

機械学習を用いた画像識別におけるリスク分析手法の提案

高橋雄志¹ 山口晋一² 石川冬樹¹

概要: IoT 時代の到来により様々なシステムが連携した新しいサービスが登場するようになった。中でも人工知能などの機械学習を用いたシステムは、我々の生活に根差したサービスを提供するようになってきている。しかし、それらのサービスを構成するすべてのシステムにおける安全性が確保されているとは限らない。また、人に代わって自動判断を行うようなシステムでは機械学習の中でも画像識別を用いるものが多く、学習モデルの精度向上はサービスの品質向上につながる。これらのことから画像識別を用いた生活に根差したサービスを構築するにあたり、安全性を考慮した学習を行うことは重要であると考え、そこで本稿では、機械学習モデルのリスク分析を学習前と学習後それぞれの特徴に合わせて行うことを提案し、提案手法を実現するためのサポートツールの開発を行ったことを報告するものである。

キーワード: 機械学習, 画像識別, リスク分析

Proposal of Risk Analysis Method in Image Recognition Using Machine Learning

YUJI TAKAHASHI^{†1} SHINICHI YAMAGUCHI^{†2} FUYUKI ISHIKAWA^{†1}

Abstract: Technical phase changing to IoT, then new service was appeared that is several services cooperative. We have a service that uses machine learning that is closely related to our life. But, not all systems that make up those services are secure. In addition, many systems that make automatic decide to behalf of humans use image identification in machine learning and improving the accuracy of learning models leads to improving the quality of services. Based on the above, we consider that it is important to learning with safety in mind when constructing a service using image identification rooted in our life. We propose to perform risk analysis of machine learning models according to the characteristics before and after learning, and report that we have developed a support tool of the proposed method.

Keywords: Machine Learning, Image Recognition, Risk Analysis

1. はじめに

様々な技術革新によって IoT (Internet of Things) と呼ばれる様々な「モノ」が相互につながり新たなシステムやサービスが登場し我々の生活は豊かになってきている。具体的には、収集したビッグデータから新しい知見を得たり、システム制御が自動化されたりといった恩恵を受けている[1]。生活面でも自動運転の実現にむけて様々な試みがなされている[2]。自動運転のような生活に根差したサービスを提供するためには安全性を考慮した設計が必要となってくる。しかし、つながったすべてのシステムの安全性が確保されているとは限らない[1]。とりわけ機械学習で構築されたモデルはブラックボックス状態でモデルの挙動がわからないものが多い。

自動運転を代表とする人の代わりに状況判断をしてサービスを提供するシステムの多くは、画像認識/識別(以降まとめて画像識別とする。)が重要な役割を果たしている。このような画像識別の多くは、機械学習による学習済みモデルを用いて識別を行っている。闇雲に学習を繰り返して

も過学習によってモデル全体の精度が下がってしまったり、狙った効果が得られなかったりといった問題がある[3]。このような問題に対して、修正対象を絞った変更をする研究もなされている[4]。特に自動運転システム(ADS)などの自動運転機能は機械学習が重要な役割を果たし、安全性の保証の必要性が Koopman らにより提唱されている[5]。また、自動運転に用いられる機械学習モデルを用いた画像によるシーンの分類問題においては、現実的なシナリオでは損失を引き起こす分類ケースの数が急速に増加する可能性があり、安全性分析を扱いやすくするためにその複雑さを効率的に軽減する技術が必要であることが言及されている[6]。以上のことから、機械学習のモデル構築において安全性を考慮し、かつ効率的/効果的な学習を行うことが重要であると考え、

我々は、画像識別の学習モデルと対象とした学習前/中(以降まとめて学習前)と学習後に分けた安全性に基づくリスク分析手法を提案する。本稿では、特に学習前のリスク分析手法について述べ、開発を行ったサポートツールの

