CF-003

ユーザーの行動パターンに注目した EC サイト上の不正取引検知
E-commerce Fraud Detection Focused on User's Behavior Pattern

野村和也 † 小川翔大 † 中野翔 † 座間味卓臣 ‡ 内田真人 †
Kazuya Nomura Shouta Ogawa Sho Nakano Takumi Zamami Masato Uchida

1. はじめに

Amazon や楽天市場など、商品を売買するための Web サイトを EC サイトという。経済産業省によると、近年ではインターネットの発達や物流網の整備拡充により、EC サイトの需要や市場規模は、2017 年までの 7 年間で 2 倍以上に成長している [1]。しかし、こうした EC サイト上の決済では、ユーザーの意図しない不正な取引が脅威となっている。例えば、日本クレジットカード協会によると、クレジットカードの不正利用による被害拡大の大きな要因の一つとして、EC サイトの普及や発展が挙げられている [2]。また、インターネット上での決済にクレジットカードを利用する人が 63%も占めているというアンケート調査結果もある [1]。さらに、2017 年におけるクレジットカード不正利用の被害総額は 236 億円であり、2015 年、2016 年と比較しても大幅に増加しているという報告もある [2]。本研究では、EC サイト上の取引のうち、クレジットカードの不正利用やアカウントのなりすまし等、不正な決済のあった取引について、その特徴を分析し、機械学習を用いて検知することを目的とする。

本研究で使用するデータは、EC サイトを運営する企業から提供された実際の取引データである。一般に、クレジットカード等の取引を記録した研究用の公開データは、プライバシー保護の観点から、実際の取引データから大幅に情報が削られている。一方、本研究で用いたデータにおいても、プライバシー保護のための適切な匿名加工処理等が施されているが、研究用の公開データと比較し、極めて詳細な情報を含んでいる。また、実際の全取引データからサンプリングされているものの、2017 年間 1 年分の約 3, 500 万件の取引データを持つ、極めて大規模なデータである。さらに、EC サイト特有の不正取引において、不正のタイプが 3 種類設定されていることも本データセットの特徴である。本研究では、このようなデータを用いて、ユーザーごとに「初回購入であるか」、「普段の商品の購入個数や取引の間隔からどれだけ違いがあるか」などの行動パターンに注目した特徴量を設計し、機械学習による不正取引検知を試みる。本論文の構成は以下のとおりである。第 2 節では関連研究について説明する。第 3 節では本研究の提案手法について述べ、またデータセットの説明、分析を行う。第 4 節では実際に本研究の提案手法に基づいた実験を行う。最後に第 5 節では本研究の結果、結論をまとめ、第 6 節では本研究の今後の課題について述べる。

†早稲田大学 基幹理工学研究科 情報理工・情報通信専攻
Department of Computer Science and Communications Engineering, Waseda University, Tokyo, Japan.
‡合同会社 DMM.com

2. 関連研究

不正取引検知に関する先行研究をいくつか取り挙げる。Bahnsen ら [3] は本研究と同様、特徴量設計というアプローチによるクレジットカードの不正取引検知を目的としている。しかし、本研究で用いるデータセットには複数タイプの不正が存在し、クレジットカード取引に関する不正はその一部である。本研究では、クレジットカード不正を含む EC サイト特有の多様な不正取引の検知を目的とする。 一方、Raju ら [4] は、同じく EC サイト上での不正取引検知について検討している。しかし、Raju らの研究は特徴量の生成に着目していない。また、データセット中の取引件数も 15 万件程度であり、本研究と比べると非常に小規模なものである。また、不正のタイプを示すラベルも 1 種類のみである。本研究では EC サイト上の 1 年間分に及ぶ約 3,500 万件もの詳細な取引データを分析し、複数タイプの不正取引の検知に有用な特徴量を設計し、機械学習による不正取引検知を試みる。

