Automatic Generation of Explanatory Text from Flowchart Images in Patents

Hidetsugu Nanba[1], Shohei Kubo[1] and Satoshi Fukuda[1]

[1] *Chuo University, 1-13-27 Kasuga, Bunkyo-ku, Tokyo 112-8551 JAPAN*

Abstract

This paper addresses the automatic generation of explanatory text from flowchart images in patents. The construction of an explanatory text generator consists of four steps: (1) automatic recognition of flowchart images from patent images, (2) extraction of text strings from flowchart images, (3) creation of data for machine learning, and (4) construction of an explanatory text generator using T5. In this study, a benchmark consisting of 7,099 images was constructed to determine whether an image in a patent is a flowchart. Furthermore, an explanatory text generator was constructed from the images using 11,188 flowchart image-explanatory text pairs. The experimental results showed that a recognition accuracy of 0.9645 was achieved for flowchart images. Although high-quality explanatory text could be generated from flowchart images, some issues remain for flowcharts with complex shapes.

Keywords

flowchart, image recognition, text generation, character recognition, patent

1. Introduction

A procedural text is a description of a set of procedures to achieve a particular objective. Our goal is to automatically extract knowledge about a series of procedures in a wide range of fields from texts and systematize them. Here, we describe the automatic generation of explanatory text from flowchart images in patents.

In automatically generating explanatory text for the flowchart images, we focus on the abstract and selected figures of the patent. A selection figure enables us to grasp the outline of the invention quickly and accurately. The applicant usually selects a diagram from among the diagrams in the patent that they consider necessary for understanding the abstract contents. If a classifier that automatically determines whether an image in a patent is a flowchart or not is constructed and only those selected diagrams that are flowcharts are extracted, a large number of pairs of flowcharts and their explanatory texts (i.e., patent abstracts) can be generated automatically. Furthermore, using these pairs, we believe it is possible to construct a system that automatically generates explanatory text from flowchart images using machine learning.

The contributions of this paper are as follows:

- To determine whether an image in a patent is a configuration diagram, flowchart, or table, we constructed a benchmark consisting of 7,099 images. We used this benchmark to achieve a classification accuracy of 0.9645.
- We constructed 11,188 pairs of flowchart images and their descriptions automatically.
- Using these pairs, we constructed a system that automatically generates explanatory text from flowchart images through machine learning.

2. Related Work

2.1. Flowchart Analysis

Services that share flowcharts, such as myExperiment and SHIWA, have started recently, which has led to a demand for techniques to search for similarities between one flowchart and another flowchart [1]. A related research project in flowchart image analysis is CLEF-IP, which refers to a task targeting patents [2]. The Conference and Labs of the Evaluation Forum (CLEF) is a workshop on information retrieval held mainly in Europe. CLEF-IP recognizes shapes, detects text, edges, and nodes that are elements of flowcharts, and recognizes flowcharts. Herrera-Cámara also worked to recognize flowchart images [3]. In addition, Sethi et al. identified flowcharts from diagram images in deep learning-related papers and further analyzed the flowcharts to build a system that outputs the sources in Keras and Caffe [4]. This research differs from theirs in that we take a flowchart image as input and output its description as a natural language sentence. We considered the availability of resources such as the CLEF-IP for our work, but as it is too small to be used as training data for the generation of explanatory texts, this study started with the creation of training data.

2.2. Generating Text from Figures and Tables

Chart to text refers to the task of generating natural language sentences to describe the important information derived from charts and tables. Zhu et al. [5] addressed this problem by building a system, AutoChart. A human and machine evaluation of the generated text and charts demonstrates that the generated text is informative, coherent, and relevant to the corresponding charts [6].

Tan and colleagues [7] generated sentences from pie charts, bar graphs, and line graphs in scientific papers, while Kantharaj and colleagues [8] generated sentences from charts using generators such as T5 [9], BART, and GPT2 based on bar and line graphs mainly describing economic, market, and social issues. Instead of graphs, this study uses flowchart images as inputs and the goal is to automatically generate explanatory text from these flowcharts.

3. Automatic Generation of Explanatory Text from Flowchart Images

The construction of the generator of explanatory text consists of the following four steps: (Step 1) automatic recognition of flowchart images; (Step 2) extraction of character strings from the flowchart image; (Step 3) creation of data for machine learning; and (Step 4) construction of an explanatory text generator using T5. Each procedure is described as follows.

