td>25</td>
<td>siberian_husky</td>
</tr>
<tr>
<td>13</td>
<td>leonberg</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Outline

• Dataset
• Model
• Experiment Results
VGG16* Model

- Prepossessing
- Convolution Parameters
- Activation Function
- Pooling
- Fully-Connected Layers

*Visual Geometry Group, University of Oxford
Prepossessing Steps

- Data Augmentation: horizontal flip (double dataset size)

*Figure. horizontal flip*
Prepossessing Steps

- Fix image size to $224 \times 224$

Figure. fix image size
Netscope (VGG16)
Convolution Parameters

- Kernel Size: $3 \times 3 \times 3$

*Figure. convolution kernel size*
Convolution Parameters

- Stride: 1 pixel

*Figure. convolution stride*
Activation Function

- ReLU

*Figure. ReLU function*
Pooling

• Max-pooling over a $2 \times 2$ pixel window with stride 2.

*Figure. max-pooling example*
Pooling

- Average-pooling over a $2 \times 2$ pixel window with stride 2.

*Figure. average-pooling example*
Fully-connected Layers

• A stack of convolutional layers is followed by three Fully-Connected layers:

• The first two have 4096 channels each.

• The third contains 1000 channels.
Dropout

- Dropout regularization for the first two fully-connected layers to reduce overfitting.

- Dropout ratio is 0.5.

Figure. dropout example

(a) Standard Neural Network  (b) Neural Net with Dropout
Multinomial Logistic Regression

- Use Multinomial Logistic Regression as the last layer of the network (like softmax)

\[
P(r(Y_i = 1)) = \frac{e^{\beta_1 X_i}}{\sum_{k=1}^{25} e^{\beta_k X_i}} \quad P(r(Y_i = 2)) = \frac{e^{\beta_2 X_i}}{\sum_{k=1}^{25} e^{\beta_k X_i}} \quad \ldots \quad P(r(Y_i = 25)) = \frac{e^{\beta_{25} X_i}}{\sum_{k=1}^{25} e^{\beta_k X_i}}
\]

\[
L(b_k) = \sum_{i=1}^{n} \left( \frac{e^{X_i b_k}}{1 + e^{X_i b_k}} \right)^{y_i} \left( \frac{1}{1 + e^{X_i b_k}} \right)^{1-y_i}
\]

\[
\hat{\beta}_k = \text{argmax}_{b_k} L(b_k)
\]
Outline

- Dataset
- Model
- Experiment Results
# Results

Table. Results of 25 classes

(Training set: 4228 images, Test set: 1108 images)

<table>
<thead>
<tr>
<th></th>
<th>Max-pooling</th>
<th>Average-pooling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.874</td>
<td>Accuracy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.866</td>
</tr>
</tbody>
</table>
## Results of Different Number of Classes

<table>
<thead>
<tr>
<th># of class</th>
<th>Accuracy</th>
<th>Training set size</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>0.874</td>
<td>4228</td>
</tr>
<tr>
<td>12</td>
<td>0.921</td>
<td>2126</td>
</tr>
<tr>
<td>6</td>
<td>0.946</td>
<td>1104</td>
</tr>
</tbody>
</table>

*Table. Results of different number of classes (Max-pooling)*
Show Errors
Show Errors
Show Errors
Future Work

• Detect mixture of breeds

• Help determine the breed of dog from shelter easily
Thank you !
References


[2]: https://www.kaggle.com/gaborfodor/keras-pretrained-models

[3]: https://keras.io/applications/#vgg16

[4]: https://en.wikipedia.org/wiki/Multinomial_logistic_regression