一方、クレジットカード等に限らない不正検知システムの構築についても、数多くの先行研究がある。 例えば Lee ら [5] は、インサイダー取引の検知や侵入検知、 サイバーセキュリティにおける障害検知や内部脅威の検出などを不正検知の例として挙げている。Wang ら [6] は、このような不正検知システム全般に関する課題や研究を幅広くサーベイしており、主に「クラス間のラベル数の不均衡 (Class Imbalanced Problem)」と「時系列によるデータの変化 (Concept Drift)」が不正検知の共通の課題であることを示している。例えば、多くの機械学習アルゴリズムでは、クラス間のラベルが不均衡であるとうまく機能しないことがある。これは、予測結果を多数派クラスのみにすれば正解率が高くなるためである。しかし、一般的に不正取引の割合は正常取引に対しごく僅かである。これが Class Imbalanced Problem である。また、不正取引を働く者は検知を逃れるために、自らの行動を変化させることがある。例えば「この IP アドレスから来たアクセスは不正取引が多い」、「このユーザーエージェントは一般的に使用されない」などの情報が予めわかっていた場合、不正取引を働く者が自らの行動を意図的に変えるのは容易である。このような時系列によるデータの特性の変化を Concept Drift という。両者は本研究で検知する不正取引検知においても課題となり得るものである。このうち、本研究ではクラス間のラベル数の不均衡 (Class Imbalanced Problem) への対処を課題の一つとして取り上げている。

3. 提案手法

本節ではまずはじめに、本研究で使用するデータセットの概要について説明する（3.1 節）。その後、このデー

タセットを用い、不正取引の検知に効果的な特徴量の設計を行う（3.2〜3.3 節)。また、複数の不正タイプを区別して検知することを可能とするために、不正タイプごとにモデルを学習し、それらを統合して不正取引を検知する手法について説明する (3.4 節)。

3.1. データセットの概要

本研究では、EC サイトを運営する企業から提供された実際の取引データを使用する。本取引データは、この EC サイト上で行われた全ての取引データから、正常取引のみをアンダーサンプリングしたものである。一方、不正取引については全データが保持されている。データの取得期間は 2017 年 1 月 1 日から 12 月 31 日までの 1 年間分であり、データセットに含まれる取引数は 35,160,546 件である。なお、本データの提供元である EC サイトの運営企業には商品のカテゴリごとに複数の事業部が存在し、それら複数事業部の商品を一括で決済、取引することができる。このように 1 度の取引で扱う商品が複数事業部にまたがる場合、本データセットにおいては、それらを事業部 (商品のカテゴリ) ごとに区別し、異なる取引であるとみなす。

第 2 節でも述べたとおり、一般的に不正取引の数は正常取引の数より圧倒的に少ない。このクラスの偏りは検知精度に大きく影響する。そこで、まず各クラスのラベル数を確認する。全取引データ 35,160,546 件のラベルの内訳を表 1 に示す。関連研究と同様、本研究で用いるデータセットにおいても不正取引が正常取引に比べ圧倒的に少ないことがわかる。また、不正タイプが 3 種類あるうち、不正タイプ 1、2、3 の順に件数が多いこともわかる。ただし、どの不正タイプがどのような種類の不正取引に対応するかについては、本データの提供元から明らかにされていない。

本データセットに含まれる変数を表 2 に示す。 表中のタイムスタンプは日時のデータを示し、具体的には年月日と秒単位での時間が記録されている。商品の購入個数を表す変数である sales_num と、タイムスタンプ値を取る変数である return_date と purchase_date は量的変数であり、それ以外は質的変数である。

3.2. 不正取引の特徴分析

本研究で扱うデータセットのサイズは、第 3.3 節で説明するような特徴量の追加や加工を施す前の元データの時点で約 2GB と非常に大きい。そこで、データの大要を把握し検知に有効な特徴を見つけるため、特徴量設計に先立ちデータセットの分析を行う。具体的には不正取引と正常取引のヒストグラムを各変数について図示し、その偏りや分布の差異を比較することで、不正取引の特徴を考察する。特に、本研究ではユーザーごとの通常時の振る舞いとの差異に注目した。具体的には、第 1 節にて述べたようにユーザーの行動パターンに注目し、「初回取引であるか」、「普段の商品の購入個数や取引の間隔からどれだけ違いがあるか」などの特徴に注目した。

表 1: 取引データのラベルづけの内訳

ラベル	取引のタイプ	取引データの数
0	正常取引	35,141,549
1	不正取引	14,605
2	不正取引	3,851
3	不正取引	541

3.2.1. 前回取引からの間隔

ユーザーごとに取引を発生時刻順に並べ、不正取引と正常取引のそれぞれについて取引の間隔を分析した。その結果、正常取引において取引の間隔が平均 18,852 分であったのに対し、不正取引の取引の間隔が平均 4,362 分であり、極めて短いことがわかった。また、不正取引における取引の間隔の分布は、正常取引の分布に比べ、より原点に近い領域にあることがわかった。

