ISSN 2466-135X
Vol.8, No.1

The 8th International Conference on

BIG DATA APPLICATIONS AND SERVICES (BIGDAS2020)

Proceeding

MULTIMEDIA
LIFE
STORAGE
NETWORK
DATABASE
SYSTEM
BIG DATA
SOCIETY
SCIENCE
CLOUD
TREND
CLUSTER
BUSINESS
ANALYSIS
GRAPHICS
VISUALIZATION

November 26-28, 2020
Busan, South Korea

Hosted by
Korea Big Data Service Society

THE KOREA
BIG DATA SERVICE SOCIETY
한국빅데이터서비스학회

CBNU
충북대학교
LINC+

충북대학교
빅데이터연구소

Table of Contents

A Generic Programming Approach to Execute NNEF Files in a Massively Parallel Way

Nakhoon Baek

School of Computer Science and Engineering
Kyungpook National University
Daegu 41566, Republic of Korea
nbaek@knu.ac.kr

Abstract. Recently, we have many research works on the neural networks and their related issues, even for network communications and information exchange. For exchangeability of neural network frameworks, the Neural Network Exchange Format (NNEF) specification is now widely used. NNEF file interpretation can be accelerated through parallel processing techniques. In this paper, we present a prototype implementation of NNEF execution system with massively parallel-processing accelerations, and also with generic programming support. Our prototype shows the possibility of this generic programming, with thrust library. We will tune the prototype acceleration to achieve more speed ups.

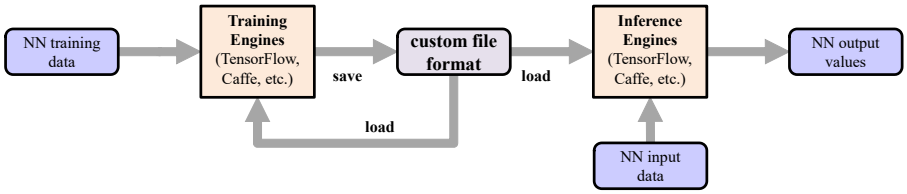
Keywords: neural network, NNEF, massively parallel processing, acceleration.

1 Introduction

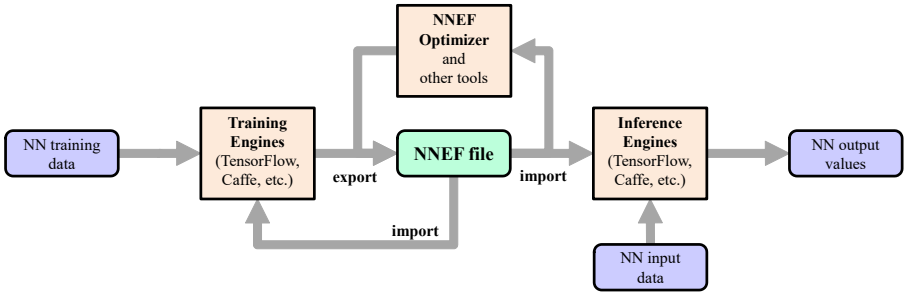
In these days, we have lots of *machine-learning* (ML) applications, to various fields, including communications and information exchange. In any application field, the fundamentals of ML will be the training and application of the deep learning process. In this paper, we will focus on the *neural network* (NN) techniques, among various fields in machine learning.

In a typical scenario, the neural network applications need two stages: *training* and *inference*, as shown in Figure 1.(a). In the training stage, a training engine will be used to generate well-computing neural network values, from the neural network training data. The resulting neural network values are stored in a disk file. Later, the stored neural network values are retrieved by an inference engine, with a new set of neural network input data, and we get the output values of the trained neural network, as the final result of this neural network full processing sequence.

Practically, we have several widely used neural network frameworks: *TensorFlow* [1], *Caffe* [2], *Keras* [3], and others. Traditionally, these neural network frameworks work as an independent way. A specific neural network framework trains a neural network, and stores the resulting vales in its own file format. Later, *the same* neural network framework reads the stored values, and process user inputs to the final output values.



(a) traditional processing



(b) with NNEF files

Fig. 1. The NNEF file changed of neural network processing.

Recently, several neural network data exchange file formats are introduced. Among them, we selected the *Neural Network Exchange Format* (NNEF) [4] as our main target. NNEF act as the standard protocols and/or file formats to exchange the neural network structures and their related data sets, we need standard protocols and/or file formats. NNEF is exactly the *de facto* standard file format, for the neural network frameworks, and managed by the *Khronos Group* [5], an industrial standard organization.

With NNEF files, the neural network processing has some changes. As shown in Figure 1.(b), the *training engines* and the *inference engines* can use NNEF as a platform-independent file format. Thus, now there is no need to use the same neural network framework as both of the training engine and the inference engine.

Although the NNEF specification provides the file format only, the file format can be parsed and even executed to simulate the specified neural network. For example, we can introduce an *NNEF optimizer*, as another independent tool, as shown in Figure 1.(b). Recently, the Khronos NNEF official *GitHub* site presented an NNEF interpreter, implemented in C++ templates [6]. To accelerate the NNEF execution, OpenCL (Open Computing Language) [7] and OpenMP (Open Multi-Processing) [8] have been used in the previous works, [9] and [10].

In this paper, we aimed to apply a massively parallel way of generic programming, to this NNEF interpreter implementation. As shown above, the NNEF interpreter can

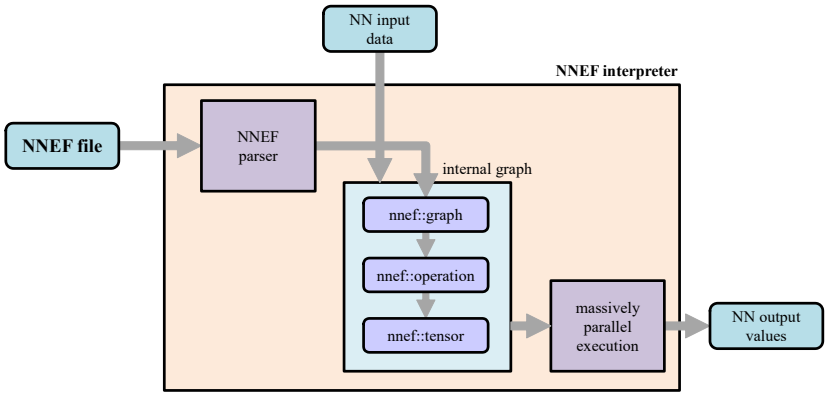


Fig. 2. The overall flow of our NNEF execution system.

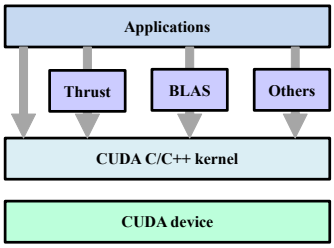


Fig. 3. Overall architecture of CUDA-related tools.

be used as a casual inference engine for the neural network computation, for various application fields, including network communications and information exchange. We will show some generic programming options in the following sections, and a simple prototype implementation will be followed. Finally, we represent the conclusion and future work.

2 Design

Based on the original Khronos NNEF interpreter [6], we used its NNEF parser routines, as is. Thus, after parsing an NNEF file successfully, we have an NNEF graph structure with the following three major components:

- **A single NNEF graph**: it contains the whole NNEF graph representation. Internally, an NNEF graph consists of NNEF operations and NNEF data nodes.

- **A sequence of NNEF operations**: this sequence represents all the operations in the NNEF graph. Each NNEF operation node has its corresponding NNEF data nodes.
- **A set of NNEF data nodes**: the set contains all the data nodes used by the NNEF operations.

A simple implementation of the NNEF execution process can be a graph traversal algorithm, which retrieves all the NNEF operations, with their corresponding NNEF data nodes. Our current implementation performs each NNEF operation, during the graph traversal process, as shown in Figure 2.

For the accelerated execution of this NNEF file traversal, we can apply *massively parallel programming* techniques. Additionally, we can consider the new trend of *generic programming* techniques. Combining these two techniques, our solutions can be generic programming features, from the massively parallel programming.

In our case, we will focus on the *CUDA* (compute unified device architecture) [11] as the major massively parallel programming paradigm, mainly due to its widespread uses and also efficiency, in comparison to other massively parallel programming paradigms.

Our candidate tools for the generic programming with massively parallel paradigms can be summarized as follows:

- **CUDA C++ templates**: Recently, CUDA implementation provides C++11 templates. It is much flexible, though its generic features are still much limited.
- **CUDA Thrust library** [12]: It is included as an independent library into current CUDA implementations. Currently, it is focused on 1D array operations. It lacks serious 2D matrix operations and 3D tensor operations.
- **CUDA BLAS library** [13]: Basic Linear Algebra Subprograms. It is one of the most famous linear algebra libraries. It offers various 2D and 3D operations. In contrast, it is too heavy and too complex to be used for casual applications.
- **SYCL** [14]: It is a higher-level programming model for OpenCL as a single-source domain specific embedded language, based on C++11. Currently, it offers the highest level abstraction to the massively parallel programming paradigms. However, at least at this time, it lacks an efficient implementation, compared to CUDA implementations.

As a prototype implementation, we tried to show the possibility of *Thrust* library, in implementing an NNEF interpreter, especially for 1D array operations, as shown in Figure 3. Later, we can add more and more operations, including 2D and 3D operations.

For the acceleration of this execution process, we used Thrust kernel programs, which are carefully designed for a set of NNEF operations or a single NNEF operation, according to the operation complexities. Our current implementation can provide limited number of NNEF operations, and some not-supported operations will be covered soon.

```
version 1.0;
graph mnist( input ) -> ( output ) {
  input = external<scalar>( shape=[784,1] );
  wih = variable<scalar>( shape=[200,784], label="mnist-wih" );
  who = variable<scalar>( shape=[10,200], label="mnist-who" );
  hidden = sigmoid( matmul( wih, input ) );
  output = sigmoid(matmul( who, hidden ) );
}
```

Fig. 4. An example NNEF file.

```
0.04075291
0.00826926
0.05436546
0.02137949
0.01371213
0.03757233
0.00205216
0.90605090
0.02076551
0.03548543
```

Fig. 5. An example result from our implementation.

3 Prototype Implementation

The key idea with our implementation can be summarized in a step-by-step manner, as follows:

- **step 1.** memory coalescing – In the original NNEF parser implementation, they allocate all the NNEF data memory to its own local space. In contrast, we change the design and also the implementation to coalesce all the NNEF memory to a single and large memory area.
- **step 2.** memory transfer to the CUDA/Thrust area – Thrust needs to use its own memory access methods and also iterators. To use these features, we transfer the main memory information to Thrust domain.
- **step 3.** Thrust executions – For each NNEF operation, we implement its corresponding Thrust template, and execute it to perform the NNEF operation.
- **step 4.** memory transfer back to the main memory – At the end of all the NNEF operations, the Thrust memory area contains the final result of the NNEF execution. We copy back the Thrust memory area to the main memory, for further processing, as same as typical Thrust applications.

After these implementations, we compared the new Thrust-based execution to the original C++-template-based official implementation [6]. As an example, we used the *MNIST* hand-writing training case [15,16], with the NNEF file shown in Figure 4.

After processing this NNEF file, with proper data sets from the MNIST case, we get the sample results, as shown in Figure 5. Including this one, our test cases show that our implementation works well, and show some remarkable speed ups, in comparison to the original C++ template implementation.

4 Conclusion

In this work, we presented an NNEF interpreter system, with Thrust-based accelerations. Since Thrust can provide massively parallel execution with GPUs, even in a generic programming way, our final result shows speed ups, with reduced development costs. In contrast, our Thrust-based implementation may have some drawbacks, originated from the underlying CUDA library. As an example, our Thrust-based implementation needs CUDA-supporting GPU support, for its acceleration. Thus, in some cases, where CUDA-supporting GPUs are not available, our work shows only limited performance. In the near future, we will release the full support framework for NNEF and its related specifications.

References

1. M. Abadi et al., TensorFlow: Large-scale machine learning on heterogeneous systems. white paper available from tensorflow.org, 2015.
2. Y. Jia et al., “Caffe: Convolutional architecture for fast feature embedding,” Proc. 22nd ACM Int’l Conf on Multimedia (MM ’14), 2014.
3. Keras Homepage, “http://www.keras.io,” retrieved in Jun 2020.
4. The Khronos NNEF Working Group, Neural Network Exchange Format, version 1.0.1. Khronos Group, 2019.
5. Khronos Group, “http://www.khronos.org/,” retrieved in Jun 2020.
6. KhronosGroup/NNEF-Tools, https://github.com/khronosgroup/nneftools, retrieved in Aug 2020.
7. Khronos OpenCL Working Group, The OpenCL Specification, Version 3.0.1. Khronos Group, 2020.
8. OpenMP Homepage, “http://www.openmp.org/,” retrieved in Aug 2020.
9. M. Yu, T. C., and J. Lee, “Accelerating NNEF framework on OpenCL devices using clDNN,” IWOCL 2020, 2020.
10. N. Baek and S.-J. Park, “An OpenMP-based parallel execution of neural networks specified in nnef,” ICA3PP 2020, 2020.
11. CUDA Zone, “http://developer.nvidia.com/cuda-zone,” retrieved in Aug 2020.
12. Thrust: CUDA Toolkit, “https://docs.nvidia.com/cuda/thrust/index.html,” retrieved in Aug 2020.
13. cuBLAS, “https://developer.nvidia.com/cublas,” retrieved in Aug 2020.
14. R. Keryell, M. Rovatsou, and L. Howes, SYCL 1.2.1 specification, Revision 5. Khronos OpenCL Working Group - SYCL Subgroup, 2019.
15. T. Rashid, Make Your Own Neural Network. CreateSpace Independent Publishing Platform, 2016.
16. makeyourownneural network, “https://github.com/makeyourownneuralnetwork/makeyourownneuralnetwork,” retrieved in Aug 2020.

Improving Object Detection in Noisy Images with Stylized Dataset

Kyu-hong Hwang[1], Myung-jae Lee[1], Young-guk Ha[1,*],

[1] Department of Computer Science & Engineering,
Konkuk University.
120 Neungdong-ro, Gwangjin-gu, Seoul, Korea
gfvxgd2k@konkuk.ac.kr, dualesspresso@naver.com, ygha@konkuk.ac.kr

Abstract. With the improving Image Classification based on Convolutional Neural Networks (CNN), object-detection also improving too. In addition, the field of autonomous driving and smart robots applying these technologies are also developing closely with people. It is closely related to the person. So, a small mistake can cause great harm to a person. In order to prevent such a big accident, the system must be strong against noise, and there should be no misinterpretation of objects due to small changes. In CNN-based image classification, which should not make mistakes, research has proved that it biased towards textures object detection. In order to secure this, we propose a method to reduce the false positive rate by diversifying learning data without changing the architecture of CNN. Through this, it is greatly reduced the misclassifiy of various textures when object detection and shows strong performance against noise.

Keywords: Deep Learning Dataset Computer vision

1 Introduction

CNN-based image classification is constantly evolving and shows better performance than people in modern times.[1] The field of object detection based on such object classification is also improving. Object detection can be used in fields such as autonomous vehicles and smart robots, which can have a beneficial effect on people. Since these are closely related to people, they should not be fast. Stability and robustness should be ensured. It becomes a problem if systems that should have these characteristics behave differently than the designer intended. If you have a car picture of an elephant textual, a person sees it and judges it to be a car, but CNN can classify it as an elephant. This is because the CNN classifies objects with texture rather than shape.[2] For this reason, object detection based on CNNs can be fatal if misclassification. In this paper, we propose a method to improve performance through diversity of training data while maintaining the structure of CNN.

* Corresponding author

2 Design and Implements

We diversify the training data of YOLO V4to improve performance. [3] Using MS COCO data for train object detection.[4] The implement was conducted with original coco data, data converted to various textures through style transfer GAN, and data extracted with edges.[5]

Fig. 1. Style transfer image

Table 1. Experiment results.

Training Set	Test O-IMG accuracy (%)	Test S-IMG accuracy (%)
O-IMG	92.9	16.4
S-IMG	82.6	79.0
O-IMG + S-IMG	90.4	83.1
O-IMG + S-IMG + E-IMG	91.4	88.4

Table 1 below shows the experimental results. In the table, O-IMG means that the training was done as O-IMG, and O-IMG + S-IMG used O-IMG and S-IMG for training. The original image was displayed as O-IMG, the style transfer image as S-IMG, and the edge extracted image as E-IMG.

3 Conclusion

We able to solve this problem by adding training data from various textures, and it is difficult to figure out the object whose texture is deformed in object detection trained with existing general data. It also showed strong performance against noise that is inevitably present in practical use. The systems used in real life prove that you need to diversify your training data against attacks and various situations.

Acknowledgments. This work was supported by the Technology Innovation Program (or Industrial Strategic Technology Development Program) (20000946 , Development of artificial intelligent computing platform technology for service robots capable of real-time processing of large-capacity, high-performance sensor fusion processing and deep learning) funded By the Ministry of Trade, Industry & Energy(MOTIE, Korea)

References

1. Mingxing Tan., Quoc V. Le.: EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. arXiv preprint arXiv:1905.11946 (2019)
2. Robert Geirhos, Patricia Rubisch, Claudio Michaelis.: ImageNet-trained CNNs are Biased Towards. arXiv preprint arXiv:1811.12231 (2018)
3. Alexy Bochkovskiy, Chien-yao Wang, Hong-Yuan Mark Liad.: YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv:2004.10934 (2020)
4. Lin, Tsung-Yi, et al.: Microsoft coco: Common objects in context. European conference on computer vision. Springer, Cham (2014)
5. Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge.: Image style transfer using convolutional neural networks. Proceedings of the IEEE conference on computer vision and pattern recognition (2016)

Impacts of Social Network Activities related to Livestock Infectious Diseases on Pork Prices in South Korea – Cases of Foot and Mouth Disease and African Swine Fever

HyungChul Rah[1.1], Hyeon-Woong Kim[2], Do-il Yoo[3], Yongbeen Cho[4], Aziz Nasridinov[1.2], Kwan-Hee Yoo[1.2]

[1.1] Department of BigData Convergence,
[1.2] Department of Computer Sciences, Chungbuk National University, Cheongju, Korea
[2] Industry Economic Development Institute, Daegu, Korea
[3] Department of Agricultural Economics and Rural Dev., Seoul Nat. Univ., Seoul, Korea
[4] Agricultural Bigdata Division, Rural Development Administration, Jeonju, Korea
hrah@cbnu.ac.kr, economisthw@naver.com, scydl8@snu.ac.kr, cho0yb@korea.kr,aziz@chungbuk.ac.kr, khyoo@cbnu.ac.kr

Abstract. When livestock epidemics have recently occurred, the livestock epidemic is a major cause of increasing volatility in livestock prices because it can reduce not only the livestock supply, but also can reduce consumer demands. In this study, we analyzed whether price transmission between wholesale price and retail price of pork differs according to SNS activity related to livestock epidemics by testing threshold effects and how it is different by analyzing impact response function. We identified two thresholds exist between wholesale price and retail price of pork according to the SNS activity related to livestock epidemics for Foot-and-Mouth disease (FMD) and African Swine Fever (ASF) by applying livestock epidemic-related SNS activities to pork prices. We estimated threshold values of ASF-related SNS as 91 and 171 whereas FMD-related SNS as 0 and 5 with three regimes respectively. Our findings could be applied to forecast the impact of livestock epidemics on pork prices.

Keywords: Pork, Agri-Food, Infectious Disease, Social Media

1 Introduction

When the African Swine Fever (ASF) broke out in Korea in 2019, the media predicted that the pork supply would decrease and pork prices would rise above the average in the end of Sept. 2019 [1]. However, at the end of Oct. 2019 when the outbreak of ASF was at a loss, pork prices fell sharply due to a decrease in pork consumption [2]. This situation is believed to be due to the shrinking consumer sentiment to purchase pork meat as news about ASF outbreak were reported through the media and the information was delivered to consumers through various media including SNS [3]. When a livestock epidemic such as ASF has recently occurred, the

livestock epidemic is a major cause of increasing volatility in livestock prices because it can reduce not only the supply of livestock products due to the purpose of stamping out, but also can reduce consumer demands. In this study, a threshold model and impact response function were used to analyze whether the price transmission between the wholesale price and the retail price of pork differs according to the SNS activity related to livestock epidemics by testing the threshold effect and to analyze how it is different.

2 Methods

The threshold model is an analysis method that classifies a sample into two or more regimes based on a certain threshold level when there is a nonlinear relationship among parameters. If analysis results show that there is no threshold effect, it indicates that the price for each distribution stage has a linear structure. When the result indicates there is more than one threshold effect, it means there is a non-linear structure in pork prices.

In order to analyze whether the price transmission between the wholesale price and the retail price of pork differs according to the SNS activity related to livestock epidemics, the threshold effect was tested. Then, in order to analyze how it is different, it was analyzed by the impact response function. The SNS activity data related to livestock epidemics was used as a threshold variable instead of the livestock epidemic data after modifying the methods described previously [4].

In order to see the trends of search terms related to livestock epidemics in Korea, we looked at trends in livestock epidemics such as ASF and FMD from 2016 to 2019 on Naver Trend (Fig. 1). The trends indicated that search frequency of FMD disease was high once every year in 2016, 2017, 2018, and 2019. The ASF showed a high trend before the outbreak in 2019 and recorded a significantly increased trend after it occurred in September.

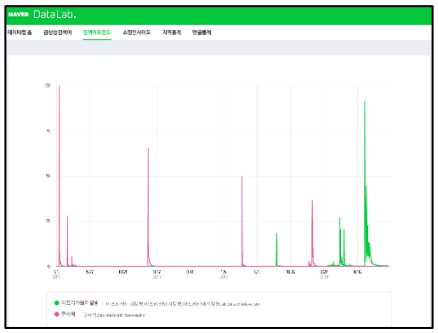


Fig. 1. Search trends of livestock epidemics including FMD and ASF.

The structured data used include daily wholesale price and retail price of pork from 2016 to 2019. The unstructured data used for the study include daily SNS data related to FMD disease between year 2016 and 2019 and ASF disease in 2019. The threshold effect test between pork wholesale and retail price according to disease-related SNS activity was performed with R's tsDyn package, and the impact response function analysis by regime according to the threshold effect was performed with STATA.

3 Results

The threshold effect of price transmission between wholesale price and retail price according to the infectious disease-related SNS activity was tested by likelihood

ratio test. In terms of ASF, the null hypothesis of linear relationship without the threshold effect between wholesale price and retail price was rejected and there were two thresholds (Table 1). The threshold values for pork prices were estimated to 91 and 171 SNS per day. In regime 1, ASF-related SNS activity was less than 91 blog posts per day (Table 2). In regime 2, ASF-related SNS activity was between 91 and 171 posts. In regime 3, ASF-related SNS activity was greater than 171 posts.

When analyzing the impact of wholesale price shock on retail price by SNS number of ASF in terms of impact response function, regimes 1 and 3 lasted 24 days, while regime 2 lasted 40 days, the longest of the 3 regimes (Fig. 2a). When analyzing the impact of retail price shock on wholesale price by SNS number of ASF in terms of impact response function, regimes 1 and 3 lasted 24 days, while regime 2 lasted 32 days, the longest of the 3 regimes (Fig. 2b).

Table 1. Results for threshold effects test.

SNS	Null hypothesis	Alternative hypothesis	Likelihood Ratio test	P-value
ASF-related SNS activity	No threshold (linear)	1 threshold	118.7185	0.000
		2 thresholds	181.4702	0.000

Table 2. Results for threshold values.

Regime	ASF-related SNS activity range
regime 1	ASF-related SNS activity ≤91
regime 2	91< ASF-related SNS activity ≤ 171
regime 3	>171 ASF-related SNS activity

In terms of FMD, the null hypothesis of linear relationship without the threshold effect between wholesale price and retail price was rejected and there were two thresholds (data to be shown later). The threshold values for pork prices were estimated to 0 and 5 SNS per day. In regime 1, FMD-related SNS activity was ≤ 0 blog posts per day. In regime 2, FMD-related SNS activity was between 0 and 5 posts. In regime 3, FMD-related SNS activity was > 5 posts.

When analyzing the impact of wholesale price shock on retail price by SNS number of FMD in terms of impact response function, regimes 2 and 3 lasted 18 days, while regime 1 lasted 24 days, the longest of the 3 regimes (data to be shown later). When analyzing the impact of retail price shock on wholesale price by SNS number of FMD in terms of impact response function, regimes 2 and 3 lasted 18 days, while regime 1 lasted 24 days, the longest of the 3 regimes.

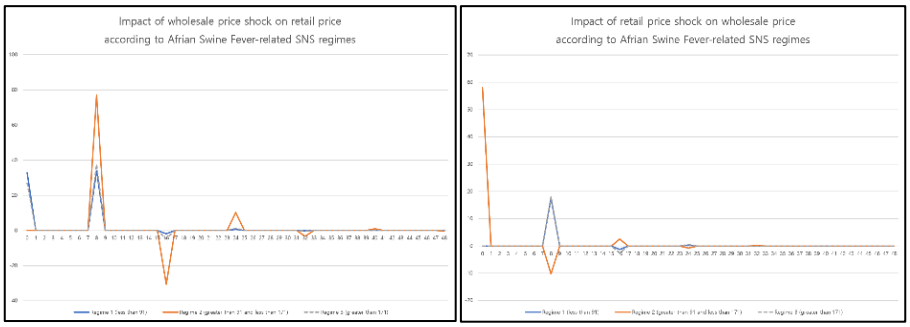


Fig. 2a. Impact of wholesale price shock on retail price. **Fig. 2b.** Impact of retail price shock on wholesale price.

4 Discussion

In our study, we aimed to analyze whether the price transmission between the wholesale price and the retail price of livestock products differs according to the SNS activity related to livestock epidemics by testing threshold effect when there is outbreak of livestock epidemics in Korea. We also wanted to analyze how it is different.

By applying livestock epidemic-related SNS activities to pork prices instead of the livestock epidemic data, we identified two thresholds exist between wholesale price and retail price of pork according to the SNS activity related to livestock epidemics for FMD and ASF indicating non-linear structure in pork prices. Our data also suggest the duration that impact lasts is different depending on the SNS activity related to livestock epidemics. Our findings could be applied to forecast the impact of livestock epidemics on pork prices and the duration of impacts, for which further studies are required.

Acknowledgments. This work was carried out with the support of "Cooperative Research Program for Agriculture Science and Technology Development (Project No. PJ015341012020)" Rural Development Administration, Republic of Korea.

References

1. Lee J-C. Domestic pork prices are expected to rise in October. "Price volatility likely to be large due to swine fever." E-Daily. 2019-09-29.
2. Seok M-S. Consumers turned away from pork due to African swine fever... Half of consumers “eat less”. KBS NEWS. 2019-10-30.
3. Kim T-P. Swine fever... Pork continues to decline in price and consumption. LAMB. 2019-11-04.
4. Kim H-W. An Analysis of the Dynamic Characteristics in Price Transmission from Farm to Retail Stages According to Infectious Diseases: The Case of Broiler. Chungbuk National University. 2018.

Application of to prediction Time Series Data based on Event-Based LSTM

Ga-Ae Ryu[1], HyungChul Rah[2], Aziz Nasridinov[1], Kwan-Hee Yoo[1*]

[1]Dept. of Computer Science, Chungbuk National University, South Korea
[2]Dept. of BigData Convergence, Chungbuk National University, South Korea
{garyu, hrah, aziz, khyoo}@chungbuk.ac.kr
[*]Corresponding Author

Abstract. Time series data are data in which observations are recorded at a certain interval with time, such as stocks. Various methods such as ARIMA and Boosting Model are used to predict these time-series data, and LSTM models using deep learning are also used a lot. However, even time series data can be difficult to predict due to events such as social issues and accidents, such as changes in observed values or patterns of data. Therefore, in this paper, using the livestock (pig) consumption amount data, we make a predicted model that is considered the event. And then, we compare to predictive models using the existing LSTM method and event-based LSTM method.

Keywords: Agricultural, Time series data, Deep Learning, LSTM, Prediction.