¹ 国立情報学研究所
National Institute of Informatics

² 慶應義塾大学大学院附属 SDM 研究所
Keio University SDM Research Institute

紹介をする。

2. 関連研究／技術

本章では、提案手法と同じように自動運転を例に安全性評価を行っている研究／技術を紹介し差分を示す。

2.1 システム全体の安全性評価に関する研究

村本らによって大阪地区での自動運転ライドシェアサービスを対象としたサービス設計ならびに評価手法の提案がなされている[7]。この研究では、実運用しているサービスに対して、経済性、利便性、安全性の3軸を評価軸としてサービス評価を行っている。安全性の評価は、リスク値を金額換算している、まず、事故発生時の車両速度を特定して運動エネルギーを算出する。次に、障害の重篤度を簡易傷害スケール（AIS: Abbreviated Injury Scale[8]）に基づき算出する。この評価の前段階として、人流データを蓄積し、走行ルートにおける自動運転車両と人との近接回数を導き出す。そしてハインリッヒの法則に基づき事故発生確率を求めている。

本稿で提案している手法とは評価する範囲と目的が異なり、この研究はサービスモデル全体を対象とした評価を行っており、提案手法は画像認識の学習モデルを対象とした評価を行っている。

2.2 安全性に着目しテストケースの削減を行う研究

Amersbach らによって、機能分解の分解アプローチとシナリオベースのアプローチを組み合わせることで冗長なテストケースを排除しテスト工数を削除してもシステムの安全性が確保されていることを示す研究が行われている[9]。

この研究では、高度な自動運転を対象として、すでに提案されている自動運転機能の6層の分解アプローチとFTAによる機能分解を使用して、関連するシナリオを特定して、特定したシナリオに基づきテストケースを導出している。導出されたテストケースから冗長なテストケースを削除したり、集約したりすることで工数の削減を実現している。

我々の提案手法は、教師データとなるデータセットのラベル情報に基づきシーンごとのリスク値を求め比較して相対的にリスクの高いシーン（またはそのシーンにおける非安全なシナリオ）を見つける手法となり、アプローチの方法が異なる。

3. 提案手法

3.1 提案概要

本研究では、画像認識に対するリスク分析にあたりセキュリティリスクの考え方の脆弱性、脅威、影響度の関係性を参考にロジックツリー（Machine Learning Risk Analysis Tree: MLRAT）を構築することを提案するものである。

脆弱性に該当する条件を画像情報に基づく環境条件（条

件 A）、脅威に該当する条件を事故条件（条件 B）、影響度に該当する事故のインパクトを影響度（条件 C）としてそれぞれの観点でリスクを求めて総合リスクを判定する。それぞれの例を表1に示す。

表1 各条件の例

	条件A	条件B	条件C
	環境条件	事故条件	影響度
例	天候	事故にならない場合があるのか	直接的な影響度合い
	時間帯	画像認識によらない事故回避はできるのか	間接的な影響度合い
	背景		
	識別すべきオブジェクト		

また、学習前と学習後ではその条件に対する要求が異なると考える。

学習前では、様々な調整が入ったり、事前の情報収集に多くの時間をかけることが難しかったりと想定されるので、厳密な発生確率を考慮しないAgilityの高いリスク分析を行うことが望ましいと考えた。一方、学習後は、構築されたモデルの安全性をモデル作成者以外にも示す必要があるため学習前よりも高い根拠性が必要になると考えた。

そこで、本提案方式では学習前と学習後に大きく分けて2パターンでのリスク分析の手法を提案する。

3.1.1 学習前のリスク分析

【条件 A】学習に用いるデータセットのラベル情報をもとにパラメータ（以下、属性）を抽出し、属性値ごとにリスク値を付与してシーンごとのリスクを算出する。ここでいうシーンとは、画像の本条件に基づく属性値が確定した組み合わせのこととなる。

