Final Project

Parallel program improve web crawler achieving AI model self-learning

Group： 24
Member: 0857214 蔡詠平
0856703 黃威竣
309551138 閻俊宇

THE MAIN CONTENTS

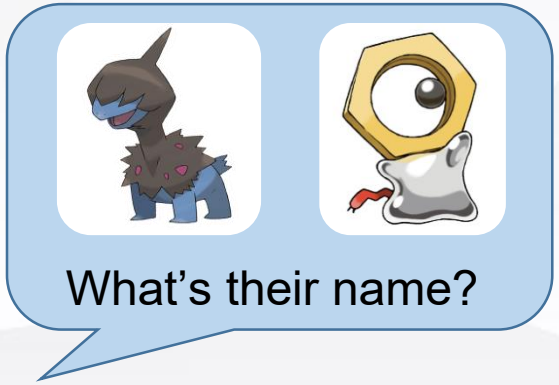
PLEASE ENTER YOUR SUBTITLE HERE

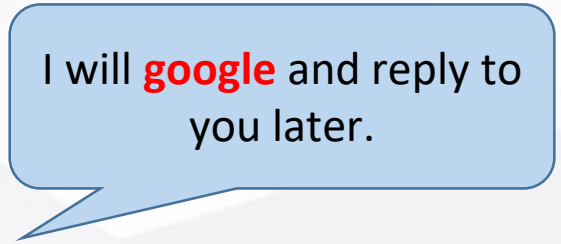
PART 1

Motivation and Introduction

01 Motivation

- Imitate the action of humans searching for information online.
- What humans can not do is that reading a lot of information at a time but machines can.
- We hope to accelerate machine autonomous learning by web crawler through parallel programming.





02 Introduction

- We trained a classifier that can classify seven categories of food images by our self.
- According to the class with weaker recognition, we using crawlers to automatically download pictures from the Internet as our training data set can effectively improve the performance of the model.

Apple pie

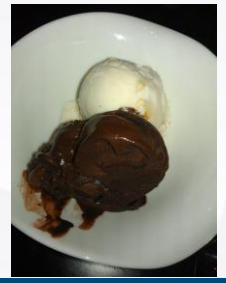
Chocolate cake

Donuts

Hamburger

Hot dog

Ice cream



Pizza

LIBSVM: A Library for Support Vector Machines

- **An easy-to-use, fast and effective SVM pattern** recognition software package developed and designed by Associate Professor Lin Zhiren of National Taiwan University.

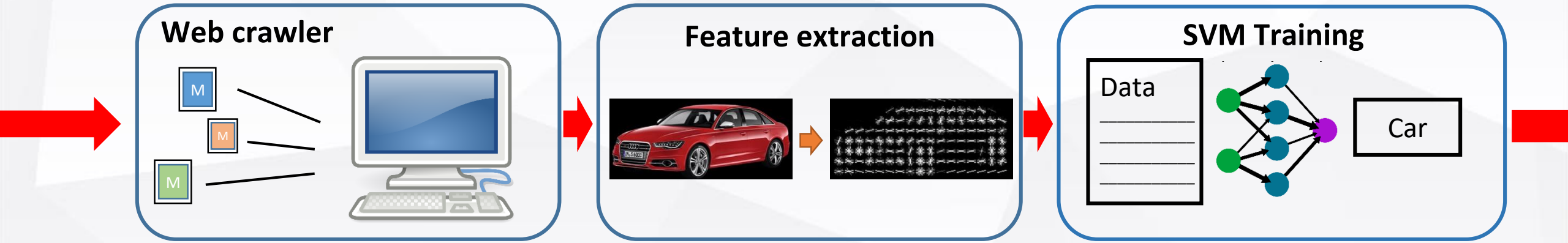
Reference: Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. 2011, 2, 1–27.

