

2024 年度（秋学期）
卒業研究 3（AH）

卒業論文

SAT ソルバーの計算時間の分類

氏名：ZHANG Jinrui

学籍番号：2600210506-2

指導教員：山下 茂 教授

提出日：2025 年 1 月 31 日

立命館大学 情報理工学部 情報理工学科
システムアーキテクトコース 4 回生

内容梗概

SAT 問題 (Satisfiability Problem: 充足可能性問題) [14] は計算理論における重要な問題の一つであり、命題論理式が与えられた際に、その論理式を真にする変数の割り当てが存在するかを判定する問題である。

SAT ソルバーは、SAT 問題を効率的に解くためのアルゴリズムを実装したソフトウェアである。与えられた命題論理式が充足可能 (SAT) か充足不可能 (UNSAT) かを判定し、充足可能であれば変数への適切な割り当てを出力する。機械学習 (Machine Learning) [8] とは、コンピュータがデータからパターンや規則を学習し、明示的なプログラムなしに予測や意思決定を行う技術である。これは人工知能の一分野であり、特に大量のデータを活用してモデルを構築し、精度の高い結果を得ることを目的とする。

本論文では、2018 年から 2024 年までの SAT Competition の過去問題から、SAT ソルバーの計算時間を影響する要因を 3 つのカテゴリに分けて数値化し、特徴として抽出する手法を提案する。その後、SVM [13] およびランダムフォレスト [4] のアルゴリズムを使用して分類モデルの構築手法を述べる。最終的に SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類するモデルを評価し、その実験結果を示す。本研究の成果は、SAT ソルバーの効率的運用や量子計算の導入を促進する基盤として寄与するものである。

目次

第1章 はじめに	1
第2章 基礎知識	3
2.1 SAT 問題	3
2.2 SAT ソルバー	4
2.3 SAT ソルバーの問題点	4
2.4 機械学習	4
2.5 先行研究	5
第3章 SAT ソルバーの計算時間を分類する手法	6
3.1 特徴抽出の概要	6
3.2 抽出する特徴	8
3.3 実行時間の測定	10
3.4 学習パターン	11
3.5 アルゴリズムの選択	11
第4章 実験結果と考察	13
4.1 取得したデータについて	13
4.2 モデルの構築と評価	15
4.2.1 評価指標	15
4.2.2 SVM のモデル	17
4.2.3 ランダムフォレストモデル	21
第5章 おわりに	25

図 目 次

4.1	TIMEOUT と NON-TIMEOUT の分布	13
4.2	SMOTE 適用後の TIMEOUT と NON-TIMEOUT の分布	15
4.3	SVM の ROC 曲線と AUC スコア	19
4.4	ランダムフォレストの ROC 曲線と AUC スコア	22

表 目 次

4.1	SVM モデルの評価結果	18
4.2	ランダムフォレストモデルの評価結果	21

第1章 はじめに

近年、SAT 問題（Boolean Satisfiability Problem：充足可能性問題）[14] の計算時間の予測が研究されている。SAT 問題は、命題論理式が充足可能か否かを判定する問題であり、計算理論の基礎としてその NP 完全性が広く知られている。SAT ソルバーとは、この SAT 問題を効率的に解くアルゴリズムを実装しているソフトウェアを指す。SAT ソルバーは多くの技術分野で活用されているが、その計算時間は入力問題の特性に大きく依存する。特に、解を求めるまでの時間が非常に長い問題も存在し、これが実用上の大きな制約となっている。

本研究の動機は、探索および計算に長時間を要する SAT 問題に対して、効率的な解決方法を模索する点にある。量子計算はそのような問題解決の手段として期待されているが、その実現に向けてはまず SAT 問題の特性に基づき SAT ソルバーの計算時間を予測し、SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類することが求められる。本研究では、効率的な SAT ソルバーの計算時間による SAT 問題の分類を実現し、将来的な量子計算の導入に向けた基盤を提供することを目的とする。

既存の研究については、GNN（Graph Neural Network：グラフニューラルネットワーク）[16] を用いて SAT 問題のグラフ構造を分析し、SAT ソルバーの探索時間を予測するアプローチが提案されている。しかし、これらの研究は大規模な SAT 問題に対して精度や汎化能力に課題を抱えている。具体的には、大規模な問題において予測精度が低下し、モデルのスケラビリティにも限界が見られる。

本研究が取り組む課題として、まずは SAT ソルバーの探索時間を短時間で解決できる問題と短時間では解決できない問題に分類し、そのモデルの汎化能力を向上すること。または、問題分類の精度を向上させ、探索時間が長い問題を効率的に特定できるようにすること。