図 1 はあるユーザーに注目した取引の間隔を示したグラフである。横軸は累計の取引回数、 縦軸は前回の取引から経過した時間 (分) である。 つまり、前回から取引の間隔が短いほど、間隔は 0 に近くなる。また、灰色の点が正常取引、赤色の点が不正取引を示す。本図からも、不正取引は短い間隔で連続して発生するという特徴をもつことを確認できる。

3.2.2. 初回取引からの日数

第 3.2.1 節の分析結果より、時間に着目した特徴が重要であると考え、 ユーザーごとに様々な特徴に関する時間変動を分析した。その結果、特に初回取引からの日数に不正取引の特徴が見られることがわかった。初回取引からの日数と不正取引の分布を図 2 に示す。図の赤色の棒が不正取引の分布、灰色の棒が正常取引の分布である。図 2 より、初回取引から 0 日目の取引、つまり初回取引が不正取引であるユーザーの割合が極めて高いことがわかった。ただし、本研究で扱うデータの取得期間は 2017 年 1 月 1 日以降のものであるため、このデータにおける初回取引が、そのユーザにとっての真の初回取引であるとは限らない。

3.3. 特徴量の生成

第 3.2 節の分析結果に基づき、

- ユーザーごとに注目した特徴量が有効
- 時間に関する特徴量が有効
- 普段のユーザーの行動からの変化に着目した特徴量が有効

という仮説を立て、特徴量の設計を行った。本研究で生成し、モデルの学習に使用した特徴量を表 3 に示す。以下では、これらの特徴量の生成方法を説明する。なお、本研究で扱う全取引データを行列 $\mathbf{T}$ で表す。この行列の各列ベクトルが個々の変数に対応し、各行ベクトルが個々の取引に対応する。

表 2: データセットに含まれる変数

変数名	値	意味
member_id	自然数	EC サイト会員固有の ID
purchase_id	自然数	1 つの取引固有の ID
sales_num	1〜21	購入した商品の個数
sales_type	1〜3	事業部を示す
pay_type	1〜3	支払い方法を示す
return_date	タイムスタンプ	返金日時 (不正取引のみ)
purchase_date	タイムスタンプ	購入日時
charge_back	0〜2	チャージバック返金の有無と手法 (2 種類)
issue_type	0〜3	正常、不正取引を示す
country_code	1〜179	国別固有のコード
lang	1〜4	取引時の言語設定
device	1〜7	使用されたデバイス
browser_flag	1〜8	使用されたブラウザ

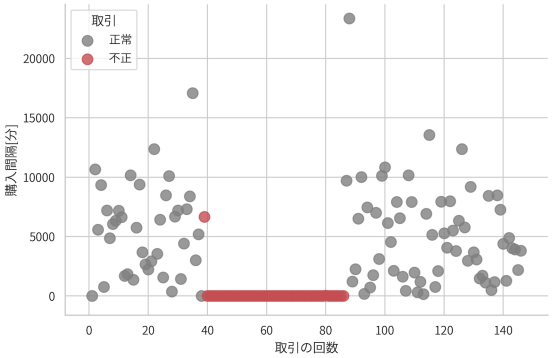

図 1: あるユーザーに注目した取引の間隔

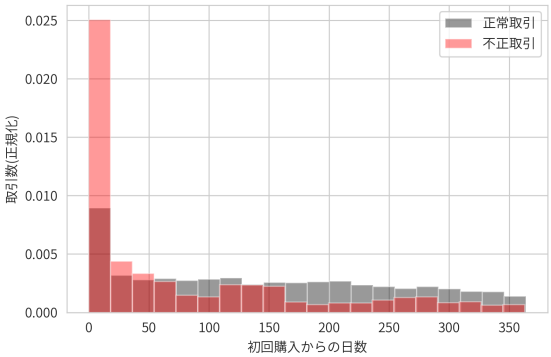

図 2: 初回取引からの日数を示す度数分布

3.3.1. タイムスタンプの処理

取引のあった日時を表す purchase_date から、日付のみを取り出したものを p_day (取引した日) とし、**T** のうち対応する列ベクトルを **p_day** とする。p_day は 1 月 1 日からの経過日数となる。例えば 12 月 31 日の取引の場合、p_day は 364 になり、1 月 1 日では 0 になる。同様の処理を取引した時刻についても行い、秒に変換する。この列ベクトルを **p_sec** とする。これは、その日の 0:00:00 秒からの経過秒数である。例えば 23:59:59 に取引した場合、p_sec は 86399 になる。