また、画像のシーンを示す情報を背景条件（条件 A-1）、その画像内で識別すべきオブジェクトを物体条件（条件 A-2）と分けて考えるものとする。条件 A-1 となる属性は、画像ごとに一意に決まるものであり、例えば、天候や時間帯といったような情報が該当する。提案手法では、この条件 A-1 の情報を基準に以降の条件を紐づけていくことになる。一方条件 A-2 となる属性は、画像ごとに必ずしも属性が存在するとは限らないし属性値が複数存在することもあり得る、例えば、障害物として、歩行者2人と車、信号機のように画像内に複数写りこんでいてそれぞれが属性値となるようなものが該当する。ただし、条件 A-2 の情報が分析対象システムのリスク分析上重要な役割を果たす場合は、属性と属性値の表現を工夫して条件 A-1 のように扱うこともできるものとする。

【条件 B】条件 A-1 において、画像認識に失敗しても実際に事故になるとは限らないケースがあると考えられる。今回の提案方式では、(1)画像認識の失敗があったとしても事故に至らない例外条件の有無、(2)その他のセンサーなどによる識別手段があり、画像認識の結果がメインの判断基準とならないような代替手段の有無もしくは画像認識が失敗した際の対策の有無、を条件 A-1 の組み合わせごとに判別

しリスク値を考える。

【条件C】条件A-1のシーンにおいて、実際に事故が発生した場合に考えられる影響の大きさを、直接的なインパクトと間接的なインパクトの両側面からリスク値を与える。

例えば直接的なインパクトでは、人身に影響がある場合に大、その他物理的・金銭的に影響がある場合に中、あまり影響がない場合に小。間接的な影響は、組織イメージが損なわれる場合に大、他のサービスに影響が出る場合に中、あまり影響がない場合に小、のように評価を行う。

条件AとBについては、各属性に対して客観的な評価を与えて、条件Cでは対象としているサービスや所属組織の意向などを加えた主観的な評価を行うことをイメージしている。

以上の概念は図1のようなロジックツリーで表すことができ、MLRATbt (Machine Learning Risk Analysis Tree for before training) と名付けるものとする。

本稿は、このMLRATbt に対しての分析のアシストツールを作成した結果を4章以降で述べる。

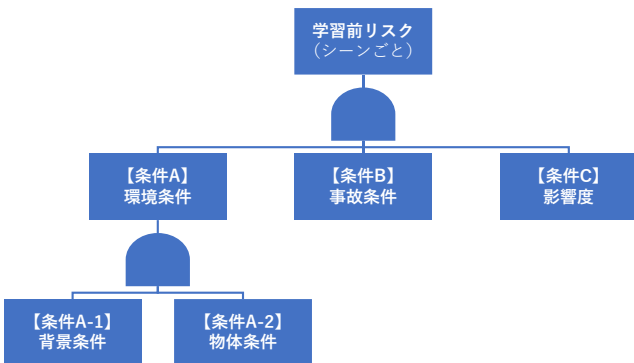


図1 Machine Learning Risk Analysis Tree before training

3.1.2 学習後のリスク分析

MLRATbt をベースに、学習結果のエラーレート (FP/FN) や、事故発生確率、シーンの遭遇率といった確率データを加えて総合的なリスク分析を行う。

まず、ラベルの示すシーンを 3.1.1 節の条件 A-1 ごとに、検出対象物が移っている該当シーンと、検出対象物が移っていない類似シーンに分ける。続けて該当シーンでは、該当シーンとなる確率 ($Pnss(i)$) と未検知 (FN: False Negative) の発生率 ($Pnsf(i)$) を掛け合わせ、類似シーンでは類似シーンとなる確率 ($Ppss(i)$) と誤検知 (FP: False Positive) の発生率 ($Ppsf(i)$) を掛け合わせてエラー発生度合い ($Pns(i)/Pps(i)$) を算出する。次にエラー発生度合いに、それぞれの事故発生確率 ($Pna(i)/Ppa(i)$) と事故発生時の影響度 ($In(i)/Ip(i)$) を掛け合わせて、FN/FP 由来のリスク値 ($Rn(i)/Rp(i)$) を算出する。算出した 2 つのリスク値を加算して条件 A-1 におけるリスク値 ($R(i)$) を算出する。この算出されたリスク値の総和が学習後のモデル全体の残存リスクとなる。略称とその意味を表 2 に示す。