1 Introduction

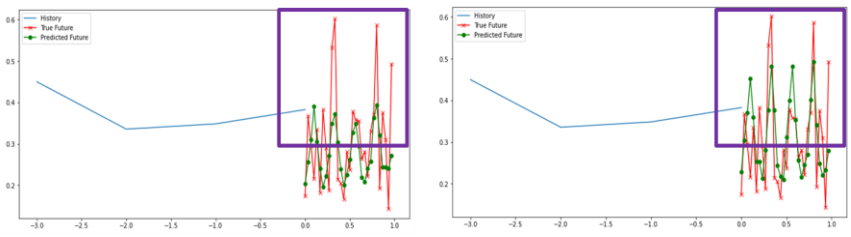
Due to the development of computer hardware, research on how to make various predictive models using machine learning and deep learning has been increasing in recent years. Among them, many methods of forecasting time-series data such as stock and dam water levels are being studied.[1-2] However, for these time series data, it is difficult to predict because the data pattern changes due to social issues, and different values are predicted from the existing forecast model. An example is the prediction of consumption in which prices change with these social issues, events, and accidents. For example, in the case of agricultural/livestock consumption forecasts, natural disasters, and diseases of plants and animals often increase or decrease the range of price changes for food. There is also an example of a recent spike in the price of pigs due to the African swine fever. Therefore, in this paper, we make a prediction model that takes into account the occurrence of these social issues. The data to be used to create the predictive model are structured and unstructured data on the amount of pork consumption purchases collected from 2010 to 2017, which is used in research by Ryu etc. [3]

2 Proposed Method

In this paper, the event-based LSTM model is proposed to make the forecast model take into account social issues, etc. The LSTM model [4] is a method to solve the vanishing gradient problem in learning through backpropagation by using the input gate, forget gate, output gate, and state storage. However, if the existing LSTM model [4] is used to learn data from 2010 to 2017, it can be seen that the predicted values converge into average values and do not reflect social issues for unstructured data. To solve this problem, a window-based learning method was applied to predict the contents of a month after the study period is limited to a specific period, and the event gate is added to the LSTM cell to reflect events on the data. When the time series data related to the event is received, the event gate enters the data and calculates it with the previously hidden value, and does not calculate with the value of the oblivion gate, so that the time series data related to the event remain in the state store without being forgotten. At this time, the event gate has weights and deviations for the event gate as well as any other gate. The time-series data identification related to the event is then given a thread on the target data to determine whether the time series data is an event or not.

3 Discussion and Future work

In this paper, the method of predicting the amount of pork consumption purchase using event-based LSTM was proposed. As shown in Figure 1, it can be seen that when the amount of consumption purchased increases, it is more predictable than when using the traditional LSTM method. It can also be seen that the event-based LSTM method is lower.



(a) (b)

Fig. 1. Predicted Amount of Purchase of Pig Consumption
(a) LSTM model (b) Event-Based LSTM model

However, it is still necessary to increase the accuracy of the forecast for consumption, and the unformatted data characteristic values for the estimate of the amount purchased should also be predictable.

Afterward, we will modify and supplement the event-based LSTM model to solve this and create a multi-step pre-diction method to predict the unstructured data feature value as well.

Acknowledgments. This work was carried out with the support of "Cooperative Research Program for Agriculture Science and Technology Development (Project No. PJ015341012020)" Rural Development Administration, Republic of Korea.

References

1. Joo, I.T., Choi, S.H.: Stock Prediction Model based on Bidirectional LSTM Recurrent Neural Network. In: Journal of Korea Institute of Information, Electronics, and Communication Technology, vol. 11, no. 2, pp. 204--208 (2018)
2. Tran, Q.K., Song, S.K.: Water Level Forecasting based on Deep Learning : A Use Case of Trinity River-Texas-The United States. In: Journal of KIISE, vol. 44, no. 6, pp. 607--612 (2017)
3. Ryu, G.-A.; Nasridinov, A.; Rah, H.; Yoo, K.-H. :Forecasts of the Amount Purchase Pork Meat by Using Structured and Unstructured Big Data. Agriculture, vol. 10, no. 21, pp.1-14 , (2020)
4. Greff, K., Srivastava, R.K., Koutnik. J., Steunebrink, B.R.: LSTM: A search Space Odyssey. In: IEEE Transactions on Neural Networks and Learning Systems, vol. 28, no. 10, pp. 2222--2232. (2017)

Anomaly detection of time series data in the cloud with Azure cognitive services

Manas Bazarbaev, Aziz Nasridinov and Kwan-Hee Yoo*

Chungbuk National University, Cheongju-si 28644, South Korea
* Corresponding authors
{manas, aziz, khyoo}@cbnu.ac.kr

Abstract. Anomaly detection has an important role when real time big data analytics is performed. However, here anomaly detection refers specifically to the detection of unexpected events, be it cardiac episodes, mechanical failures, hacker attacks, or fraudulent transactions. Recent advances in technology allow us to collect a large amount of data over time in diverse research areas. Time series data mining aims to extract all meaningful knowledge from the data, and several mining tasks (e.g., classification, clustering, forecasting, and outlier detection) have been considered in the literature. Our goal is to use cloud computing service to find the anomalies in the time series from sensor data. We used the Microsoft Azure cognitive services to achieve our goal. As result we have REST API real-time service for detect anomalies in sensors data.

Keywords: Univariate anomaly detection, sensor data, time-series, cloud services.

1 Introduction

Outlier detection has become a field of interest for many researchers and practitioners and is now one of the main tasks of time series data mining. Outlier detection has been studied in a variety of application domains such as credit card fraud detection, intrusion detection in cybersecurity, or fault diagnosis in industry. In particular, the analysis of outliers in time series data examines anomalous behaviors across time.

The unexpected character of the event means that no such examples are available in the data set. Classification solutions generally require a set of examples for all involved classes. So, how do we proceed in a case where no examples are available? It requires a little change in perspective.

There are a lot of services which provides anomaly detector in the cloud. For example, one is anomaly detector in the Azure cognitive services [5] and the other is from the Amazon Sagemaker AI services [6].

The paper is structured as: Section 2 describes related studies which about anomaly detection methods in the cloud, Section 3 describes our dataset which is from steel making company data and used method which we mentioned earlier, Section 4 describes experimental results and explain experimental environment of our implementation. Finally, Section 5 contributes discussion on our result.

2 Related Work

In this section, we research several studies focusing on how outlier detection methods are used in cloud, particularly in Azure Cognitive Services. To use a cognitive service, we need to navigate to the Azure portal and then to cognitive services. We will need to create an instance and you will receive an API key and endpoint for billing. We can use it for free until you use up your free quote. After that, we can move on to payments [1]. Ren *et. al*. made some online time-series anomaly detection at Microsoft based on their Azure Services [2].

3 Dataset and method

3.1 Dataset

Regarding the dataset, the data was collected from steel companies. The data is 593962 records from 4th February 2020 to 10th February 2020. The data were received from programmable logic control (PLC). The PLC transmits data from sensors or machines in the plant to a database at the end of the day, possibly midnight. All records are sorted by time stamp every second.
From the data we used *elepower* column. This feature is one of the most important factors in metal melting. Elepower is the electric energy consumed to melting metal at time t, Table 1.

Table 1. Description of data.

Column name	Count	Mean	Standard deviation	min	max
Elepower	10080	23932.75	7412.77	7438.44	32887.06

3.2 Data Preparation

Since this service requires no more than 8640 records and the minimum can process data per minute, we used an average for each minute. After this converting, we have only 10080 data Figure 1. We analyzed 8640 records from this data.

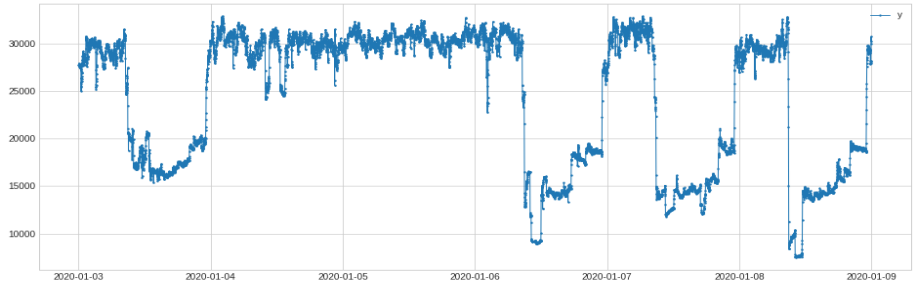


Figure. 1. Prepared data visualization

3.3 Method

Generally, there are a lot of the definitions on anomaly detection for time series data. In this paper, we are using the anomaly detection problem proposed by Ren *et.al* [2]. They adopted a simple yet powerful **Spectral Residual** (SR) [3] based on Fast Fourier Transform (FFT) [4]. They define the problem of detecting time series anomalies as follows.
Problem: *For a given sequence of real values, i.e.,* $x = x_1, x_2, ..., x_n$*, the task of detecting time series anomalies is to generate an output sequence,* $y = y_1, y_2, ..., y_n$*, where* $y_i = \{0, 1\}$ *, here 1 means that* x_i *is an anomaly point, 0 others.*
We have tried to use that cloud service which provides that anomaly detection method to solve our problem.

4 Experiment results

4.1 Experimental environment

We have used computer with specification such as CPU Intel® Core™ i7-4790 3.60GHz, RAM 20 GB, Graphics card NVIDIA GeForce 745, Operating System Manjaro Linux 20.0.1 with Linux kernel version 5.4 and Integrated Development Environment Jupyter notebook for Python.

4.2 Results

In this section, we show results of Azure Cognitive Services outlier detection method. They have made an API service where we must send our data and parameters with requests and receive the result as response.
For our data we used the following request:
Request(series=series, granularity=Granularity, minutely, sensitivity=90, period=18)
Figure 2 shows our original data as blue line and predicted data as orange line.

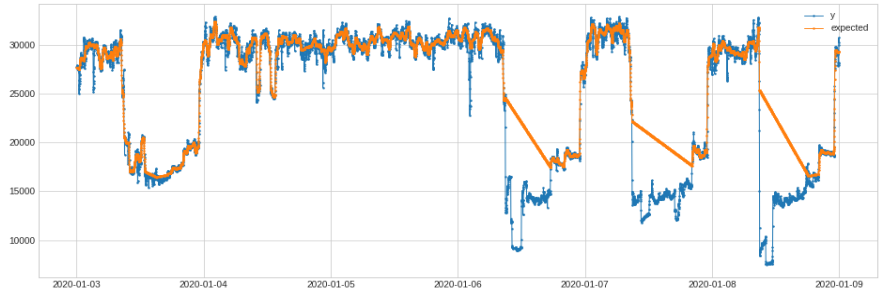


Figure. 2. Original and expected data visualization

Based on predicted data the algorithm detects the outliers. The outlier data are shown as red points on the Figure 3.

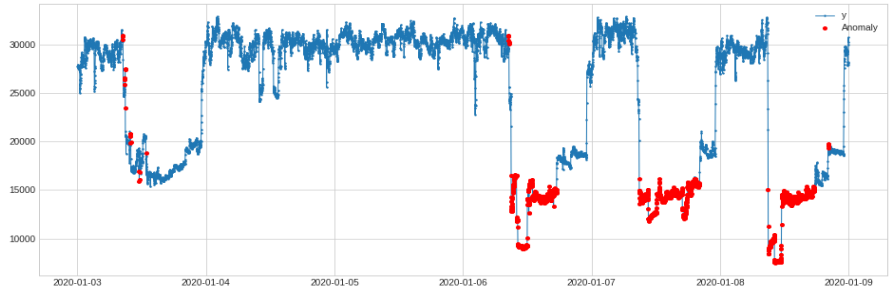


Figure. 3. Detected outlier data visualization

5 Discussions

Time-series anomaly detection is a critical module to ensure the quality of online services. An efficient, general and accurate anomaly detection system is indispensable in real applications. In this paper, we have introduced a time-series anomaly detection with Azure Cognitive Services. The cloud services like Azure Cognitive Services or Amazon Sagemaker gives us an opportunity to create machine learning algorithms or use already created algorithms to make predictions. All we need to do only pass data, selected method and select mode.

Acknowledgments: This material is based upon work supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program, No.10082578, 'Development of intelligent operation system based on big data for production process efficiency and quality optimization in non-ferrous metal industry and this work was supported by the Technology Innovation Program (2004367, Development of cloud big data platform for the innovative manufacturing in ceramic industry) funded By the Ministry of Trade, Industry & Energy(MOTIE, Korea)

References

1. Anomaly detection from the edge to the AWS and Azure cloud, https://www.researchgate.net/publication/34008a5547 (2020)
2. Hansheng Ren, Bixiong Xu, Yujing Wang, Chao Yi, Congrui Huang, Xiaoyu Kou, Tony Xing, Mao Yang, Jie Tong, Qi Zhang.: Time-Series Anomaly Detection Service at Microsoft, Microsoft Beijing China. KDD '19: The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 3009–3017, (2019).
3. Xiaodi Hou and Liqing Zhang.: Saliency detection: A spectral residual approach. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on. IEEE, pp. 1–8, (2007).
4. Charles Van Loan.: Computational frameworks for the fast Fourier transform. Vol. 10. Siam. (1992).
5. Azure cognitive services, https://azure.microsoft.com/en-us/services/cognitive-services/
6. Amazon SageMaker, https://aws.amazon.com/sagemaker/

Benchmarking on Distributed File Systems for Scientific Big Data Processing

JunYeong Lee[1], Moonhyun Kim[1], Seo-Young Noh[1*]

Department of Computer Science, Chungbuk National University
1, Chungdae-ro, Seowon-gu, Cheongju-si, Chungcheongbuk-do,
28644, Republic of Korea
lee1238234@chungbuk.ac.kr, moonhyun.kim@cern.ch, rsyoung@cbnu.ac.kr

Abstract. Data is becoming important in numerous research fields. As the era of such data pandemic research environment, it is important to handle big data efficiently and effectively. Due to the nature of velocity and volume of big data, handling big data results in critical challenges such as storing and processing tremendous data. RAID is one of successful technology to safely handle large amount of data with less cost. However, it can be only vertically scalable and requires RAID-featured hardware. In this paper, we benchmarked three different distributed file systems such as EOS, Ceph and GlusterFS which can horizontally scalable and does not require specific hardware. In our evaluation, we mainly focus on their performance targeting to scientific big data and discuss their usability in scientific data processing environments.

Keywords: Big Data, Distributed File System, EOS, Ceph, GlusterFS

1 Introduction

Data is getting important nowadays and playing vital roles in all around areas. There is a report that the revenue of global big data market was 7.6 billion dollars in 2011, and is forecasted to 103 billion dollars by 2027[1]. It is natural that we can expect it requires more data storage capacity for data. In 2018, it was reported that 124 exabytes were produced and it will grow up to 403 exabytes in 2021[2]. Data does not only influence to commercial fields, but also to the fields of scientific communities.

As data size increases, storing the data is getting more important. In the past, we have RAID(Redundant Array of Independent Disks) technology to store data safely and securely. But RAID needs specific hardware to run, and it can only expand its capacity vertically, resulting in requiring additional expensive RAID-support hardware. Furthermore, if there are some disk failures, RAID must rebuild entire data structure. If failed, data can be loss permanently. To prevent such disastrous events, distributed file system has been developed and widely used in many data-intensive environments. In this paper, we benchmark three distributed file systems such as Ceph, EOS and GlusterFS for data intensive science. In our benchmark, we evaluate read/write performance and discuss their suitability in scientific data processing.

The rest of this paper is organized as follow: In section 2, we will review three distributed file systems. In section 3 and 4, we will discuss evaluation environment setup and results, respectively. Finally we conclude this paper in section 5.

* Correspondence Author

2 Review on Distributed File Systems

Ceph is software-defined storage platform that provides object, block and file based storage. Such a file system is based on RADOS(Reliable Automatic Distributed Object Store)[3]. It consists of Monitor (MON) for managing storage cluster and Manager(MGR) for keeping track status of Ceph cluster. Object storage(OSD) for storing data and handles replication, recovery and rebalancing. Ceph uses CRUSH algorithms to compute information about object location.

EOS is an open-source storage solution developed by CERN(Conseil Européene pour la Recherche Nucléaire). Its main target is to provide cost-effective and large disk-only storage space for LHC(Large Hadron Collider)[4] experiments. EOS is composed of three components: MGM, FST and MQ. MGM is a management server which manages namespace, authentication, file placement and location. FST is a file storage server which stores file and calculates checksum for consistency. MQ is message broker for communication between MGM and FST. EOS supports not only XRootD protocol, but also HTTPS and FUSE.

GlusterFS is open-source distributed file system developed by RedHat[5]. It can be scaled out to several petabytes using commodity hardware. It can handle thousands of clients. Unlike the other distributed file systems, GlusterFS has no central metadata or namespace server. if client requests a file, Gluster's client uses hashing algorithms to deterministically find the correct node where file is stored[6].

3 Evaluation Environment

In order to benchmark, we have setup the evaluation environment using hardware and software. For hardware, there are four servers used: 1 for master server and 3 for slave. Table 1 shows the specification of servers in our test environment.

Table 1. Distributed File System Server Specifications.

	Master Server	Slave Server
Chassis	HP ProLiant ML350 G9	HP ProLiant ML350 G6
CPU	E5-2609v3 1.90Ghz	E5606 2.13Ghz
RAM	40GB	4GB
Disk	120GB SSD (Boot)	500GB HDD (Boot)
	2 * 1TB HDD (Data)	2 * 1TB HDD (Data)
Network	3Gbps (1Gbps * 3)	1Gbps

It should be noted that booting disk and data disk are split in order to prevent interference between OS and distributed file system. It is also noted that master server uses bond interface consisting three 1Gbit NICs to communicate all slave servers without bottleneck.

Regarding software, we installed three targeting distributed file systems on all servers. All file systems are configured to use six data disks and two parity disks as shown in Figure 1. File systems are mounted to the master server using native FUSE client for benchmarking.

For benchmark, we used FIO to measure the performance of each file system. Table 2 describes benchmark options used in our evaluation. Each test increases thread size and job size to evaluate multi-threaded performance of the file system.

Figure 1. File System Configuration.

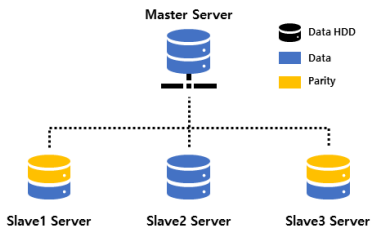


Table 2. FIO benchmark options.

Block Size	1024K
Job Size	1GB
Number of Jobs	1, 2, 4, 8
Number of Thread	1, 2, 4, 8
IO Depth	32
Tests	Seq/Random Read, Write

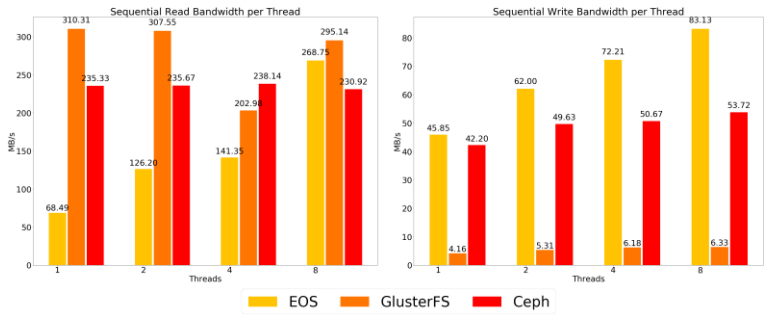
In order to prevent from affecting successor evaluation by unflushed data of precedent test, all servers and disks were reset before evaluating another file system.

4 Performance Evaluation

4.1 Sequential Read/Write

Figure 2 shows sequential read/write bandwidth results for each file systems. EOS's read and write bandwidth is increasing depending on the number of threads. And it shows high write bandwidth for all threads. GlusterFS shows relatively high bandwidth on read, but writing bandwidth is very poor compared to the other file systems. Ceph shows consistent for reading regardless of threads while writing performance is growing slightly. However, the performance gap between EOS and GlusterFS is noticeable in writing.

Figure 2. Sequential Read/Write benchmark results.

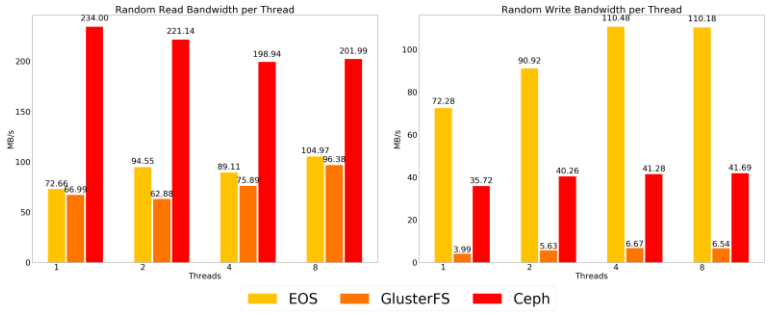


4.2 Random Read/Write

Figure 3 shows random read/write bandwidth results for each file systems. Unlike sequential read result, Ceph shows high read bandwidth compared to the other file systems. However, as thread increases, Ceph's read bandwidth drops slightly. The read performance of EOS and GlusterFS is significantly improved when the number of threads increase. In random write bandwidth test, EOS shows two times higher than Ceph in all tests. Like sequential write test, GlusterFS shows the worst

performance. It performed lower than 10MB/s. Ceph gains bandwidth as thread increases, but bandwidth is saturated when more than 2 threads are used.

Figure 3. Random Read/Write benchmark results



5 Conclusion

Our benchmark showed that the file systems have different characteristics. EOS has great write performance, but lacks read performance. GlusterFS showed the best performance for sequential read, but both writing benchmark showed relatively worse performance compared to the other file systems. Ceph shows reasonable performance only except random read benchmark, but it showed downgraded performance for both reading benchmarks as thread increases. As shown in our experiment, the three file systems have their strength and weakness in specific evaluation configurations. Therefore, it is important for researchers to choose a proper file system for their projects.

Acknowledgments. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2008-00458). and Korea Institute of Science and Technology Information.

References

1. Big data market size revenue forecast worldwide from 2011 to 2027, https://www.statista.com/statistics/254266/
2. Volume of big data in data center storage worldwide from 2015 to 2021, https://www.statista.com/statistics/638621/
3. Architecture – Ceph Documentation, https://ceph.readthedocs.io/en/latest/architecture/
4. Introduction – EOS CITRINE Documentation, http://eos-docs.web.cern.ch/eos-docs/intro.html
5. Gluster Docs, https://docs.gluster.org/en/latest/
6. Davies, A., & Orsaria, A.: Scale out with GlusterFS. Linux Journal, vol. 2013(235), pp. 72--74, Belltown Media, Houston (2013)

IIoT Cloud Platforms Survey – Focused on MindSphere and Predix

Won-jun Lee[1], Sun-jong Bong[2], Su-young Chi*

[1] Yanbian University of Science & Technology, 3458 Chaoyang St, Yanji, Yanbian Korean Autonomous Prefecture, Jilin, China
[2] University of Science and Technology, 217, Gajeong-ro, Yuseong-gu, Daejeon, Republic of Korea
* Electronics and Telecommunications Research Institute, Gajeong-ro, Yuseong-gu, Daejeon, Republic of Korea
wjdnjswns@gmail.com, sunjong@ust.ac.kr, chisy@etri.re.kr

Abstract. After the announcement of Industry4.0, the number of companies providing cloud-based IIoT platforms and solutions has increased due to the smart factory construction policies of each country. Under these circumstances, users are having difficulty choosing the right platform for the situation. The purpose of this study is to describe the underlying Industry4.0, the concept of smart factory, and edge computing, the next generation of distributed processing technology in cloud computing, and to introduce and analyze Siemens' MindSphere, and General Electric's Predix, to help users choose in the future.

Keywords: Industry 4.0, Smart Factory, Edge Computing, IIoT Platform

1 Introduction

Industry 4.0 was first announced by Klaus Schwab at the Davos Conference in Switzerland in 2016. It refers to a national strategy to maximize productivity and efficiency by connecting advanced ICT technologies such as AI, IoT, and cloud computing to the economy and society in general.

Since the announcement of Industry 4.0, not only Germany but also the U.S., Japan, China, and other countries have been competing to dominate Industry 4.0. In the case of South Korea, the government has announced a plan to build 30,000 smart factories by 2025 under the "Smart Manufacturing Innovation Vision 2025" and is making efforts. Also, as the strategy for manufacturing innovation is being promoted as a

national issue, efforts and research are being made to anticipate the IIoT platform based on cloud.

Therefore, this paper first introduces the concept of smart factory and the current state and outlook of the market, and then introduces edge computing, the related technology that is the next generation of core cloud computing used. Secondly, Introduce and analyze Siemens' MindSphere and General Electric's Predix, representative of cloud-based IIoT solutions, the aim is to help users make the right choice to use the solution.

2 Related Work

2.1 Definition of Smart Factory

First, a smart factory is a concept that has improved existing factory automation. The concept of factory automation is similar in that it automates production facilities and automates management. [1] However, unlike conventional process automation, artificial intelligence (AI) is given to each process, and there is a difference in connecting, collecting, and analyzing data autonomously.

In other words, smart factory means an advanced factory with a production system in which all phases of the process are digitized, automated, and all processes are real-time linked to each other by combining sensors attached to machines, the Internet of Things (IoT), and Cyber-Physical System (CPS) by applying ICT to all processes, including product planning, development, sales, and production. [2]

According to the results of the Markets & Markets survey, the global smart factory market is expected to grow by an average of 9.76 percent annually from 2019 to 2024 and will grow by 9.3 percent annually until 2022, creating a market size of $205.42 billion. Especially in the case of Korea, the market size is expected to be $7.83 billion by 2020 and $12.76 billion by 2022, showing a high annual growth rate of 12.2%.[3]

2.2 Edge Computing

Gartner has selected "The empowered edge" and "The distributed cloud" as the top 10 strategic technology trends to look out for in 2020. [4] Edge computing is becoming increasingly important every day.

Edge computing is not an alternative to existing cloud computing, but a win-win relationship that complements and enhances cloud computing challenges. [5] The reason for the advent of edge computing is that traditional cloud computing mechanisms alone cannot handle the vast amount of data coming out of multiple manufacturing processes. There is a problem that arises if insist on the existing cloud computing structure.

The first problem is the data delay. Data transmission/reception time increases due to the bottle-neck phenomenon as the data center is physically far away or the amount of data to be processed increases.

Secondly, due to security concerns, Cloud Computing is vulnerable to external attacks such as hacking and D-Dos, and there is a risk of information leakage data loss.

When Edge Computing is combined into an existing Cloud architecture, it becomes a distributed computing structure and only processes data generated by that device. This can reduce the load relatively. Security is also better than Cloud Computing because it is self-handling, and it is much faster to handle failures on its own rather than in large quantities.

3 IIoT Solutions

Introduce and analyze Siemens' MindSphere, which was selected as the industrial IoT software platform leader in Forrester Wave: Evaluation of Industrial IoT Platform in Q4 2019, and GE's Predix.

3.1 MindSphere

Siemens, founded in 1847, is a world-class electric and electronics company based in Germany. Power generation, power transmission, smart grid solutions, and efficient applications of power energy are doing business in medical imaging and clinical diagnosis along with the entire power generation value chain.[6]

MindSphere[7] is a cloud-based open IoT system developed by Simens that includes tools, applications, and services for developers based on cloud servers. It consists of MindApps that provide applications, MindSphere, the underlying Cloud service OS, and MindConnect that can be associated with third-party products. Based on this, any manufacturing environment, such as machinery and plants, can easily and quickly connect to other companies' assets as well as their own, and store data safely. Also, stored data can be optimized through analysis algorithms to increase productivity.

Gämmerler[8] is a manufacturer of short-term compressors, introduces Siemens solutions to apply mind-speaker systems to production facilities and visualizes collected data in the form of a dashboard that provides information on data management services and resource optimization through monitoring and data analysis, and configures it to provide an immediate alarm in the event of operational errors. By detecting operational errors in advance, unexpected downtime was reduced by 10%, which increased the operating time of production facilities, and increased service sales by 10%.

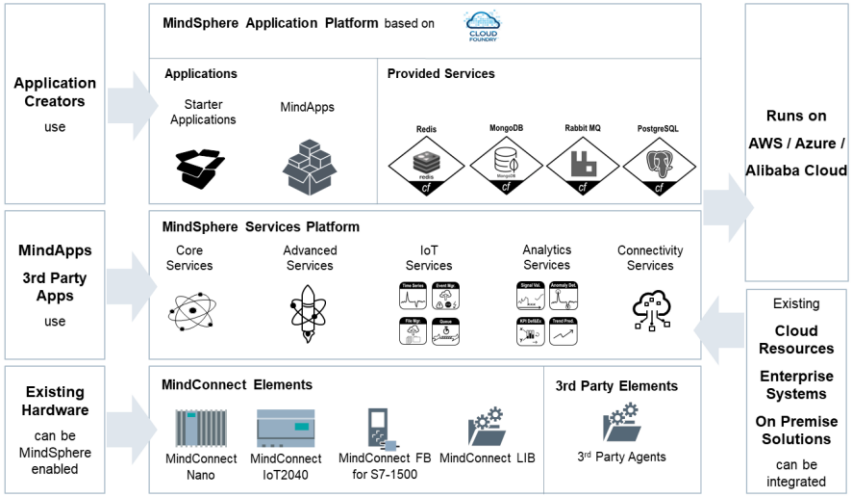


Fig. 1. The structure of MindSphere[7]

3.2 Predix

General Electric (GE) is a traditional manufacturing company that manufactures essential equipment for a variety of industry groups, including aircraft engines, power generation turbines, and medical equipment.[10]

Predix[11] is an industrial platform developed by General Electric for digital innovation. The purpose of Predix development is to make it easier for companies to develop, distribute and operate industrial applications to reduce the number of outages caused by unexpected mechanical failures or to improve asset output and maximize operational efficiency. It consists of Predix Cloud, which integrates and collects various types of OT data from industrial sites into IT data and transfers it to Predix Cloud, which stores the industrial data based on Cloud Foundry and provides services necessary to manage and analyze the data in the form of API (Application Programming Interface), and Predix Applications, which provides HTML5 based data analysis results for field facility operators, factory operators, and business users to use for their work, as well as for mobile devices.