3.3.2. 取引の曜日

タイムスタンプをもとに、取引のあった曜日を 0〜6 の 7 つの自然数に割り当てて表現した。具体的には月曜日を 0 と割り当て、日曜日を 6 と割り当てる。生成した列ベクトルを **p_weekday** とする。

3.3.3. 累計取引回数

ユーザごとの各取引について、その取引までの取引回数を特徴量として追加した。ユーザー u のその取引までの取引回数を n とすると、この特徴量 $\mathbf{pc\_csum}_u$ は、

$$\mathrm{pc\_csum}_{un} = n$$

を要素とする列ベクトルで表すことができる。以降、ユーザー u のその取引までの取引回数を n で示し、ユーザ u の n 回目の取引に対応する行を添字 $_{un}$ で示す。

3.3.4. 初回取引からの日数

ユーザごとの各取引について、初回取引からの経過日数を表す特徴量を考える。ユーザー u について、この特徴量 $\mathbf{p\_day\_diff}_u$ は、

$$\mathrm{p\_day\_diff}_{un} = \begin{cases} 0, & (n = 1) \\ \mathrm{p\_day}_{un} - \mathrm{p\_day}_{u1}, & (n > 1) \end{cases}$$

を要素とする列ベクトルで表すことができる。

3.3.5. 前回の取引からの間隔

ユーザごとの各取引について、その取引と m 回前 ($m = 1, 2, \cdots, 5$) の取引との間隔を表す特徴量を考える。単位は分である。ユーザー u について、この特徴量 $\mathbf{pi}_u^m$

表 3: 生成した特徴量

列名	説明
p_weekday	取引があった曜日
pc_csum	累計取引回数 (ユーザーごと) [回]
p_day_diff	データ上の初回取引からの日数 (ユーザーごと) [日]
pi_1,pi_2,⋯,pi_5	1〜5 回前からの取引間隔 (ユーザーごと) [分]
pi_average	取引間隔の平均 (ユーザーごと) [分]
sn_average	1 取引での購入個数の平均 (ユーザーごと) [個]
pi_diff	取引間隔の平均との差分 (ユーザーごと) [分]
sn_diff	1 取引での購入個数の平均との差分 (ユーザーごと) [個]

は、

$$\mathrm{pi}_{un}^{m} = \begin{cases} 0, & (n \le m) \\ (\mathrm{p\_day}_{un} - \mathrm{p\_day}_{u(n-m)}) \times 60 \times 24 & \\ \quad + \dfrac{(\mathrm{p\_sec}_{un} - \mathrm{p\_sec}_{u(n-m)})}{60}, & (n > m) \end{cases}$$

を要素とする列ベクトルで表すことができる。

3.3.6. 商品の購入個数と取引の間隔の平均

ユーザごとに、取引の平均間隔と、1 取引あたりに購入する商品の平均個数を特徴量として追加する。ユーザ u について、前者を $\mathbf{pi\_average}_u$、後者を $\mathbf{sn\_average}_u$ と表すと、これらはそれぞれ、以下に示す要素で構成される列ベクトルで表される。

$$\mathrm{pi\_average}_{un} = \begin{cases} \mathrm{pi}_{u1}^{1}, & (n = 1) \\ \sum_{i=1}^{n} \mathrm{pi}_{ui}^{1} \times \frac{1}{\mathrm{pc\_csum}_{un}}, & (n > 1) \end{cases}$$

$$\mathrm{sn\_average}_{un} = \begin{cases} \mathrm{sales\_num}_{u1}, & (n = 1) \\ \sum_{i=1}^{n} \mathrm{sales\_num}_{ui} \times \frac{1}{\mathrm{pc\_csum}_{un}}, & (n > 1) \end{cases}$$