これらの課題を解決することで、SAT 問題の効率的な解法選定や計算資源の最適化が可能となり、量子計算を含む新たな技術への橋渡しが可能になる。その目的を実現するために、2018 年から 2024 年までの SAT Competition の過去問題を分析し、問題の特徴を抽出

した。そして、SVM (Support Vector Machine) [13] やランダムフォレスト [4] といった機械学習手法を用いて予測モデルを構築した。これにより、SAT 問題の特性に基づき探索時間を高精度で予測し、問題を分類することを可能にした。

最終的に、探索時間に基づいた SAT 問題の効果的な分類モデルを構築することができた。このモデルは、SAT ソルバーの計算時間予測において高い精度を示し、特に大規模な SAT 問題に対しても有用であることが実験によって確認された。本研究の成果は、SAT ソルバーの効率的な活用や量子計算の導入に向けた実用的な基盤を提供するものである。

本論文は 5 章で構成される。以下、第 2 章で SAT 問題、SAT ソルバー、機械学習の基礎知識について述べる。第 3 章は提案手法やモデルの構築手法について述べる。そして、第 4 章で実験結果やモデルの性能評価について述べる。最後に第 5 章でまとめと今後の課題について述べる。

第2章 基礎知識

本章では、SAT 問題、SAT ソルバーおよび SAT ソルバーの問題点、機械学習、そしてこの分野の先行研究について述べる。

2.1 SAT 問題

SAT 問題 (Satisfiability Problem: 充足可能性問題) は、論理式が少なくとも一つの解を持つかどうかを判定する問題であり、計算理論や人工知能、組合せ最適化において重要な役割を果たしている。特に、ブール変数と論理演算 (論理和、論理積、否定) を用いて記述された論理式を満たす解が存在するかどうかを判定することを目的とする。以下は SAT 問題の一例である。

$$(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_3) \wedge (\neg x_3 \vee x_2)$$

ここで、 x_1, x_2, x_3 はそれぞれ真 (True) または偽 (False) の値を取り得るブール変数であり、 \vee は論理和、 \wedge は論理積、 \neg は否定を表す。

この例では $x_1 = \text{False}$, $x_2 = \text{True}$, $x_3 = \text{True}$ の場合に論理式全体が True となる。そのため、対象の論理式は充足可能であることがわかる。

SAT 問題は NP 完全であり、任意の NP 問題は多項式時間で SAT 問題に帰着可能である。つまり、SAT 問題を多項式時間で解くアルゴリズムが存在すれば、すべての NP 問題を多項式時間で解くことが可能である。この特性により、SAT 問題は NP 完全問題の原型として知られ、他の多くの問題の研究の基盤となっている。

2.2 SAT ソルバー

SAT ソルバーとは、SAT の解を効率的に求めるアルゴリズムまたはソフトウェアの総称である。SAT ソルバーは、入力された命題論理式を満足する解が存在するかどうかを判定し、存在する場合は解となる変数の真理値割り当てを返すものである。代表的な SAT ソルバーには、MiniSat [15]、CaDiCaL [6]、Glucose [1] などが存在し、いずれも効率的な探索手法を使用し、高い性能を実現している。

2.3 SAT ソルバーの問題点

SAT 問題は NP 完全 (2.1 章) であり、入力の規模が増大するにつれて問題の難易度が指数的に増加する。つまり、問題の規模が大きくなると、解を求めるための探索時間が急激に増加することがある。また、SAT ソルバーの性能は問題の特性に強く依存するため、大規模な CNF (Conjunctive Normal Form: 連言標準形) [14] 式や変数の依存関係が複雑な問題に対しては、解法の探索に膨大な時間がかかる場合がある。このような場合、ソルバーが解法を探索し続けることで計算リソースを消費し続けるが、解に到達する保証はない。特に、現在主流の CDCL (Conflict-Driven Clause Learning) [11] ベースのソルバーは効率的な探索戦略を提供するが、探索空間のサイズが膨大な場合には実用的な時間内に解を得ることが困難となる。

一方、量子計算の進展により、SAT 問題の解法に新しい可能性をもたらしている。量子アルゴリズム (例えば、QAOA [2]) を利用することで、特定の種類の SAT 問題に対して古典的なアルゴリズムよりも高速に解を求めることが可能となる。しかし、既存の SAT ソルバーは古典コンピューティングに基づいて設計され、量子アルゴリズムを直接利用することは難しい。本研究では、SAT ソルバーの計算時間に基づき、SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類する。「時間がかかる問題」の解決は将来の量子計算の活用を期待し、その基盤を提供することは本研究の目的である。