GE Aviation has developed analytical tools for aircraft engine conditions, and flight management efficiency using Predix. Measured 16 times per second with 26 sensors attached to the engine, 300 metrics were calculated to analyze flight performance, engine conditions, and efficiency to establish a more effective flight path, and the benefits of using Presidents exceeded $175 million. Also, fault detection accuracy has improved by 10%, and the number of unnecessary cancellations of commercial aircraft has decreased by more than 1,000. [12]

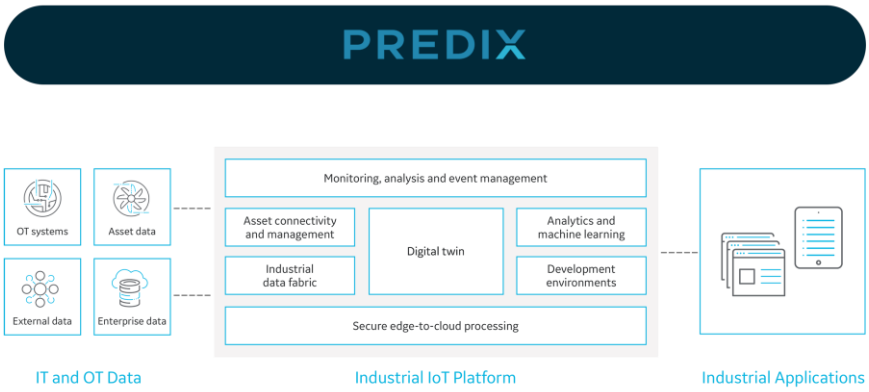


Fig. 2. The Structure of Predix[11]

4 Component for Smart Factory

To facilitate the understanding of the functional aspects of the two representative IIoT platforms described above, a summary of the basic information for each platform was prepared in Table 1. If you look at it from Table 1, you can see that MindSphere and Predix have many things in common. Here we can understand the core technologies that make up the smart factory.

First, it is constituted of an edge computing structure. Edge computing provides an accurate and efficient data collection system in real-time to enhance user convenience with rapid data processing and flexible incident response.

Second, it uses IaaS platforms such as Amazon AWS and Microsoft Azure to help store and compute large-scale industrial data.

Third, it consists of a hybrid cloud platform. Service provision is implemented on the cloud in the form of data storage sensitive to security, local services are managed by own infrastructure, flexibility, economy, speed, security, and stability of physical servers, service drive is on the cloud, and data storage and local services are secured by own infrastructure.

Fourth, flexible connection and development are possible using various SDKs and protocols. Both companies utilize MQTT, OPC Unified Architecture (OPC), and Modbus to enable flexible integration of industrial automation applications as well as consolidation with their assets. Each platform also offers a variety of SDKs for easy and quick connection of hardware devices or mobile applications.

Fifth, provide a variety of service solutions tailored to each industry to provide optimized industrial applications for specific industries and specific scenarios. Currently, there are more than 160 industrial apps each in the Predix and MindSphere app stores.

Table. 1. Comparison of MindSphere and Predix Platforms Information [7], [10]

	MindSphere	**Predix**
Deployment	PasS, IasS, SasS	PasS, IasS, SasS
Cloud types	Public, Private, Hybrid	Public, Private, hybrid
Computing Structure	Edge Computing	Edge Computing
Data Collection	MindConnect	Predix Machine
Cloud Platform	MindSphere	Predix Cloud
Applications	MindApps	Predix Apps
SDK	Java, Node.is, Python	Python, Java
Protocols	OPC UA, Modbus TCP, MQTT, Rockwell	OPC UA, Modbus TCP, MQTT
Component	Open Standard, Plug & Play, Cloud Infrastructure, Open Interfaces, Transparent Price Model	Digital Twin, Analytics & Machine Learning, Application Catalog, Developer Productivity
Benefits	Increase service efficiency, Enable new business models, Optimize assets, Rapidly develop apps, Scalable Development environment	Innovate Quickly, Optimize asset performance, Invest Wisely, Plan for the future, Immediate cost benefits

5 Specialized field

Siemens' MindSphere and GE's Predix have similar industries, including energy, aviation, and engine production, but each has its specialized areas.

First of all, MindSphere is specialized in manufacturing, smart factory, with a focus on integrating and digitizing its strengths, factory automation, and software. In the manufacturing industry of Coca-Cola, a beverage manufacturer, and Gering Technologies, an abrasive processing machinery company, MindSphere was used to reduce maintenance costs and reduce downtime time to improve the uptime of facilities and increase production. [7]

Predix focuses on the production of self-produced devices such as aircraft engines, generators, and medical devices, and the efficient production and management of industrial goods produced by GE such as engines, generators, medical containers, and locomotives. Predix shows strength in solutions such as GE Aviation[12], GE Digital

Power Plant [15], and GE Mine Performance. GE Aviation has refined commercial aircraft through Predix and established effective flight paths by analyzing engine performance and efficiency, and combined Predix with its power generation plants to improve production by making the probability of failure predictable. GE also introduced GE Mine Performance in the mining industry, increasing availability from 70-85% and reducing production losses due to reduced failures. [16]

Table. 2. Comparison of Representative Solutions Offered [7][11]

MindSphere	Predix	Same Type	Different Type
Wind	Wind Power	✓	
Mobility	Transportation	✓	
Healthcare	HealthCare	✓	
Power and Gas, Power Generation Services	Oil and Gas	✓	
Digital Factory	Intelligent Environment		✓
Energy Management	Power Generation	✓	
Building Technologies	Water		✓
Smart Cities	Mining		✓
Process Industries and Drive	Automotive	✓	
	Aviation		✓
	Power Distribution		✓

6 Conclusion

Since the announcement of Industry 4.0, the number of companies that provide cloud-based IIoT solutions is increasing exponentially. Under these circumstances, it is important to choose the right platform for each company's needs.

We compare the characteristics of Mindsphere and Predix, which are representative of cloud-based IIoT platforms, are derived from edge computing structures, core technologies, and solution fields, and based on this. it is possible to identify and compare cloud types, structures, functions, and benefits in selecting IIoT platforms suitable for the industrial structure that will be applied in the future.

References

1. Korea Institute for Industrial Economics & Trade : Issues and Assignment of Smart Factory for Future Manufacturing Innovation. 620, 1--12 (2015)
2. radziwon, A., Bilberg, A., Bogers, M., & Madsen, E. S.: The smart factory: exploring adaptive and flexible manufacturing solutions. Procedia engineering. 69, 1184--1190(2014)
3. Markets and Markets, https://www.marketsandmarkets.com
4. Gartner, https://www.gartner.com
5. Telecommunications Technology Associations, https://www.tta.or.kr

6. https://new.siemens.com/kr/ko/company/about-us.html
7. MindSphere, https://siemens.mindsphere.io
8. Gämmerler, https://www.gammerler.de
9. MindSphere Architecture, https://iot5.net/
10. GE, https://www.ge.com/digital/iiot-platform
11. Predix, https://www.ge.com
12. GE Aviation, https://www.geaviation.com
13. GE Predix Architecture, https://www.ge.com
14. GE Aviation, https://www.geaviation.com/
15. GE Digital Power Plant, https://www.ge.com/power/software
16. GE Mine. https://www.ge.com/in/water/mining

Toward Performance Improvement of RocksDB by Tuning Parameters

Jiwon Kim[1], Hyeonmyeong Lee[1], Sungmin Jung[1], Heeseung Jo[1],

Chungbuk National University, Chungdae-ro 1, Seowon-Gu, Cheongju, Chungbuk 28644, Korea
{jw9502, myeong58, jjsm9595, heesn}@cbnu.ac.kr

Abstract. Performance optimization through application parameter adjustment has the advantage of obtaining performance improvement without any other tools. However, it is difficult for the user to predict an appropriate value for their workloads. This paper proposes a performance optimization scheme through the internal parameter tuning of RocksDB and shows the evaluation result for each workload. We offer a method to find the optimal parameter combination, and the method is scalable at a low cost.

Keywords: RocksDB, File systems, Optimization, Parameter tuning

1 Introduction

As applications are becoming more complex and diverse, the range of performance improvements available to users has expanded. Tuning the application through internal parameters does not require any other application and can improve service performance without any modification of the application. However, it is difficult for the user to predict an appropriate value considering the varied workload and the various parameters of the applications. Because the parameters are affected by executing workload, and the correlations between the parameters are complicated. In this paper, we offered methods for performance optimization in RocksDB, a key-value store optimized for high-speed storage [1], through internal parameters.

2 Design

This chapter describes the parameter tuning method designed considering the number of physical cores (c) of the machine used. For optimization through tuning of RocksDB, the following three methods are proposed.

V1. This method executes changing the value of a single parameter to 1 to 2*c in sequence and checks the parameter with the highest performance and the corresponding value. In several other papers tuning application in a similar way. [3, 4, 5]

V2. V1 can be said to be a method that does not consider the correlation between internal parameters for application execution. We predict that tuning two or more parameters would be better in improving performance, so we developed a method of tuning all parameters. For efficiency, we first execute each parameter to 1 to 2*c in sequence. Then combine only the highest and lowest performance values that have the biggest impact on performance and default value to reduce the number of executions.

V3. Finally, v2 obtained better efficiency than the brute force algorithm in parameter combination, but there is still a need to reduce search cost. Through the experimental results of V1 and V2, we found that if the effect of a single parameter on the performance is insignificant, the combination with other parameters does not significantly affect the performance. So we filter parameters according to the performance effect on the application. Unlike the above two methods, instead of sequentially executing 1 to 2*c, the filtering process is minimized by executing only 1, c, c*2 values.

3 Experiment

In this Section, based on the previously suggested, we conduct parameter tuning experiments in RocksDB, and check the highest performance improvement rate per workload.

In this paper, for performance evaluation, we use Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz (4 physical core) and set 14GB of 16GB RAM to use 2GB. The Samsung SSD 840 EVO 120GB (SSD Model) was used for storage, with Hyper-threading and swap keep off. And Linux kernel 4.15.0-115-generic and ext4 file system were used, with RocksDB version 6.8.

Workloads. We use RocksDB's own benchmark db_bench for performance evaluation, readrandom (: read N times in random order), fillseq (: write N values in sequential key order in async mode), and readseq (: demand in time) [2] are selected among various workloads. Because fill requires reading, the database was created with fillseq for readrandom and readseq, and read data.

Also, the database size is set to 1GiB, 2GiB, and 4GiB by setting the parameter of db_bench. With the value_size fixed to 8192 bytes, by adjusting the num to 131072, 262144, and 524288 respectively, which is the number of times the value is input.

Parameter Select. Among various internal parameters, seven parameters in Table 1 were selected for tuning by referring to [3].

Table 1. Selected parameter and their default value

Parameter	Default Value	Parameter	Default Value
max_bytes_for_level_multiplier	10	max_background_flushes	-1
max_write_buffer_number	2	base_background_compactions	-1
threads	1	max_background_compactions	-1
		subcompactions	1

Experimental results. From the results of Table 2, the upper part of each workload with version means the performance improvement ratio (times) compared to the default performance in the experiment result, and the lower part means the number of attempts to find optimized parameters. There was the best performance improvement in all cases when adjusting threads, and there was a performance impact when adjusting max_background_flushes in fillseq workload. In comparison by version, v2 showed the highest performance improvement overall, with the attempt to find the parameter value was also the most - up to 1458 times when performing readseq with the size of 2GiB data. The best performance improvement rate for cost was v3, which tried an average of 38 times per parameter.

Table 2. performance improvement rate and number of attempts by version

	readrandom			fillseq			readseq		
	V1	V2	V3	V1	V2	V3	V1	V2	V3
1GiB	3.82	4.25	3.82	1.56	1.81	1.71	4.47	4.61	4.35
	56	488	31	56	704	53	56	288	43
2GiB	3.90	4.00	3.89	1.42	1.64	1.39	4.59	4.79	4.59
	56	1028	43	56	488	41	56	1458	31
4GiB	3.77	3.94	3.77	1.41	1.65	1.42	4.75	1.98	4.75
	56	1028	31	56	488	41	56	972	31

4 Conclusion and Future Work

In this paper, we propose an efficient tuning method for the RocksDB by using parameters and confirm the performance improvement rate through the db_bench, a RocksDB benchmark. The previous related work has been limited to adjusting one or two parameters and suggested manual parameter optimization by experts. [3, 4, 5] This paper is meaningful in that it has suggested a method to find the best performance automatically at the lowest cost through a combination of various parameters.

The evaluation results show that the proposed scheme can increase the performance of the RocksDB by 1.39 times at least and 4.98 times at most. Although the method of tuning all parameters shows the greatest performance improvement, the search cost and time are also higher than other methods. We can confirm that the method of tuning after parameter filtering was the most effective compared to the search cost and time. As the number of selected parameters for tuning or the number of cores of the machine increases, the search cost largely decreases compared to other methods. In other words, it has good scalability and is more effective when tuning a high-performance system or the application through various parameters.

For future work, we plan to expand the method to other workloads within the RocksDB to confirm the performance gain for real workloads and devise a general method for automatic parameter tuning of applications in addition to the RocksDB.

5 Acknowledgement

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2014-3-00035, Research on High Performance and Scalable Manycore Operating System), and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2020R1F1A1075941).

References

1. RocksDB, https:// https://rocksdb.org/
2. RocksDB Benchmarking tools, https://github.com/facebook/rocksdb/wiki/Benchmarking-tools
3. Hwajung Kim, Heon Young Yeom, Yongseok Son.: An Optimal Resource Allocation Scheme for Increasing RocksDB Parallelism on High-Performance Computing Systems. KIISE Transactions on Computing Practices, 26(3), 144--149 (2020)
4. Keren Ouaknine, Oran Agra, and Zvika Guz.: Optimization of RocksDB for Redis on Flash. In Proceedings of the International Conference on Compute and Data Analysis (ICCDA '17). Association for Computing Machinery, New York, NY, USA, 155--161 (2017)
5. MEYER, Stefan; MORRISON, John P.: Impact of single parameter changes on Ceph cloud storage performance. Scalable Computing, 17.4: 285--298 (2016)

Proposal for Saying CCTV Technology for Effective CCTV Management

Eun-ji Lee[1], So-hyun Park[2], Young-ho Park[1,*],
Department of IT Engineering, Sookmyung Women's University,
04310 Seoul, Korea
{lej9031, shpark, yhpark}@sookmyung.ac.kr

Abstract. In this paper, we propose a Saying CCTV technology that automatically detects dangerous situations occurring in CCTV and converts them into text, and whenever a specific event occurs, a description of the dangerous situation is generated in sentences and delivered to the administrator. The proposed technology enhances the efficiency of CCTV monitoring by establishing a video database to enable hourly, location, and contextual retrieval and by communicating the relevant information to the administrator only when a critical situation occurs.

Keywords: intelligent CCTV, abnormal event Detection, Text Generator

1 Introduction

The recent series of violent crimes has increased the public's desire for safety, increasing the number of CCTV installations, and making it important to monitor the video efficiently[1]. However, since a person can't monitor all CCTVs visually, technology is needed to solve existing CCTV management problems and efficiently manage large amounts of CCTV [2, 3].

The problems of the existing CCTV management are as follows:

• Video retrieval is not possible. When you need to track past images with a keyword, the data is not texted, so it takes a lot of time for the administrator to manually watch the video[3].

• There is a limit to video data storage. Since data is stored in analog format and images are stored only inside the monitoring device, there is a limit to saving past images in the long term[4].

• Incident accident detection is impossible. It is impossible for humans to detect incidents and accidents while watching all the numerous and ever-increasing CCTV[2].

Recently, the development of intelligent CCTV to express and search video content like text data is actively progressing[3, 6, 7]. Examples of these methods are annotation-based search, which uses the meaning of the video to be identified by humans and expressed in natural language for search, and feature-based search system that extracts the meaning of the video using image analysis techniques and used for search[5]. However, many existing studies have focused on video summarization or

search functions, and systems to cope with dangerous situations have not been implemented.

Therefore, to improve the inefficiency of such CCTV monitoring, this study develops a "Saying CCTV" system that can text the entered image data and explain the situation to the manager in case of a hazardous situation through automated image analysis.

The composition of this paper is as follows. Chapter 2 introduces related research. Chapter 3 introduces the proposed Saying CCTV method. Chapter 4 introduces future research and concludes.

2 Related Work

This chapter introduces related research. The study in [3] extracts an object-based keyframe through image background removal and motion detection, summarizes video, and proposes a semantic-based search system. The proposed system enables detailed retrieval of the summary degree of videos captured in the surveillance area, according to the threshold, enabling a more efficient summary retrieval of vast amounts of surveillance area videos.

The research in [6] proposes a search system that supports user's various semantic search for large-capacity video data using annotation-based search and feature-based search. The system compares the keyword extracted from the user's query with frame information in the database and displays a similar keyframe to the user.

The video ontology system proposed by [7] constructs a scene name ontology and a scene model ontology that has information on feature information of the scene by structuring keywords for scene content. The scene name ontology stores words in a tree structure to enable semantic search for indexing content. And the scene model ontology enables semantic-based search by overcoming the semantic difference between low-level information such as color, shape, and material and high-level information such as objects and events. This study can search for all similar scenes by analyzing the meaning of words defining scenes beyond video scenes and text matching. As such, various video-based search systems have been studied, but studies that have actually applied it to CCTV to detect dangerous situations and deliver dangerous situations to managers are insufficient.

The study in [8] creates the best sentences by combining visual object recognition and text mining techniques in the video to identify subject Verb Object (SVO) and ranking it using statistical language models trained in web-scale data to create video-based sentences. This system is capable of generating video annotations without requiring large amounts of corpus collection and annotations, but it is difficult to determine whether the generated sentence is a dangerous situation because it simply generates only the sentence and the meaning of the sentence is not understood.

Therefore, this paper proposes the "Saying CCTV" technology, which increases the efficiency of storage by combining video search system and video feature-based sentence generation technology, and enables rapid response by explaining the risk situation to the manager by creating sentences.

3 Saying CCTV

This chapter describes the proposed "Saying CCTV" model. The overall structure of Saying CCTV is as shown in Figure1. First, it is determined whether a sentence needs to be generated according to a predefined saying time rules for labeled input data. The saying time rule is a rule to prevent repeating the same sentence in the same repeated situation. In this paper, if there is no change in the object or behavior information, it is judged as the same situation and only one sentence is created. On the other hand, when a change in one of the object and behavior information is detected, it is determined that a change has occurred in the video, and a new sentence describing the situation is created. Frame-by-frame object, action, and time information extracted from the video, and the generated sentence are stored in the database. Then, using the stored data, it is visualized and displayed in the web application to communicate the risk situation to the manager.

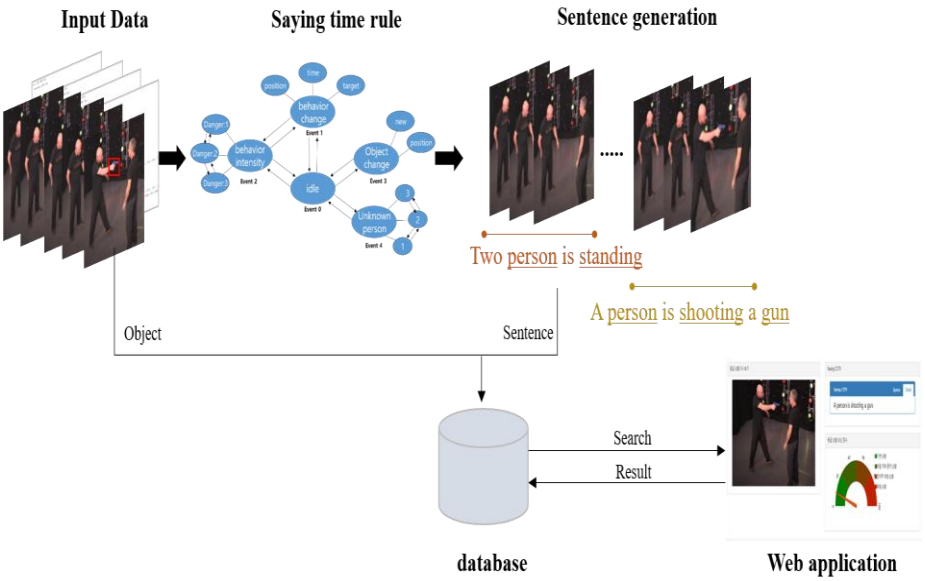


Fig. 1. Proposed Saying CCTV Process.

3.1 Dataset

This chapter describes the dataset used in the paper. Abnormal behavior can be defined in various ways, but this paper examines the four major violent crimes: sex crimes, murder, assault, and robbery, violence and murder events[9]. These are important events because they are directly related to the criminal situation. Table 1 summarizes the dataset used in the paper. The behavior of the assault situation has to

punch, kick, and the behavior of the murder situation is classified as swing, shoot. In the case of a murder situation, it includes bat, Knife, handgun, and ripple, which are mainly used as murder weapons.

Table 1. Dataset

Situation	Behavior	Object
Assault	Punch	-
	Kick	-
Murder	Swing	Bat, knife
	Shoot	Handgun, rifle

3.2 Saying Time Rule

This chapter describes the saying time rules. Continuing to create and store sentences in meaningless situations is inefficient. Thus, in this paper, the Saying time rule is defined to generate a sentence to be delivered to the manager only when a change is detected in the video.

Looking at Figure 2, when a behavior change, the intensity of behavior, change in the object, or change in the number of people occurs, it is recognized as the time point when the notification is required to the manager and a sentence is created. Looking at the event1 in Figure 2, when a change in the behavior of a person appearing in the video is detected, the first sentence is generated by identifying the time, location, and person of the action. event2 generates a sentence according to the intensity of the dangerous situation detected in event1. For example, the situation in which a person with a weapon approaches another person is more dangerous than when he or she is simply holding a weapon. In such a case, the score is selected from 1 to 3 and the manager is notified. Event3 sentences generate sentences that describe which objects were moved from which position and which direction they were moved when new objects appeared or when the objects extracted from the video were moved. Finally, when a new person appears, it generates an event4 sentence indicating how many people appeared.

Table 2 shows an example of how a sentence of a murder scene is produced, which is threatened with a gun. Looking at no1, the behavior and object to the 0 to 14th frame to appear as stand and the person. In this way, when the behavior and the object are the same, it is inefficient to generate the same sentence 14 times, so only one sentence is created by grouping 14 frames into one identical situation. Looking at no2, the 15th to 24th frames has the same behavior compared to no1, but the object information is gun added. In this case, it is recognized that a change has occurred in the video. Likewise, no3 also has the same object, but because the behavior information has changed, we can detect that a change has occurred in the video. When information changed in behavior or object is detected like this, it is judged that a change has occurred in the video, and a sentence is created by grouping it into a group of new situations.

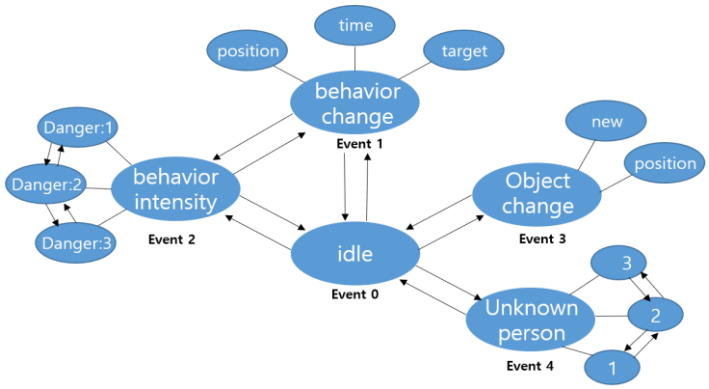


Fig. 2. Saying time rule.

Table 2. Murder Situation Saying time rule

No	Video frame	Start frame	End frame	Behavior	Object	Sentence
1		0	14	stand	person	Person is standing
2		15	24	stand	person, gun	Person is holding a gun
3		25	33	threat	person, gun	Person threatens with a gun.

3.3 Sentence Generation

This chapter describes how to create sentences. When an event interval is detected, it generates a descriptive sentence for that area. The basic word order for English grammar consists of SVO (Subject Verb Object) structure[8, 10]. Among the extracted objects, the most reliable object is used as the subject of the sentence, and the second-highest object is used as the object. By using Verb of the most reliable of the dynamic characteristics of the behavior sentence, it extracts the objects and behaviors that best suit the situation and organizes them into sentences. Figure 3 is an example of outputting and visualizing sentences by applying Saying CCTV technology, which proposes video data of a dangerous situation when shooting a gun. When a scene of a man threatening the other with a gun was detected, the sentence “A person is shooting a gun” was displayed on the web application.

Fig. 3. Proposed Saying CCTV Web application.

4 Conclusion

This paper proposes the Saying CCTV method of the automation of CCTV monitoring. Saying CCTV proposed in this paper converts visual information on video into text. Then, if there is a change in the object or behavior in the video, a statement is generated to communicate that content with the administrator.

When the text of the video becomes possible, it is possible to build a video database to enable search by time, location, and situation, and increase the efficiency of CCTV monitoring by transmitting the information to the manager only when a dangerous situation occurs. Future studies plan to directly implement the "Saying CCTV" system proposed in this paper and demonstrate how effectively the model in question detects dangerous situations and produces sentences through comparative analysis with other algorithms.

Acknowledgments. This work was supported by Institute for Information &communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.2016-0-00406, SIAT CCTV Cloud Platform).

References

1. Sun-Young Heo, Tae-Heon Moon.: An Analysis on the CCTV Location Appropriateness and Effectiveness for the Crime Prevention. Journal of The Korean Association of Regional Geographers, pp.739--750.(2015)
2. Kwang Jung Kim, Jin Wook Park, Eun Som Jeon, Jae Cheol Kwon.: Abnormal event detection for intelligent CCTV services. Korea Institute of Information & Telecommunication Facilities Engineering.pp58--61(2001)

3. HyeYoung Kwon, Kyoung-Mi Lee.: Object-based video summarization in a wide-area surveillance system. The Korean Institute of Information Scientists and Engineers.pp544--548(2006)
4. Tae-Woo Jang, Jong-Bae Kim.: Automatic CCTV Control System based on Ubiquitous Computing. The Journal of Korean Institute of Communications and Information Sciences. vol37, pp96--102(2012)
5. Ki-Byoung Kim, Hyoung-Joo Kim.: Design and Implementation of a Video Data Model Integrating Content-Based Retrieval and Annotation-Based Retrieval. The Korean Institute of Information Scientists and Engineers.pp115--126(1997)
6. Jong-hee Lee.: A Semantics-based Video Retrieval System using Annotation and Feature. The Institute of Electronics and Information Engineers.pp413--410(2004)
7. Min Young Jung, Sung Han Park.: Semantic-based Scene Retrieval Using Ontologies for Video Server. The Institute of Electronics and Information Engineers.pp32--37(2008)
8. Niveda Krishnamoorthy, Girish Malkarnenkar, Raymond Mooney, Kate Saenko, Sergio Guadarrama.: Generating Natural-Language Video Descriptions Using Text-Mined Knowledge. Association for Computational Linguistics.pp10--19(2013)
9. Sangwook Park, Seon Ho Oh, Su Wan Park, Kyung Soo Lim, Bum Suk Choi, So Hee Park, Sang Won Ghyme, Seung Wan Han, Jong-Wook Han, Geonwoo Kim.: Trends in Dynamic Crime Prediction Technologies based on Intelligent CCTV. Electronics and Telecommunications Trends. vol.35, pp17--27(2020)
10. Wahab Khan1, Ali Daud, Jamal A. Nasir, Tehmina Amjad.: A survey on the state-of-the-art machine learning models in the context of NLP. Kuwait Journal of Science. vol43, pp95--113(2016)

Deriving Software Core Functions by Source Network Structures and Execution Frequency

SangMoo Huh[1], Woo-Je Kim[2],

[1] Department of Industrial and Systems Engineering, Seoul National University of Science and Technology, Gongneung-ro 232, Nowon-gu, Seoul, South Korea, norhuh@naver.com
[2] Department of Industrial and Systems Engineering, Seoul National University of Science and Technology, Gongneung-ro 232, Nowon-gu, Seoul, South Korea, wjkim@seoultech.ac.kr

Abstract. An efficient method of improving software quality is to improve the source code that significantly affects software quality. Static analysis prioritizes code with a significant place in the source network, while dynamic analysis prioritizes code with high execution frequency. Both methods have disadvantages due to their characteristics. To solve disadvantages, this study integrated source network structure and execution. It was applied to functions of Notepad++ source code as an experiment, and the function ranking was derived. For verification, the Spearman correlation between the ranking of derived function and the ranking of static and dynamic analysis was analyzed. Analysis shows that the above 10% of the functions derived have high correlations with execution frequency and include the network structure. Like Pareto law, twenty percent of functions can be selected as important functions that affect software quality.

Keywords DEA (Data Envelopment Analysis), social network analysis, software profiling, execution frequency, Pareto law, Notepad++, software quality

1 Introduction

Software is essential for many companies. Software defects can cause fatal damage to companies. Therefore, removing defects in software has become an important task. Pareto law states that 80% of defects occur in 20% of the source code. A method that intensively manages core sources that significantly affect software quality is being presented as a cost-efficient strategy to eliminate defects [1]. The method of deriving important source code is a static analysis technique that derives the code that occupies an important position in the network structure of the source code [1]. This method has the disadvantage of overestimation that does not match execution [2]. The second is a dynamic analysis technique that derives functions that execute frequently by running software [1]. This method has the disadvantage of underapproximation in which important functions in the source network are not derived [2]. To solve these disadvantages, this study conducts research to find important source codes in which the network structure and execution are integrated.