PART 2

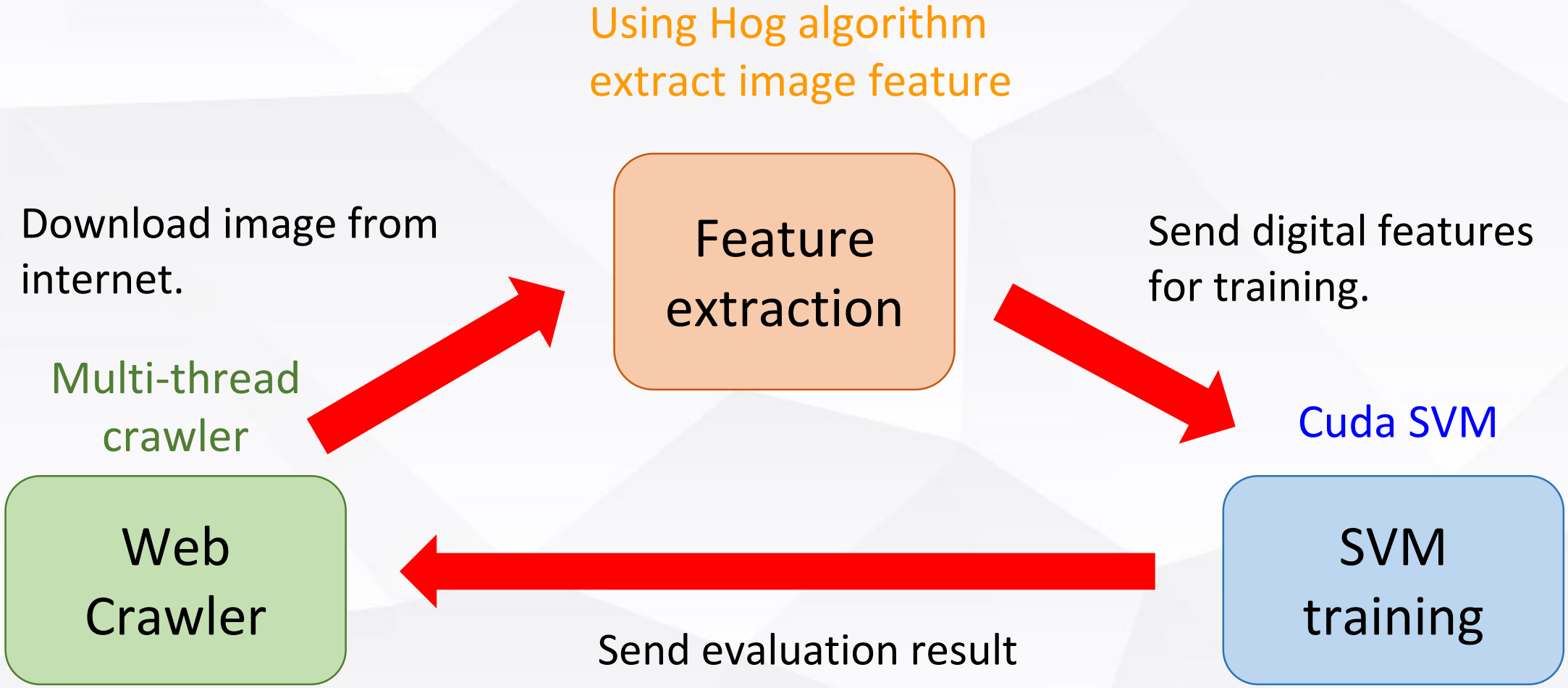
Problem and Solution

04 Problem

- The three training stages of the system always wait for each other.
- Especially the SVM training stage takes a lot of time so that the other two stages are always waiting.
- Feature extraction and neuron computation use a lot of Independent loops to calculate. They are all able to be parallel.
- Web Crawler's task can also be speedup by parallel programing.