3.3.7. 平均取引回数と間隔との差分

ユーザーごとの各取引について、商品の購入個数の平均、および 1 取引前との取引の間隔の平均との差分をとる。具体的には、$\mathbf{pi\_average}_u$ と $\mathbf{pi}_u^1$、$\mathbf{sn\_average}_u$ と $\mathbf{sales\_num}_u$ の差分をとる。これをそれぞれ $\mathbf{pi\_diff}_u$、$\mathbf{sn\_diff}_u$ とすると、

$$\mathrm{pi\_diff}_{un} = \mathrm{pi\_average}_{un} - \mathrm{pi}_{un}^{1}$$
$$\mathrm{sn\_diff}_{un} = \mathrm{sn\_average}_{un} - \mathrm{sales\_num}_{un}$$

を要素とする列ベクトルで表すことができる。

3.4. 複数の不正タイプへの対策

不正の種類が複数あることへの対策として、不正のタイプごとに検知モデルを分離した。具体的には、不正のタイプ 1〜3 のうちいずれか 1 つに着目し、その着目したタイプを検知するためのモデルを不正のタイプごとに構築し、最後に予測結果を統合するという手法を採った。統合の際には、3 種類の予測結果でどれか一つでも不正と予測されれば不正と見なすものとした。

		予測値	
		不正 (Positive)	正常 (Negative)
真値	不正 (Positive)	True Positive	False Negative
	正常 (Negative)	False Positive	True Negative

図 3: 混同行列の見方

4. 実験・評価

4.1. データの前処理

モデルの学習と評価を行うため、全データセットを学習データとテストデータに分離した。その際、取引の発生順序を無視し、学習データ：テストデータ = 2 : 1 の比率で無作為に分離した。その上で、不均衡なクラスのデータを扱うため、学習データにおける正常取引の数と不正取引の数の比率をサンプリングにより調整した。具体的には、まず、正常取引の数：不正取引の数 = 50 : 1 となるように、正常取引に対するアンダーサンプリングを行った。次に、正常取引の数：不正取引の数 = 10 : 1 となるように、不正取引に対するオーバーサンプリングを SMOTE により行った。SMOTE とは、k-近傍法に基づきデータを合成し水増しする手法である。

4.2. モデルの学習と評価

モデルの学習には XGBoost[7] を採用した。その際、決定木の数 100 本、決定木の最大深さは 60 とした。モデルの評価には図 3 に示す混同行列を用いた。混同行列を用いることで、不均衡なデータを用いた学習における適正な評価が可能となる。また、特徴量の重要度を評価する指標として、Python の XGBoost ライブラリに付属する 'feature_importances_' を用いた。この指標は、生成された全ての木の中にその変数が分岐として存在する個数を表す指標である。

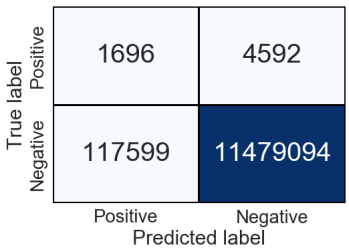

図 4: 混同行列 (ベースライン)

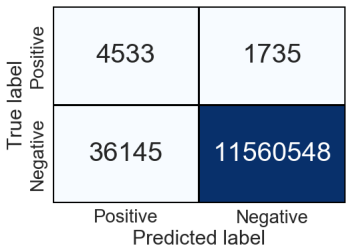

図 5: 混同行列 (提案特徴量追加後)

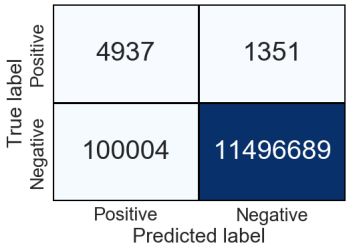

図 6: 混同行列 (不正タイプごとのモデル構築)

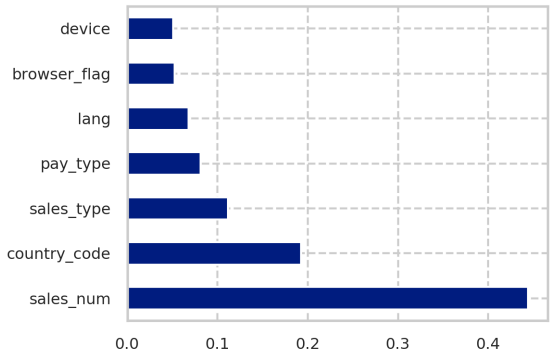

図 7: 特徴量の重要度 (ベースライン)