2.4 機械学習

機械学習 [8] は、データを基にしてアルゴリズムが自律的にパターンを学習し、新たなデータに対して予測や意思決定を行う手法を指す。従来のプログラミング手法では、問題解決の手順や規則を人間が設計する必要があったが、機械学習は大規模なデータを活用し

てモデルを構築し、データに潜む規則やパターンを見つけ出す。これにより、従来のプログラミング手法では困難であった複雑なタスクにも適応可能である

2.5 先行研究

近年の研究では、SAT ソルバーの実行時間を予測するための新しいモデル「TESS」(Time Estimator for SAT Solver) [9] が提案された。TESS は、GNN (Graph Neural Network : グラフニューラルネットワーク) [16] を活用し、CDCL のアルゴリズムの動作を模倣した手法を採用している。結果として、文節が 50000、変数が 17000 以下の SAT に対して高い精度を示した。しかし、学習データへの依存や複雑なデータセットの性能限界が課題として残る。本研究では、汎化能力が高い、効率的な SAT ソルバー計算時間分類モデルの構築を目的とする。

第3章 SAT ソルバーの計算時間を分類する手法

本章では、深層学習を活用し、ある SAT 問題が与えられた場合に、SAT ソルバーを使用して短時間内で解決可能かどうかを分類する手法について述べる。以下では、この分類手法の実現に必要な特徴抽出の概要、抽出した特徴、実行時間の測定、学習パターン、そしてモデル構築の流れについて述べる。

3.1 特徴抽出の概要

SAT 問題は通常、CNF (Conjunctive Normal Form : 連言標準形) [14] という形式で表現する。CNF 形式では、複数のリテラル (変数またはその否定) が含まれる節 (ブール式) の集合として問題を記述する。この形式を用いることで、SAT ソルバーによる効率的な処理が可能である。

本研究では、SAT 問題の計算時間を効率的に分類することを目的として、CNF から多角的な特徴を抽出する。これにより、SAT ソルバーの性能に影響を与える要因を数値的に定量化し、分類モデルの精度向上を図る。以下はどのような観点で特徴を抽出するかについて述べる。

1. **問題の規模を表す基本的な特徴** : SAT 問題の基本的な構造を示す特徴として、変数の数、節の数、変数の総数 (CNF の長さ) などの特徴を抽出した。これらは、問題のスケールや計算負荷を直接的に反映し、SAT 問題全体の大まかな特性を把握するための基盤を提供する。
2. **節と変数の分布に関する特徴** : 節と変数の分布に注目し、節の長さや変数の出現頻度の標準偏差、変数の出現頻度などの特徴を加えた。

標準偏差 [10] とは、データの値が平均からどの程度ばらついているかを示す統計量である。具体的には、すべてのデータ点と平均値との差の二乗平均平方根として計算される。節の長さの標準偏差は、各節に含まれる変数のばらつきを表し、問題の節の構

造が均一であるか、それとも多様性が高いかを示す。一方、変数の出現頻度の標準偏差は、特定の変数が他の変数に比べてどの程度頻繁に使用されているかの偏りを数値化する。この情報を取り入れることで、問題の内部構造の多様性や複雑性を表現し、一部の変数や節に偏りが生じる場合の影響をモデルに反映させることが可能となる。

3. **問題の制約密度 [12] に関する特徴**：制約密度が問題の計算難易度に密接に関与することは、1992 年の研究 [12] によって証明されている。

制約密度とは、問題に含まれる節の数を変数の数で割った値であり、問題の構造および計算難易度に大きく影響を及ぼす指標である。

制約密度が低い場合：変数に対して節が少ないため、解を満たす割り当てが多く存在し、問題は容易に解ける。

制約密度が高い場合：変数に対して節が多すぎるため、ほとんどの問題が充足不能となり、この判定も容易である。

制約密度が中間的な場合：問題が充足可能であるかどうか曖昧になり、一部の問題は「充足可能」、一部は「充足不能」である。この範囲では、どちらの状態かを判定するために膨大な探索が必要となり、計算コストが大幅に増加する。このため、制約密度が中間的な場合に問題の計算難易度が最も高くなる。

本研究では、この論文を参考し、節と変数の比率、変数の密度や節の密度などの特徴を抽出した。これらの特徴は、SAT 問題の制約の複雑性や分布を数値的に表現し、計算負荷を効率的に予測するための重要な手がかりを提供する。

