

# 製造業における製品品質の オンラインリアルタイム診断ツール

佐々木 大知<sup>†3,a)</sup> 福田 浩章<sup>†3</sup> 高橋 和也

**概要:** 製造業において、産業用ロボットの導入により、生産性の向上が著しい。さらに生産性を向上させるためには、検査工程の自動化を進めることが不可欠である。近年、AI技術の一つであるディープラーニングが急速に進歩している。検査工程の自動化の手法として、ディープラーニングを用いた画像認識による外観検査が主流となっている。しかし、この手法では製品の良否を判定することはできるが、不良品が発生した原因がわからない。

そこで本研究ではIoT技術を用いて、製造過程のデータ収集を行い、産業用ロボットの動作をグラフ化し、可視化することで製品の良否判定をリアルタイムで行い、不良品が発生した原因を視覚的に表示するツールの実装と提案を行う。

**キーワード:** 製造業, エッジコンピューティング, IoT, データマイニング

## Online and Real-time diagnostic tool for product quality in the manufacturing industry

**Abstract:** In the manufacturing industry, the introduction of industrial robots has significantly improved productivity. In order to further improve productivity, it is essential to automate the inspection process. In recent years, deep learning, which is one of the AI technologies, is making rapid progress. As a method of automating an inspection process, an appearance inspection by image recognition using deep learning has become mainstream. However, with this method, the quality of the product can be determined in sequence, but the cause of the defective product is not known.

In this research, we use IoT technology to collect data on the manufacturing process, graph and visualize the operation of industrial robots, and determine the quality of products in real time by visualizing the cause of defective products. Implement and propose tools to be displayed

**Keywords:** Manufacturing, Edge Computing, IoT, Data Mining

### 1. はじめに

近年、市場のグローバル化が急速に進み、製造業ではグローバル市場における競争力を確保するため、従来よりも高い品質管理と生産性が求められている [1]。日本では、ロボットや自動機械の導入により製造や生産工程において生産性の向上が著しい。さらに生産性を向上させるためには、

検査工程の自動化を行うことが不可欠である。しかし、現状では製造工程毎に検査員が目視で製品の良否判定を行ない、検査員が不良品と判断したものをその場で廃棄している。そのため、疲労などによるヒューマンエラーや、検査員ごとに良否判定基準が異なることが原因で、本来不良品と判断されるべき製品を良品と判断してしまうといった課題がある。これらの課題の主な原因として、製造業における人材不足の深刻化が挙げられる [2]。図1に経済産業省製造産業局の製造現場の人材不足を調査した結果を示す。図1からわかるように製造業の大企業、中小企業の約94%が人材不足の課題が顕在化、約32%がビジネスに影響が出ていると回答している。製造業における人材不足が原因で、検査

<sup>†1</sup> 現在、芝浦工業大学工学部情報工学科  
Presently with Shibaura Institute Technology

<sup>†2</sup> 現在、芝浦工業大学工学部情報工学科准教授  
Presently with Shibaura Institute Technology

<sup>†3</sup> 現在、Edecross コンソーシアム会員  
Presently with Edgexcross Consortium

a) al16047@shibaura-it.ac.jp

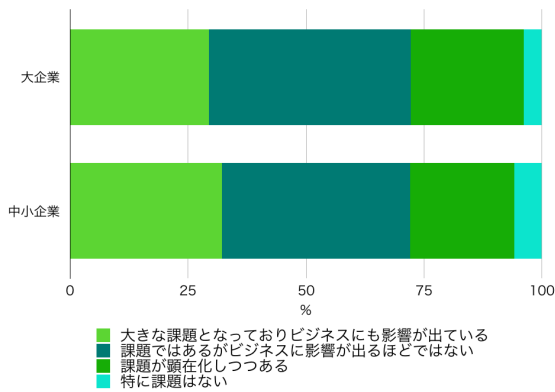


図 1 人材不足の現状

員一人あたりの作業時間の増加により、疲労によるヒューマンエラーが起りやすい。また、新人検査員の育成が困難となり、いわゆる職人と呼ばれる検査員との間に差が生まれ、良否判定基準にばらつきが出てしまう。