2 Methods

2.1 Static Analysis Technique : Source Network Structure Analysis

Social Network Analysis (SNA) is a static analysis technique that analyzes the network structure to derive elements that occupy an important position [3]. The modules(functions in this study) that comprise the software call each other and form a network. An analysis of the source network structure using SNA techniques can derive functions that occupy important positions.

2.2 Dynamic Analysis Technique : Execution Frequency Analysis

Software profiling is a dynamic analysis technique that can measure the execution frequencies of functions [4]. All the source functions for the study were modified, and the execution frequencies of the functions were collected by executing the software.

2.3 Data Envelopment Analysis : Integrating Network Structure and Execution

DEA is a non-parametric technique that can calculate the relative efficiency of input elements by analyzing the relationship between input and output [5]. By setting function ranking derived by the SNA method as input values and an execution frequency as an output value, new functions ranking the integration of the network structure and execution can be derived.

3 Research Procedure

Step (1.1) research source code selection: Research software requires an adequately sized source that can perform every menu function. Operating systems and databases too large to run every menu function and small-scale software were deemed unsuitable for research. Notepad++ version 10.0.0.136, which is a well-known open-source software that can execute every menu function, was selected for the study.
Step (1.2) static analysis: Derivation of network function ranking through network structure analysis using 8 indicators of 5 SNA.
Step (1.3) dynamic analysis: All menu functions of Notepad++ were executed only once, and the execution frequencies of source function were measured.
Step (1.4) derivation of new ranking of functions that combines network structure and execution frequency by DEA.
Step (2.1) Using DEA, the functional rankings of 8 network indicators of Step (1.2) are integrated, and the function ranking of integrated SNA only was derived.
Step (2.2) Three developers actually used Notepad++ to collect the execution frequency, and the function ranking of the execution frequency was derived.

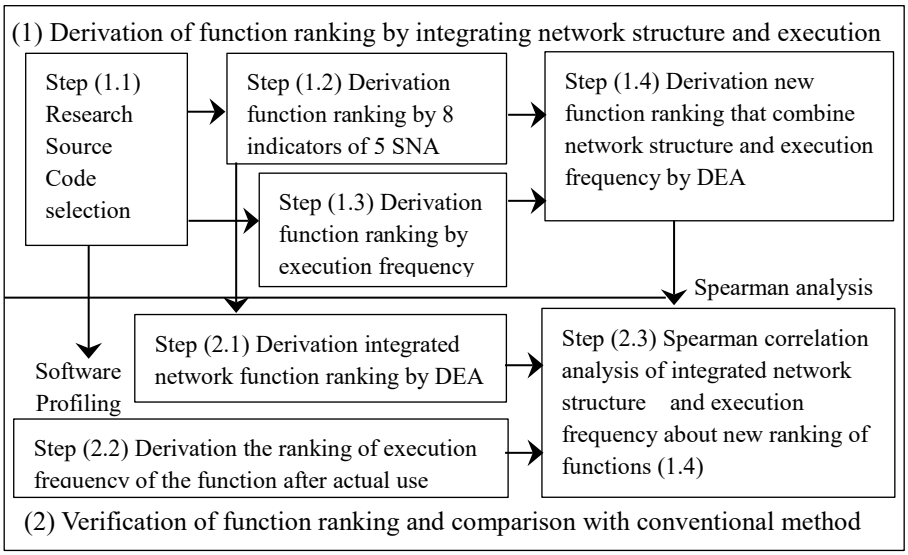


Fig. 1. Overall study process

4 Research Result

Step (2.3) Spearman correlation analysis: Verification of the integrated function ranking should be compared with the actual function ranking. However, it is unknown. Therefore, it was verified through the Spearman correlation to determine whether the execution frequency and the network structural characteristics were included. The integrated function rank was set on the x-axis and the Spearman correlation was set on the y-axis, expressed in percentages.

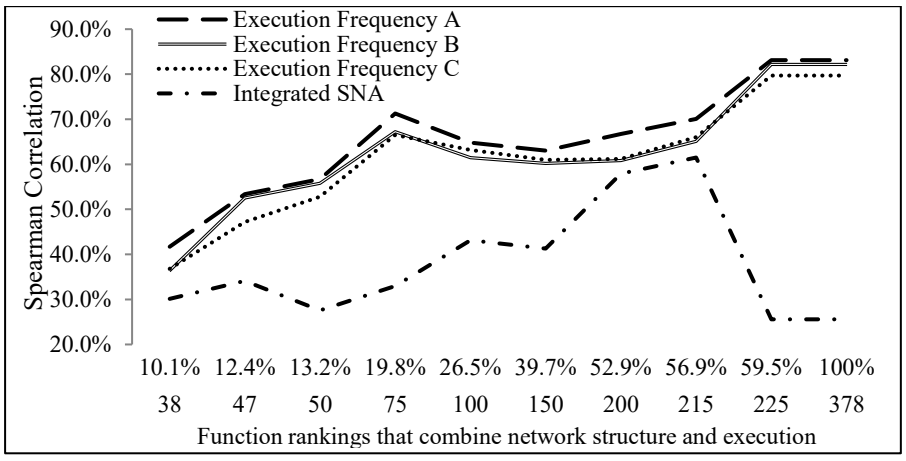


Fig. 2. The graph of Spearman correlation by function rankings

Based on the P-value, the function rankings below 10% were not correlated, while those above 10% were correlated. The Spearman correlation with the execution frequency (dynamic analysis) was higher than that of the integrated network structure (static analysis). At the level of 20% of Pareto law, a high correlation with execution frequency was found, with a little correlation with the network structure. In the conventional static analysis, only the network structure was analyzed. In the dynamic analysis, only the execution frequency was analyzed. In this study, the integrated result of the execution frequency and network structure was derived by overcoming the disadvantages of conventional techniques. Moreover, 20% functions can be selected as important functions that affect software quality, like Pareto law.

5 Conclusion

In this study, the network structure and execution were integrated using the DEA technique, and the results were analyzed to possess correlations with the execution and network structure. The significance of this study, first, is that the network structure and execution were integrated using the DEA technique. Second, since important functions can be derived more accurately, performance can be improved and defects can be eliminated more accurately and cost-effectively. Third, since important functions can be derived, it can be used as a theoretical basis for performing inspection and refactoring. For future research, it is necessary to study the degree to which performance is improved in the case that the upper function derived through this study is improved.

References

[1] Minhazur Rahman. 2007. Application of Social Networking Algorithms in Program Analysis: Understanding Execution Frequencies. Ph.D. Dissertation. Colorado State University Libraries, Colorado.

[2] Julia Rubin and Marsha Chechik. 2013. A survey of feature location techniques. In *Domain Engineering*, Springer, Berlin, Heidelberg, 29–58.

[3] Yonghak Kim and Youngjin Kim. 2019. Social Network Analysis. Retrieved from http://www.riss.kr/link?id=M15388176.

[4] Manos Renieres and Steven P. Reiss. 2003. Fault localization with nearest neighbor queries. In 18th IEEE International Conference on Automated Software Engineering. IEEE, 30–39. DOI: https://doi.org/10.1109/ASE.2003.1240292

[5] Min-Soo Choi, Woo-Je Kim, Hyun-Ki Cho, and Se-Jung Park. 2012. A study on an evaluation method for LCD TV products using axiomatic design based hybrid AHP/DEA model. *Korean Manag. Sci. Rev.* 29, 1 (Mar. 2012), 33–56. DOI: https://doi.org/10.7737/KMSR.2012.29.1.033

Image Classification on Agriculture Using Transfer Learning

Borin Min[1], Hyoseok Oh[1], Ga-Ae Ryu[1], Sang Hyun Choi[1], Carson Kai-Sang Leung[2], Kwan-Hee Yoo[1]

Dept[1]. Of Computer Science, Chungbuk National University, Cheongju, South Korea
Dept[2]. Of Computer Science, University Of Manitoba, Manitoba, Canada
{minborin, garyu, gyzmdh, chois, khyoo}@chungbuk.ac.kr
kleung@cs.umanitoba.ca

Abstract. We aimed to detect and classify the labels of normal and abnormal types from images of agriculture product using computer vision method. The purpose of this work was to operate throw computer vision's algorithm on the input image to give the outcome could be normal or abnormal of corn, cucumber, pepper, rice strawberry. Here, we proposed to construct an ideal method to deal with label classification task using transfer learning model[1] which could work on similar task, minimal dataset and incredible less time to train.

Keywords: Image Analysis, Computer Vision, Transfer Learning Method.

1 Introduction

Defective agriculture plants are the primary concerned issue in agriculture field. And those things are the biggest risks that can be destroyed plants and unidentifiable easily the type of diseases. This difficulty could be inspired more researchers in seeking the ideal ways that could be responded effectively. In our agriculture, we have 5 category of plants which is grouped into normal and abnormal type and within that has: corn, cucumber, rice, pepper and strawberry. We proposed method to classify normal and abnormal type by applying transfer learning method[1] (pretrained weight) of Inception-v3[2] which was trained very well on ILSVRC2015 over 15 million different label high-resolution images. Transfer learning method [1] is useful for performing similar tasks that is much reduce model learning time, allow small dataset to get fine prediction results.

Fig.1 Normal and abnormal images of agriculture

2 Dataset Preparation

Dataset preparation, images were grouped into 9 different category folders and the name of folder represent of label of image. Each image has resolution 4032x3024 pixels equally and had scaling down to the resolutions of 299x299 pixels before input to training. Also, the dataset was split as three parts: 80% of dataset used for training, 10% of dataset used during model training and the rest of the dataset are used to test trained model. The size of training and validation have essentially affecting on result of training.

Table 1. The number of datasets.

Category	Total image
Normal corn	2,351
Normal cucumber	2,883
Abnormal cucumber's leaf	414
Normal pepper	6,550
Abnormal pepper's leaf	881
Abnormal pepper's fruit	106
Normal rice	8,727
Normal strawberry	7,575
Abnormal strawberry's leaf	164

3 Methodology

In this section, we will be describing the involved methods used to study image classification which separated into three main parts. First part, Inception-v3[2] is the convolutional neural network base architecture for image feature extraction. Second part, transfer learning[1] is the technique used to share learning experiences of existed

task to new similar task. For our case, we apply the knowledge of Inception-v3[2] that learned from ILSVRC2015. In the final part, where we link the learned knowledge to our problem, by creating a bottleneck layer to operating the learned result and new input image. Next, forwarding bottleneck layer into training process for gradient descendant.

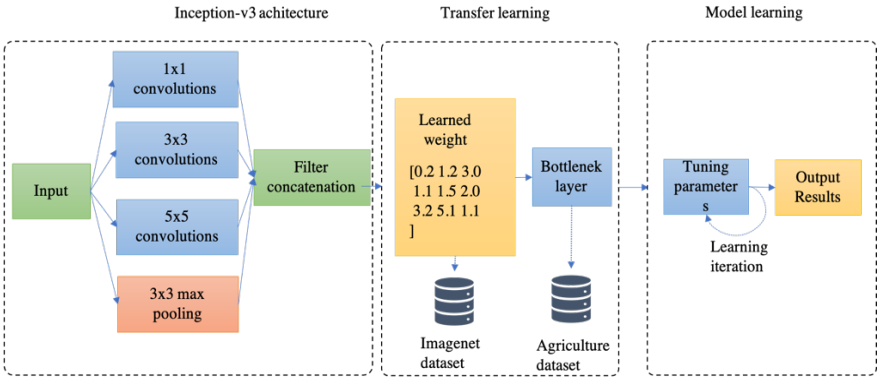


Fig.2 The construction of transfer learning with Inception-v3 architecture.

4 Experimental Results

In this section, showing the experimental results of transfer learning from inception-v3[2] on and ResNet101[3] on ILSVRC2015. After training through various epochs, test and validation set size, we got that the highest training accuracy was inception-v3[2] which is about 97.00 in Fig.3.

Table 2. Table of showing training various parameters.

Architecture	Epochs	Validation set %	Test set %	Test acc	Validation acc	Train acc
Inception-v3	100	10	10	92.22	74.00	85.00
Inception-v3	200	10	10	92.66	87.99	87.99
Inception-v3	300	10	10	94.22	88.99	91.00
Inception-v3	400	10	10	94.22	77.99	92.00
Inception-v3	500	10	10	94.80	81.99	87.00
Inception-v3	100	20	20	90.37	79.00	88.99
Inception-v3	200	20	20	93.00	83.99	89.99
Inception-v3	300	20	20	94.49	82.99	93.00
Inception-v3	400	20	20	95.64	83.99	97.00
Inception-v3	500	20	20	95.53	86.00	94.99

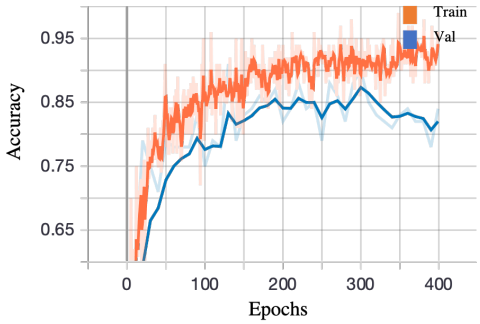


Fig.3 Training and validation accuracy of transfer learning.

5 Conclusions and Future work

In this paper, we constructed a method to deal with label classification task by using transfer learning. Transfer learning could be able to apply for similar task, especially when the dataset is not enough size. Rather than this, time is always considered as significant point in training, with inception-v3[2] took time around 20 minutes to finish bottleneck and 20 seconds for tuning. For next training, it took around 20 to 30 seconds to complete 500 epochs. Also encountered the long training time on ResNet101[3]. Last but not least, we are planning to be multi-label bounding boxes on every defect point of images using transfer learning with bounding boxes methods.

6 Acknowledgment

This research was supported by the Bigdata Platform and Center Establish Program(communication)(2020-데이터-위 07) supervised by the NIA(National Information Society Agency), Republic of Korea. This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program(IITP-2020-0-01462) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation). This research was supported by the ITRC(Information Technology Research Center) support program (IITP2020-2015-0-00448) supervised by the IITP (Institute for Information and Communications Technology Planning & Evaluation), grant funded by the Korea government.

References

[1] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A Survey on Deep Transfer Learning," *arXiv:1808.01974 [cs, stat]*, Aug. 2018, Accessed: Nov. 12, 2020. [Online]. Available: http://arxiv.org/abs/1808.01974.

[2] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," *arXiv:1512.00567 [cs]*, Dec. 2015, Accessed: Nov. 12, 2020. [Online]. Available: http://arxiv.org/abs/1512.00567.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv:1512.03385 [cs]*, Dec. 2015, Accessed: Nov. 12, 2020. [Online]. Available: http://arxiv.org/abs/1512.03385.

Point-Based Rectangle Clustering with DBSCAN

Tin Hok, Dimang Chhol, Kwan-Hee Yoo

Computer Science Department, Chungbuk National University, South Korea
{hoktin, dimangchhol, khyoo}@chungbuk.ac.kr

Abstract. A thousand data categories in all research fields are increasingly collected and analyzed every second of the time. This paper proposed a data integration technique for clustering rectangle dataset with DBSCAN, which is useful for the approach in many use cases. For instance, using this technique to cluster rectangle-based defects for recognizing the patterns of defective areas of damaged products.

Keywords: Point-based rectangle clustering, Representative points of rectangles, DBSCAN

1 Introduction

The data integration aims to explore the usefulness of every level of data and seamlessly compute the data preparation of the related information from sources. This paper has a robust data integration ability of clustering point-based rectangles with DBSCAN [1]. In this way, the rectangles data is arranged and translated into an understandable dataset structure for the DBSCAN algorithm before learning to be clustered.

2 Rectangle Data Preparation Approach for DBSCAN and Result

Although DBSCAN is great at clustering high density and handling outliers within a given point-based dataset, it still has a limitation in evaluating the shape dataset.

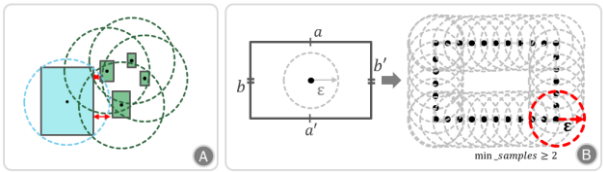


Figure 1: (A) Misconception of clustering center-based rectangles in DBSCAN. (B) Translation from rectangle to representative points

As shown in Figure 1, we introduce a dependable approach to generate points along with rectangle edges due to having a constraint to work with a shape-based dataset. The equation required the parameters of DBCAN to be calculated the step from point to point. The following formula defines the interval step:

$$interval\_step = \frac{\varepsilon}{\min\_samples}$$

We, therefore, can explain the total edges of a rectangle L by:

$$L_{a=a';\, b=b'} = \sum_{i=0}^{n} \mathrm{i} * \frac{\varepsilon}{\min \_\mathrm{samples}} \quad \text{that} \quad n_{a=a';\, b=b'} = \frac{edge\_length}{interval\_step}$$

for $0 \leq i \leq n$ with a length of edge n

As a result, every given point is evaluated, clustered, and identified by different colors, as illustrated in Figure 2. Notably, if there is a group of relative points representing only one rectangle, it is considered as a noise rectangle.

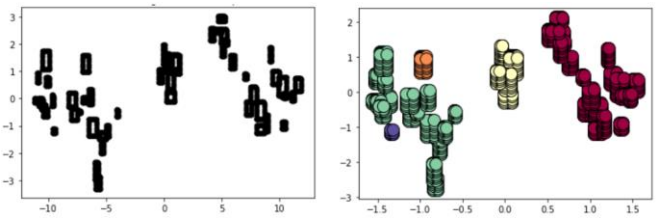


Figure 2: Result of point-based rectangle rectangles grouped by colors

In conclusion, we found a beneficial approach suable to cluster the point-based rectangle dataset. Interestingly, this proposed concept navigates to an idea of clustering different dynamic shapes in the context of DBSCAN or in other clustering algorithms.

3 Acknowledgements

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program(IITP-2020-1711120023) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation) and this research was supported by the ITRC(Information Technology Research Center) support program (IITP2020-2015-0-00448) supervised by the IITP (Institute for Information and Communications Technology Planning & Evaluation), grant funded by the Korea government.

Reference

[1] “DBSCAN clustering algorithm — scikit-learn 0.23.2 documentation.” https://scikit-learn.org/stable/auto_examples/cluster/plot_dbscan.html (accessed Nov. 13, 2020).

Knowledge Graph-based Matching of Industrial Wastes and Raw Materials for Closed Loop Recycling

Seon Kyu Park, Sang Lok Yoo, Keon Myung Lee

Dept. of Computer Science, Chungbuk National University,
Cheongju, Chungbuk 28644, Korea, kmlee@cbnu.ac.kr

Abstract. Industrial manufacturing consumes various resources such as raw materials, intermediate products and energies, and produces wastes and by-products which might be used for other manufacturing. Closed loop recycling is to make a manufacturing factory use wastes and by-products from other manufacturing factories. To activate such recycling, it is important to find matching pairs of factories in which a factory uses some wastes or by-products from other factory. This paper presents a knowledge graph-based method to find such matching pairs using an ontology for closed loop recycling.

Keywords: knowledge graph, ontology, closed loop recycling, manufacturing

1 Introduction

Modern society heavily depends on industrial manufacturing which fabricates industrial and household products from raw materials, intermediate products along with machinery and energy. During a manufacturing process, some wastes, by-products, slug, and heat are produced and thrown away. If such thrown-away stubs can be used for another manufacturing, it will be very helpful in saving cost and valuable resources. Closed loop recycling denotes such process that by which a product is used, recycled, and then made into a new product, therefore, not ever entering landfill. For example, the cold energy generated during the evaporation of LNG can be used in cold storage. For cost-saving and resource-conservation, it is desirable to make the best recycling of such wastes, by-products, slug, and heat in the manufacturing. To enable such virtuous recycling, we have to find the pairs of matching factories one of which produces recyclable stubs and the other of which uses them in its manufacturing.

It would take considerable amount of time to manually search for the pairs of matching factories because factories do not usually maintain in a standardized vocabulary and format their data about raw materials, intermediate products, energy, wastes, slug, final product, and so on for their manufacturing process. Hence it is needed to have a method to handle non-standardized vocabularies and data for automatic matching pair search. An ontology is a good mechanism for handling

communication of non-standardized vocabularies. A knowledge graph is a flexible framework to represent some information and knowledge for a domain. This paper presents a method to support matching pair search for closed loop recycling.

2 Related Work

Knowledge is a valuable asset to be maintained and used in organizations or enterprises for service and management. Knowledge management is an important activity for organizations that defines, categorizes structures, retains, and shares the knowledge and experience of employees and organization to improve their efficiency and save their knowledge. Hence effective knowledge representation is essential for modeling domains of knowledge to retain more context and meaning, as information is parsed and abstracted.

Various knowledge representation methods have been developed such as rules, frame, semantic network, script, and so on. To represent domain and cross-domain knowledge, knowledge graph can be used. Knowledge graph is a kind of semantic network which represents a collection of interlinked descriptions of objects, events, and concepts in which data are put in context via linking and semantic metadata.[1] In a knowledge graph, meanings are expressed as structure, all identified entities including types and relations are identified using global identifiers with unambiguous denotation, and a limited set of relation types is used.

When vocabularies used for description is not standardized, it is difficult to communicate information in an intended meaning. To handle this issue, an ontology can be used for specifying shared concepts that represents a view of the world that can be used to structure information.[2] In other word, an ontology is a shared vocabulary for a domain which gives information about conceptual hierarchies, synonyms, and related attributes.

In closed loop recycling, supply chain management is important and difficult task which specifies the factories to give and receive some materials and their capacity.[3] Designing such supply chains is known as an NP-hard problem. Hence, the supply chain design relies on field experts and some optimization techniques like meta-heuristics. In many countries, proliferation of closed loop recycling is an industrial policy for environmental recovery and respect. To build a closed loop recycling supply chain, it is first needed to pairs of factories that give and receive materials. The search for such matching factories has been done manually by investigating the manufacturing process and evaluating the expenses and profit. This paper is interest in assisting such search process with the help of information services.

3 Knowledge Graph-based Matching Factory Pair Search for Recycling

A manufacturing process takes in raw materials, intermediate injecting material, puts out by-products, waste, slug, heat while final products are produced, as shown in Figure 1. Some intermediate products can be used in other manufacturing process

with some purification and chemical treatments. Such recycling process helps enhances expenses reduction and environmental protection.

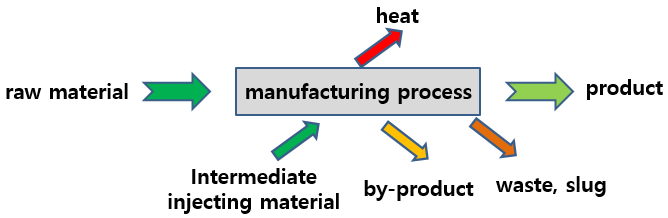


Fig. 1. A typical manufacturing process with inputs and outputs.

For recycling, we first find which factory consumes which materials and discharges which subs in manufacturing process. Among many factories, it is inconvenient to find manually such pairs of matching factories because it is time-consuming and not effective. We are interested in an approach of helping find such pairs with the assistance of information services.

Each factory has its own databases in its own vocabulary and format. Such databases might manage data about raw materials, by-products, intermediate injecting material, wastes and slug, products, and manufacturing process-related information such as water, ejecting heat, steam, and so on. Some materials are named differently, some materials are a subclass of other material, some materials are product of other materials, and some materials are a branch of materials. These kinds of variety in description make it difficult to find the pairs of matching factories. In addition, the databases of factories do not maintain the same kind of items; some items of a factory are maintained in other factory.

To find automatically matching pairs, we need to have a system that maintains knowledge and data for factories and manufacturing processes. To handle the heterogeneity of data, we use knowledge graph-based representation because it is flexible and has well-established manipulation operations for knowledge handling. Knowledge graph represents related information in terms of nodes and edges; Nodes corresponds entities which can be anything related to manufacturing process, materials, by-products, products, wastes and so. Edges represents relations between entities. Knowledge graph is used to represent data and information in terms of factories. Knowledge graphs are stored in triples of (entity, relation, entity) and retrieved from the triple databases by SQL-like queries.

To handle the diversity of vocabularies, we use an ontology for the entities, relations, and attributes. For matching pair search, we define an ontology for synonyms, subclasses, brands, constituents, substitute in raw materials, by-products, intermediate injecting material, wastes and slug, products, and manufacturing process-related information. With such ontology, the matching factories can be searched by generating and executing such queries as ‘find <factory A>.waste is <factory B>.raw_material’. Such queries are augmented by modifying them with help of the ontology. The augmentation can be done as follows: Terms can be replaced by synonym terms, terms can be replaced with their brand, terms can be replaced with subclasses, terms can be replaced with their raw materials, and so on. Such queries are

generated in terms of raw materials to be used a specific factory, or outcomes of a specific factory.

4 Pilot Studies of the Proposed Method

The proposed method has been applied in a pilot study to search matching factories in an industrial complex of Korea. Knowledge graphs have been constructed for factories in the industrial complex. Manufacturing processes, materials and wastes used and produced are treated somewhat as business secrets at companies. The constructed knowledge graphs have not contained detailed information for factories, yet are useful to search for candidate matching pairs. An ontology has been developed to handle heterogeneity of vocabularies used in knowledge graph representation. Figure 2 shows a snapshot of the ontology which describes terms used in concrete manufacturing.

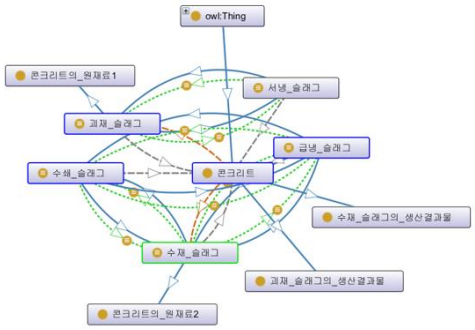


Fig. 2. A snapshot of the constructed ontology for concrete manufacturing.

5 Conclusions

For closed loop recycling, it is required to first find factories that use industrial wastes from other factories. To help such search process, it is needed to effectively represent related data and knowledge and to provide searching mechanism. This paper proposed a knowledge graph-based representation method of factories' manufactories process and an ontology-based method for handling term heterogeneity. The proposed methods has been applied to a pilot study of an industrial complex. We has seen that it is feasible and effective. There remains to acquire factories' data which are regarded as somewhat business secrets.

Acknowledgments.
This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program(IITP-2020-0-01462) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation)

References

1. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic Web 8 (2016): 489–508.
2. Euzenat, J., Shvaiko, P: Ontology matching (Vol. 18). Heidelberg: Springer (2007).
3. Soleimani, H., Kannan, G.: A hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks. Applied Mathematical Modelling, 39(14), 3990-4012. (2015).

Application of Optimal Process Prediction Method for Non-ferrous Metal Operation

Ga-Ae Ryu[1], Hyoseok Oh[2], Kwan-Hee Yoo[1*]

[1]Dept. of Computer Science, Chungbuk National University, South Korea
[2]Dept. of BigData, Chungbuk National University, South Korea
{garyu, gyzmdh, khyoo}@chungbuk.ac.kr
[*]Corresponding Author

Abstract. As the 4th industrial revolution progresses in the non-ferrous metal manufacturing industry, various services are being researched and manufactured for the purpose of building a smart factory environment. The smart factory environment aims to provide an efficient work environment for workers and increase production efficiency by analyzing data and providing results using various techniques of machine learning and deep learning. In this paper, to achieve the above objectives, we propose a system that predicts optimal values for various factors that affect nonferrous metal operations and visualizes the results. If the proposed method is used, it is possible to increase production efficiency and improve the working environment because it works under optimal conditions. If the proposed method is used, it is possible to increase production efficiency and improve the working environment because it works under optimal conditions.

Keywords: ,Non-ferrous metal, Golden Recipe, Smart Factory, Deep Learning, Prediction.

1 Introduction

Recently, manufacturing processes in various types are focusing on building a smart factory environment. The smart factory environment aims to increase production and work efficiency by using various techniques such as machine learning and deep learning. The smart factory environment aims to increase production and work efficiency by using various techniques such as machine learning and deep learning. For example, by using actual data of the manufacturing process, the condition of the machine can be determined in advance and provided to the operator to enable the machine to take action. [1] In addition, more useful information can be delivered to workers through various analysis of the manufacturing process. [2]
In this paper, in order to increase production efficiency even in non-ferrous metal manufacturing processes, the optimal process is predicted using deep learning and visualized in real time. Non-ferrous metal operation is carried out by controlling the tap by using electric power and voltage in the electrode, and by controlling the amount of electricity using three electrode bars(EPI A, EPI B, EPI C) through tap

movement to generate heat to melt the non-ferrous metal. At this time, the operator can adjust the positions of the tap and electrode bars by viewing the flow of power and voltage. Therefore, we predict and provide future power, current, tap, and electrode bars positions that can be optimal processes to workers using a learning model. The learning model is trained using Long Short-term Memory (LSTM)[3], which is a time-series data prediction method. Because we need to predict multiple factor values, we train using the LSTM[3] model for multi-factor prediction based on time. The usage data is from 2019 to the present of the non-ferrous metal company'S'. The prediction results are visualized on the web and provided.