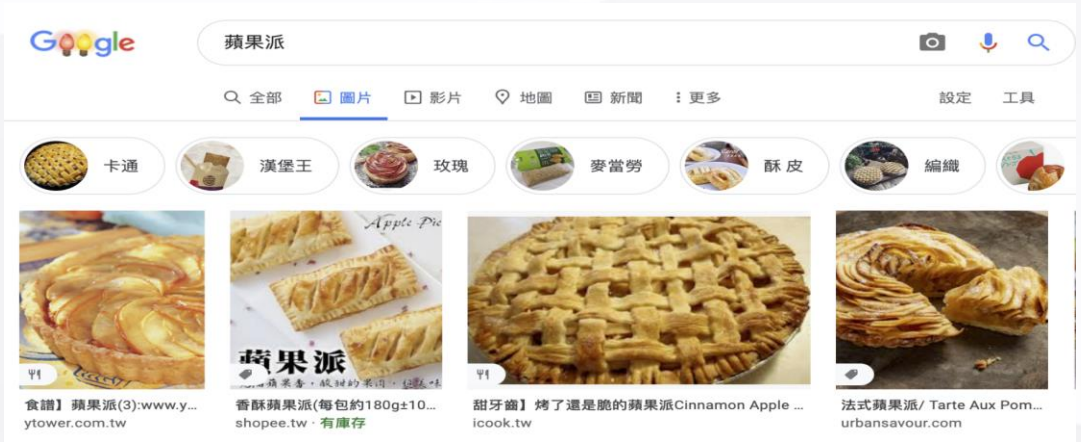


System Flow


Using Hog algorithm
extract image feature
Download image from
internet.
Feature
extraction
Send digital features
for training.
Multi-thread
crawler
Cuda SVM
Web
Crawler
SVM
training
Send evaluation result
to web crawler stage.

Web crawler parallel

Task



Method

Selenium is a automated testing framework used to validate web applications. We use **Selenium** to simulate human click、scroll 、hover operation.

Squid is a caching proxy server, we use **Docker** to Deploy Squid Server.

We parallel here, **using threads to upgrade**.

Challenges encountered

1. How to simulate human operation webpage.
2. Frequent request will be considered an attack

Metrics

	one-thread	one-thread use proxy	multi-thread use proxy
time(s)	56.1	827.34	**12.1**

crawler 700 images

Image preprocess

Method

We use HOG method(histogram of oriented gradients) to extract image feature

First we translate image to eigen vector. Because the image and image resolution is too big, so it takes a lot of time to convert data

So we parallel here, **using threads** to convert huge data for decrease convert time.

Metrics

Threads	times
1	1290s
2	673s
4	353s
8	169s
16	86s
32	48s
48	28.5s

SVM cuda parallel

Method

- Store information in .pkl format speedup load data time.
- CUDA accelerate model forward and back propagation.
- Iteration: 6000
- lr initiation: 0.001
- Batch size: 2500
- GPU memory usage: 3300 MB

Flow	Before	After opt
Initialization	0.987 s	0.987 s
Load Data	60.995 s	0.713 s
Training	719.805 s	49.566 s
Evaluation	0.2608 s	0.2792 s

SVM cuda parallel

Comparison to libsvm (Related work)

Test result accuracy (%)									
	apple pie	chocolate cake	donuts	hamburger	hot dog	ice cream	pizza	**mean acc.**	**Spend time (s)**
libsvm	19.0	29.5	26.5	16.5	20.5	35.5	42.0	**27.1**	4284.5
cuda svm	21.0	31.0	25.0	18.5	29.0	47.5	52.0	**32.0**	**49.5**