これらのことから、検査工程の自動化を行うための要求として、検査員が不良品をその場で廃棄しているという点から、リアルタイム性が求められる。また、検査員ごとに良否判定基準に差があるということから、検査員の技術レベルに依存しないツールが求められる。

これらの要求を満たす検査工程の自動化手法として、画像認識による外観検査がある [3][4][5]。画像認識による外観検査はディープラーニングを用いて、過去に作成した製品の画像を大量に機械に学習させ、データマイニングモデルを作成し、作成した製品の画像をデータマイニングモデルに入力として与えることでリアルタイムで良否判定を行う。しかし、この手法では、完成した製品の画像のみを比較し、良否判定を行なうため、不良品が発生した時、原因の推測や特定ができない。

そこで本研究では、生産設備の製造過程のデータ収集および蓄積を後述する Edgecross [6] を用いて行う。そして、蓄積データを元に良否判定を行うデータマイニングモデルを作成および標準化し、収集したデータと比較して良否判定を行う。また、製造過程のデータを収集することで、生産設備の動作をグラフ化し、可視化することで製品の良否をリアルタイムで判定し、不良品が発生した原因を視覚的に表示する。

本論文の構成は以下の通りである。2章では、本研究で利用した技術や関連研究について述べ、3章では、本研究の提案手法および提案システムの機能について述べ、4章では、実験の内容および実験結果に基づく考察について述べ、5章では本研究のまとめと今後の課題について述べる。

## 2. 関連研究

### 2.1 Edgecross

Edgecross とは産業用 PC 情で動くエッジコンピューティング領域のプラットフォームである。近年、製造業の進化に向け、世界では様々な IoT 活用の動きが見られる。製造業で IoT を活用するためにはエッジコンピューティング技術の活用が重要である。エッジコンピューティング技術を活用することにより、生産現場のデータをクラウドや IT システムにあげるべきデータに一時処理することで、通信量の低減やセキュリティの確保ができる。また、生産現場に近い場所でデータの管理、処理およびフィードバックを行うことで設備保守等のリアルタイム性が必要な業務の最適化ができる。しかし、製造業の IoT 化の現状として、IoT 化に向けたデータ連携では様々な技術要素が混在しているため、システムの複雑さが増している点や、生産現場で蓄積しているデータの中には生産現場を管理するために必要なデータと必要のないデータが混在しているため、生産現場のデータを活用した IoT 化を実現するためにデータの整理が必要となるという課題があった。これらの課題を解決するために、エッジコンピューティング領域のプラットフォームが必要となる。エッジコンピューティング領域のプラットフォームを作成することにより、生産現場の IT システムの異なる技術要素を吸収することにより、シンプルな IoT システムの実現を可能にすることができる。また、生産現場のデータを抽象化および階層化し管理することで業務プロセスごとに必要なデータの抽出を容易にすることができる。エッジコンピューティング領域のプラットフォームを実現するためには企業と産業の枠を超えた協力と協働が必要となる。理由として、生産現場からデータを収集するためには生産設備を管理している Programmable Logic Controller(以下 PLC) に接続する必要がある。しかし、PLC を製造する企業は多数存在し、企業ごとに独自仕様のネットワーク規格を採用しているため、企業間での連携が取れずデータの収集が困難であることが挙げられる。また、同一の企業の PLC でも製造時期が異なれば PLC 間の連携が取れないという課題があるからである。このような経緯から Edgecross が作成された。Edgecross の機能を以下に述べる。

#### 2.1.1 生産現場のあらゆるデータの収集

Edgecross の機能の一つとして、生産現場のあらゆるデータの収集を行う機能がある。Edgecross は生産現場のデータを一定周期 (最大 1ms) で収集することができる。そして収集したデータをデータベースや CSV 形式で蓄積することや、後述する MQTT 通信により、生産現場の稼働監視アプリケーションや予防保全アプリケーションなどの様々なエッジアプリケーションおよび IT システムにリア