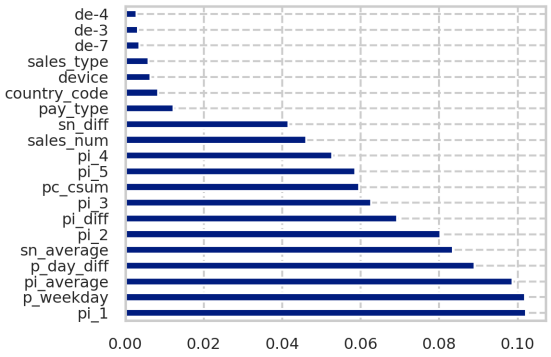

図 8: 特徴量の重要度 (提案特徴量追加後)

4.3. 実験結果・考察

4.3.1. 特徴量に関する有効性

本研究で生成した特徴量の有効性を確認するために、まず、第 3 節に示した手順のうち、第 3.2 節に示した特徴量の生成を行わずに学習した場合（以下、ベースラインと呼ぶ）の分類性能を評価する。図 4 は、特徴量の生成を行わなかった場合の混同行列を示し、図 7 は、データセットに元々含まれている特徴量（表 2 に示す特徴量）の重要度を示す。なお、評価においては不正取引のタイプを区別せず、正常取引であるか不正取引であるかのみを学習・予測した。図 4 より、特徴量生成を行わない時点で実際に不正取引を検知できる件数は 1,696 件と低く、また間違えて検知してしまう取引数 (FP) も約 110,000 件と多い。全不正取引の中で検知できた割合 (以下、Recall とする)、すなわち TP/(TP＋FN) は 27.0%である。また、図 7 より、購入した商品の個数を表す sales_num の重要度が高いことがわかる。

次に、本研究で生成した特徴量を用いて学習した場合の分類性能を評価する。この学習・評価においても、不正取引のタイプを区別せず、正常取引であるか不正取引であるかのみを考慮した。このときの混同行列を図 5 に示し、また、使用した特徴量の重要度を図 8 に示す。特徴量の生成を行わなかったベースラインの場合と結果を比較すると、不正取引を検知できた件数は 4,533 件に改善し、また間違えて検知してしまう取引数 (FP) も 36,145 件とおよそ 3 分の 1 に減少した。Recall は 72.4%である。また、図 8 より、データセットに元々含まれている特徴量と比べ、提案手法で生成された特徴量の重要度が高い傾向にあることがわかる。

第 3.2 節では、特徴量の生成に関して以下の仮説を立てた。

- ユーザーごとに注目して特徴量を生成する
- 時間に関するデータが有用と見込まれる
- 普段のユーザーの行動からの外れ具合を表現する事が重要である

図 7、8 を参照すると、 特に平均を示す二種の特徴量 **pi_average** と **sn_average** が重要であるということがわかる。 このことから不正検知に関わる特徴量として、普段の行動からどの程度外れているかということを示す特徴量が重要だといえる。また、購入間隔を表す特徴量 $\mathbf{pi}^{1,\ldots,5}$、初回購入からの経過日数 **p_day_diff** の重要度も高いことがわかる。第 3.2 節において「不正取引は短時間で連続して発生する」という特徴を見出したが、実際に不正検知に効果があることが示された。初回購入からの経過日数については、長い間ログインや使用のなかったアカウントが不正ログイン等の被害にあったものと考えられる。以上より、時間に関する特徴が重要であるという仮説も立証された。

以上より、特徴量の生成に関して、 分析時に立てた方針に則った特徴量の生成は、不正検知に効果があったと結論づけられる。

4.3.2. 不正タイプごとのモデル構築に関する有効性

不正タイプごとにモデル構築を行った場合の混同行列を図 6 に示す。図 5 の結果と比較すると、検知できた件数が 4,937 件に向上したことがわかる。Recall は 78.5%であった。ただし、誤検知数 (FP) は約 100,000 件と大幅に増えている。この原因として、不正タイプ (**issue_type**)=3 の取引データの分類が非常に難しいことが考えられる。不正タイプ 1 の検知率が 79.5%、不正タイプ 2 の検知率が 66.6%であったのに対し、不正タイプ 3 の検知率は 37.6%と大幅に低下した。また、不正タイプ 3 の取引は評価データ中に 178 件しかないにも関わらず、誤検知が約 40,000 件にも達する結果となった。不正タイプ 3 のデータは全データでも 500 件程度であり、十分な特徴の分析ができなかった可能性が高い。しかし、図 4 の結果と比較すれば大きな精度向上が達成されており、不正タイプ後にモデル構築を行うことの有効性を確認することができた。