表 2 略称と意味

R	Risk
$Rn(p)$	Risk of false Negative(Positive)
$Pn(p)s$	Percent of false Negative(Positive) at the Scene
$Pn(p)S$	Percent of false Negative(Positive) at the Scene using Scene frequency
$Pn(p)sf$	Percent of false Negative(Positive) at the Scene using Failure rate
$Pn(p)a$	Percent of false Negative(Positive) at the Accident
$In(p)$	Impact of false Negative(Positive)

その概念は図2のようなロジックツリーで表すことができ、MLRATbt に対して MLRATat (Machine Learning Risk Analysis Tree for after training) と名付けるものとする。

本稿では、MLRATat は概念のみの紹介とし今後検討を進めて詳細を発表の予定である。

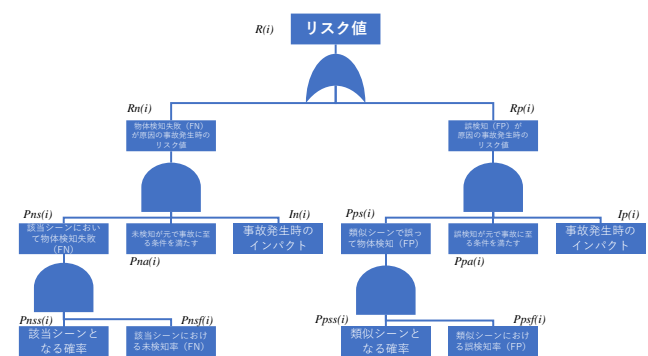


図2 Machine Learning Risk Analysis Tree after training

4. アシストツール

MLRATbt を用いた分析を行う際に、手作業で属性ごとのパラメータを集計してまとめあげるのは手間がかかり、ヒューマンエラーや計算ミスなどの入り込む余地が多いことが考えられる。そこで、アシストツールの開発を行い、リスク分析の労力を削減することを考えた。作成したツールはロジックツリーと区別するため以降は、MLRA-ATbt (Machine Learning Risk Analysis Assist Tool for before training) と呼称する。

本稿で作成したツールは、公開されている自動運転向けのデータセット BDD100k[10]をベースに GitHub 上に公開されているフォーマット[11]に則り、属性と属性値を設定してツールの構築を行った。ツールは Excel VBA で開発を行い、データストレージとして Access DB を使用した。以下、ツールの動作手順を追いながら各種設定について述べる。

4.1 初期設定

初期値として、適用システム名、使用するデータセット名、ストレージのパスを与える。同じデータセットを使用して学習する場合でも、適応先のシステムが異なることで要件が変わることが想定されるので、分析結果はシステムとデータセットの組で管理を行うようにした。また、複数のシステムやデータセットに関する分析を行うことができ

るようするため、分析結果を DB に蓄積できるようにしている。

4.2 条件 A の設定

条件 A-1 と条件 A-2 として使用する属性と属性値を設定する。

【条件 A-1：背景条件】

シーンを識別するための環境条件として、attributes の情報を使用した。今回は、条件 A-1 に使用できる条件数を入力項目数の爆発的増加をさせないために 3 種類に限定した。詳細を表 3 に示す。

表 3 BDD100k に基づく条件 A-1

属性	属性値						
weather	rainy	snowy	clear	overcast	undefined	partly cloudy	foggy
scene	tunnel	residential	parking lot	undefined	city street	gas stations	highway
timeofday	daytime	night	dawn/dusk	undefined			

【条件 A-2：物体条件】

各シーンに移っている検出対象物として、Road object categories の情報に、検出対象物が何も移っていない状態 [none] を追加したものを使用した。その際に、サブパラメータを持つ属性もあるが、今回は使用せず属性のみを使用することとした。条件数も 1 種類だけとした。詳細を表 4 に示す。

表 4 BDD100k に基づく条件 A-2

属性	属性値					
Road object categories	bike	bus	car	motor	person	rider
	trafficlight	trafficsign	train	truck	[none]	