(Step 1) Automatic recognition of flowchart images

Convolutional neural networks (CNNs) are used to recognize flowchart images in patents. Our method uses seven CNN models trained on a large image data set called "ImageNet" to construct a learning model by fine tuning, and its effectiveness is verified through the experiments described in Section 4.

(Step 2) Extraction of character strings from the flowchart image

An optical character recognition function in Google Cloud Vision (https://cloud.google.com/vision) is used to extract text strings from flowcharts. An example of a flowchart image and the character recognition result are shown in Figures 1 and 2 respectively. Here, "\n" indicates a line break.

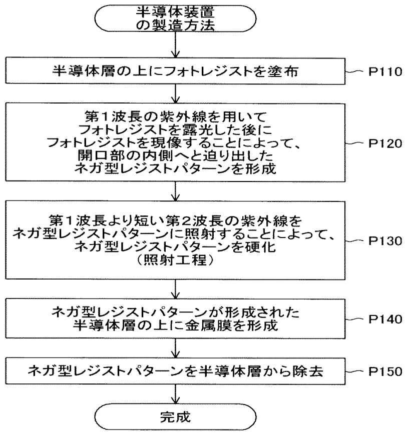
半導体装置
の製造方法
半導体層の上にフォトレジストを塗布
P110
第1波長の紫外線を用いて
フォトレジストを露光した後に
フォトレジストを現像することによって、
開口部の内側へと迫り出した
ネガ型レジストパターンを形成
P120
第1波長より短い第2波長の紫外線を
ネガ型レジストパターンに照射することによって、
ネガ型レジストパターンを硬化
（照射工程）
P130
ネガ型レジストパターンが形成された
半導体層の上に金属膜を形成
P140
ネガ型レジストパターンを半導体層から除去
P150
完成

Figure 1: Example of Flowchart Image Included in a Patent

[original]
半導体装置\n の製造方法\n 半導体層の上にフォトレジストを塗布\n 第 1 波長の紫外線を用いて\n フォトレジストを露光した後に\n フォトレジストを現像することによって、\n 開口部の内側へと迫り出した\n ネガ型レジストパターンを形成\n 第 1 波長より短い第 2 波長の紫外線を\n ネガ型レジストパターンに照射することによって、\n ネガ型レジストパターンを硬化\n(照射工程)\n ネガ型レジストパターンが形成された\n 半導体層の上に金属膜を形成\n ネガ型レジストパターンを半導体層から除去\n 完成\nP110\nP120\nP130\nP140\nP150
[translation]
Semiconductor devices \n Manufacturing method for \n photoresist is applied over the semiconductor layer \n using ultraviolet light of the first wavelength \n After exposing the photoresist \n by developing the photoresist, \n The photoresist is developed to form a negative resist pattern that extends into the aperture. \n Negative resist pattern is formed. \n By exposing the negative resist pattern to ultraviolet rays of the second wavelength, which is shorter than the first wavelength \n by irradiating the negative resist pattern, \n curing the negative resist pattern by irradiating it with ultraviolet light of the second wavelength, which is shorter than the first wavelength. \n (Irradiation process) \n The negative resist pattern is formed \n Metal film is formed on top of the semiconductor layer \n Negative resist pattern is removed from the semiconductor layer \n Completion \n P110 \n P120 \n P130 \n P140 \n P150

Figure 2: Character Recognition Results for the Image in Figure 1.

(Step 3) Creation of data for machine learning

We build an explanatory text generator by machine learning, using pairs of character recognition results and explanatory text from a large number of flowchart images. In this process, we consider that data with large differences between the character recognition results and the manually written explanatory texts (patent abstracts) are inappropriate as training data[1]; therefore, we exclude these data. In this process, we calculate the similarity between the character recognition result and the explanatory text of the flowchart image using Gestalt pattern matching [10] and use only the pairs that are above a threshold value for training.

(Step 4) Construction of an explanatory text generator using T5

We build an explanatory text generator by the language model T5. With respect to the flowchart image in Figure 1, the input and output of T5 are Figure 2 for input and Figure 3 for output.

[1] Figures 2 and 3 show examples of a character recognition result and a manually written explanatory text (patent abstract). In this case, the similarity between them is so high that we use them as machine learning data.