2 Proposed Method

In this paper, we predict and visualize the optimal process recommendations for non-ferrous metal working processes. In the non-ferrous metal operation process, the optimal process refers to an operation with a low power per product quantity value and a high production quantity. The values to be predicted for the optimal operation process are tap, voltage, power, and electrode bars position, and to predict this, an LSTM model[3] that can predict multiple factors is created. The LSTM model [3] is a method to solve the vanishing gradient problem in learning through backpropagation by using the input gate, forget gate, output gate, and state storage. The training data extracts good working days from 2019 to the present and creates time-series data according to the working hours of that date. At this time, the extracted data refers to real-time data coming through the operation (voltage, current, power, tapping temperature, electrode bars location, exhaust gas, etc.). To predict multiple factors using the training data, we build a learning model using LSTM cells. The constructed learning model is shown in Fig 1.

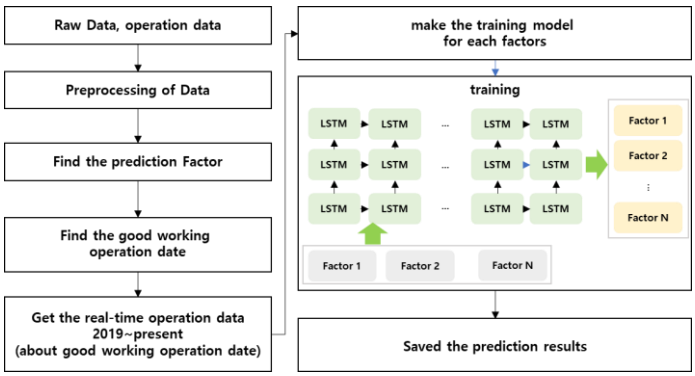


Fig. 1. Structure of data learning methodl

A learning model is constructed for each factor to be predicted, and predicted values of each factor are summed by prediction time. Using the proposed learning method, tap, voltage, power, and electrode bars positions can be predicted so that the operation becomes an optimal process

3 Discussion and Future work

In this paper, we proposed a method for predicting the optimal process of non-ferrous metal operations using a learning method for multiple factors. Fig 2 is the result of visualizing the predictions made using the proposed method. The visualization of the prediction results is to visualize the prediction results after learning by finding data that fits the conditions when the operator enters conditions for the operation.

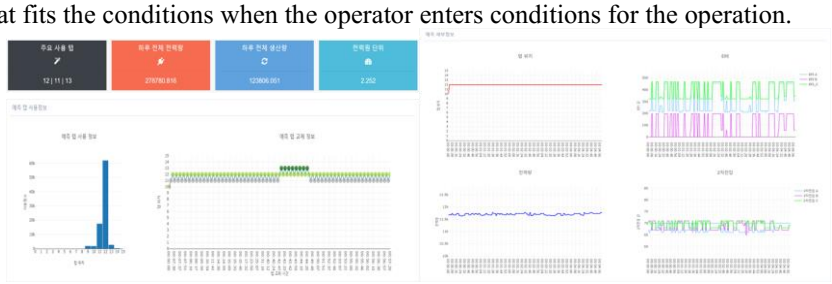


Fig. 2. Visualize the results of training a multi-factor prediction model

In the future, a method of providing a value for an optimal process to an operator in real time is supplemented, and an accurate value is provided by determining whether the predicted value is meaningful.

Acknowledgments. This material is based upon work supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program, No.10082578, 'Development of intelligent operation system based on big data for production process efficiency and quality optimization in non-ferrous metal industry.

References

1. BorithJoo, I.T., Choi, S.H.: Stock Prediction Model based on Bidirectional LSTM Recurrent Neural Network. In: Journal of Korea Institute of Information, Electronics, and Communication Technology, vol. 11, no. 2, pp. 204--208 (2018)
2. Doung Chankhihort, Byung-Muk Lim, Gyu-Jung Lee, Sungsu Choi, Sun-Ock Kwon, Sang-Hyun Lee, Jeong-Tae Kang, Aziz Nasridinov , Kwan-Hee Yoo, “A Visualization Scheme with a Calendar Heat Map for Abnormal Pattern Analysis in the Manufacturing Process”, International Journal of Contents, Vol.13, No.2(2017)
3. Greff, K., Srivastava, R.K., Koutnik. J., Steunebrink, B.R.: LSTM: A search Space Odyssey. In: IEEE Transactions on Neural Networks and Learning Systems, vol. 28, no. 10, pp. 2222--2232. (2017)
4. Tran, Q.K., Song, S.K.: Water Level Forecasting based on Deep Learning : A Use Case of Trinity River-Texas-The United States. In: Journal of KIISE, vol. 44, no. 6, pp. 607--612 (2017)

Product Quality Prediction by Multivariate Timeseries Anomaly Detection

Manas Bazarbaev[1], Hyoseok Oh[1], Aziz Nasridinov[1], Kwan-Hee Yoo[1]

[1] Chungnuk National University, Computer Science Department
{manas, gyzmdh, aziz, khyoo}@chungbuk.ac.kr

Abstract. In this paper we were tried to predict product quality by number of founded anomalies in real time data from sensors. Initially we have real time data from sensors, and we made anomaly detection model for this data. Then we have product information data and from this data we are generating data with product quality and number of anomalies during producing of this product. Last, we are trying to make classification of product quality. The goal of our research is to understand is it possible to predict product quality from anomalies information.

Keywords: Anomaly detection, Machine learning, Sensors data, Product quality.

1 Introduction

The product quality one of the most important criterion for consumers. Consequently, every manufacturer desire to produce high quality products with minimum expenses. To produce high quality products, we need to discover reasons of producing products with low quality. Further we should try to prevent occurring the situations where manufacturer can produce product with low quality.

The purpose of our research is to make predictions of making low quality product based on number of anomalies occurred during production of the product. For our research we have the timeseries sensors data from metal melting manufactory and product information data. We desire to perform multivariate anomaly detection method. As the anomaly detection method, we are going to select one model from Anomaly Detection Tool Kit (ADTK) library[1]. In the next place our goal is to make machine learning model to predict product quality with number of detected anomalies. We can use here any classification model from Scikit-Learn library[2]. Hereafter we will compare the results of classification, then select the best method with best result. Finally, we expect that based on that results we can predict product quality.

2 Data Preparation

The anomaly detection model there have been used timeseries data from with 2 features. The training data from 18th February 2020 to 25th February 2020 have been used for anomaly detection algorithm (Fig. 1).

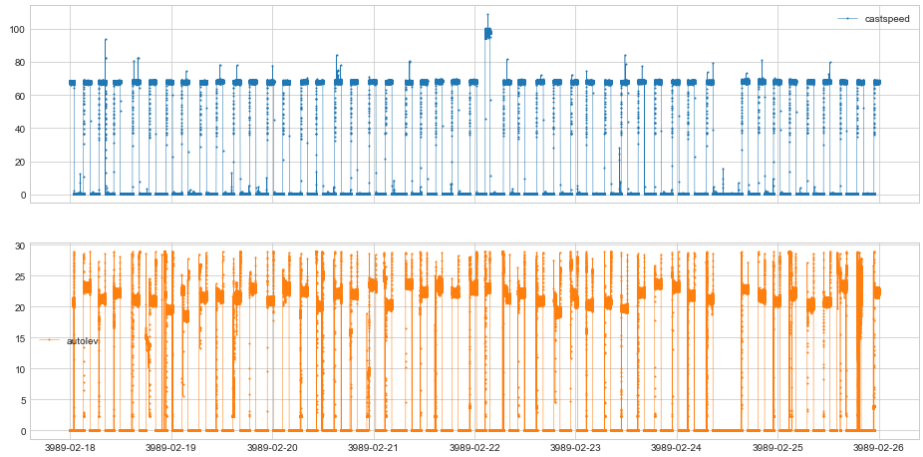


Fig. 1. Sensor data from with 2 features (castspeed and autolev) used for train anomaly detection algorithm (the data range is 1 week).

As the data of classification model by product quality have been used data from 2nd January 2019 to 6th November 2020. We have used data from casting start time and casting end time and detected anomalies for this product. The number of anomalies between casting start time and casting end time has been added as a new column for the table. The data used for classification consists two columns: the first is number of anomalies and the second is categorical data where 0 means high quality product and 1 means low quality product (Table 1).

Table 1. The product quality classification dataset information.

Dataset type	High quality (0)	Low quality (1)	Total
Initial	3596	454	4050
Train	2688	349	3037
Test	908	105	1013

3 Method

In this chapter described Machine Learning methods for anomaly detection and then methods for product quality classification.

3.1 Multivariate Timeseries Anomaly Detection

For multivariate timeseries anomaly detection have been used the method called OutlierDetector[3] from ADTK library with EllipticEnvelope[4] model from Scikit-Learn[2] library with parameters: *contamination=0.006*, *support_fraction=0.35* and *assume_centered=True*. The data has been feed into the OutlierDetector model and with calling *fit_detect* function. The method found as anomaly 4094 records from our timeseries training data (Fig. 2).

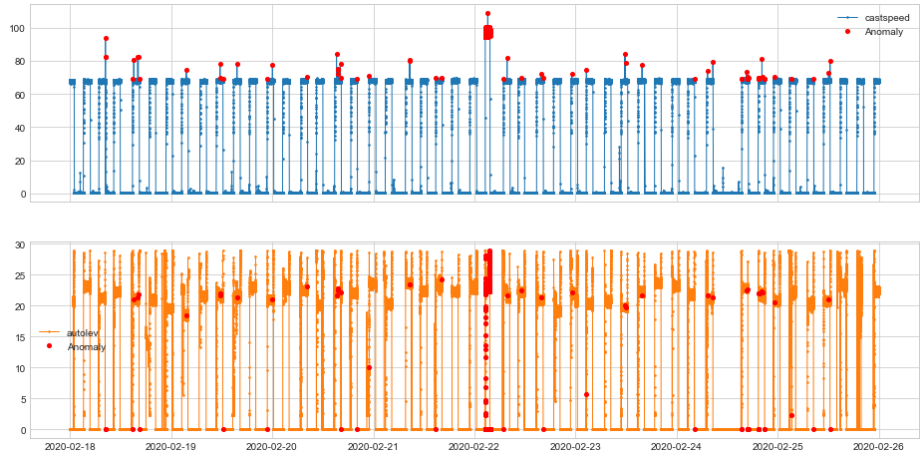


Fig. 2. Outlier detector result from the multivariate time series data (1 week data from the company sensors). The detected anomaly records have been shown as red markers.

3.2 Product Quality Prediction

As the product quality prediction method can be used any classification model from Scikit-Learn library. There have been used several methods such as: DecisionTreeClassifier, Support Vector Machines (SVC), RandomForestClassifier and GradientBoostingClassifier. Since our classification data is not complicated and the results of these methods are not so various (Table 2).

Table 2. The product quality classification methods result.

Method	Train set accuracy (%)	Test set accuracy (%)
DecisionTreeClassifier	90.1	88.9
Support Vector Machines (SVC)	88.5	89.6
RandomForestClassifier	90.1	88.9
GradientBoostingClassifier	89.1	89.2

The accuracy of these methods can be better if we try to find better anomaly detection method and labeled data for the anomaly detection.

4 Conclusion

We have found good machine learning anomaly detection method. And based on that anomaly detection method we generated the data with product quality and number of anomalies while producing this product. We have used that data to classification of product quality.
As we noticed from classification result, it is possible to predict product quality by number of anomalies. In the next we can try to use some deep learning methods for anomaly detection, and it may improve the product quality prediction. Based on this knowledge, in the future we can make product quality analysis to exclude the situations that can affect to produce low quality product.

5 Acknowledgements

The work is supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under i-Ceramic manufacturing innovation platform technology development business. No.2004367, 'Development of cloud big data platform for the innovative manufacturing in ceramic industry and by the MSIT(Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program(IITP-2020-0-01462) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation)"

References

[1] "Anomaly Detection Toolkit (ADTK) — ADTK 0.6.2 documentation." https://arundo-adtk.readthedocs-hosted.com/en/stable/index.html (accessed Nov. 15, 2020).

[2] "scikit-learn: machine learning in Python — scikit-learn 0.23.2 documentation." https://scikit-learn.org/stable/index.html (accessed Nov. 15, 2020).

[3] "Detectors — ADTK 0.6.2 documentation." https://arundo-adtk.readthedocs-hosted.com/en/stable/api/detectors.html#adtk.detector.OutlierDetector (accessed Nov. 15, 2020).

[4] "sklearn.covariance.EllipticEnvelope — scikit-learn 0.23.2 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.covariance.EllipticEnvelope.html (accessed Nov. 15, 2020).

A Study on the GSDC-LDG System for Efficient Resource Utilization

Sangwook Bae[1], Geonmo Ryu[2], Byungyun Kong[3] and Heejun Yoon[4*]

[1,2,3,4] Korea Institute of Science and Technology Information (KISTI), 245, Daehak-ro, Yuseong-gu, Daejeon, Republic of Korea
{wookie[1], geonmo[2], kong91[3], k2[4]}@kisti.re.kr

Abstract. The emergence of large-scale research facilities and research equipment has brought challenging and large-scale computational and data processing requirements. LIGO and Virgo generate 20 terabytes of astrophysical strain data and 1 petabyte of raw data (environmental, instrument monitors) per instrument year of observation. These facilities and equipment require absolute support for computing resources such as CPU, storage, GPU, etc. KISTI-GSDC is a data center that provides these computing resources and was established as a national funding project to promote fundamental research activities in South Korea. KISTI-GSDC supports domestic and foreign researchers by establishing a gravitational wave data analysis computing environment through a memorandum of understanding with the LIGO Scientific Collaboration (LSC). In particular, the first direct observation of gravitational waves for the first time in 2015, demand for gravitational wave research has increased, supporting Tier 2 service(global services) at the request of KGWG in 2019. In addition, as the KAGRA has been in full operation recently, the demand for KISTI-GSDC resources (CPU, storage, network etc.) has been increasing. Therefore, in this paper, we study a method of efficiently operating limited KISTI-GSDC computing resources.

Keywords: Data Center, Efficient Resource Utilization, LDG, Global Service

1 Introduction

The Korea Institute of Science and Technology Information's Global Science Experimental Data Hub Center(KISTI-GSDC) has been supporting domestic and foreign researchers since 2010 by establishing a gravitational wave data analysis computing environment called GSDC-LDG(LIGO Data Grid)[1-4]. GSDC-LDG also supports Laser Interferometer Gravitational-Wave Observatory(LIGO) global services(Tier2) at the request of Korean Gravitational Wave Group(KGWG) in 2019. Recently, with increasing computing demand for gravitational wave research, the

* Corresponding author

demand for Kamioka Gravitational Wave Detector(KAGRA)[5] global services(Tier1) has also been increasing. However, it is impossible to reflect all the requirements in limited resources, and efficient resource utilization is essential as the requirements are satisfied. First, support for efficient computing cores for gravitational wave research should be possible. Second, demand for data continues to grow, but storage resources are limited. A way to solve this is needed. Third, there should be rapid data transmission of KAGRA data. Therefore, this paper studies how to support efficient computing cores in GSDC-LDG, how to use storage effectively for LIGO and KAGRA data storage, and how to test and optimize KAGRA data for rapid data transmission.
The remainder of this paper is organized as follows: Chapter 2 describes the status of GSDC-LDG, identifies, and applies considerations. The paper concludes with Chapter 3.

2 A Study on the GSDC-LDG System for Efficient Resource Utilization

There are currently three considerations for the efficient use of GSDC-LDG resources. First, Bayesian inference calculations of gravitational wave data analysis are embarrassingly parallel[6]. Second, GSDC-LDG must simultaneously accept LIGO and KAGRA data. Third, there should be rapid data transmission of KAGRA data.

First, as described above, Bayesian inference calculations of gravitational wave data analysis are embarrassingly parallel. Therefore, computing resources should be available efficiently. Most modern calculations are performed on the central processing unit(CPU) or graphics processing unit(GPU). The CPU consists of a relatively small number of cores optimized to perform all the tasks required by the computer in series. The GPU, on the other hand, is composed of hundreds or thousands of cores that can be used to compute many numbers at the same time[7]. Therefore, by allocating additional GPU servers in addition to the existing 996 core CPU resources, resources can be used more efficiently than before. In order to apply GPU servers to existing HTCondor[8] pools in GSDC-LDG, the following settings shall be made:

```
use feature : GPUs
GPU_DISCOVERY_EXTRA = -extra
STARTD_ATTRS = $(STARTD_ATTRS), WantGPU, HAS_GPU
START = $(START) && (MY.HAS_GPU =!= True || TARGET.WantGPU =?= True)
```

Second, the demand for data continues to grow, but storage resources are limited. here is a limit to deleting existing data for the persistence of research. The solution to this problem is the application of CernVM File System(CVMFS)[9]. Currently, LIGO can efficiently utilize GSDC-LDG's storage by making frames and software available through CVMFS[10]. The following are the settings for CVMFS that apply to GSDC-LDG:

```
CVMFS_REPOSITORIES="oasis.opensciencegrid.org"
```

```
CVMFS_HTTP_PROXY="http://cms-squid.sdfarm.kr:3128|http://cms-squid2.sdfarm.kr:3128"
CVMFS_QUOTA_LIMIT="42500"
CVMFS_CACHE_BASE="/cvmfs_cache"
```

Third, data generated by KAGRA should be quickly stored in GSDC-LDG and easily accessible to users. For this, KISTI's KREONET[11] was used. We conducted a network test for fast data transfer between GSDC-LDG and ICRR(Institute for Cosmic Ray Research)[12].

Table 1. Network test between GSDC-LDG and ICRR

	GSDC-LDG -> ICRR	ICRR -> GSDC-LDG
Test 1	[root@ldr tmp]# tracepath -n 157.82.231.173 1?: [LOCALHOST] pmtu 1500 1: 134.75.124.2 0.287ms 1: 134.75.124.2 0.342ms 2: no reply 3: 203.250.102.5 0.353ms asymm 2 4: 134.75.105.177 0.550ms asymm 3 5: 134.75.105.241 0.380ms asymm 4 6: 134.75.105.82 110.202ms asymm 5 7: 207.231.240.8 110.263ms asymm 6 8: 162.252.70.82 113.963ms asymm 7 9: 162.252.70.85 127.325ms asymm 8 10: 162.252.70.71 134.235ms asymm 9 11: 150.99.199.93 158.088ms asymm 9 12: 150.99.92.4 158.209ms asymm 8 13: 150.99.91.222 158.167ms asymm 9 14: 150.99.190.98 158.642ms asymm 10 15: 133.11.127.94 158.592ms asymm 11 16: no reply 17: 157.82.254.251 160.029ms asymm 13 18: 157.82.231.173 160.310ms reached Resume: pmtu 1500 hops 18 back 51	[tier-1.kisti@pegasus-01full]$ tracepath -n 134.75.124.244 1?: [LOCALHOST] pmtu 1500 1: 157.82.231.161 0.201ms 1: 157.82.231.161 0.146ms 2: 157.82.254.254 19.677ms pmtu 9140 2?: [LOCALHOST] pmtu 1500 2: 157.82.254.254 16.991ms pmtu 9140 3?: [LOCALHOST] pmtu 1500 3: 133.11.130.117 2.297ms 3: 133.11.130.117 2.677ms 4: 133.11.127.93 3.165ms 5: 150.99.190.97 2.623ms asymm 7 6: 150.99.91.223 13.801ms asymm 8 7: 150.99.89.255 2.679ms asymm 9 8: 202.179.241.77 55.813ms asymm 10 9: 202.179.241.110 53.094ms asymm 11 10: 134.75.105.17 161.160ms asymm 15 11: 134.75.105.242 160.362ms asymm 16 12: 134.75.105.178 160.340ms asymm 17 13: 203.250.102.6 160.522ms asymm 18 14: 134.75.124.244 160.348ms !H Resume: pmtu 1500
Test2	[root@ldr archive]# tracepath -n 157.82.231.173 1?: [LOCALHOST] pmtu 1500 1: 134.75.124.2 27.564ms 1: 134.75.124.2 7.850ms 2: no reply 3: 203.250.102.5 0.359ms asymm 2 4: 134.75.105.177 0.466ms asymm 3 5: 134.75.105.241 0.484ms asymm 4 6: 134.75.105.18 34.148ms asymm 5 7: 203.181.248.201 34.268ms asymm 6 8: 203.181.194.178 84.459ms asymm 7 9: 203.181.249.93 87.842ms asymm 8 10: 203.178.137.193 84.467ms asymm 9 11: 203.178.141.141 84.454ms asymm 10 12: 203.178.136.102 86.352ms asymm 11 13: 133.11.125.201 84.888ms asymm 10 14: 133.11.127.94 84.950ms asymm 11 15: no reply 16: 157.82.254.251 86.547ms asymm 13 17: 157.82.231.173 86.552ms reached Resume: pmtu 1500 hops 17 back 51	[tier-1.kisti@pegasus-01 ~]$ tracepath -n 134.75.124.244 1?: [LOCALHOST] pmtu 1500 1: 157.82.231.161 0.199ms 1: 157.82.231.161 0.148ms 2: 157.82.254.254 11.624ms pmtu 9140 2?: [LOCALHOST] pmtu 1500 2: 157.82.254.254 18.815ms pmtu 9140 3?: [LOCALHOST] pmtu 1500 3: 133.11.130.117 2.102ms 3: 133.11.130.117 2.043ms 4: 133.11.127.93 5.619ms 5: 150.99.190.97 2.612ms asymm 7 6: 150.99.91.223 2.800ms asymm 8 7: 150.99.89.255 3.716ms asymm 9 8: 202.179.241.77 53.248ms asymm 10 9: 202.179.241.110 53.058ms asymm 11 10: 134.75.105.17 86.417ms asymm 12 11: 134.75.105.242 86.532ms asymm 13 12: 134.75.105.178 86.523ms asymm 14 13: 203.250.102.6 88.144ms asymm 15 14: 134.75.124.244 86.590ms !H Resume: pmtu 1500

We can confirm through test1 that the data is sent to the TEIN[13] route. It was confirmed that there was a problem with the route as it was via the United States, and the speed was improved by setting the correct route through KREONET. This is a route via the United States, which identified the problems with this route and changed it to

the best route through KREONET(Test2). Currently, it has been set as the best route via Hong Kong (KREONET2 and APAN-JP).

3 Conclusion

The emergence of large-scale research facilities and research equipment has brought challenging and large-scale computational and data processing requirements. KISTI-GSDC has been providing GSDC-LDG since 2010 as a data center for such large-scale computation and data processing. In addition, as KAGRA is fully operational, the demand for KISTI-GSDC resources (CPU, storage, network, etc.) is also increasing. However, it is impossible to reflect all the requirements in limited resources, and efficient resource management is essential as the requirements are satisfied. In this study, GSDC-LDG's GPU, CVMFS, and network path optimization measures were proposed and applied to support efficient computing resources for gravitational wave research.

Acknowledgments. This study was supported by the National Research Foundation of Korea (NRF) through contract N-20-NM-CR01-S01 and the Program of Construction and Operation for Large-scale Science Data Center (K-20-L02-C03- S01).

References

1. S U Ahn, A Jaikar, B Kong, I Yeo, S Bae and J Kim.: Experience on HTCondor batch system for HEP and other research fields at KISTI-GSDC. Journal of Physics: Conference Series, Conf. Series 182938 (2017)
2. Korea Institute of Science and Technology Information, https://www.en.kisti.re.kr/
3. LIGO, https://www.ligo.caltech.edu/.
4. Sangwook Bae, Geonmo Ryu, Seo-young Noh and Heejun Yoon.: Study of LIGO Tier-2 System with Priorities. The 7th International Conference on BIGDAS2019 (2019)
5. KAGRA, https://gwcenter.icrr.u-tßokyo.ac.jp/en/
6. C. Talbot, R. Smith, E. Thrane, G. Poole.: Parallelized Inference for Gravitational-Wave Astronomy. Phys. Rev. D, 100, 043030, arXiv:1904.02863 (2019)
7. Masafumi Niwano, Katsuhiro L. Murata, Ryo Adachi et al.: GPU-accelerated Image Reduction Pipeline. Publications of the Astronomical Society of Japan, psaa091, 06 October (2020)
8. HTCondor, https://research.cs.wisc.edu/htcondor/
9. CVMFS, https://cernvm.cern.ch/fs/
10. P Paschos, B Riedel, M Rynge et al.: Distributed Computing Software and Data Access Patterns in OSG Midscale Collaborations. 10.13140/RG.2.2.24836.48001. (2020)
11. KREONET, https://www.kreonet.net/eng/
12. ICRR, https://www.icrr.u-tokyo.ac.jp/en/
13. TEIN, https://www.tein.asia/

Mark Detection of Various Size Using YOLO

Gijin Hong[1], YoungBong Kim[1],

[1] Dept. of IT Convergence and Application Engineering, Pukyong National University, 48513 Busan, Rep. of Korea
{gjhong, ybkim}@pknu.ac.kr

Abstract. In this research, we will try to detect the crosshair mark at corner positions of each OLED substrate in order to separate individual OLEDs one by one from a very large OLED etching plate. To get the corner points of each OLED substrate, we employ a machine learning-based YOLO algorithm that has excellent performance in object detection. For giving an enough training set, a total of 1000 data sets are obtained from the etching plate, and in the test, the resolution was increased to facilitate detection of small objects. Through a result of experiments with various sized images, meaningful results were shown in the detection rate.

Keywords: object detection, crosshairs, various sizes, YOLO

1 Introduction

Since the OLED substrate itself stretches or bends differently from the existing LCD substrate, there is a possibility that the OLED has a slightly different shape rather than the same rectangle. Therefore, in order to cut out one OLED from the OLED substrate, it is necessary to accurately detect the deformed location through the crosshairs on the four corners. However, since there are many cases where foreign substances are buried on the OLED substrate through various processes, the image of the mark is quite often inaccurate. In this study, we intend to accurately detect the crosshairs at the corner points of the OLED substrate even in the presence of foreign substances.

Many researches related to conventional object detection that recognizes marks are done by using a classifier or localizer to split the image into multiple smaller images to detect the object to be found and then compare it to the original object [1], [2]. However, the YOLO system overrides the object detection with single regression problem that applies a single neural network to the entire image and simultaneously predicts the bounding boxes and class probabilities from the image pixels [3].

In this research, we try to develop an algorithm that accurately detects a given mark in an image covering many pollutants by using YOLO, a group of artificial intelligence libraries, which are recently becoming important issues. In addition, since the OLED substrate has a very large size, a study was attempted on a method to make this system well applied to an oversized image.

2 Research Method and Experimental Results

The YOLO system stands for 'You Only Look Once' and is an algorithm released at CVPR2016. The YOLO system divides the input image into S*S cells and the grid cell detects the object if the center of the object exists in the grid cell. Each grid cell predicts C conditional class probabilities. Then, at test time, we multiply the conditional class probabilities by the confidence scores of the bounding box to get the class-specific confidence scores. These scores determine both whether the class exists in the box and how well the box fits the object [4].

When observing the mark on the OLED etching plate, it is covered with various foreign substances as shown in Figure 1, so more than 50% of the entire mark is not clear, making it impossible to distinguish.

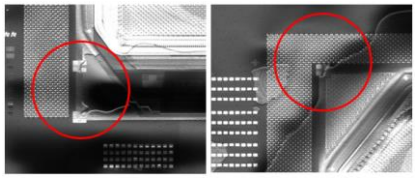


Fig. 1. OLED etching plate contaminated with foreign substances

In addition, we want to detect a bounded crosshair to separate the OLEDs one by one from the very large OLED etching plate of 16384x81920, as shown in Figure 2. For this, we used Tiny YOLO, a light model suitable for predicting a small number of classes, in our experiments. The full version of YOLO uses 24 convolutional layers, whereas the Tiny version uses 9 and applies 6 max pooling [5].

Fig. 2. OLED etching plate and crosshair

We secured a total of 1000 image data sets from the original large etching plate for the collection of data to be used in model training. To improve the test results, we constructed a data set consisting of various image sizes, lights, and backgrounds and also used negative samples without bounding boxes.

In the test, we set the object detection threshold to 0.2 and modified the cfg file to increase the network resolution to improve the precision and facilitate detection of small objects. Figure 3 shows the results of testing using images of different sizes, where we can see that the detection rate drops slightly when the image size is over 1500*1500, but the overall results are meaningful.

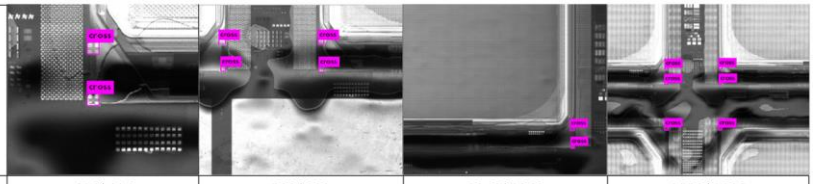
Test image				
Image size	416*416	832*832	1248*1248	1500*1500
Detection rate	94%	96%	95%	80%

Fig. 3. Detected results at different image sizes

3 Conclusion

It has often been observed that the shape of a rectangle is deformed by stretching or bending in the manufacturing process, such as an OLED substrate. To solve this problem, we designed a learning system for the recognition of accurate markers using the YOLO system and conducted learning using more than 1,000 data. As a result, parts that are covered with contaminants and which are difficult to recognize even with human eyes were recognized well, resulting in meaningful research results with a recognition rate of 95% or more.

