CLOUD CLASSIFICATION OF CELLPHONE PHOTOS BY MACHINE LEARNING

Kun Yue, Jing Zhuang, Lili Ding
Moji Co., Ltd, Beijing, China
2017.12.31
Nowcasting by Machine Learning

Forecast by Location in Minute Level
Nowcasting by Machine Learning

Radar Images

Radar Images Preprocessing

Radar Images Denoising

Spatiotemporal Forecasting

Input the Latest radar images

A → A → ... → A

Forecast the future radar images

T1 → T2 → T9 → T10 → T11 → T12 → T20

Results Merging

Display

TensorFlow

OpenCV

Python

Caffe

pandas

NVIDIA

NumPy

Machine Learning

Moji Co., Ltd
Mainly forecast the precipitation for the next two hours in minutes level, the forecast data is updated every 6 minutes. Users request data according to the latitude and longitude.

Observed data inputed in the nowcasting system include:

- Radar echo Image (updated per 6 minutes)
- Satellite Image (updated per 10 minutes)
- Numerical Model (updated per hour)

Problems in the nowcasting system:

- Radar echo images is inaccurate sometimes because of noise, topography or hardware problem
- Satellite Image cannot reflect the amount of rainfall accurately
- Numerical model and Observed station is distribute sparsely
- Interpolation reduces accuracy and updates slowly
Nowcasting by Machine Learning

Is there any other data that can be used as a supplement?

Demand: Timeliness, Geographic accuracy

Mobile Phone data could match two points above

What UGC data Moji has?

- Depend on 500 million users
- Depend on 50 million Daily active users
- 50,000 ~ 100,000 images/day
- 50,000 ~ 60,000 commons/day
- 60,000,000 cellphone barometers/day

The most intuitive data is images

There are various categories of images

Cloud maps can accurately analyze the weather conditions in the area where the user is located

What we need to do is to identify the information in the cloud through image recognition
Nowcasting by Machine Learning

- History Image Server
- Classification Model
- Web Server

Offline:
- Backup
- Update Online Model
- Retrain Model
- Check Classification Result Manually

Online:
- Image Server
- Classification Model
- Nowcasting Model
Nowcasting by Machine Learning

- Infrequent dataset for cloud classification
- Less public hand-classified network dataset
- Hard to classify by non-professional people

- Image data collection by engineers
- Cloud image tagging by meteorologist

In the first phase, our meteorologist spent 1 month tagging nearly 5000 cloud images, including the type and the position.

However, there are 13 categories, with an average of fewer than 500 in each category. For machine learning and training, the number of images is still small, and it costs a lot to continue tagging.
Data Preparation for Cloud Classification Model

1. First Dataset
2. Classify Images Manually
3. Train Model
4. New Datasets
5. Check Result Manually
6.
Training for Cloud Classification Model

Difficulties in step from 0 to 1:
- Infrequent training data
- Too much cloud categories
- Some categories have very small differences

☑ Simple network for rapid training and testing in early stage
  CNN Network (10 layers)
  Network parameters Xavier initialization
  Training data for each type of 500 images (less than the number of categories on the image were randomly intercepted or repeat complement)

☑ Cyclically repeated training and check the prediction results, the similar types of cloud images were merged, the different types of cloud split, the final establishment of 13 categories of clouds as a training target

☑ However, the training effect is still not satisfactory, Top1 accuracy can only reach 40-50%
Training for Cloud Classification Model

Transfer learning

For the short time, we can not mark large quantities of available training samples, we adopt the transfer learning, improve the model accuracy, and achieve a better prediction level.

We used the Resnet network trained for the IMAGENET Large Scale Visual Recognition Challenge data set (containing the 1000 categories and 1.2 million images) as a pre-training model.
Training for Cloud Classification Model

Train ImageNet Model

Model1 Training

Input Layer

Share Weight

Output Layer

Train Cloud Model

Model2 Training

Input Layer

Output Layer

Classify Cloud Image

Model2 Trained

Input Layer

New Weight

Output Layer
Model Accuracy Comparison

<table>
<thead>
<tr>
<th></th>
<th>Top 1</th>
<th>Top 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN - 13 categories</td>
<td>0.61</td>
<td>0.94</td>
</tr>
<tr>
<td>CNN - 10 categories (Transfer Learning)</td>
<td>0.77</td>
<td>0.95</td>
</tr>
<tr>
<td>CNN - 13 categories (Transfer Learning)</td>
<td>0.73</td>
<td></td>
</tr>
</tbody>
</table>
Only cloud classification model can not be deployed in our online environment: Users of our application upload all kinds of images they took.

We need a preprocess model to distinguish the non-cloud images which will be filtered by the pre-model. The pre-model is simple, only need to predict two categories. We use the cloud images we prepared for the cloud model and some non-cloud images that our app users upload to train the pre-model and get an accuracy of 95% easily.
Application scenarios and Future Work

- Application on consumer sides and business sides
- Post for product on Exhibit Hall 3(ACC)  Tuesday 9th Jan 2018
- Expand the image dataset  Jan - June  2018
- Improve accuracy with cellphone barometers, satellite datas  Mar - Sep 2018
- Expand recognition type, like: sunny, hail, rain, snow  Sep-Dec 2018