[original]

半導体装置の製造方法は、半導体層の上にフォトレジストを塗布する工程と；第１波長の紫外線を用いてフォトレジストを露光した後にフォトレジストを現像することによって、開口部の内側へと迫り出したネガ型レジストパターンを、形成する工程と；第１波長より短い第２波長の紫外線をネガ型レジストパターンに照射することによって、ネガ型レジストパターンを硬化させる照射工程と；照射工程を行った後、ネガ型レジストパターンの開口部から露出する半導体層の上に、ニッケル（Ｎｉ）から主に成る金属膜を形成する工程と；ネガ型レジストパターンを半導体層から除去する工程とを備える。

[translation]

The method of manufacturing a semiconductor device comprises the steps of: applying a photoresist onto a semiconductor layer; forming a negative resist pattern, which is pressed inwards into an aperture, by developing the photoresist after exposing the photoresist using a first wavelength of ultraviolet light; forming a negative resist pattern by irradiating the negative resist pattern with ultraviolet light of a second wavelength that is shorter than the first wavelength; and The negative resist pattern is hardened by irradiating the negative resist pattern with ultraviolet light of a second wavelength shorter than the first wavelength; forming a metal film mainly comprising nickel (Ni) on the semiconductor layer exposed from the opening of the negative resist pattern after performing the irradiation process; and removing the negative resist pattern from the semiconductor layer. The process of removing the negative resist pattern from the semiconductor layer.

Figure 3: Explanatory Text (Patent Abstract) Correspongind to the Image in Figure 1

4. Experiments

We performed some experiments to confirm the effectiveness of our method.

4.1. Automatic Recognition of Flowchart Images

Data

Using 7,099 randomly selected images from the 2018 edition of the Japanese Patent Public Gazette, we manually identified whether they were flowcharts or not and obtained 1,120 flowcharts from the 7,099 cases.

Alternative methods

As a baseline method, we used Keras, a deep learning library, to build CNN training models with three layers for Conv2D and two layers for MaxPooling2D. As a comparison method, we used seven CNN models: VGG16, VGG19, ResNet50, InceptionV3, MobileNet, DenseNet169, and DenseNet121 trained on a large image data set called ImageNet.

Evaluation

The seven methods and the baseline method were evaluated using Precision, Recall, and F-measure.

Results

The experimental results are shown in Table 1. Among the compared methods, DenseNet121 was the most accurate in detecting flowcharts in terms of Precision. The results from DenseNet121 were used in the subsequent experiments.

Table 1
Flowchart Recognition Results with Eight Models

	Precision	Recall	F-measure
Baseline	0.8508	0.8902	0.8701
VGG16	0.8750	**0.9711**	0.9205
VGG19	0.9227	0.9653	0.9435
ResNet50	0.8698	0.9653	0.9151
InceptionV3	0.9422	0.9422	0.9422
MobileNet	0.9326	0.9595	0.9459
DenseNet169	0.9593	0.9538	**0.9565**
DenseNet121	**0.9645**	0.9422	0.9532

4.2. Automatic Generation of Explanatory Text from Flowchart Images

Data

Among the Japanese patents published from 2010 to 2019, 11,188 patents that included flowcharts and with a similarity of 0.1 by Gestalt pattern matching were used in our experiments. Of these patents, 90% were categorized as training data and the remainder as validation and evaluation data.

Hyperparameters

The following hyperparameters were used in the generation of explanatory texts by T5.

- Max input length: 280
- Max target length: 256
- Train batch size: 8
- Eval batch size: 8
- Num train epochs: 6

Evaluation

Our method was evaluated using the following measures:

- ROUGE-N: This is the most basic index and is a method of taking the degree of agreement in N-gram units. In this case, N = 1, 2 were used for evaluation (https://github.com/pltrdy/rouge).
- ROUGE-L: Evaluates the maximum sequence that matches the generated summary and the manually generated summary.
- BERTScore [11]: An automatic evaluation metric using the language model BERT [12], which calculates the similarity between texts using vector representations obtained from pretrained BERT.

Results

The results of Recall, Precision, and F-measure for ROUGE-1, ROUGE-2, ROUGE-L, and BERTScore.

Table 2
Evaluation Results for the Generation of Explanatory Texts

	Recall	Precision	F-measure
ROUGE-1	0.47	0.72	0.55
ROUGE-2	0.26	0.46	0.32
ROUGE-L	0.41	0.64	0.49
BERTScore	0.74	0.77	0.75

Discussion

For simple geometries with no branches in the flowchart (see Figure 4), we obtained good analytical results. Figures 5 and 6 show the explanatory text and the patent summary (correct answer) generated by our method, respectively.