4.3 条件 A のパラメータ設定

条件 A-1 の各属性値に対して遭遇（使用）頻度（以下、頻度）、重要度／優先度（以下、重要度）を入力する。例えば、通勤バスに採用する自動運転システムであれば、会社への行き帰りの時間帯では頻度が高く、深夜や就業時間中などは低い値となる。通勤経路として街中を通り高速道路使用しない場合は、街中での重要度が高く、高速道路の重要度は低いといったような入力を行う。

今回は、頻度と重要度について表 5 で示すような値を採用した。この値は条件 A-2 と共通の値となる。入力画面を図 3 に示す。

表 5 条件 A 入力値

【頻度】					
日常的に遭遇する	遭遇する	たまに遭遇する	ほとんど遭遇しない	対象外	
10	5	3	1	0	
【重要度】					
非常に高い	高い	中程度	低い	対象外	
10	5	3	1	0	

weather	日常的に遭遇する	遭遇する	たまに遭遇する	ほとんど遭遇しない	対象外	リスク値	非常に高い	高い	中程度	低い	対象外	リスク値	リスクの和
rainy	10	5	3	1	0	36	10	5	3	1	0	36	36
snowy													0
clear							10	5	3	1	0	10	20
overcast													0
undefined													0
partly cloudy													0
foggy													0

図 3 条件 A-1 の入力画面

個別パラメータの入力後、全体をリスト表示して特殊な

組み合わせでリスク値に変更が必要な場合は変更を行う。

例えば、「雨と夜はそれぞれの遭遇頻度も低い組み合わせの場合にニーズや危険度増す」というような想定がなされる場合は、図 4 で示すリスト画面からリスク値を修正して情報を更新する。

条件1	【頻度】	【重要度/優先度】	【合計】	条件2	【頻度】	【重要度/優先度】	【合計】	条件3	【頻度】	【重要度/優先度】	【合計】	【小計】	
weather	0~10	0~10		scene	scene	0~10	timeofday	和	timeofday	0~10	0~10	和	総和
rainy	3	5	8	tunnel	3	5	8	daytime	10	10	20	20	36
rainy	3	5	8	tunnel	3	5	8	night	3	3	6	6	24
rainy	3	5	8	tunnel	3	5	8	dawn/dus	3	3	6	6	26
rainy	3	5	8	tunnel	3	5	8	undefined	1	1	2	2	18
rainy	3	5	8	residential	10	10	20	daytime	10	10	20	20	46
rainy	3	5	8	residential	10	10	20	night	3	3	6	6	36
rainy	3	5	8	residential	10	10	20	dawn/dus	3	3	6	6	38
rainy	3	5	8	residential	10	10	20	undefined	1	1	2	2	30

図 4 条件 A-1 リスト画面

続けて条件 A-2 の各属性値に対しても、条件 A-1 と同じように頻度と重要度を入力する。条件 A-2 を加えた状態で特殊な組み合わせがある場合は条件 A-1 の時と同じように図 5 で示すリスト画面からリスク値を修正して条件 A 全体のリスク値を確定させる。

条件 A-1：背景条件			【小計】	条件 A-2：物体条件		【総計】	標準化
条件1	条件2	条件3		Road object categories	リスク値		
weather	scene	timeofday		属性値	リスク値	80.00	
rainy	tunnel	daytime	36	bike	15	51	63.75
rainy	tunnel	daytime	36	bus	6	42	52.50
rainy	tunnel	daytime	36	car	20	56	70.00
rainy	tunnel	daytime	36	motor	15	51	63.75
rainy	tunnel	daytime	36	person	20	56	70.00
rainy	tunnel	daytime	36	rider	15	51	63.75

図 5 条件 A-2 リスト画面

最後に、リスク値の総和を 100 点満点換算（以下、標準化）して各シーンにおける条件 A のリスク値とする。

条件 A-2 として使用している属性は各画像に複数件登場する可能性があるが、1 件分のリスクのみ取り扱う仕様としている。詳しくは課題の章で述べるが、複数件登場する場合や特殊な組み合わせなどについては、今後の課題としている。