リアルタイムでデータの送信を可能にする。生産現場のデータをリアルタイムで送信することにより、エッジアプリケーション内で生産現場のリアルタイム診断および診断した結果を生産現場にフィードバックすることができる。また Edgecross はデータコレクタを搭載している。データコレクタは、CC-Link IE, EtherNet/IP, EtherCAT などの PLC で使用されているネットワーク規格を搭載している。データコレクタにより、生産現場のデータを各 PLC のネットワーク規格に沿ってデータを収集することを可能にする。また、様々な PLC から収集したデータをインターフェースを通じて共通フォーマットに変換し、Edgecross に受け渡す機能や、Edgecross からインターフェースを通じてデータを取得し、生産現場にフィードバックすることを可能にする。データコレクタにより、前述した PLC の課題を解決し、PLC の製造企業、PLC のネットワーク規格にとらわれず、生産現場のあらゆる生産設備、装置からデータを収集することが可能となる。

### 2.1.2 データモデル管理

Edgecross の機能の一つとして、生産現場のデータを抽象化し、階層化して生産現場を管理する機能がある。生産現場から Edgecross が収集する際のデータ名は、D100:100 といった生産現場の状態を知らなければ理解できない様なデータになっている。そのため、このデータをそのままエッジアプリケーションに送信しても、アプリケーション側はこのデータが何のデータなのかを理解することが難しい。そのため、Edgecross が収集したデータに対し、人がタグ付けなどの抽象化を行うことで、電流 (A):100 の様にその生産現場の状態を知らなくてもそのデータの意味がわかる様にする事ができる。また、Edgecross が取得したデータがどこの生産設備のデータなのかを知るために取得したデータを生産設備 X->電流 (A):100 の様に階層化することでデータ同士の関係性がわかる様になる。これらの機能により、膨大な生産現場のデータを人やアプリケーション側で容易に活用することができる。Edgecross は生産設備の製造過程データをリアルタイムで収集、および IoT 通信における標準的な通信プロトコルである Message Queue Telemetry Transport[8](以下 MQTT) 通信によるデータの送受信、DB や CSV 形式で蓄積する機能を有する。

## 2.2 Message Queue Telemetry Transport(以下 MQTT)

MQTT とは、IoT 通信における標準的な通信プロトコルである [8]。MQTT は、Publish/Subscribe 型の通信モデルを採用している。この通信モデルでは、メッセージの送信者側を Publisher、受信者側を Subscriber とし、MQTT サーバがメッセージの中継を行う。本研究では Edgecross を Publisher、提案システムを Subscriber として MQTT 通信を使用する。MQTT は、プロトコルヘッダ長が最小で 2

オクテットと小さく、帯域幅が狭いネットワークや通信遅延の大きいネットワークといった信頼性の低いネットワークでの使用を目的としている。MQTT 通信の機能を以下に述べる。

### 2.2.1 Quality of Service(以下 QoS)

MQTT 通信の機能の一つである QoS とは送信するメッセージの到達保証をサポートする機能である。QoS にはメッセージの到達を 3 段階に分けて保証する。

- (1) QoS0: メッセージの到達が最高で一回。
- (2) QoS1: メッセージの到達が最低でも一回。
- (3) QoS2: メッセージの到達が必ず一回

本研究では、Edgecross が MQTT 通信 QoS2 で生産設備のデータを、提案システムに送信する。本研究では、工場内の生産設備のデータをリアルタイムで大量に取得する必要があり、通信回数が増加するため、通信プロトコルには MQTT を用いる。

## 2.3 推論モデル

製品の特徴を元に品質の良否判定の基準を記述したデータマイニングモデルを推論モデルと呼ぶ。推論モデルの作成方法について述べる。推論モデルを作成するためには、大量の生産設備の過去データが必要となる。そのため Edgecross が蓄積した過去データを元に推論モデルを作成する。推論モデルを作成する例として射出成形機をあげる。射出成形機とは、溶かした金属を金型に注入し、冷やし固め製品とする生産設備である。射出成形機の推論モデルを作成するためには、溶かした金属を金型に注入する際の金属の射出速度、金型内の真空度、鋳造圧力などのデータが必要になる。しかし、Edgecross が蓄積したデータの中には、製品を作成した日時、製品番号、欠損データ等の製品品質の良否判定に不要なデータが混在しているため蓄積データからそのまま推論モデルを作成することができない。そのため、推論モデルを作成する人は、製品の良否判定を行うために有効なデータの抽出、欠損地の補完方法の設定、抽出したデータに対して特徴量抽出 (標準偏差, 分散) などのデータ変換を行い、そのデータを元に推論モデルを作成する。しかし、推論モデルは作成した分析ツールに依存しているため、例えば Python を用いて推論モデルを作成した場合、この推論モデルは Python の実行環境でしか使用することができない。そのため、実行環境が Java であれば推論モデルを Java で再実装する必要があるといった課題がある。