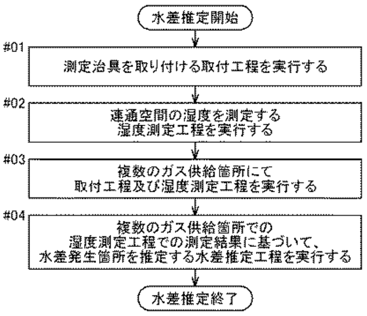
水差推定開始
#01
測定治具を取り付ける取付工程を実行する
#02
連通空間の湿度を測定する
湿度測定工程を実行する
#03
複数のガス供給箇所にて
取付工程及び湿度測定工程を実行する
#04
複数のガス供給箇所での
湿度測定工程での測定結果に基づいて、
水差発生箇所を推定する水差推定工程を実行する
水差推定終了

Figure 4: Example of Target Image for Generation

[original]
測定治具を取り付ける取付工程と、複数のガス供給箇所にて取付工程及び湿度測定工程を実行する湿度測定工程と、複数のガス供給箇所での湿度測定工程での測定結果に基づいて、水差発生箇所を推定する水差推定工程と、を含む。
[translation]
The process includes a mounting process to install the measurement jig, which is a humidity measurement process to perform the mounting and humidity measurement processes at multiple gas supply locations, and a water-difference estimation process to estimate the location of water-difference occurrence based on the measurement results of the humidity measurement process at multiple gas supply locations.

Figure 5: Exploratory Text Automatically Generated from the Image in Figure 1

[original]
ガス供給箇所へ供給されるガスの流量を測定するガスメータの一次側へ連通接続する連通空間を有すると共に当該連通空間の湿度を測定する湿度測定部を有する水差推定治具を取り付ける取付工程と、湿度測定部にて連通空間の湿度を測定する湿度測定工程と、湿度測定工程にて測定される湿度に基づいて、ガス管に水差しが発生しているか否かを推定する水差推定工程とを実行する。
[translation]
The following processes are performed: An installation process in which a water-difference estimation jig is attached to a gas meter with a connecting space that is connected to the primary side of the gas meter that measures the flow rate of gas supplied to the gas supply point and a humidity measuring section that measures the humidity in the connecting space; a humidity measurement process in which the humidity in the connecting space is measured by the humidity measuring section; and a water-difference estimation process in which whether a water drop occurs in a gas pipe is estimated based on the humidity measured in the humidity measurement process.

Figure 6: Patent Summary for the Image in Figure 1 (Correct Answer)

Flowcharts with complex shapes, such as the one shown in Figure 7, tended to generate low-quality explanatory text. The dash line boxes in the figure were added by the author for the purpose of explanation. The description generated by the process in Figure 7 is shown in Figure 8.

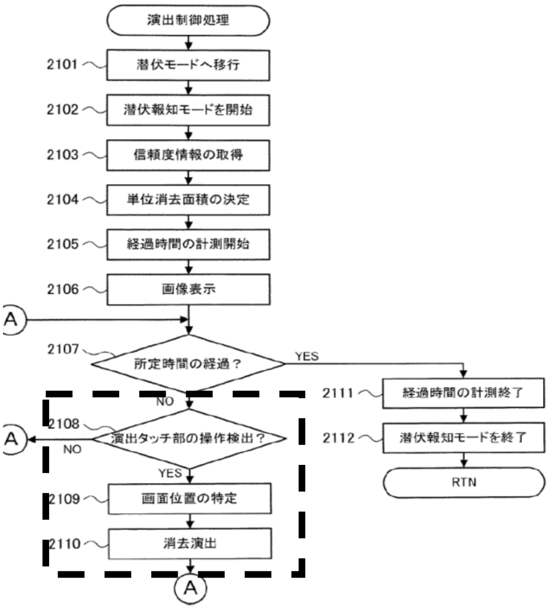

Figure 7: Example of a Flowchart with Conditional Branching

[original]
潜伏モードへ移行し(s2110)、信頼度情報を取得し(s2110)、変倍率を決定し(s2110)、経過時間の計測を開始する(s2112)。そして、画像表示部が所定時間経過しているか否かを判定し(s2112)、所定時間が経過すると(s2112 にて yes)、潜伏報知モードを終了する(s2112)。潜伏報知モードを終了すると(s2112 にて yes)、潜伏報知モードを終了する。

[translation]
The system moves to the latent mode (s2110), obtains the reliability information (s2110), determines the variable magnification factor (s2110), and starts measuring the elapsed time (s2112). The image display then determines whether or not the predetermined time has elapsed (s2112). When the predetermined time elapses (yes at s2112), the latent report mode is terminated (s2112). When the latent report mode is terminated (yes at s2112), the latent report mode is terminated.