4.4 条件 B のパラメータ設定

条件 A の組み合わせごとに値を設定していくことになるが今回のケースでも 2,156 (7×7×4×11) 通りの組み合わせがあるためあらかじめ標準設定を決めておき、異なる場合にのみ値を入力するものとした。入力値としては(1)例外条件の有無、(2)代替手段（もしくは対策）の有無をそれぞれ入力する。今回それぞれの組み合わせで取る値を表 6 に示す。

表 6 条件 B 入力値

	代替手段			
	無	有		
	3	1		
例外条件	無	3	9	3
	有	1	3	1

標準設定と入力値を確定したら、続けて個別設定が必要なシーンに対して図 6 で示すリスト画面から設定を行い、リスク値を更新する。条件 B についても標準化をして各シーンにおける条件 B のリスク値とする。

条件A-1: 背景条件			条件A-2: 物体条件		条件B: 事故条件		リスク値	標準化
条件1	条件2	条件3	代替手段	例外条件	無	有		
weather	scene	timeofday	Road object categories					
rainy	residential	daytime	bike	無	無	無	9	100.00
rainy	residential	daytime	bus	無	無	無	9	100.00
rainy	residential	daytime	car	無	有	有	3	33.33
rainy	residential	daytime	motor	無	無	無	9	100.00
rainy	residential	daytime	person	有	無	無	3	33.33

図6 条件Bリスト画面

4.5 条件Cのパラメータ設定

条件Bと同様に標準設定を決めて、条件Aの組み合わせごとに(1)直接的な影響と(2)間接的な影響に分けて考えてそれぞれの影響度を設定していく。

今回は、(1)(2)共に影響度を大中小の3段階で入力するものとした(表7)。

表7 条件C入力値

		直接的な影響			
		大	中	小	
間接的な影響	大	3	9	6	3
	中	2	6	4	2
	小	1	3	2	1

それぞれの値を選択する基準としては、以下のようなものが目安になると考えている。

(1) 直接的な影響

- 大：人身事故に至ると予測される場合
- 中：物損事故に留まると予測される場合
- 小：影響は少ない(主だった被害がない)場合

(2) 間接的な影響

- 大：企業イメージを損なう恐れがある場合
- 中：無視できない影響が出る場合
- 小：影響は少ない(主だった影響はない)場合

条件Cでも条件Bと同様に、標準設定と入力値を確定したら、続けて個別設定が必要なシーンに対して図7で示すリスト画面から設定を行い、リスク値を更新する。条件Cについても標準化をして各シーンにおける条件Cのリスク値とする。

条件A-1: 背景条件			条件A-2: 物体条件		条件C: 事故のインパクト		リスク値	標準化
条件1	条件2	条件3	直接的な影響	間接的な影響	無	有		
weather	scene	timeofday	Road object categories					
rainy	residential	daytime	bike	小	大	大	3	33.33
rainy	residential	daytime	bus	中	小	小	2	22.22
rainy	residential	daytime	car	大	大	大	9	100.00
rainy	residential	daytime	motor	中	小	小	2	22.22
rainy	residential	daytime	person	大	中	中	6	66.67