PART 3

Evaluation

Platform

Device

CPU: Intel(R) Xeon(R) Gold 6136 CPU @ 3.00GHz

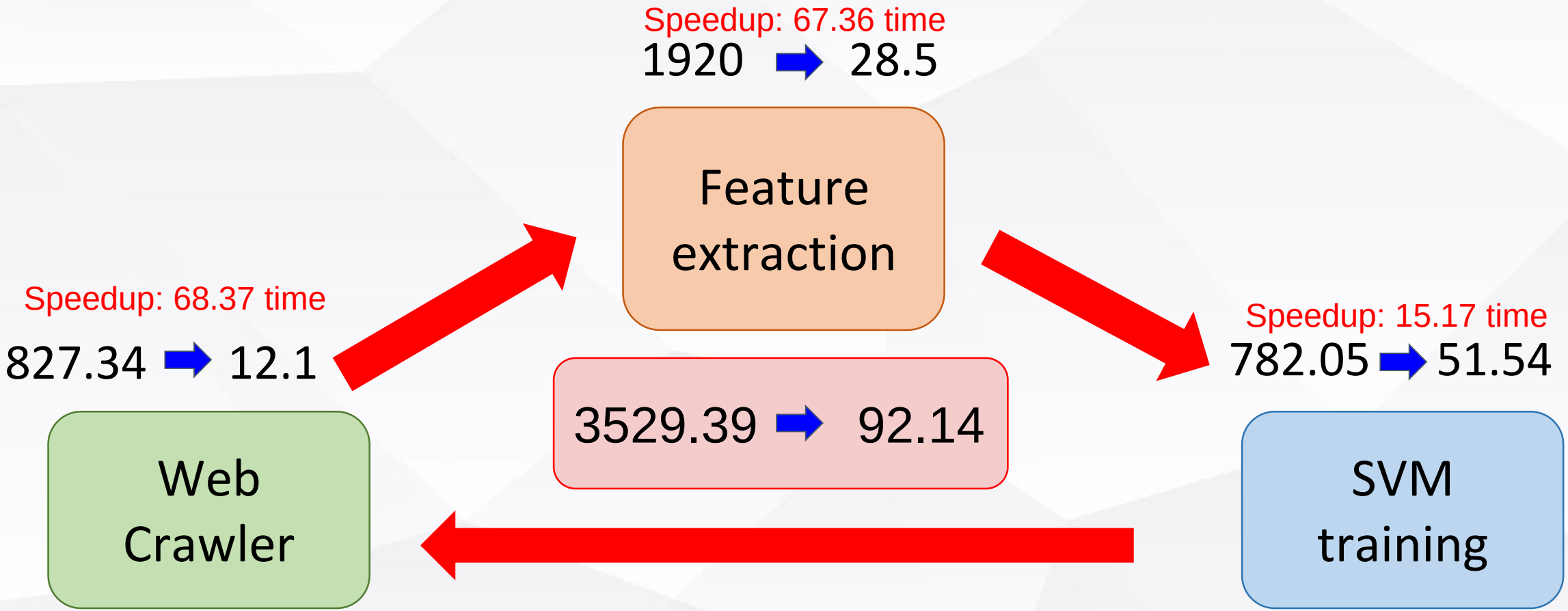
Core: 2 * (12 cores 24 threads)

GPU: RTX 2080 Ti 12GB

OS

1. Centos 8
2. Ubuntu 16.04

Metrices


Speedup: 67.36 time
1920 28.5
Feature extraction
Speedup: 68.37 time
827.34 12.1
Speedup: 15.17 time
782.05 51.54
3529.39 92.14
Web Crawler
SVM training

Evaluation

Round 1

```
Chocolate_cake accuracy: 31.000000 %
Donuts accuracy: 25.000000 %
Ice_cream accuracy: 47.500000 %
Hot_dog accuracy: 29.000000 %
Hamburger accuracy: 18.500000 %
Pizza accuracy: 52.000000 %
apple_pie accuracy: 21.000000 %
```

Round 2

```
Chocolate_cake accuracy: 25.500000 %
Donuts accuracy: 23.000000 %
Ice_cream accuracy: 44.500000 %
Hot_dog accuracy: 29.500000 %
Hamburger accuracy: 22.000000 %
Pizza accuracy: 54.500000 %
apple_pie accuracy: 22.000000 %
```

Round 3

```
Chocolate_cake accuracy: 27.000000 %
Donuts accuracy: 25.000000 %
Ice_cream accuracy: 42.000000 %
Hot_dog accuracy: 31.000000 %
Hamburger accuracy: 24.000000 %
Pizza accuracy: 50.000000 %
apple_pie accuracy: 21.500000 %
```

13 Conclusion

- We use parallel programs including multi-threading, cuda and other techniques to significantly improve the efficiency of the overall system. Compared with the sequential method, **our optimized system will be 35 times faster.**
- Compared with Libsvm, our model has similar performance and is faster a lot when using cuda.
- For weaker categories, adding data from web crawlers has grown in our case.
- As network data will be constantly updated, maybe our model performance will get better and better from time to time.

14 Contributions

Web Crawler: 閻俊宇

Image preprocess: 黃威竣

SVM: 蔡詠平

Integration work: 蔡詠平、黃威竣、閻俊宇

PPT: 蔡詠平、黃威竣、閻俊宇

Q & A

Final Projection

Thank you for listening