Figure 8: Exploratory Text Automatically Generated from the Image in Figure 7

Looking at Figure 8, overall, step IDs such as s2110 do not correspond to the explanatory text, but this is because this time the coordinates of each string in the figure are not considered at all. The first conditional branch is “When the predetermined time elapses (yes at s2112), the latent report mode is terminated.” The correct sentence is generated except for the step ID (s2112) (see Figure 8). However, the dashed box in Figure 7 is not included in the explanatory text. Currently, the character strings output by Google Cloud Vision’s character recognition results are used as input to T5 as they are, but in the future, it will be necessary to perform preprocessing such as considering the coordinate information of the character strings and reordering them appropriately.

5. Conclusions

In this study, 11,188 flowchart image-description pairs were obtained from patents and these data were used to construct a system that automatically generates descriptions of flowchart images using T5. The experimental results showed that for the detection of flowchart images, an accuracy of 0.9645 was achieved with a fine-tuned model using DenseNet121. In the generation of explanatory text from flowchart images, it was found that high-quality explanatory text could be generated, although some issues remain for flowcharts with complex shapes. In the future, we will examine the possibility of generating appropriate explanatory text for flowcharts with complex shapes, such as those containing multiple conditional branches, by considering the positional information of each character string in the image, rather than using the character strings in the flowchart as is.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Numbers JP22K12154 and JP20H04210.

References

[1] J. Starlinger, B. Brancotte, S. Cohen-Boulakia, and S. Leser, Similarity Search for Scientific Workflows, Proceedings of the VLDB Endowment, Vol. 7, No. 12, pp.1143-1154, 2014.

[2] F. Piroi, M. Lupu, and A. Hanbury, Overview of CLEF-IP 2013 Lab Information Retrieval in the Patent Domain, Information Access Evaluation. Multilinguality, Multimodality, and Visualization. CLEF 2013. Lecture Notes in Computer Science, Vol. 8138. Springer, Berlin, Heidelberg, 2013.

[3] J. I. Herrera-Cámara, FLOW2CODE - From Hand-drawn Flowchart to Code Execution, Master Thesis, Texas A&M University, 2017.

[4] A. Sethi, A. Sankaran, N. Panwar, S. Khare, and S. Mani, DLPaper2Code: Auto-generation of Code from Deep Learning Research Papers, Proceedings of the 32th AAAI Conference on Artificial Intelligence, 2018.

[5] J. Zhu, J. Ran, R. K. Lee, Z. Li, and K. Choo, AutoChart: A Dataset for Chart-to-Text Generation Task, Proceedings of the International Conference on Recent Advances in Natural Language Processing, pp. 1636-1644, 2021.

[6] J. Obeid and E. Hoque, Chart-to-Text: Generating Natural Language Descriptions for Charts by Adapting the Transformer Model, Proceedings of the 13th International Conference on Natural Language Generation, pp. 138-147, 2020.

[7] H. Tan, C. Tsai, Y. He, M. and Bansal, Scientific Chart Summarization: Datasets and Improved Text Modeling, Proceedings of the AAAI-22 Workshop on Scientific Document Understanding, 2022.

[8] K. Kantharaj, R. T. Leong, X. Lin, A. Masry, M. Thakkar, E. Hoque, and S. Joty, Chart-to-Text: A Large-Scale Benchmark for Chart Summarization, Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, pp. 4005-4023, 2022.

[9] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Journal of Machine Learning Research, Vol. 20, No. 140, pp. 1-67, 2020.

[10] J. W. Ratcliff and D. Metzener, Pattern Matching: The Gestalt Approach, Dr. Dobb’s Journal, pp. 46, 1988.

[11] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y, Artzi, BERTScore: Evaluating Text Generation with BERT, arXiv:1904.09675 [cs.CL], 2019.

[12] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p. 4171-4186, 2019.