図7 条件Cリスト画面

4.6 グラフ表示

MLRA-ATbtでは、すべてのパラメータ入力完了後に、シーンを選んで図8で示すようなバブルチャートを用いたグラフ表示をすることができる。

グラフのx軸は条件A、y軸は条件B、z軸は条件Cの標準化したリスク値を示している。グラフ表示できるのは、条件A、B、Cを検索条件として絞り込んで比較したいシー

ンを最大20件までとしている。

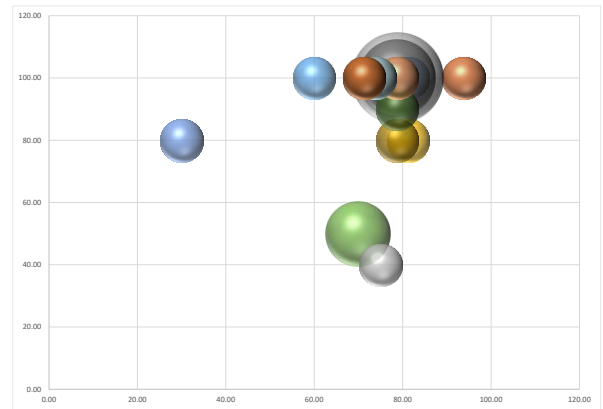


図8 分析結果のバブルチャート

このようにバブルチャートで表示することで原点からの距離やバブルのサイズで直感的にシーン間のリスクの差を捉えることができるようになった。また、数値入力段階で予想していなかった組み合わせが、グラフの右上に表示されることもあり、潜在的にリスクの高いシーンを見つけることができると予想される。

5. 考察と今後の課題

本稿では、MLRA-ATbtの条件Aの内容を決めるために仕様が公開されているBDD100kを使用した。しかし、BDD100kを基準に安全性の議論を進めるにあたり属性情報が不足しているのではないかと考える。例えば、同じ夕暮れ時でも逆光であるとか、人が写っている場合であっても交差点などを渡っている瞬間であるとか直感的に事故につながると認識できる情報がラベル上で表現されているべきであると考えた。

条件A-1とA-2として区別する際に、条件A-1は各画像の必須項目、条件A-2は必須ではなくオプション項目の位置づけとなっているが、事故に特化した画像情報とするためには、信号機や人などの属性も必須項目(条件A-1)とすることも考えられる。しかし、信号機や人などは各画像に1つのオブジェクトとならず、どのオブジェクトの情報も条件A-1とできるのかという課題がある。例えば、信号機であれば「識別すべき信号機」のような属性を追加して属性値として色などの情報を持たせることなどが考えられる。同様に人も「注目すべき人」のように属性を追加して属性値として「車両通行帯(道路)上、歩行者通行帯(歩道)上、交差点横断中」のような値を持たせることで提案方式でも条件として扱えるのではないかと考える。ただし、どのようにして、「識別すべき信号機」や「注目すべき人」を選ぶのかという別の課題を解決する必要がある。

本稿では、条件A-2に該当するオブジェクトが複数写っている場合に最もリスク値が高いオブジェクトの情報を用いることとしている。しかし、複数のオブジェクトが写っ

ている際、特殊な組み合わせとなる場合にリスク値が変更できる方がよいのではないかと考えた。本稿のツールでは、条件 A-2 の属性値として複数入っている状態の属性値をあらかじめ登録しておくことで対応が可能であるが、事前に特殊な組み合わせを定義する必要がある。例えば、**bike** と **rider** が一緒に写っている場合は自転車と同じように動くことがあるオブジェクトとして認識し、**bike** 単独で移っている場合は停車しているため動かない障害物として認識することで、そこから想起される事故の状況が異なると推察できる。

また今回は、条件 A-1 と条件 A-2 に対する入力を共通で頻度と重要度とし、条件 B の入力を代替手段と例外条件の有無としたが、条件 A-2 の物体識別に関して未認識(以下、FN : False Negative) と誤認識(以下、FP (False Positive)) とでは発生する事故や対策状況が異なることが予測される。自動運転の例で考えれば、人に対する FN が起こると人身事故に発展すると想像できるが、人を他の障害物と識別する FP が起こった場合に障害物に対する回避や警戒をするというアルゴリズムがあれば事故には発展しないと判断することができる。条件 A-2 や条件 B に対する入力エラー内容を加味した入力値とすることでより正確なリスクを測ることができるのではないかと考える。

提案方式では、属性値ごとにリスクを割り振っていくので潜在的にリスクが高い組み合わせ(シーン)を見逃すことがなかった。また、ツールでサポートすることですべての組み合わせのリスク値を自動計算されるため網羅的にリスク値を出すことができた。