Acknowledgments. This research was financially supported by the Ministry of Trade, Industry, and Energy (MOTIE), Korea, under the "Regional Innovation Cluster Development Program (R&D, P0004797)" supervised by the Korea Institute for Advancement of Technology (KIAT).

References

1. J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders.: Selective Search for Object Recognition. International journal of computer vision, 104(2):154--171 (2013)
2. R. Girshick, J. Donahue, T. Darrell, and J. Malik.: Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, pp. 580--587. IEEE (2014)
3. Jeseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi: You Only Look Once: Unified, Real-Time Object Detection. CVPR 2016, OpenCV People's Choice Award
4. Joseph Chet Redmon's blog, "YOLO CVPR 2016", https://goo.gl/Xj2Eik
5. Darknet, "Implementation", https://github.com/pjreddie/darknet

Development of Chinese cabbage cultivation strategy using statistical analysis method and machine learning method

Myung Hwan Na[1] , Wanhyun Cho[1], Sooram Kang[1]

[1] Dept. of Statistics, Chonnam National University, Gwang-Ju, South Korea
{nmh, whcho}@chonnam.ac.kr, slkakng2001@gmail.com

Abstract. In this paper, we developed the best cultivation strategies for growth factors that affect the yield of Chinese cabbage grown on field. First, we considered twenty-five environmental factors such as mean temperature, soil surface temperature, mean humidity, soil surface humidity, etc. as explanatory variables, and also the leaf length and leaf widths representing the growth states of cabbage were used as the response variables. Second, we consider three statistical analysis methods such as Correlation Analysis, Canonical Correlation Analysis, Partial Least Square Regression and two machine learning methods such as Support Vector Regression, Random Forest to investigate the relationship between environmental factors and growth variables. Third, through experimentation, it was confirmed that factors related to ground temperature have a close influence on growth of Chinese cabbage. And in terms of direction, daily temperature difference and maximum temperature have a negative relationship with Chinese cabbage. In particular, it was confirmed that proper control of temperature related factors is a very important factor in controlling cabbage production. Finally, the results studied so far are expected to provide an important cultivation strategy for farmers who grow Chinese cabbage in the field.

Keywords: Chinese Cabbage, Cultivation Strategy, Environmental Factors, Growth Factors, Statistical Analysis Method, Machine Learning Method.

1 Introduction

In general, Kimchi is one of Korean's favorite foods. Kimchi is also the best well-being food that contains a variety of nutrients such as vitamins, calcium, minerals, and lactic acid bacteria. Furthermore, Kimchi is used as a material for various dishes such as kimchi stew and potato soup. Therefore, Koreans eat kimchi as their favorite food in all seasons.

Here, Kimchi is made using Chinese cabbage produced in the open field. As winter approaches, the temperature is lowered, making it impossible to produce cabbage. Therefore, Koreans soak kimchi using cabbage harvested from the open field before winter comes. In this case, if a lot of cabbage is produced, the price of cabbage drops and farmers who produce cabbage lose a lot. On the other hand, if a cabbage is produced less, the price of cabbage increases so much that consumers who buy cabbage have to spend a lot of money. Therefore, if an appropriate amount of Chinese

cabbage is produced, both farmers producing cabbage and consumers buying cabbage do not lose equally with each other [1]. At this time, both the farmers who produce cabbage and the officials who manage them are interested in the cultivation strategy to produce a good amount of good cabbage. Therefore, we need a cultivation strategy that can produce high-quality Chinese cabbage at an appropriate level.

To find out this problem, let us first briefly review previous studies on cultivation strategies and yield prediction of Chinese cabbage. Burt and his colleagues [2] have published a paper on the cultivation of cabbage in Western Australia. In this paper, they introduced the western region, climate, soil, and cultivation preparations for Chinese cabbage cultivation. In addition, they considered the problems of re-arrangement, such as the decision of the time of study, the supply of pest management water from sowing to harvest. In the introduction to his book, Balliu [3] described the historical background of Chinese cabbage cultivation, mentioned its influence effect, and described the efficacy of Chinese cabbage. In the following chapters, he introduces the overall contents of cabbage varieties, sowing and cultivation methods, pest management, and production value of cabbage. Laczi and colleagues [4] investigated the effects of various treatments on the growth and yield of Chinese cabbage and the quality of the final harvested product in organic agriculture. In addition, these treatments included the development of a variety of fertilized cabbage types, multiple culture sites, and multiple hybridized varieties. Research has shown that all factors considered have an important influence on the growth characteristics of Chinese cabbage, such as the length and diameter of Chinese cabbage. Kim and colleagues [5] proposed a mathematical modeling of Chinese cabbage and white radish and verification of growth status using a drone capable of capturing red, green, and blue images to measure the biological properties of vegetables. They estimated the plant height using a motion structure algorithm to generate a 3D surface model from the crop canopy data. They also applied a multiple linear regression model for three explanatory variables and four response variables representing fresh weight, leaf length, leaf width and number of leaves.

In this paper, we consider the effects of various environmental factors on the growth factors that determine the growth of cabbage. To do this, we consider three statistical analysis methods such as Correlation Analysis (CA), Canonical Correlation Analysis (CCC), Partial Least Square Regression (PLSR) and two machine leaning methods such as Support Vector Regression (SVR), Random Forest (RF). We conducted an experiment to confirm how environmental variables affect the growth of Chinese cabbage.

2 Dataset and Analysis Method

2.1 Dataset

In general, autumn cabbage, which is used to make winter kimchi in Korea, is sown in mid-August and harvested after the end of October. Therefore, it takes about 80 days from sowing to harvesting cabbage. Figure 1 shows the growth stages from cabbage sowing to harvesting.

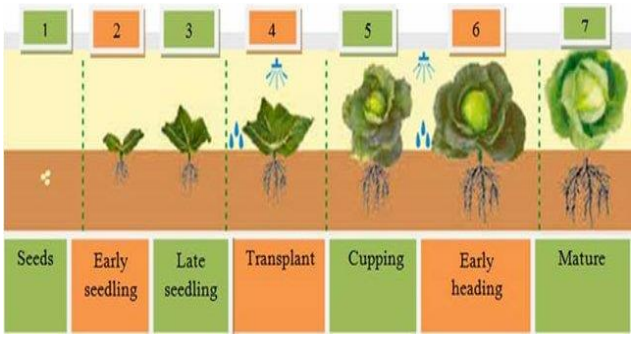


Fig. 1. Process of cultivating cabbage in growth stages

In this study, biological data representing two growth characteristics such as leaf length and leaf width of Chinese cabbage collected from farms in various regions of Korea from September 2019 to December 2019, and twenty-five environmental characteristics observed by the Meteorological Agency during the growth period were used. We used a total of 243 data collected from 8 farms living in Kangwon-do and Chungcheong-do in Korea. Each environmental variable we considered here was observed on a weekly basis and repeated for 9 weeks. Table 1 below shows the number of data collected by region used in the experiment.

Table 1. Number of observations corrected by two regions

Region (farmers)	Kangwon-do (5)	Chungcheong-do (3)	Total (8)
Number of obs.	153	90	243

We used twenty-five environmental factors related with temperature, humidity, ground temperature, ground humidity, solar radiation, etc. as explanatory variables and the growth factors of Chinese cabbage such as leaf length, leaf width as response variables. A description of explanatory variables and response variables used in model is given in Table 2 below.

Table 2. Growth factors and Environmental Factors

Role in model	Variable Name
Growth Factors (Response Variables)	Leaf Length Leaf Width
Environmental Variables (Explanatory Variables)	Mean Temperature Ground Temperature Mean Humidity Ground Humidity

2.2 Analysis Method

Here, we have used three statistical analysis methods such as Correlation Analysis, Canonical Correlation Analysis, Partial Least Square Regression and two machine learning methods such as Support Vector Regression, Random Forest to investigate the relationship between environmental factors and growth variables. We would like to briefly review these methods.

3 Experimental Result

3.1 Scatter plot and correlation analysis

First, in order to visually examine the relationship between growth factors and environmental factors, we considered the scatter plot and conducted correlation analysis. Figure 2 shows the scatter plot between environmental factors and growth factors. From the Figure 2, we can roughly see that among the environmental factors, the average temperature and the ground temperature are negatively correlated with the growth factors, but the average humidity or the average ground humidity are a little positively correlated with the growth factors.

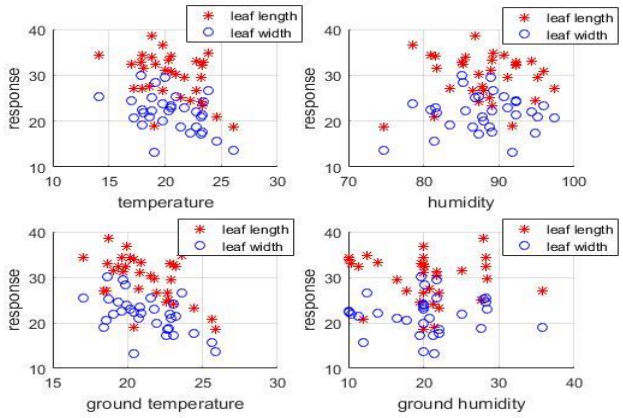


Fig. 2. Scatter plot between environmental factors and growth factors

Next, we performed a correlation analysis to numerically confirm the results obtained above. Table 3 shows the correlation coefficients between environmental factors and growth factors. From Table 3 given, we can numerically confirm the results similar to those given in Figure 3.

Table 3. Correlation matrix between environmental factors and growth factors

(a) Gangwon-do

	Mean Temperature	Mean Humidity	Mean Ground Tem.	Mean Ground Hum.
Leaf length	-0.4123	0.0322	-0.5357	0.1042
Leaf width	-0.4164	0.0868	-0.5283	0.0209

(b) Chungcheong-do

	Mean Temperature	Mean Humidity	Mean Ground Tem.	Mean Ground Hum.
Leaf length	-0.8123	0.2940	-0.8514	0.4510
Leaf width	-0.8230	0.3077	-0.8556	0.3620

3.2 Canonical Correlation Analysis

We conducted a canonical correlation analysis to determine whether there is a relationship between growth factors and environmental factors. Table 4 and 5 show various statistics related to the results of canonical analysis on the growth data of Chinese cabbage. From the results in Table 4, we can see that both regions can sufficiently explain the data given by one canonical correlation variable. First, in the case of Gangwon-do, it can be seen that the leaf length variable among the growth variables has a negative correlation with the ground temperature and groun humidity among the environmental variables, while the leaf width has a positive correlation with these two environmental factors. Second, in the case of Chungcheong-do, both growth variables have a positive correlation with average temperature and average humidity, but on the contrary, it can be seen that these variables have a negative correlation between ground temperature and ground humidity.

Table 4. Canonical Correlation Analysis and its related statistics for growth data

(a) Gangown-do

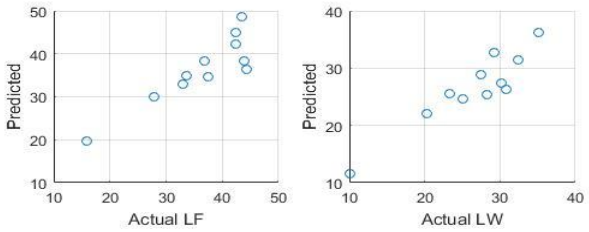
No	Correlation	Eigenvalue	Wilks Statistic	F	Sig.
1	0.875	3.283	0.176	9.353	0.000
2	0.497	0.329	0.752	3.070	0.044

(b) Chungcheong-do

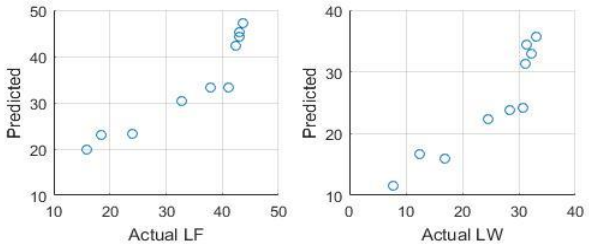
No	Correlation	Eigenvalue	Wilks Statistic	F	Sig.
1	0.715	1.045	0.465	3.037	0.007
2	0.224	0.053	0.950	0.476	0.702

3.3 Partial Least Square Regression

From Figure 3, we can see that there is not much difference between the actual observed values and the predicted values given by the two-component regression model.



(a) Gangwon-do



(b) Chungcheong-do

Fig. 3. Plot for actual values vs predicted values

3.4 Support Vector Regression

SVR model for leaf length increase tuned to 0.1 epsilon, 1 cost, 0.2 gamma by grid search. RMSE was 1.209 and R-square was 0.889. SVR model for leaf width increase tuned to 0.3 epsilon, 100 cost, 0.1 gamma by grid search. RMSE was 1.155, R-square was 0.864. SVR models showed good performance. Figure 4 show the real data and prediction of SVR models. They have trouble with subtle changes, but it is well following the major trend.

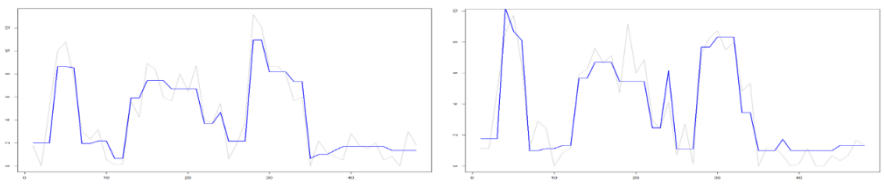


Fig. 4. It is comparing prediction of SVR model and real data. Left image is for leaf length height increase and right image is for leaf width increase.

3.5 Random Forest Analysis

Random Forest model for leaf length increase tuned to 100 ntree and 7 mtree. 89.63% variable was explained. RMSE was 1.247. Random Forest model for leaf width increase tuned to 200 ntree and 18 mtree. 84.67% variable was explained. RMSE was 1.219. The performance of Random Forest models was comparable to that of other techniques. Below figure shows the importance of factors in models.

Figure 5 shows the importance of factors expressed on two scales. Mean Decrease Accuracy(%IncMSE) is determined during the out of bag error calculation phase. The more the accuracy of the random forest decrease due to exclusion of a single variable, the more important that variables are deemed.

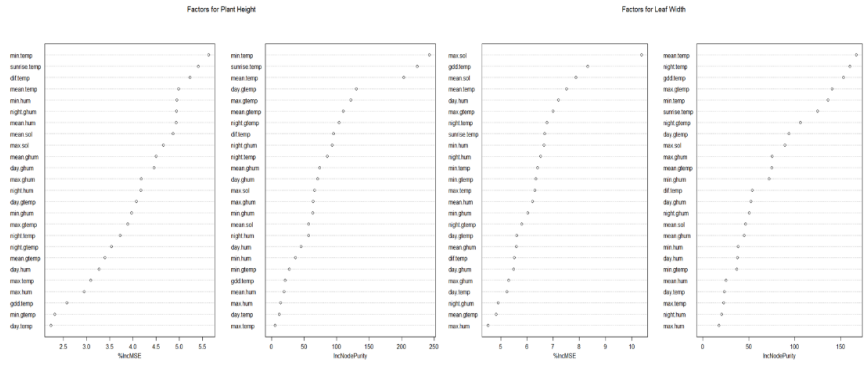


Fig. 5. Two measurement of importance of variables.

Mean Decrease Gini(IncNodePurity) is the total decrease in node impurities from splitting on the variable, averaged over all trees. Fore regression, it is measured by residual sum of squares. Compared to Correlation coefficient and SVR coefficient, there are some differences. In Mean Decrease Accuracy method, the importance of ground temperature is relatively low. Instead, temperature, solar radiation and humidity were important. On the other hand, in the Mean Decrease Gini method, the importance of temperature and ground temperature were high and the solar radiation and humidity were low. We conducted a canonical correlation analysis to determine whether there is a relationship between growth factors and environmental factors.

4 Conclusion

In this paper, several statistical analysis methods and machine learning methods were applied to analyze the relationship between environmental factors and growth factors for Chinese cabbage cultivation in two regions. We performed correlation analysis, canonical correlation analysis, partial least squares regression, support vector regression, random forest analysis to confirm the relationship between factors.

From the results of the five analyzes, we can see that the relationship between the four environmental variables affecting the growth factors of Chinese cabbage is given slightly differently according to the two regions. Temperature and humidity, which affect growth factors, are generally positively correlated in cold regions such as

Gangwon-do, but environmental factors such as temperature and humidity have negative correlations with growth factors in temperate regions such as Chungcheong-do.

In conclusion, the growth of Chinese cabbage can be represented by the leaf length and leaf width, and it can be seen that the effect of temperature and humidity on growth has a moderate rate. However, ground temperature and ground humidity were found to have a great influence on the growth of Chinese cabbage. In particular, it was confirmed that proper control of ground temperature and ground humidity is a very important factor in controlling cabbage cultivation.

Acknowledgments. This work was partially supported by the Research Program of Rural Development Administration (Project No. PJ015361012020), and the Korea National Research Foundation (Project No. 2020R1F1A1067066).

References

1. Capinera, J.L. (2011). " Cabbage Looper, Trichoplusia ni (Hübner) (Insecta: Lepidoptera: Noctuidae)." Featured Creatures EENY-116. Entomology and Nematology Department, University of Florida
2. Burt, J., D. Phillips and D. Gatter (2006). Growing Chinese cabbage in Western Australia. Department of Agriculture. South Perth. Western Australia. Bulletin No. 4673:1-23.
3. Astrit Ballu (201$). In book: Handbook of Vegetables Vol. III, Chapter Cabbage, Publisher: Studium Press, pp.79-120
4. Laczi, E.; Apahidean, A.; Luca, E.; Dumitraş, A.; Boancă, P. Headed Chinese cabbage growth and yield influenced by different manure types in organic farming system. Hort. Sci. 2016, 43, 42–49.
5. Kim, D.W.; Yun, H.; Jeong, S.J.; Kwon, Y.S.; Kim, S.G.; Lee, W.; Kim, H.J. Modeling and testing of growthstatus for Chinese cabbage and white radish with UAV-based RGB imagery. Remote Sens.2018,10, 563.
6. Bobko, P. (2001). Correlation and regression: Applications for industrial organizational psychology and management (2nd ed.). Thousand Oaks, CA: Sage Publications.
7. Kendall, M. G., & Gibbons, J. D. (1990). Rank Correlation Methods (5th ed.). London: Edward Arnold.
8. Härdle, Wolfgang; Simar, Léopold (2007). "Canonical Correlation Analysis". Applied Multivariate Statistical Analysis. pp. 321–330.
9. Knapp, T. R. (1978). "Canonical correlation analysis: A general parametric significance-testing system". Psychological Bulletin. 85 (2): 410–416.
10. Wold, S; Sjöström, M.; Eriksson, L. (2001). "PLS-regression: a basic tool of chemometrics". Chemometrics and Intelligent Laboratory Systems. 58 (2): 109–130
11. Abdi, H. (2010). "Partial least squares regression and projection on latent structure regression (PLS-Regression)". Wiley Interdisciplinary Reviews: Computational Statistics. 2: 97–106.

Prediction of fresh raw weight of onion using spatiotemporal autoregressive moving average (STARMA) model

Myung Hwan Na[1] , Wanhyun Cho[1], YunJeong Kang[1]

[1] Dept. of Statistics, Chonnam National University, Gwang-Ju, South Korea
{nmh, whcho}@chonnam.ac.kr, yooon3220@naver.com

Abstract. In this study, the spatiotemporal autoregressive moving average model was used to predict how the fresh raw weight of onions varies by region and time. First, we have collected the dataset from farms who cultivate onions in three regions, including Jeolla-do, Chungcheong-do, and Gyeonggi-do, and the raw weight of onions was constructed by measuring each region by time difference. Second, the analysis was performed by applying the three-time series models most commonly used to analyze spatiotemporal data: spatiotemporal autoregressive, spatiotemporal moving average and spatiotemporal autoregressive moving average models. Through the analysis experiments, we found that the spatiotemporal autoregressive model and the spatiotemporal autoregressive moving average model can predict the fresh bulb weight of onions relatively accurately, but the spatiotemporal moving average model shows a slight difference in predicting the fresh bulb weight of onions.

Keywords: Onions, Raw weight, Spatiotemporal autoregressive moving average model, Maximum likelihood estimation, Forecast

1 Introduction

In general, onions grown in the open field are greatly affected by various environmental factors such as temperature, humidity, rainfall and sunlight. In particular, when it rains a lot like this year, the growth of onions is poor and the yield is very low. Therefore, the supply of onions cannot keep up with the demand, leading to a large increase in onion prices.

For this reason, farmhouses that grow onions and agricultural organizations that manage them frequently check the growth status of onions from transplanting to harvesting in order to produce an appropriate amount of onions. In addition, through these activities, it is possible to check the growth status of onions, and furthermore, to determine how good onions can be produced at harvest time.

Therefore, in this study, the growth status of onions is confirmed by measuring the weight of fresh bulbs representing the growth status of onions at regular intervals during the cultivation period. Also, based on this, we intend to provide a method to predict the yield of onions produced at harvest time by applying three spatiotemporal time series models.

2 Dataset and Analysis Method

2.1 Dataset

The data used for the analysis were collected from 18 farms growing onions in three regions, including Jeonnam, Gyeongbuk, and Gyeongnam. The data were collected by measuring the fresh bulb weight of onions at 6 lags every week from mid-April, 2020 to early June, 2020. The distribution of observation data corrected from regions is given in Table 1 below.

Table 1. Number of data collected by regions.

Province Region	County Region	Number of farmhouse
Gyeongnam	Changryeong	2
	Hamyang	1
	Hapcheon	1
Gyeongbuk	Gimcheon	2
Jeonnam	Muan	4
	Sinan	3
	Hampyeong	4
	Haenam	1

2.2 Analysis Method

Three models were considered the spatio-temporal model used in the analysis of the measured data as follows. Let's consider briefly about them.
First, the spatial temporal autoregressive moving average (STARMA) model is expressed for measurements representing the fresh bulb weights $Z_i(t)$ of onions observed at $i(i = 1, \cdots, N)$ fixed spatial locations on $t(t = 1, \cdots, T)$ time periods. Also, the N spatial locations can be the geological location where onions are grown. Furthermore, the spatial correlation between N locations is represented by a weight matrix W of size $(\mathrm{N} \times \mathrm{N})$.
Here, the spatiotemporal autoregressive moving average model is represented by STARMA $(p, \alpha_k, q, \beta_k)$, which is expressed by the following matrix equation.

$$\mathrm{Z}(t) = \sum_{k=1}^{p} \sum_{l=0}^{\alpha_k} \phi_{kl} W^{(l)} \boldsymbol{Z}(t-k) - \sum_{k=1}^{q} \sum_{l=0}^{\beta_k} \theta_{kl} W^{(l)} \boldsymbol{\varepsilon}(t-k) + \boldsymbol{\varepsilon}(\mathrm{t}) \qquad (1)$$

where $\boldsymbol{Z}(t) = (Z_1(t), \cdots, Z_N(t))^T$ is the $(N \times 1)$ vector representing the fresh bulb weights of the onions observed at the time point $t(t = 1, \cdots, T)$, p is the order of autoregressive, and q is the order of the moving average. And α_k is the spatial order in the autoregressive term, and β_k is the spatial order in the moving average term. In addition, ϕ_{kl} is the parameter of the autoregressive model in the temporal lag k and

spatial lag l , and likewise θ_{kl} is the parameter of the moving average model in the temporal lag k and spatial leg l. The matrix $W^{(l)}$ is a spatial weight matrix of the $(N \times N)$ size of the spatial order l defined above. The main diagonal elements of the matrix $W^{(l)}$ are zero and the non-diagonal elements indicate the relationship between two different points. Finally, random error vector $\boldsymbol{\varepsilon}(\mathrm{t})$ follows a multivariate normal distribution with a mean vector and a covariance matrix given as

$$\mathbf{E}\big(\boldsymbol{\varepsilon}(\mathrm{t})\big) = \mathbf{0}, \quad \boldsymbol{E}(\boldsymbol{\varepsilon}(t)\boldsymbol{\varepsilon}(t+s)^T) = \begin{cases} \sigma^2 I_N & s = 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Second, as methods of estimating the parameters of the spatiotemporal autoregressive moving average models, the momentum estimation method, least squares estimation method, maximum likelihood estimation method, and nonlinear estimation method are widely used. Here, since the nonlinear least squares method and the most likelihood estimation method are generally the same, we consider a method of estimating the required parameters using the nonlinear least squares method.
Here, given the observed value $\boldsymbol{Z}(t)$ and the model parameter $\boldsymbol{\Phi} = (\boldsymbol{\phi}, \boldsymbol{\theta})$ to be estimated, the difference between the observed value and the model value including parameters is the error. This is given by:

$$\begin{aligned} \boldsymbol{r}(\boldsymbol{\Phi}; t)^T &= (r_1(\boldsymbol{\Phi}; t), \cdots, r_N(\boldsymbol{\Phi}; t))^T \\ &= (\boldsymbol{Z}(t) - f(\boldsymbol{\Phi}, \boldsymbol{Z}(t-1), \cdots, \boldsymbol{Z}(t-k)) \end{aligned} \tag{3}$$

At this time, the method of determining the model parameter that minimizes the sum of squares of errors is called the least squares method.

$$\begin{aligned} \boldsymbol{\Phi}^* &= \underset{\boldsymbol{\Phi}}{\operatorname{argmin}} \textstyle\sum_{t=1}^{T} \boldsymbol{r}(\boldsymbol{\Phi}; t)\boldsymbol{r}(\boldsymbol{\Phi}; t)^{\boldsymbol{T}} \\ &= \underset{\boldsymbol{\Phi}}{\operatorname{argmin}} \textstyle\sum_{t=1}^{T} \sum_{i=1}^{N} r_i^2(\boldsymbol{\Phi}; t) \end{aligned} \tag{4}$$

Here we will use the Levenberg-Marquardt method to find the least square estimators. This method starts from an appropriate initial value $\boldsymbol{\Phi}_0$ of model parameters $\boldsymbol{\Phi}$ and searches for a solution by repeating the update until the model parameters $\boldsymbol{\Phi}$ converges according to the following equation.

$$\boldsymbol{\Phi}_{k+1} = \boldsymbol{\Phi}_k - (\boldsymbol{J}_{\boldsymbol{r}}^T \boldsymbol{J}_{\boldsymbol{r}} + \mu_k \mathrm{Diag}(\boldsymbol{J}_{\boldsymbol{r}}^T \boldsymbol{J}_{\boldsymbol{r}}))^{-1} \boldsymbol{J}_{\boldsymbol{r}}^T \boldsymbol{r}(\boldsymbol{\Phi}_k; t)^T, k = 1,2, \cdots \tag{5}$$

where μ_k is the damping factor, and $\boldsymbol{J}_{\boldsymbol{r}}$ denotes the Jacobian matrix for $\boldsymbol{r}(\boldsymbol{\Phi}; t)$ given as

$$\boldsymbol{J}_{\boldsymbol{r}} = \begin{bmatrix} \frac{\partial r_1(\boldsymbol{\Phi}; t)}{\partial \phi_{10}} & \cdots & \frac{\partial r_1(\boldsymbol{\Phi}; t)}{\partial \theta_{q\beta_q}} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_N(\boldsymbol{\Phi}; t)}{\partial \phi_{10}} & \cdots & \frac{\partial r_N(\boldsymbol{\Phi}; t)}{\partial \theta_{q\beta_q}} \end{bmatrix} \tag{6}$$

Third, we will consider the problem of predicting or forecasting how much raw bulb weight of onions will be given at a future point in time using past observational values and estimated models. Using the time series data observed up to the current time point t, the minimum mean square error predicted value $\hat{Z}_{t+l}$ of the future value Z_{t+l} at the future time point $t+l$ is defined as follows.

$$\hat{Z}_{t+l} = E(Z_{t+l} |\ z_t, z_{t-1}, \cdots, z_1) \tag{7}$$

At this case, using the recursive equation of the conditional expectation and the STARMA (1,1,1,1) model, the optimal predicted values in ahead of one step and two steps are given as follows.

$$\begin{aligned} \hat{Z}_{t+1} &= E(Z_{t+1} |\ z_t, z_{t-1}, \cdots, z_1, \boldsymbol{\Phi}) \\ &= \hat{\phi}_{10} z_t + \hat{\phi}_{11} W^{(1)} z_t - \hat{\theta}_{10} \varepsilon_t - \hat{\theta}_{11} W^{(1)} \epsilon_t \ , \end{aligned} \tag{8}$$

and

$$\begin{aligned} \hat{Z}_{t+2} &= E(Z_{t+2} |\ z_t, z_{t-1}, \cdots, z_1, \boldsymbol{\Phi}) \\ &= E[E(Z_{t+1} |\ z_t, z_{t-1}, \cdots, z_1, \boldsymbol{\Phi}), z_t, z_{t-1}, \cdots, z_1, \boldsymbol{\Phi}] \\ &= \hat{\phi}_{10} \hat{z}_{t+1} + \hat{\phi}_{11} W^{(1)} \hat{z}_{t+1} - \hat{\theta}_{10} \hat{\varepsilon}_{t+1} - \hat{\theta}_{11} W^{(1)} \hat{\varepsilon}_{t+1}. \end{aligned} \tag{9}$$

Finally, as a special case of the space-time autoregressive moving average (STARMA) model, the following two models, the space-time autoregressive (STAR) model and the space-time moving average (STMA) model, can be obtained.