## 2.4 PMML

Predictive Model Markup Language(以下 PMML) とはデータマイニングのマークアップ言語である [7]。PMML は XML を用いて、データマイニングモデルを様々なベンダのアプリケーション間で共有するための規格である。本研究では、R や python で作成したデータマイニングモデル

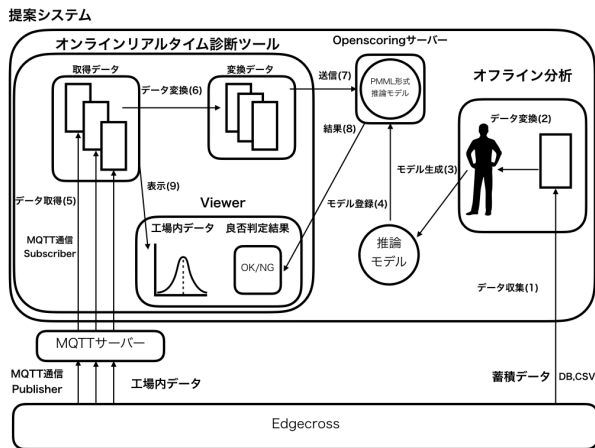


図 2 提案システムのアーキテクチャ

を PMML 形式に変換することで、異なるアプリケーション間でデータマイニングモデルの標準化を行う。

### 3. 提案手法

本研究では、既存の推論モデルを PMML 形式に変換し、Openscoring サーバ [10] に登録することで、アプリケーション間で推論モデルの共有を実現し、柔軟性のあるシステムを実現する。Openscoring サーバは、PMML 形式の推論モデルをデプロイする機能、およびデプロイした推論モデルにデータを与えることで良否判定の計算を行う機能を有する。提案システムではロジスティック回帰分析を用いて作成した推論モデルを使用する。提案システムでは Edgecross から 1ms ごとに製品を作成する過程で生じる生産設備のデータを収集し、製品が完成したタイミングで収集したデータ群に対しデータの変換を行う。その後、Openscoring サーバに変換したデータを送信し、計算を行うことで製品の良否判定をリアルタイムで行う。提案システムのアーキテクチャを図 2 に示す。提案システムが製品の良否判定をリアルタイムで行う手順について述べる。提案システムでは、Edgecross がデータベースや CSV 形式で蓄積したデータを収集し (図 2(1))、外部の分析ツールを使用して製品の良否判定を行うために適したデータ変換をし (図 2(2))、推論モデルを作成して PMML 形式に変換する (図 2(3))。その後、PMML 形式の推論モデルを Openscoring サーバに登録する (図 2(4))。提案システムは、Edgecross から生産設備のリアルタイムデータを MQTT 通信により取得し (図 2(5))、特徴量抽出等のデータ変換を行う (図 2(6))。その後、Openscoring サーバに変換データを送信し (図 2(7))、Openscoring サーバに登録した推論モデルに変換データを与えることで良否判定の計算を行う。最後に、提案システムは Openscoring サーバから良否判定の結果を取得し表示し (図 2(8))、Edgecross から取得したデータ群を波形グラフとして表示する (図 2(9))。提案システムの機能についていかに述べる。