本稿では、MLRATbt を中心に議論とツール開発を行ったが、学習後のモデル全体を評価するための MLRATat についても同様に議論、ツール化を進めていきたい。学習後に使用する MLRATat では、学習後モデルのテスト結果から FN (False Negative) /FP (False Positive) のエラーレートや事故発生確率、事故発生時にかかるコストなどより現実的に即した値を採用することでモデル全体の評価につながるかと考えている。

また本稿では、題材として自動運転で用いる場合を取り上げて提案方式の議論を進めてきた。今後は、自動運転以外の画像識別に対しても提案方式が同じように適用できるのかということも検証していく。

6. おわりに

画像識別を題材に、機械学習モデルの学習前と学習後に分けて教師データのラベル情報に基づくリスク分析手法の提案を行った。また、学習前のリスク分析をサポートするためのツールの作成を行った。

提案方式を用いることで、すべてのシーンに対して個別にリスクを考える場合に対して潜在的にリスクの高いシー

ンの取りこぼしやリスク値の設定もれを防止することができた。

そして、ツールによるサポートで効率的なリスク値の算出、グラフ表示による可視化をすることでシーン間のリスク値の比較を直感的に実現することができた。

また、ツール化や実際に BDD100k のデータをサンプルとして運用をすることにより 5 章で述べたように多くの考察や課題を得ることができた。

今後は、5 章で述べた課題に取り組み MLRATbt の改良を目指し、MLRATat についても議論、ツール化を進めていく。

謝辞 eAI プロジェクト 交通 PoC ワーキンググループの皆様には、本稿をまとめるにあたりご助言をいただきました。心より感謝申し上げます。本研究は JST 未来社会創造事業 JPMJMI20B8 の支援を受けたものである。

参考文献

- [1] つながる世界の開発指針～安全安心な IoT の実現に向けて開発者に認識してほしい重要ポイント～第 2 版, 独立行政法人情報処理推進機構 (IPA) 社会基盤センター
- [2] “自動走行ビジネス検討会 - 報告書「自動走行の実現及び普及に向けた取組報告と方針 Version5.0」～レベル 4 自動運転サービスの社会実装を目指して～”, https://www.meti.go.jp/shingikai/mono_info_service/jido_soko/20210430_report.html, (参照 2021-9-9)
- [3] AI が学習しすぎる? 「過学習」問題とそれを抑制する方法, AI 研究所, <https://ai-kenkyujo.com/kagakushu/> (参照 2021-12-10)
- [4] Shogo Tokui, Susumu Tokumoto, Akihito Yoshii, Fuyuki Ishikawa, Takao Nakagawa, Kazuki Munakata, and Shinji Kikuchi. NeuRecover: Regression-controlled repair of deep neural networks with training history. In The 29th IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER 2022), 2022
- [5] P. Koopman and M. Wagner, “Challenges in autonomous vehicle testing and validation,” SAE International Journal of Transportation Safety, vol. 4, no. 1, pp. 15–24 (2016)
- [6] R. Salay, M. Angus, and K. Czarniecki, “A safety analysis method for perceptual components in automated driving,” IEEE 30th Int. Symposium on Software Reliability Engineering (ISSRE), pp. 24–34, (2019).
- [7] 村本衛一, 河本弘和, 東島勝義, 古川量也, 澤井薫, 知念賢一, 篠田陽一, 三浦基嗣: 人の生活圏での自動運転電動車両の運行のサービス設計・評価手法の一考察, 情報処理学会研究報告コンシューマ・デバイス&システム (CDS), Vol.2021-CDS-32 No.14, pp.1-8, (2021.9.3)
- [8] “AIS”, <https://www.aaam.org/abbreviated-injury-scale-ais/>, (参照 2021-9-9)
- [9] C. Amersbach and H. Winner, in 8. Tagung Fahrerassistenz. Munich: Technical University of Munich, (2017)
- [10] BDD100K A Diverse Driving Dataset for Heterogeneous Multitask Learning, <https://www.bdd100k.com/> (参照 2021-12-9)
- [11] <https://github.com/ucbdrive/bdd100k/blob/master/doc/format.md> (参照 2021-12-9)