5. まとめ

本研究では、EC サイト上の不正取引について、1 年間分の取引データに基づいてその特徴を分析し、機械学習を用いて検知を行なった。はじめに、データを分析した結果から、特徴量の生成に関して以下の仮説を立てた。

- ユーザーごとに注目した特徴量が有効
- 時間に関する特徴量が有効
- 普段のユーザーの行動からの変化に着目した特徴量が有効

そして、以上の方針に基づき、実際に特徴量の生成を行った。具体的には、商品の購入個数の平均や初回購入からの経過日数などの特徴量をユーザごとに生成した。

また、SMOTE による不正取引のオーバーサンプリングを行い、不均衡データに対処した上で、XGBoost を用いて検知モデルの構築を行なった。その結果、特徴量の生成により、大幅な検知精度の向上が見られた。

さらに、EC サイトにおける複数の不正タイプの検知のため、検知モデルを不正タイプごとに分けて学習し、不正 3 タイプそれぞれ 3 つのモデル構築を行なった。その結果、不正取引を検知できる割合が 8 割弱を記録した。

以上のことから、分析結果に基づいて立てた方針を基にした特徴量の生成、ならびに提案手法には一定の効果があることがわかった。

6. 今後の課題

本研究では取引の発生時刻順にデータを切り出しておらず、完全にランダムにデータを取り出して学習評価を行なっている。しかし、Concept Drift (第 2 節を参照) に対応するためには、時間変化するデータの特性を捉える必要がある。そのためには、例えば、オンライン学習を導入することが考えられる。オンライン学習とは、入力データがリアルタイムに次々と発生するデータに対し、学習器を逐次更新する手法である。これにより、Concept Drift への対応できる可能性がある。

また、本研究においては提案した不正のタイプごとに検知モデルを分離する手法については、タイプごとに、その特徴に関する分析を深める必要である。本研究において提案した特徴量は、不正の大部分をしめるタイプ 1 の特徴に強く影響を受けている可能性が高い。したがって、タイプごとにより特徴を精査し、またタイプごとに特有の特徴量の生成をすることなども課題に挙げられる。

さらに、不正タイプごとの検知モデルによる予測結果の統合手法も課題としてあげられる。本研究においては、どれか一つのモデルでも予測結果が 不正 (Positive) であれば、統合後の予測結果も不正と予測するという単純な手法を採用している。この統合手法については工夫の余地がある。

謝辞

本研究は、合同会社 DMM.com との共同研究により実施した。また、本研究の一部は、日本学術振興会における科学研究費補助金基盤研究 (C)(課題番号 17K00135) による支援を受けている。ここに記し謝意を表す。

参考文献

[1] 経済産業省. 平成 *29* 年度 我が国におけるデータ駆動型社会に係る基盤整備（電子商取引に関する市場調査）報告書. 2018. (Visited on 11/30/2018).

[2] 日本クレジット協会. 日本のクレジット統計　*2017* 年 *(*平成 *29* 年*)* 版. 2018. (Visited on 11/30/2018).

[3] Bahnsen A et al. “Feature engineering strategies for credit card fraud detection”. In: *Expert Systems with Applications*. June 2016.

[4] Amitha R Raju. “Predicting Fraud in Electronic Commerce: Fraud Detection Techniques in E-Commerce”. In: *International Journal of Computer Applications*. Aug. 2017.

[5] Yu-Chiang Frank Wang Yuh-Jye Lee Yi-Ren Yeh. “Anomaly Detection via Online Oversampling Principal Component Analysis”. In: *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*. Vol. 25. 7. July 2013, pp. 4802–4821.

[6] Xin Yao Shuo Wang Leandro L. Minku. “A Systematic Study of Online Class Imbalance Learning With Concept Drift”. In: *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*. Vol. 29. 10. Jan. 2018, pp. 4802–4821.

[7] Tianqi Chen and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System”. In: *KDD ’16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016.