### 3.1 データ変換の自動化

提案システムでデータを変換するためには、外部の分析ツールを使用して推論モデルを作成する際に行なったデータの変換方法を知る必要がある。理由として、変換されたデータを元に推論モデルを作成しているため、その変換方法を行なったデータを入力として推論モデルに与えなければ、適切な計算が行われず、良否判定結果が正しく出力されないためである。しかし、データの変換方法は複雑なため、変換方法を提案システム内で人が入力するのは困難かつヒューマンエラーを起こす原因になりうる。そこで、提案システムはデータの変換を自動で行う。提案システムは、データ変換の種類として、最小値、最大値、標準偏差、分散、合計、範囲の計 6 種類に対応している。提案システムがデータの変換を自動で行う方法について述べる。まず、Edgecross が提案システムに送信するデータ名と推論モデルに記述されているデータ名を一致させる。また PMML 形式の推論モデルに記述されているデータ名をデータ名\_変換方法\_処理区間にする。これらの条件を満たすことにより、提案システムで PMML 形式の推論モデルのデータ名を取得することにより、Edgecross から取得したデータに対し、どのようなデータ変換をどの区間に行うかを把握することができる。この手法により、データの変換を自動化する。

#### 3.1.1 データの欠損補完

提案システムは、Edgecross がリアルタイムで取得した生産設備のデータをから MQTT 通信により取得する。しかし、生産現場には帯域幅が狭いといった特徴があるため、Edgecross から送信されるデータが欠落する可能性がある。そのため、提案システムが取得するデータに抜けが出る可能性があり、製品の良否判定を正常に行えないといったことが予想される。そこで提案システムは、送信されるデータに抜けが出た際、それらを検知し補完する機能を持つ。Edgecross がデータを提案システムに送信する際、生産設備のデータの他にシーケンスナンバを含め送信する。シーケンスナンバは現在、送信されているデータが何個目のデータかを示している。提案システムは、一つ前に送信されたシーケンスナンバを記憶し、次に送られてきたシーケンスナンバと比較する。比較した結果、記憶していたシーケンスナンバに+1 した値より大きかった場合、データが欠落していると判断して補完を行う。補完の方法は、欠落したデータが製品を作成し始めてから一番初めのデータであった場合、0 で埋め、そうでない場合は、一つ前に送信されたデータを複製する方法をとる。この方法により、製品の良否判定を正常に行うことができる。

#### 3.1.2 Viewer

提案システムは、Edgecross から送信される生産設備のデータおよび製品の良否判定結果の分布をグラフとして表示する。製品の良否判定結果の分布グラフを図 3 に Edgecross から送信される生産設備データの波形グラフを図 4