In the space-time autoregressive moving average model given above, if the moving average order is set to zero, the following space-time autoregressive model (STAR) is obtained.

$$\boldsymbol{Z}(t) = \sum_{k=1}^{p} \sum_{l=0}^{\alpha_k} \phi_{kl} W^{(l)} \boldsymbol{Z}\,(t-k) + \ \boldsymbol{\varepsilon}(\mathrm{t}) \tag{10}$$

In the mode above, k symbolizes time leg, l symbolizes lag. Therefore, if p is temporal and α is spatial order,this model is a STAR(p, α) process.

Similarly, in the space-time autoregressive moving average model given above, if the autoregressive order is set to zero, the following space-time moving average (STMA) is obtained.

$$\boldsymbol{Z}(t) = \boldsymbol{\varepsilon}(\mathrm{t}) \ - \ \sum_{k=1}^{q} \sum_{l=0}^{\beta_k} \theta_{kl} W^{(l)} \boldsymbol{\varepsilon}\,(t-k) \tag{11}$$

As above, if q is temporal and β is spatial order, this model is a STMA(q, β) process.

3 Experimental Result

3.1 Performance Comparison of Three Spatial-Temporal Models

The average RMSE of the STAR, STMA and STARMA models using each spatial weighting matrix is shown in Table 2 below. From the results of this table, it can be seen that the distance-based matrix for the STAR model, K-NN(3) for the STMA model, and the spatial weighting matrix for setting the threshold for the STARMA model are most appropriate.

Table 2. RMSE of three Spatiotemporal Models according to Model-Specific Weight Matrix

Weight Matrix	RMSE		
	STAR(1,1)	STMA(1,1)	STARMA(1,1,1,1)
Inverse Distance	**2068.678**	19308.25	2104.729
Limit Inverse Distance	2098.012	18653.92	**2101.385**
K-Nearest(2)	2111.245	17947.53	2103.656
K-Nearest(3)	2117.611	**17881.76**	2124.183

3.2 Prediction of Fresh Bulb Weight of Onion Using Three Models

The estimated coefficients fitted using appropriate spatial weighting matrices for each model and their significance test results are shown in the table below. From the results of Table 3 given, only the coefficient ϕ_{10} is significant in the STAR(1,1) model, only the coefficient θ_{10} is significant in the STMA(1,1) model, and only the coefficient $\phi_{10}$is significant in the STARMA(1,1,1,1) model.

Table 3 Estimation and test results of coefficients according to spatial weighting matrix

Models	ESTIMATE	STD.ERROR	P-VALUE
STAR(1,1)	$\phi_{10} = 1.073$	0.131	0.000^{***}
	$\phi_{11} = 0.204$	0.134	0.13
STMA(1,1)	$\theta_{10} = 0.945$	0.384	0.015^{*}
	$\theta_{11} = 0.324$	0.401	0.421
STARMA(1,1,1,1)	$\phi_{10} = 1.069$	0.180	0.000^{***}
	$\phi_{10} = 0.187$	0.186	0.317
	$\theta_{10} = -0.001$	0.356	0.998
	$\theta_{11} = 0.032$	0.422	0.94

Using the estimated coefficients, the result of comparing the predicted value suitable for each region and the actual weight of onion greens is as shown in the following figure. From these three figures, we can see that in all three regions, the spatiotemporal autoregressive model or the spatiotemporal autoregressive moving average model predict the fresh weight of onions relatively well, but the spatiotemporal moving average model has a lot of errors between the predicted values and the actual observed values.

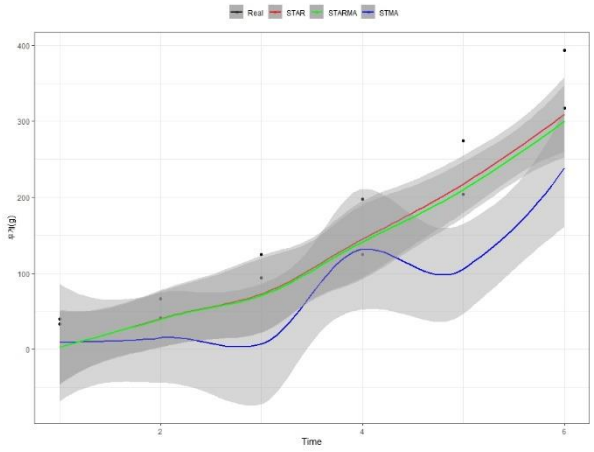


Fig. 1. Predicted results of Youngman area: Green, Red, Blue lines mean STARMA, STAR, STMA.

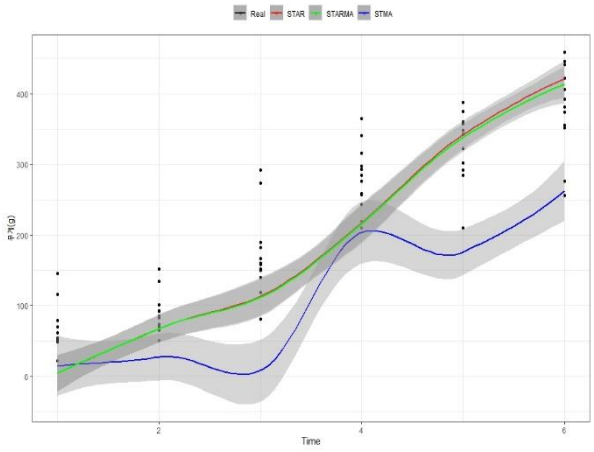


Fig. 2. Predicted results of Gyeongbuk: Green, Red, Blue lines mean STARMA, STAR, STMA

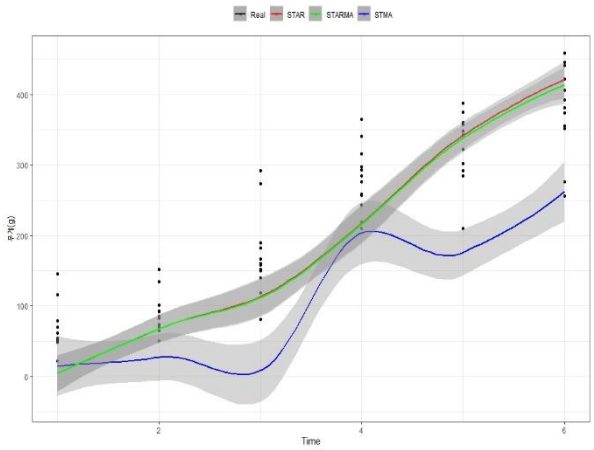


Fig. 3. Predicted results of Jeonnam: Green, Red, Blue lines mean STARMA, STAR, STMA

4 Conclusion

In this paper, we tried to fit the weight of onions in three models: a spatiotemporal autoregressive model, a spatiotemporal moving average model, and a spatiotemporal autoregressive moving average model for prediction.

According to the results that were adapted for each region, the spatiotemporal autoregressive model and the spatiotemporal autoregressive moving average model did not show a significant difference from the actual raw weight, but the spatiotemporal moving average model showed a significant difference from the actual raw weight.

In the future research direction, we plan to research and develop a method to find out how various environmental factors affect the raw weight of onions.

Acknowledgments. This work was partially supported by the Research Program of Rural Development Administration (Project No. PJ015361012020), and the Korea National Research Foundation (Project No. 2020R1F1A1067066).

References

1. Rathod, S., Gurung, B., Singh, K. N., Ray, M.: An improved Space-Time Autoregressive Moving Average (STARMA) model for modelling and Forecasting of Spatio-Temporal time-series data. Journal of the Indian Society of Agricultural Statistics, vol. 72, no. 3, 239-253 (2018)
2. Ma, C.: Spatial autoregression and related spatio-temporal models. Journal of Multivariate Analysis, vol. 88, 152-162(2004)

3. Kurt, S., Tunay, K. B.: STARMA Models Estimation with Kalman Filter: The Case of Regional Bank Deposits. Procedia – Social and Behavioral Sciences, vol. 195, 2537-2547(2015)
4. Park, M. S., Heo, T.-Y.: Seasonal Spatial-temporal Model for Rainfall Data of South Korea. Journal of Applied Science Research, vol. 5, bo. 5, 565-572(2009)
5. Sigrist, F., Kunsch, H. R., Stahel, W. A.: An autoregressive spato-temporal precipitation model, Procedia Environmental Sciences, vol. 3, 2-7(2011)
6. Yao, Q., Brockwell, P.J.: Gaussian maximum likelihood estimation for ARMA models II: Spatial processes. Bernolli, vol. 12, no. 3, 403-429(2006)

Top-k Keyword Extraction from News Articles: The Case of Pork in South Korea

Yifan Zhu[1], Tserenpurev Chuluunsaikhan[2], Kwanhee Yoo[2], HyungChul Rah[3], Aziz Nasridinov[2]

[1] Department of Big Data, Chungbuk National University, Cheongju 28644, South Korea
[2] Department of Computer Science, Chungbuk National University, Cheongju 28644, South Korea
[3] Department of Management Information System, Chungbuk National University, Cheongju 28644, South Korea

Abstract. In recent years, the pork market has been heavily influenced by a variety of important events. Such as hand-foot and mouth disease, African swine fever, and so on. Many factors affect the pork price. The research topic of this paper is to analyze the correlation between keywords and news topics, and top-k keywords are extracted to pave the way for the achievement of accurate price prediction in the future. We need to analyze the news to understand whether the news impacts the pork price. Considering that the news is non-structural, we use Topic Modeling to process the data and then use term frequency–inverse document frequency (TF-IDF) to convert it into structural data. We will finally display various kinds of price curves and tables of keywords and importance (TF-IDF value) through the web page we developed.

Keywords: news, keywords, TF-IDF, top-k

1. Introduction

People's consumption habits in a country are determined by their culture, income, and religious beliefs. In South Korea, pork is the most popular meat among all kinds of meat products because of its low price and good taste, which leads to the popularity of South Korea barbecue restaurants. South Korea's annual per capita consumption of pork is 30.6kg. South Korea also ranked second in the world in 2018. World per capita annual pork consumption is 12.3kg, Organization for Economic Cooperation and Development (OECD) per capita consumption is 23.0kg. South Korea's annual per capita consumption of pork is 2.4 times the world average and 1.3 times the OECD average. By these data, we can conclude that stability of the pork consumption market is very important for South Korean and the stability of the meat consumption market.

It is essential to know the pork price trend in advance if we want to keep the pork consumption market stable. It is not easy for the average person to predict price in advance in our daily lives, but some critical news can help the average person predict how price will change. Positive news, such as increased government investment in the pig industry and a study that eating pork is good for your health, will promote pork consumption. Moreover, bad news like African swine fever, hand-foot and mouth disease, and swine flu can reduce people's pork consumption. Because when these pig disease events occurred, consumer can get the news in a short time by TV or media tweets and reduce their consumption about pork. Hence, we can conclude that the impact of news on pork price is significant.

This paper focuses on achieving two goals. The first goal is to analyze the correlation between keywords and news topics and calculate the importance of all keywords. By observing the importance of all keywords, we can judge the news topic in a particular period. And we also can use calculated TF-IDF value(importance) and pork price to create a neural network model to predict prices. For example, we can establish Long short-term memory (LSTM) model to do price prediction, this model is very useful for time series data. The second goal is to extract the top-k keywords from the group of keywords and display them on the web page. The reason why we extract top-k keywords is that when news is too much, there will be too many keywords, which will bother users to get valuable information. And in the future, we will classify the emotions of keywords into positive and negative. Top-k keywords may be the words with the greatest positive or negative impact, based on which we can make price prediction more accurate.

2. Related Study

In recent years, more and more scholars are aware of the impact of news on the prices. Many papers that achieve price prediction based on news headlines or content have been published. And Many companies have put them into the application to solve some specific problems. However, many research results have some limitations, such as low prediction accuracy and limited application scene. Next, I will briefly introduce three papers about stock price prediction based on news and price. These three papers have established three different models and set price and news as input. After they finish training the models, they all got acceptable results.

Mohan et al. [3] collected five years' daily stock prices and 265,000 articles about S&P500 companies' financial news. The authors input financial news and prices into the recurrent neural network (RNN) model and uses cloud computing to train the RNN model. Finally, the trained model is used to predict future stock prices, and the prediction results show that the stock prices correlate with financial news.

Liu et al. [2] collected daily stock prices for S&P500 companies and news headlines for the same period on Yahoo Finance. The authors used a joint model as a price prediction model. A joint model that including translating embeddings models that used to study characteristics and convolutional neural network (CNN) that used to extract valuable

information from financial news articles. This method solves the problem of text sparsity in feature extraction. Moreover, the result proved that this model is better than the traditional machine learning model in a short time prediction. It also can offer some help to investors.

Liu et al. [1] established the hierarchical complementary attention network (HCAN) model to extract the headlines and content of valuable complementary information to predict the stock prices. They adopted a two-level attention mechanism to quantify the importance of the words and sentences in given news. Moreover, a novel measurement is designed to calculate the attention weights to avoid capturing redundant information in headlines and contents. The result shows that their model also achieves an excellent price prediction.

In this paper, we focus on the analysis of keywords in the news. We will analyze the correlation between news keywords and news topics. Unlike the previous paper, we do not establish a model for price prediction. We intend to get news topics and extract and quantify keywords in the news first. Then we can infer price trend by observing these keywords and topics. As for price prediction model and keywords emotion analysis will be our future work.

3. Proposed Method

This section will introduce the methods proposed in the paper. We will briefly introduce the principle of the methods and explain why we use them. After that, we will use tables to show the results got by these methods and evaluate the results. First, we will show an overview of this section. Then, we will introduce proposed methods and show the results according to the order of the workflow in Figure 1.

3.1 Overview of proposed method

As shown in Figure 1, the web crawler is used to collect news from the Pigtimes(http://www.pigtimes.co.kr/) website. The collected data is then processed. The processing process is generally divided into two steps. First is to use the Topic Modeling preprocessing method to pre-process the collected news into a bag of words and second is to train the decomposed data through the latent dirichlet allocation (LDA) model to generate new articles that are most similar with the raw news. After we finished Topic Modeling, we use the TF-IDF formula to calculate the TF-IDF value(importance) of each keyword in the topics, representing the correlation between this keyword and the news topic. In the last step, we use the top-k function to filter the k keywords with the highest TF-IDF value in topics, which are the keywords with the highest correlation with the topics. These are all for this section.

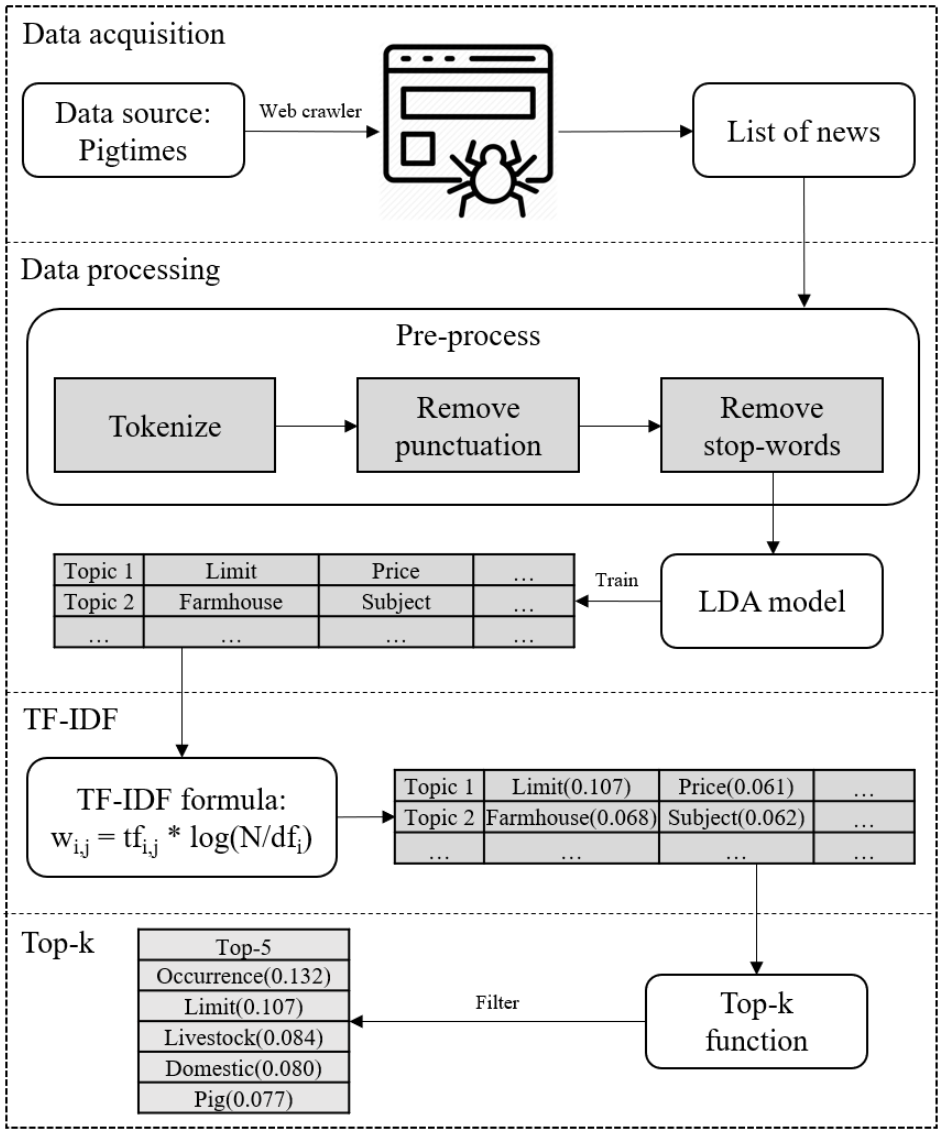


Figure 1. Workflow of proposed method

3.2 News collected by web crawler

We use web crawler for collecting news dataset from Pigtimes pages. Since this paper's research object is pig, we need to ensure that the dataset is closely related to the research object. The news published in Pigtimes is only related to the pig and pork market, so we use web crawler to get the news from 2010 to 2019 on the Pigtimes website.

In web crawling, we load web pages by the URL of the web page and then convert the web page's content into XML or HTML format and stored them in the mongo database. This paper uses python Requests and Beautifulsoup packages to collect XML and HTML files of Pigtimes.

3.3 Keywords' importance calculated by TF-IDF

TF-IDF is a statistical measure used to calculate a word's correlation with a particular article in all articles. We can get the TF-IDF value of a word by multiplying the number of times the word appears in the text and the reciprocal of the number of articles in which the word exists (taking the logarithm). The larger the TF-IDF value is, the higher the correlation between this word and a particular article. We can generally understand the content of the article by observing keywords and topics.

- Information retrieval:
  TF-IDF can be used to retrieve information that is most correlate with what you are searching for.
- Keyword extraction:
  TF-IDF can also be used for keyword extraction. The word with the highest IF-IDF value is the word with the highest correlation with the article so that that word can be considered as one of the keywords of the article.

Table 1. Parts of 2019.Jan each topic's keywords and their importance

2019.Jan ---Keywords importance						
Topic 1	Limit(0.107)	Price(0.061)	Import(0.054)	Delivery(0.045)	Region(0.043)	···
Topic 2	Farmhouse(0.068)	Subject(0.062)	Livestock(0.058)	Execution(0.056)	Import(0.055)	···
Topic 3	Numbers(0.074)	Limit(0.062)	Rise(0.054)	Consumption(0.048)	Breed(0.045)	···
Topic 4	Occurrence(0.132)	Domestic(0.080)	Market(0.065)	Import(0.051)	Pass(0.042)	···
Topic 5	Livestock(0.084)	Association(0.056)	Pig(0.056)	Limit(0.052)	Management(0.052)	···
Topic 6	Pig(0.077)	Consumption(0.045)	Breed(0.043)	Market(0.042)	Import(0.038)	···

Table 1 shows the TF-IDF value of the news keywords in January 2019. The left 1,2,3,4,5 and 6 represent six news topics, and the right side consists of the keywords that belong to some topics and the TF-IDF value (importance) of these keywords. In the following section, we will use the top-k to display the chart. Because so many keywords and TF-IDF values are displayed in front of the eyes, it is difficult for people to have a general understanding of the news in January 2019 in a short time.

3.4 Filter top-k keywords from group of keywords

Top-k is the k highest TF-IDF value keywords. In the previous section, we calculated the TF-IDF values of all words in a particular article. In this section, the keywords will be sorted by value size. When the soring work is over, we will extract the k highest keywords to achieve top-k keywords.

We choose top-k because when a piece of news appears, we can independently select the TF-IDF value with the k highest TF-IDF value and then judge the topic of the news by observing these keywords. Because we may not be able to judge an article's topic immediately by a lot of keywords, the k here means that we can choose how many keywords to observe within a reasonable range. When we cannot understand an article's topic by the 5 keywords with the highest TF-IDF value, we can judge by the 10 or 15 keywords with the highest TF-IDF value.

Table 2. Filter out the top-k keywords from all topics in 2019 Jan.

2019.Jan ---Keywords importance						
Topic 1	Limit(0.107)	Price(0.061)	Import(0.054)	Delivery(0.045)	Region(0.043)	···
Topic 2	Farmhouse(0.068)	Subject(0.062)	Livestock(0.058)	Execution(0.056)	Import(0.055)	···
Topic 3	Numbers(0.074)	Limit(0.062)	Rise(0.054)	Consumption(0.048)	Breed(0.045)	···
Topic 4	Occurrence(0.132)	Domestic(0.080)	Market(0.065)	Import(0.051)	Pass(0.042)	···
Topic 5	Livestock(0.084)	Association(0.056)	Pig(0.056)	Limit(0.052)	Management(0.052)	···
Topic 6	Pig(0.077)	Consumption(0.045)	Breed(0.043)	Market(0.042)	Import(0.038)	···

Here, we modified the previous section of the chart and marked top-5, top-5~10, top-10~15, and top-15~20 with different colors, respectively. As shown in Table 2, top-5 is red, top-5~10 is green, top-10~15 is yellow, and top-15~20 is orange. Top-5 consists of occurrence, limit, livestock, domestic, pig these five words. Top-5~10 consists of numbers, farmhouse, market, subject, limit these five words. Top-10~15 consists of price, livestock, association, execution, pig these five words. Top-15~20 consists of import, rise, import, limit, management, these five words. Using these color areas, we can judge the price trend

in January 2019. For example, when we look at Limit and import in topic 1, we may think that the government has issued a policy to limit pork imports. If this policy is true, domestic pork price may rise for some time.

4. Web interface

We developed the web page for prices, number of news and keywords. This page now can display the prices and news curve from January 1, 2010 to December 31, 2019. We can choose to look at prices in yearly, monthly, weekly and daily. Moreover, the curve of the number of news changing with time is displayed on this web page. By observing this curve, we can judge which period has important things and which period is relatively calm. We also can choose the number of keywords we want to observe. Currently, there are four options: top-5, top-10, top-15, and top-20.

Figure 2. Top-k, various kinds of price and news chart on the website

Figure 2 shows the function of the line. This is a double Y-axis curve of prices and number of news over time, the Y-axis on the left is the pork price and the Y-axis on the right is the number of news. The X-axis is the time that we chose. When we move the mouse over a time point on curve, a table will pop up. This table consists of two parts. The first part is the price and the number of news, the prices include retail price, distributor price and traditional price. As shown in figure 2, Jan 2019 retail price is 17230, distributor price is 15882, traditional price is 19254 and number of news is 73. The second part is the top-5 keywords and their importance to a particular topic. Such as Figure 2, top-5 keywords are occurrence, limit, livestock, domestic and pig. And their corresponding importance is 0.132, 0.107, 0.084, 0.080 and 0.077 relatively.

5. Conclusion

This paper introduces the analysis between keywords and news topics. We collected the news from the Pigtimes by web crawler and generated the topic and keywords through preprocessing and the LDA model. Then the TF-IDF formula is used to calculate the importance of these keywords. Finally, select k highest TF-IDF value keywords. Based on these keywords, we can judge price trend over a period and know what events happed during that period. Users can observe the curve of prices and the number of news over time and choose the k keywords that they want to observe on the web page. In another paper, we input the prices and the value of TF-IDF of these keywords as the dataset into the LSTM model for training and finally achieve the price prediction through this model.

Although we calculated the TF-IDF values of the news keywords and through them, we can achieve the judgment of price trend, understanding of the news topics during a particular period and predict the price by LSTM model. But for the impact of these keywords on prices is positive or negative, we did not make a distinction. In the coming months and years, we will divide keywords into positive and negative categories. Expand the application scenario of news keywords and achieve a excellent price prediction model.

Acknowledgments. This work was carried out with the support of the "Cooperative Research Program for Agriculture Science and Technology Development" (Project No. PJ015341012020), Rural Development Administration, Republic of Korea.

6. Reference

1. Qikai, L., Xiang, C., Sen, S., Shuguang, Z.: "Hierarchical Complementary Attention Network for Predicting Stock Price Movements with News": CIKM '18: Proceedings of the 27th ACM International Conference on Information and Knowledge Management (2018)
2. Yang, L., Qingguo, Z., Huanrui, Y., Adrian, C.: "Stock Price Movement Prediction from Financial News with Deep Learning and Knowledge Graph Embedding": Knowledge Management and Acquisition for Intelligent Systems pp 102-113(2018)
3. Saloni, M., Sahitya, M., Sudheer, S., Parag V., David, C. A.: "Stock Price Prediction Using News Sentiment Analysis": 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)(2019)
4. OECD 2019 Report

A Survey on Voice Identification of Singers using Deep Learning

So-Hyun Park[1], Young-Ho Park[1,*]

[1] Department of IT Engineering, Sookmyung Women's University, Seoul 04310, Korea
{shpark, yhpark}@sm.ac.kr

Abstract. Voice identification is very important in various applications. Among them, this study focuses on the research on the classification of singer's voices and speaker's voices. A lot of research has been done to distinguish human's voices. However, it is very difficult to classify the voices of singers because human voices are similar to each other, and the singer's song contains not only songs but also accompaniment or noise. In addition, there are studies that categorize human's voices, but there is a lack of research papers on voice identification related studies. In this paper, studies related to Singer identification and Speaker identification were surveyed. In both fields, there are two steps: removing noise and classifying Singer and Speaker. With the development of deep learning, performance has improved in both fields, but further research is still needed due to the difficulty of noise presence and human voice being similar.

Keywords: Singer identification, Speaker identification, Deep learning

1 Introduction

Voice identification is very important in various applications such as voice gender classification [1], celebrity voice classification [2], and singer voice classification [3-4]. Among them, this study focuses on the research on the classification of singer's voices and speaker's voices. A lot of research has been done to distinguish human's voices. However, it is very difficult to classify the voices of singers because human voices are similar to each other, and the singer's song contains not only songs but also accompaniment or noise [5]. In addition, there are studies that categorize human's voices, but there is a lack of research papers on voice identification related studies. In this paper, we would like to survey related studies of speaker identification and singer identification. Singer identification and speaker identification are closely related in that they must distinguish human voices in noisy environments. The noise of singer identification means accompaniment, and the noise of speaker identification refers to sounds other than voices such as coughing.

The structure of this paper is as follows. Chapter 2 surveys singer identification and speaker identification studies. Chapter 3 introduces data sets for both fields. In Chapter 4 introduces the limitations of this study and future studies.

2 Related Works

This chapter introduces research related to voice identification such as speaker identification and singer identification. First, chapter 2.1 introduces the study of singer identification. Later, in section 2.2, speaker identification method is introduced.

2.1 Singer identification

The reasons why Singer identification is difficult are as follows [5]. First, since singing and accompaniment are mixed, it is difficult to extract only the singer's voice characteristics excluding accompaniment. Second, there is a part where only the accompaniment comes out without a voice, and the deep learning model may not learn this part well when learning. In the singer identification study, studies were conducted to effectively remove the accompaniment that interferes with the identification of the singer's voice. In addition, studies to increase the accuracy of singer identification using deep learning techniques have recently appeared.

Recently, studies have emerged to separate background music from solo instrument sounds or separate background music from solo singers' voices using deep learning technology. Stoter et al. provided an open source that separates music sources. Music source separation refers to separating a song by parts such as vocals, bass, and drums. Music source separation is a very difficult problem, but performance has improved with the recent advances in deep learning. Stoter et al proposed a deep learning-based music source separation. It is meaningful that it has helped researchers in follow-up research, such as providing programs that end users and artists can try out pre-trained models [6].