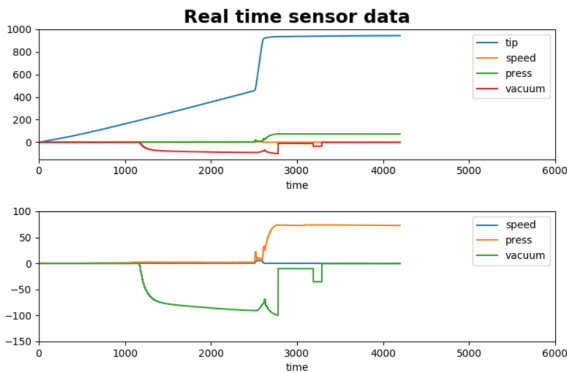


図 3 生産設備データの波形グラフ

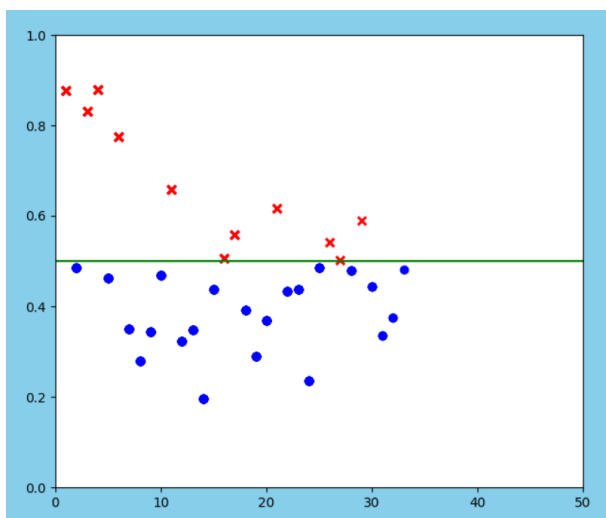


図 4 良否判定結果の分布図

に示す。

Edgexross から送信される生産設備のリアルタイムデータを波形グラフとして表示することにより、完成した製品が不良品と判断された場合、この波形グラフを見ることにより、生産設備の稼働状況を視覚的に確認することで不良品と判断された原因の特定や推測を可能にする。またこのグラフは、Edgexross が送信したデータ群の中に、他のデータと比べ値が大きすぎるなどの突出したデータが存在する場合は、データ群全体の波形グラフの表示の他に突出したデータを除いたデータ群を別グラフとして表示する。これにより、一つのデータが突出した値を持つため他のデータが見れないといったことがないようにした。図 4 は横軸を良否判定を行なった製品の個数、縦軸を不良品の確率としている。製品の良否判定結果の分布をグラフとして表示することにより、過去に作成された製品の良否判定結果を確認することができる。過去に作成された製品の良否判定結果を確認することで不良品と判断された製品が連続した場合、生産設備が故障したのではないかと原因を推測することができる。

次に提案システムの構成について述べる。提案システムが Edgexross に依存するのは Edgexross が送信したデー

表 1 実験環境

CPU	CPU Intel Core i5 2.6GHZ
OS	MacOS Mojave ver 10.14.6
メモリ	8GB

表 2 処理時間の平均と分散

処理時間の平均(s)						
処理の種類/個数	最小値	最大値	標準偏差	分散	合計	範囲
10	0.064	0.065	0.076	0.078	0.069	0.068
20	0.070	0.076	0.097	0.098	0.080	0.079
30	0.079	0.086	0.117	0.118	0.091	0.092
40	0.088	0.092	0.137	0.139	0.101	0.102
50	0.097	0.102	0.158	0.158	0.112	0.115

処理時間の分散

処理の種類/個数	最小値	最大値	標準偏差	分散	合計	範囲
10	1.03-e <sup>5</sup>	8.48-e <sup>6</sup>	2.82-e <sup>5</sup>	5.06-e <sup>5</sup>	2.87-e <sup>5</sup>	3.05-e <sup>5</sup>
20	2.53-e <sup>5</sup>	1.87-e <sup>5</sup>	1.04-e <sup>4</sup>	1.25-e <sup>4</sup>	4.79-e <sup>5</sup>	9.44-e <sup>5</sup>
30	9.60-e <sup>6</sup>	4.71-e <sup>5</sup>	1.12-e <sup>4</sup>	1.48-e <sup>4</sup>	3.35-e <sup>5</sup>	1.24-e <sup>4</sup>
40	4.46-e <sup>5</sup>	4.73-e <sup>5</sup>	1.11-e <sup>4</sup>	2.14-e <sup>4</sup>	5.22-e <sup>5</sup>	1.20-e <sup>4</sup>
50	1.60-e <sup>5</sup>	3.35-e <sup>5</sup>	7.65-e <sup>5</sup>	8.42-e <sup>5</sup>	7.65-e <sup>5</sup>	2.28-e <sup>4</sup>

表 3 Openscoring サーバとの通信時間の平均と分散

サーバとの通信時間の平均(s)

処理の種類/個数	最小値	最大値	標準偏差	分散	合計	範囲
10	0.010	0.008	0.007	0.011	0.006	0.006
20	0.011	0.009	0.008	0.011	0.007	0.007
30	0.011	0.009	0.007	0.011	0.007	0.007
40	0.011	0.010	0.008	0.011	0.008	0.007
50	0.012	0.010	0.009	0.007	0.009	0.007

サーバとの通信時間の分散

処理の種類/個数	最小値	最大値	標準偏差	分散	合計	範囲
10	6.01-e <sup>5</sup>	1.10-e <sup>5</sup>	9.44-e <sup>7</sup>	6.96-e <sup>5</sup>	7.83-e <sup>7</sup>	1.36-e <sup>6</sup>
20	1.73-e <sup>4</sup>	1.36-e <sup>5</sup>	2.11-e <sup>6</sup>	7.46-e <sup>5</sup>	1.06-e <sup>6</sup>	2.36-e <sup>6</sup>
30	6.92-e <sup>5</sup>	1.18-e <sup>6</sup>	1.49-e <sup>6</sup>	1.02-e <sup>4</sup>	1.20-e <sup>6</sup>	1.90-e <sup>6</sup>
40	7.31-e <sup>5</sup>	1.06-e <sup>5</sup>	1.70-e <sup>6</sup>	1.45-e <sup>4</sup>	1.06-e <sup>5</sup>	2.66-e <sup>5</sup>
50	1.34-e <sup>4</sup>	1.09-e <sup>5</sup>	2.42-e <sup>5</sup>	1.06-e <sup>5</sup>	2.89-e <sup>4</sup>	2.88-e <sup>5</sup>