Singer identification research based on deep learning technique is as follows. First, Nasrullah et al proposed a Convolutional Recurrent Neural Network (CRNN) that learns all the spatiotemporal features of Mel-Frequency Cepstral Coefficient (MFCC) for singer identification. As a result of performing a performance comparison experiment with existing studies, it showed superior results than the existing method. As a result of measuring the singer identification classification accuracy by extracting the audio features by frame level and song level, the classification accuracy of the song level feature was relatively higher than that of the existing method of 0.937. Frame level treats each frame as independent sample data, and singer level treats multiple samples belonging to a specific song as one data [6]. Hsieh et al also proposed a deep learning-based singer identification method [5]. First, the accompaniment and song were separated, and then singer identification was conducted. In the source separation step, the melody and accompaniment are separated using open-unmix, a deep learning-based source separation open api [7]. In the singer identification stage, a singer identification model was built based on the implementation of convolutional recurrent neural network, a state-of-the-art singer identification method [6]. The difference from existing methods is that the melody was extracted with CREPE [8], which is a melody extraction api, and the Data Augmentation technique was used. The data augmentation technique is a remixing method in which voices are artificially placed on various accompaniment songs to alleviate the confusion caused by the accompaniment. As a result of the artist

classification experiment (song level), the result was an f1 score of 0.75, which exceeded the f1 score of 0.67 of [5].

2.2 Speaker identification

This chapter introduces speaker identification methods in various fields. Even in the field of speaker identification, removing noise is an important issue.

Lukic et al proposed cnn-based speaker identification that automatically extracts feature points, unlike MFCC and Gaussian Mixture Model(GMM)-based speaker identification methods that require manual feature selection. Also, the process of transferring the trained speaker identification network for speaker clustering is described in detail [2].

Chowdhury applies the attention-based model, which has excellent performance in speech recognition, machine translation, and image captioning, to the speaker recognition system. The authors find an attention model for optimal speaker recognition by applying various topologies, attention layers, and different pooling methods. The proposed method showed 14% improvement compared to the existing LSTM-based method [3].

Shi et al proposed a cascading speech enhancement method to improve the performance of speaker recognition when speech signals are disturbed by noise. Existing methods separately performed speech enhancement to increase the quality of speech signals by removing noise for speaker recognition and speaker processing to recognize speakers, but in this paper, deep learning techniques were used to integrate the two modules into one. In addition, a multi-stage attention model was proposed to emphasize speaker relevant information [1].

3 Dataset

Chapter 3 introduces datasets that are often used in singer identification and speaker identification. The first dataset to be introduced is artist20, and the second dataset to be introduced is VoxCeleb.

Artist20 is a dataset for singer identification and consists of 1413 songs by 20 singers. It includes songs from musicians such as Aerosmith, Beatles, Dave matthews band, Depeche mode, Fleetwood mac, and Garth brooks. Each artist contains 6 albums, and one album contains about 11 songs. It also provides MFCC files as well as mp3 files [3].

VoxCeleb is a dataset for speaker recognition. The existing speaker recognition data set has a disadvantage that it is difficult to apply to the real world because it is data collected in a limited environment or in a noise-free environment. In addition, there is a disadvantage that human labeling may not be accurate. The VoxCeleb dataset consists of 100,000 impressions of audio spoken by 1,251 celebrities extracted from the YouTube platform. The gender ratio is uniformly distributed, and the speakers come from a wide range of races, accents, occupations, and ages [2].

4 Conclusion

In this paper, studies related to Singer identification and Speaker identification were surveyed. In both fields, there are two steps: removing noise and classifying Singer and Speaker. With the development of deep learning, performance has improved in both fields, but further research is still needed due to the difficulty of noise presence and human voice being similar. As a future study of this study, we intend to build a deep learning-based model that can simultaneously perform Singer identification and Speaker identification.

References

1. Gender Recognition by Voice dataset, https://www.kaggle.com/primaryobjects/voicegender
2. VoxCeleb dataset, http://www.robots.ox.ac.uk/~vgg/data/voxceleb
3. Artist20 dataset, https://labrosa.ee.columbia.edu/projects/artistid
4. Common Voice, https://github.com/mozilla/common-voice
5. Hsieh, T.H., Cheng, K.H., Fan, Z.C., Yang, Y.C., Yang, Y.H.: Addressing the confounds of accompaniments in singer identification. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1--5. IEEE Press (2020)
6. Nasrullah, Z., Yue, Z.: Music artist classification with convolutional recurrent neural networks. In: 2019 International Joint Conference on Neural Networks (IJCNN). pp. 1--8. IEEE Press (2019)
7. Open-Unmix, https://github.com/sigsep/open-unmix-pytorch
8. Crepe, https://github.com/marl/crepe
9. Lukic, Y., Vogt, C., Dürr, O., Stadelmann, T. Speaker identification and clustering using convolutional neural networks. In: 2016 IEEE 26th international workshop on machine learning for signal processing (MLSP). pp. 1--6. IEEE Press (2016)
10. Rahman Chowdhury, F. R., Wang, Q., Moreno, I. L., Wan, L. Attention-based models for text-dependent speaker verification. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5359-5363. IEEE Press (2018)
11. Shi, Y., Huang, Q., Hain, T. Robust speaker recognition using speech enhancement and attention model. arXiv preprint arXiv:2001.05031, 2020.

Acknowledgments. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology [NRF-2018R1D1A1B07046550].

Classification of Moving Patterns in Crowds

Uju Gim[1], Jaeyoung Lee[2], Aziz Nasridinov[1], Yoo-Sung Kim[2]

[1] Department of Computer Science, Chungbuk National University,
[2] Department of Information and Communication Engineering, Inha University

kwj1217@chungbuk.ac.kr, aopad984@gmail.com, aziz@chungbuk.ac.kr, yskim@inha.ac.kr

Abstract. This paper reports a classification analysis of moving patterns in crowd videos by using a 3 dimensional convolution network for feature extractions with several classification models. For analysis experiments, crowd videos are collected and selected, as those are very similar to the crowd situations interested in Korea, from the previous other studies and some extra videos collected by ourselves from the Internet. According to the experiment results, the deep learning architecture can distinguish each interesting moving pattern from other different patterns up to 73% of the classification accuracy.

Keywords: Social security, Korean popular crowd situation, crowd behavior analysis, moving patterns, 3D convolution network, classification model.

1 Introduction

For developing intelligent CCTV systems, computer vision community has concentrated on to develop from simple object detection schemes to more complex event detection schemes such as loitering detections, violence detections, up to crowd behavior analysis [1]. As the object detection schemes, the series of Yolo [2] and the ones of R-CNN [3] have proposed and used very popularly these days. For more complex event detections, loitering detection methods [4], violence detection methods [5], and crowd behavior analysis methods [6-8] have proposed.

Since the crowd behavior analysis differs in the sizes of the crowds and the analyzing purposes, previous related studies have used different data sets of different situations of different sizes, and different features with different methods [1,6]. Previous study [1] proposed a pipeline framework for describing the previous related works, which consists of 4 continuous stages; detection, tracking, feature extraction, and crowd behavior classification and anomaly detection. Using this pipeline framework can be helpful to understand the previous studies themselves and to figure out the differences between the previous studies. Another previous study [6] pointed out the weaknesses of the previous data sets used in the previous studies and proposed a data set named 'crowd-11' as a more generally fine grained data set that can be commonly used in many related studies. Crowd-11 data set consists of 11 classes such as lamina flow, turbulent flow, crossing flow, merging flow, diverging flow, gas free,

gas jammed, static calm, static agitated, interacting crowd, and no crowd. These classes are defined based on how a crowd can evolve across time within a video.

In this study, for developing the crowd behavior analysis against crowd situations which are interested and are highly likely in Korea, we looked into the crowd videos on the social network services and selected 6 classes; crossing flow, merging flow, static calm, static agitated, interacting crowd, and no crowd. According to the moving patterns in crowds, crossing flow, merging flow, static agitated, interacting crowd are considered as the abnormal cases, while static calm and no crowd classes are regarded as the normal cases. For this study, we have collected about 3,100 videos of 6 classes and do experiments for classification analysis of moving patterns in these crowd videos.

To correctly classify the moving pattern of crowds, we need to extract the appropriate features from crowd videos, which are useful for discriminating each pattern from others. To extract the useful features from crowd videos, a 3D convolution network which is well known and widely used for extracting the spatiotemporal motion features from the continuous frames in a video is used as in [6]. We also design and optimize the moving pattern classifier based on the comparisons of the accuracies of different classification models with the extracted motion features.

The rest of this paper is as follows. Section 2 describes the previous related works on the crowd dataset and on crowd behavior analysis. In Section 3, the data set which is collected and used in this study is introduced, and the classification analysis is discussed in Section 4. This study is concluded with a short introduction of future studies in Section 5.

2 Related Works

Previous study [1] proposed a taxonomic pipeline framework, which consists of 4 continuous stages such as detection, tracking, feature extraction, and crowd behavior classification and anomaly detection, and where sub-tasks in last stages benefit from the results in previous ones. Using this pipeline framework can be helpful to understand the previous studies and to find easily the differences between the previous studies. The detection stage is to localize the crowds in each frame. For this purpose, several detection schemes have proposed [2,3]. The tracking stage aims at re-identifying the specific persons and crowd trajectories over the continuous frames. Many previous studies have tackled this sub-task successfully. In the feature extraction stage, a set of metrics that describes the dynamics, topological structures of the crowd are computed. Examples of features are velocity, direction, density, collectiveness, and these features are computed by optical flow, histogram optical flow, and so on. In the last stage, crowd behavior and anomaly detection are performed. In general, behavior classification is performed in a supervised manner while anomaly detection is tried to identify a priori unknown.

Another previous study [6] proposed a data set named ‘crowd-11’ as a more generally fine grained data set that can be commonly used. Crowd-11 data set consists of 11 classes such as lamina flow, turbulent flow, crossing flow, merging flow, diverging flow, gas free, gas jammed, static calm, static agitated, interacting crowd,

and no crowd. These classes are defined based on how a crowd can evolve across time within a video. For classifying the classes in the dataset as shown in Figure 1, fluid or gas dynamic descriptions are used. According to the classification criteria in [6], a lamina flow occurs when the persons of crowd follow a smooth stream, a turbulent flow is for flow that undergoes a disturbance, a crossing flow is intertwined streams in opposite directions, a merging(diverging) flow is characterized by a compression(expansion), a gas free is for very scattered persons in crowd, a gas jammed is for the crowd where the trajectories of the persons disturb another. Two classes are for static situations. Static calm is with no movement and static agitated is with moving individuals in crowd. And an interacting crowd is for group of persons who move towards each other.

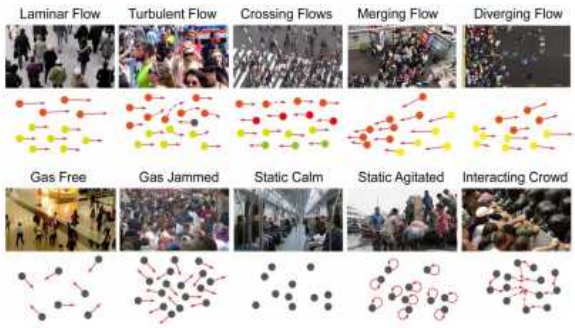


Fig. 1. Illustration of the classes in Crowd-11 dataset, this figure is from [6].

3 Crowd Dataset

For developing the crowd behavior analysis against crowd situations possible in Korea, we looked into the crowd videos on the social network services and we compared the moving patterns in the videos with those in the crowd-11 dataset. As shown in Figure 2, according to the similarity of the crowd moving patterns, for the crowd situations in Korea, we selected 6 classes from the crowd-11 dataset; crossing flow, merging flow, static calm, static agitated, interacting crowd, and no crowd. According to the moving patterns in crowds, crossing flow, merging flow, static agitated, interacting crowd are considered as the abnormal cases, while static calm and no crowd classes are regarded as the normal cases.

As shown in Table 1, we have collected about 3,100 videos of 6 classes. The major videos are from the selected classes from the crowd-11 [6], and some are from Real Life Violence Set(RLVS) [9] which is popularly used for developing violence detectors, and some are collected from the Internet by ourselves. All videos are longer than 3 seconds of playing time and have 5 or more persons.

Fig. 2. Selected moving patterns based on the similarity from the crowd situations in Korea.

Table 1. The number of videos according to the classes and the sources.

Classes	Crowd-11[6]	RLVS [9]	From SNS	Total
Crossing flow	741	-	54	795
Interacting crowd	200	231	-	431
Merging flow	276	-	54	330
Static Agitated	378	10	45	433
Static Calm	688	14	33	735
No crowd	378	-	-	378
Total	2,661	255	186	3,102

4 Moving Pattern Analysis

First of all, to correctly classify the moving pattern of crowds, the appropriate features which are useful for discriminating each pattern from others should be defined and extracted from the input videos. To extract the useful features from crowd videos, a 3D convolution network which is widely used for extracting the spatiotemporal motion features from the continuous frames in a video is used as in [6]. According to the descriptions of C3D in [6], C3D [10] is a well-known deep learning network to extract the spatiotemporal features from videos and consists of a succession of 3D convolutions. The third dimension of the input corresponds to a temporal stack of images that form a clip which is defined as a pack of 16 continuous frames [6]. The network follows a configuration of five 3D convolution + 3D pooling layers followed by three FC(fully connected) layers [6]. In general, to extract spatiotemporal features, using C3D is more efficient than using both 2D CNN(convolution neural network) and RNN(recurrent neural network) since it needs smaller amount of computation, and it has simpler and compact architecture, and C3D can tackle the gradient vanishing problem which are frequently occurred in 2D CNN + RNN networks.

As shown in Figure 3, from the spatiotemporal feature map extracted by C3D network, the deep learning classifier is used to determine the moving patterns in each crowd video. The deep learning classifier uses Label-smoothing Cross Entropy

(LCRE) for activation to resolve the imbalance problems in the training dataset. During the training, we use batch size = 50, and learning rate=1e-1, weight decay = 1e-4, momentum=0.9 for stochastic gradient descent(SGD) optimization. Also to overcome the overfitting problems, we use also batch normalization(BN) and leaky ReLU(LReLU).

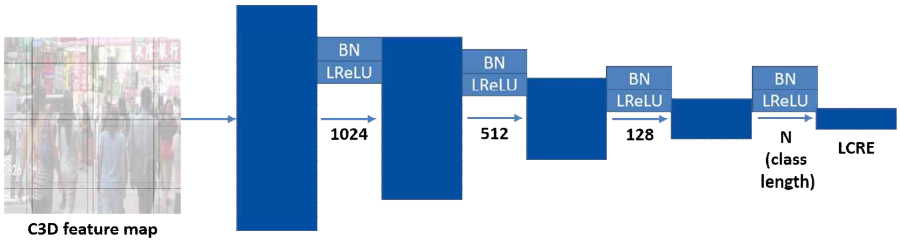


Fig. 2. Deep learning classifier architecture.

Table 2 shows the classification analysis results for the interesting classes selected in Section 3 in terms of classification accuracy according to the loss functions used in the last step in the deep learning classifier. According to the results, when we use a label smoothing cross entropy loss function instead of others such as cross entropy and margin loss functions, the avearge classification accuracy is about 73%, which is higher than those of other cases.

Table 2. The classification analysis results according to the loss functions.

Classes		# Train	# Test	Accuracy with CrossEntropyLoss	Accuracy with MarginLoss	Accuracy with LabelSmoothing
Abnormal	Crossing Flows	570	142	85%	90%	94%
	Merging Flow	209	52	45%	27%	31%
	Static Agitated	280	70	69%	59%	56%
	Interacting Crowd	67	17	40%	42%	27%
Normal	Static Calm	533	133	56%	65%	66%
	No crowd	294	73	95%	90%	95%
Total		1,953	488	70%	70%	73%

5 Conclusions

In this paper, we analyze the classification of moving patterns by using deep learning architectures from crowd videos which are very similar to the interesting situations in Korea. For this, we collect crowd videos which can be in Korea from the previous studies and also from the Internet by ourselves. To extract the appropriate spatiotemporal features, a 3D convolution network, C3D is used. And we design and optimize the deep learning classifier architecture with different loss functions. According to the experiment results, the deep learning architecture can distinguish each interesting moving pattern in crowd videos from other different patterns up to

73% when the label smoothing cross entropy loss function is used for resolving the imbalance problem of the training data.

As the further study, we will develop a crowd behavior analysis scheme which is well working for the crowd situation in Korea and is very helpful to improve the social security in Korea.

Acknowledgments. This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2020-0-00237, Development of Video Security Edge Technology with General Intelligence supporting 5G-based Mobility).

References

1. Sanchez, F. L., Hupont, I., Tabik, S., Herrera, F.: Revisiting crowd behavior analysis through deep learning: Taxonomy, anomaly detection, crowd emotions, datasets, opportunities and prospects. Information Fusion, vol. 64, 318--335 (2020)
2. Bochkovskiy, A., Wang, C., Liao, H. M.: YOLOv4: Optimal Speed and Accuracy of Object Detection. https://arxiv.org/abs/2004.10934, (2020)
3. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. https://arxiv.org/abs/1506.01497, (2015)
4. Kim, Y., Kim, Y-S.: Optimizing Neural Network to Develop Loitering Detection Scheme for Intelligent Video Surveillance Systems. International Journal of Artificial Intelligence, vol. 15(2), 30--39 (2017)
5. Joo, H-S., Kim, Y-S.: Violence detector using both CCTV videos and extracted skeleton images. Proceedings of Fall Conference of Korea Information Processing Society, (2020)
6. Dupont, C., Tobias, L., Luvison, B.: Crowd-11: A Dataset for Fine Grained Crowd Behavior Analysis. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2184 -- 2191 (2017)
7. Ramchandran, A., Sangaiah, A. K.: Unsupervised deep learning system for local anomaly event detection in crowded scenes. Multimedia Tools and Applications, (2019). https://doi.org/10.1007/s11042-019-7702-5
8. Sultani, W., Chen, C., Shah, M.: Real-world Anomaly Detection in Surveillance Videos. Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 6479—6488 (2018)
9. Soliman, M., Kamal, M., Nashed, M., Mostafa, Y., Chawky, B., Khattab, D.: Violence Recognition from Videos using Deep Learning Techniques. Proceedings of 9th International Conference on Intelligent Computing and Information Systems. 79--84 (2019) https://www.kaggle.com/mohamedmustafa/real-life-violence-situations-dataset
10. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M,: Learning spatiotemporal features with 3d convolution networks. https://arxiv.org/abs/1412.0767, (2015)

Data integration model for predicting PM index

Menghok Heak[1], Doorheon Jeong[1], Aziz Nasridinov[2], Sang-Hyun Choi[1],

[1] Department of Management Information System, Chungbuk National University, Cheongju, South Korea
[2] Department of Computer Science, Chungbuk National University, Cheongju, South Korea
menghok.heak@gmail.com, djccnt15@gmail.com, aziz@chungbuk.ac.kr, chois@cbnu.ac.kr

Abstract. PM index has become an interesting topic for researchers. There are many researches related to PM index prediction have been conducted. Data integration is an essential part of building the predictive model. In term of PM Index, the meteorological data and traffic data are usually be used in the research addiction to the expected output data of PM index. In this paper, we propose a solution to collect and integrate the data from multiple data sources in order to build up a model for predicting the index of PM. The paper will introduce the data to be work on in section 1, processes of collecting, pre-processing and integrating the data and technical implementation in section 2 and the result section 3.

Keywords: Data integration; PM index; Data collection;

1 Introduction

Air pollution is one of the most serious issues which causes a lot of harmful diseases to both human physical and mental illness [1]. The root cause of the pollution is believed to be from industrial emissions, vehicle engine generations and especially meteorological factors [2][3].

There are numerous researches have been conducted related to air pollution. Predicting the index of Particle Matters (PMs) is one of the interesting topics which has been researched in order to help people in knowing the level of PM and be prepared in advance. Researchers usually use different algorithms and data to do forecasting. However, in general, meteorological data and traffic data are usually been used. Those data are collected from different sources using different methods and later be integrated together. This paper focuses on the processes of collecting and integrating the data to build a data warehouse which is useable for building a predictive model of the PM index.

2 Materials and Methods

In this section, we focus on the processes of collecting, pre-processing and integrating the data as well as the technical implementation. The raw data are collected from

different data sources In this research, we use Meteorological data from the Korea Meteorological Administration portal (web.kma.go.kr), Daejeon Transportation Data Warehouse (tportal.daejeon.go.kr) and Korea Air Quality portal (airkorea.or.kr).

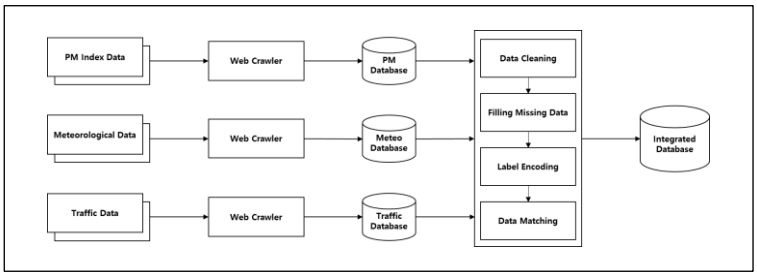


Fig. 1. Overall process of collecting and integrating data

The data were captured from the internet sources using a small computer program called Web Crawler which was written using Python Programming Language and BeautifulSoup Library [4]. The web crawler collected the data by sending requests to the web addresses or URLs that contain data and accessed over the response web page contents using HTML selectors [5]. After receiving the data, the data were stored in an individual database before getting to the pre-processing stage.

The pre-processing stage contains multiple sub-processes such as data cleaning, filling the missing data, label encoding and data matching. The dataset may contain some invalid values which are required to be cleaned before other processing. After cleaning the invalid data, some missing values in the dataset may be filled using a predicted value from a simple model. As the datasets are in time-series, the interpolation estimation is recommended to be used. After the missing values are filled, categorical variables are subjected to be converted into multiple numerical variable columns. This process is called label-encoding. And Finally, the 3 datasets are matching and integrate into a complete dataset using the 'date-time' column.

3 Results

This paper described a simple solution for collecting, processing and integrating the data for building the PM Index predicting model from multiple data sources. The integrated dataset is ready for building the model of PM index prediction. Even the processes could be altered depending on a specific condition, the mentioned methods and processes are capable for the integration processes of data from common data sources.

References

1. Pope, C.A., Burnett, R.T., Thun, M.J., et al: Lung Cancer, Cardiopulmonary Mortality, and Long-Term Exposure to Fine Particulate Air Pollution. Jama-J. Am. Med. Assoc. 2002, 287, 1132–1141 (2002)
2. Chang Z., Guojun S.: Application of data mining to the analysis of meteorological data for air quality prediction: A case study in Shenyang. doi :10.1088/1755-1315/81/1/012097 (2017)
3. Zhao C. X., Wang Y. Q., Wang Y. J., Zhang H. L., Zhao B. Q.: Temporal and Spatial Distribution of PM2.5 and PM10 Pollution Status and the Correlation of Particulate Matters and Meteorological Factors During Winter and Spring in Beijing, Environmental Science, 2014, 35(2):418-427, (2014)
4. BeautifulSoup Website. Retrieved on November 20th, 2020 from https://www.crummy.com/software/BeautifulSoup/
5. WikiPedia: Web Crawler. Retrieved on November 20th, 2020 from https://en.wikipedia.org/wiki/Web_crawler

Case analysis of solar monitoring facility failure

Woo-seok Choi[1], Da-bin Choi[2], Aziz Nasridinov[3], Sang-hyun Choi[2],
[1]Dept. Bigdata, Chungbuk National University, Cheongju, South Korea
[2]Dept. Management Information System, Chungbuk National University, Cheongju, South Korea
[3]Dept. Computer Science, Chungbuk National University, Cheongju, South Korea
{cdt3017,choidb1018}@naver.com, {aziz,chois}@cbnu.ac.kr

Abstract. In PV system, losses are increased due to defects in design and construction, such as inverters and junction boards, resulting in additional costs and time to identify and repair failures or defects. Therefore, it is very important to analyze existing failure cases for areas, timing, etc. where failures occur to maintain the quality of PV systems and to detect prior failures. Therefore, this study described the type of solar power facility failure and visualized the actual failure case as data.

Keywords : Solar Power, Facility Failure, Failure Detection, Case Analysis

1 Introduction

Due to social issues such as climate change response, fine dust reduction, nuclear power plant safety, and the unstable supply and demand of energy due to changes in international crude oil prices, as well as the problem of energy depletion, interest and investment in renewable energy have been rapidly increasing not only in Korea but also around the world[1]. Such solar power generation is inherently sensitive to external factors such as weather information, making it difficult to predict accurately and contains intermittent output variability, which will have a significant impact on grid stability[2]. Recently, the importance of early abnormal detection to monitor power generation facilities in real time and respond to changes in power generation has been increasing in order to resolve uncertainties in power generation data. Therefore, in this study, we would like to describe the types of failures that may occur in solar installations and analyze failure cases with actual data collected from the facilities.

2 Failure Type

There are four main types of failures in solar power systems. First, It is a 'Hotspot Phenomenon'. Hotspot is caused by surface pollution accumulation and shadow, and when a part of a solar module is obscured by a shadow, the solar module operates under load, greatly reducing the output of the entire solar power generation. Second is 'Junction box fault'. The junction box plays an important role in connecting

electricity generated from solar modules, and since it is exposed to the external environment, the risk factor for weather environmental factors is high. In addition, if power loss occurs inside a diode mounted within the junction box, these losses are released into heat and affect surrounding parts. Third one is 'Inverter fault'. Inverter is a very important facility for PV power generation, and in the field, inverter failure is the most vulnerable problem for power generation. Because it is mostly installed in outdoor environments, it is exposed to a lot of electricity and heat during operation. The last one is an 'Arc-Fault'. Fire often occurs in junction boxes and inverters in solar power generation systems, of which fire caused by arc generation is the main cause. Such failure detection is analytically possible by calculating the data generated on the installation at a quantitative value[3].

3 Case Analysis

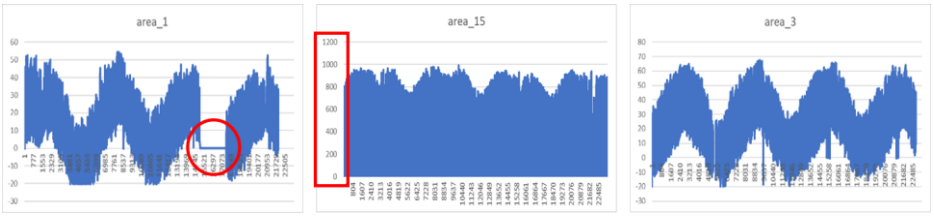


Fig. 1 Facility Failure Case 1

In this study, we analyzed data-based failures through actual PV data. <Fig.1> is the result of visualizing the module temperature sensor values collected from each of the three power plants in a time series between 2016 and 2020. Area_1 can confirm that the values for a particular interval have not been collected, which can be assumed to be malfunctioning or facility failure of the monitoring system. Area_15 can confirm that the units of the collected values are located at 600 to 800 degree, which is considered a failure of the module temperature sensor. Finally, Area_3 has no problem, and it can be seen that the values of the module temperature have been collected normally.

Fig. 2 Facility Failure Case 2

In addition, abnormalities in solar power generation can be seen through forecasts of power generation. <Fig.2> illustrates the actual and predicted power generation of the area_1 plant on August 26, 2020. The graph shows that the predicted value and the actual value were similar until 11 o'clock, but the actual value dropped sharply from 12 o'clock. Based on the collected power generation data, abnormalities can be determined as facility failure such as inverter, junction board or communication failure.

4 Conclusion

Internationally, the importance of renewable energy is higher every year, especially among energy, solar power generation is the most effective generation facilities recognized as the environment[4]. However, the efficiency of power generation is not higher than that of existing power generation facilities such as thermal power, hydro power, and nuclear power, and solar power generation is lost not only due to installation and control, but also due to environmental factors such as weather and regional features. Therefore, for the solar industry to grow in the future, stable power generation forecasts and early detection of abnormalities should be possible[5]. In this study, we checked the existing type of facility failure and analyzed the actual data-based facility abnormality detection to confirm the possibility of data-based early abnormal detection. In the future, we will actually increase the accuracy of forecasting power generation and design early abnormal detection models.

Reference

1. Dong-su Park.: Reduction Plan of Electrical Fire through Analysis of the Electric Failure and Accident Cases and Application of ETA in PV System (2018)
2. Jong Kwan Seo, Tae Il Lee, Whee Sung Lee, Jeom Bae Park.: A study on the outlier data estimation method for anomaly detection of photovoltaic system. In: Journal of IKEEE, pp.32--37 (2020)
3. Kyu-kwang Kim.: A Study on Fault Detection of PV System Using Multivariate Analysis (2019)
4. Korea Energy Economics Institute.: Promoting Eco-friendly Energy Policy in Major Countries and Changing the Role of Renewable Energy (2018)
5. Yongsu Kim, Sanghyun Lee, Howon Kim.: Prediction Method of Photovoltaic Power Generation Based on LSTM Using Weather Information. In: The Journal of Korean Institute of Communications and Information Sciences, Vol.44 No.12, pp.2231—2238 (2019)

THE KOREA
BIG DATA SERVICE SOCIETY
한국빅데이터서비스학회

N13 404-2, Chungbuk University, Chungdae-ro 1
Seowon-Gu, Cheongju, Chungbuk 28644, Korea

Tel: 043-261-3636 / Fax: 043-266-3637

Email: kbigdataservice@gmail.com

Homepage: www.kbigdata.or.kr