タを受信するデータ取得部分だけである(5)。そのため、Edgexross のバージョンが変わりデータの取得方法が変更された場合でも、その取得方法に応じたデータ取得部分を作成するだけでよく、データ変換部分と良否判定部分の変更は行わずに使用することができる。

#### 4. 実験と評価

提案システムでは、生産設備のデータを取得してから良否判定の結果を取得するまでに、特徴量抽出等のデータ変換と Openscoring サーバとの通信を必要とする。そのため、製品が完成してから良否判定結果を取得するまでに遅延が生じる。遅延により、良否判定の結果を取得する前に、完成した製品が次の工程に流れてしまう可能性がある。そこで、提案システム内で行う処理と Openscoring サーバとの通信にかかる時間の平均と分散を計測した(表 2, 3)。

表 1 に実験環境を示す。表 2, 3 にはそれぞれ、処理の種類(最大値、標準偏差等)と処理の個数に対して、処理にか

かる時間の平均と分散, および Openscoring サーバとの通信時間の平均と分散を記述している. 表 2, 3 から良否判定の結果を取得するために処理にかかる時間と Openscoring サーバとの通信時間は共に線型の関係であることがわかる. このことから, 製品の良否判定を行う処理の種類と個数に応じて製品が次の工程に流れるまでの時間を変更すればよい.

## 5. まとめと今後の課題

本研究では, 生産設備の製造過程のデータを元に, 製品の品質をリアルタイムで良否判定, および不良品と判定された原因を可視化するためグラフを表示するツールの実装を行った. 実験の結果, 良否判定の結果を取得するための処理にかかる時間と Openscoring サーバとの通信時間は共に線型の関係にあることから, 提案システムは問題なく動作していることから製品の良否判定を行う処理の種類と個数に応じて製品が次の工程に流れるまでの時間を変更すればよい.

今後の課題として, 現在の提案システムでは 6 種類の汎用的なデータ変換しか対応していない. データ変換の種類は汎用的な処理だけでも数十種類, 専門的な変換になると無数に存在するため, 推論モデルを作成する人からヒアリングを元に汎用的なデータ変換の種類, および現場で使われる専門的なデータ変換を把握し, それらに対応できるようにしていく必要がある.

## 参考文献

- [1] 辻 敏夫, “ものづくり企業の生産現場における検査の自動化促進可能性調査報告書,” 公益財団法人ちゅうごく産業創造センター, 2016, pp.1-24.
- [2] 製造業における人材不足の現状および外国人材の活用について経済産業省 製造産業局, 2019, pp1-30
- [3] 大竹 守, 日下 裕幸, 柏木正造, 黒澤 公紀, “ディープラーニングの製造への応用,” フジクラ技報, 第 132 号, 2019, pp.36-39.
- [4] 高澤 悠介, 堀江 一央, “ディープラーニングを利用したガラス製品の自動検査システム,” 人工知能学会全国大会論文集, 第 33 回全国大会, 2019, pp.1-3.
- [5] 全 邦釘, 嶋本 ゆり, 大窪 和明, 三輪 知寛, 大賀 水田生, “ディープラーニングおよび Random Forest によるコンクリートのひび割れ自動検出手法,” 土木学会論文集, 73 巻 2 号, 2017, pp.297-307.
- [6] 藤田 正弘, “e-F@crory を支える最新の FA 技術,” 三菱電機技法, Vol.91, No.4, 2017, pp.204-208.
- [7] PMML, <http://dmg.org/pmml/pmml-v4-2-1.html>
- [8] MQTT, <http://mqtt.org/>.
- [9] ダイカストマシンと射出成形の相違点とは?, <http://www.taiyoparts.co.jp/blog/4458>
- [10] Openscoring, <https://openscoring.io/>