これらの特徴抽出の観点は、SAT 問題の特性を多角的に捉え、SAT ソルバーの計算時間を影響する要因を選定した。特に、問題の規模やばらつき、密度といった要素を統合的に考慮することで、分類モデルの精度を高め、SAT 問題の短時間での解決可能性の予測を実現する基盤を構築している。

3.2 抽出する特徴

本研究では、SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類するために、前述の観点に基づいて、以下の 3 つのカテゴリで特徴を抽出した。

- n : 問題に含まれる変数の総数 (num_vars)。
- m : 節 (clause) の総数 (num_clauses)。
- l_i : 節 i に含まれる変数またはその否定の数。
- v_j : 変数 j の出現回数。

特徴量の計算方法

1. 問題の規模を表す基本的な特徴

- num_vars: SAT 問題に含まれる変数の総数。問題の大きさを示す基本的な指標。

$$n$$

- num_clauses: SAT 問題を構成する論理式の総数。制約の多さを示す。

$$m$$

- literal_count: CNF 全体に含まれる変数またはその否定の総数。問題全体の計算負荷を反映する。

$$\sum_{i=1}^m l_i$$

- avg_clause_length: 節に含まれる変数の平均数。制約の複雑さを表す。

$$\frac{\sum_{i=1}^m l_i}{m}$$

- max_clause_length: 最も長い節に含まれる変数の数。複雑な制約がある場合を検出する。

$$\max_{i=1}^m l_i$$

- `min_clause_length`：最も短い節に含まれる変数の数。単純な制約がある場合を検出する。

$$\min_{i=1}^m l_i$$

2. 変数と節の分布に関する特徴

- `clause_length_std`：節の長さの標準偏差。節の構造のばらつきを示す。

$$\sqrt{\frac{1}{m} \sum_{i=1}^m \left(l_i - \frac{\sum_{j=1}^m l_j}{m} \right)^2}$$

- `var_occurrence_std`：変数出現回数の標準偏差。変数の使用頻度のばらつきを表す。

$$\sqrt{\frac{1}{n} \sum_{j=1}^n \left(v_j - \frac{\sum_{k=1}^n v_k}{n} \right)^2}$$

- `var_counts_sum`：すべての変数の出現回数の総和。変数の使用頻度を全体的に把握する指標。

$$\sum_{j=1}^n v_j$$

- `max_var_occurrences`：最も頻繁に出現する変数の出現回数。特定の変数に偏りがあるかを示す。

$$\max_{j=1}^n v_j$$

- `min_var_occurrences`：最も少ない出現回数の変数。特定の変数がほとんど使用されない場合を示す。

$$\min_{j=1}^n v_j$$

3. 制約密度に関する特徴

- `clause_to_var_ratio`：節の数を変数の数で割った値。問題の全体的な制約密度を示す。

$$\frac{m}{n}$$

- `clause_density`：節に含まれる変数の密度。節の内容の複雑さを表す。

$$\frac{m}{\sum_{j=1}^n v_j}$$

- `var_density`：変数がどの程度均等に節に分布しているかを示す指標。

$$\frac{\sum_{j=1}^n v_j}{n}$$

- `literal_to_clause_ratio`：変数の数を句の数で割った値。句ごとの平均的な制約量を示す。

$$\frac{\sum_{i=1}^m l_i}{m}$$

これらの特徴は、SAT 問題の構造的な特性と SAT ソルバーの性能に密接に関連し、分類モデルの精度向上に寄与することが期待できる。

3.3 実行時間の測定

SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類するために、実行時間の測定が必要である。本研究では、過去の SAT Competition で高い評価を受けた SAT ソルバー CaDiCaL [6] を使用し、2018 年から 2024 年までの SAT Competition の過去問題、計 2271 問を実行し、解決時間を測定する。

- **制限時間**：各 SAT 問題に対して最大 **1800 秒**（30 分）の実行時間を設定し、それを超える場合は TIMEOUT として扱う。
- **測定方法**：各 SAT 問題を CaDiCaL で実行し、NON-TIMEOUT または TIMEOUT に至るまでの時間を記録する。NON-TIMEOUT とは制限時間内でその SAT 問題は充足可能（SAT）または充足不可能（UNSAT）と判定できたことを意味し、TIMEOUT とは制限時間に至るとき、判定できなかったことを意味する。

この測定データは後述のモデル構築においてラベルとして使われ、重要な役割を果たす。

3.4 学習パターン

SAT 問題を「短時間で解決が可能な問題」と「時間がかかる問題」に分類するために、適切な機械学習のパターンを選択する必要がある。機械学習 [8] には主に以下の三種類が存在する。

1. **教師あり学習 (Supervised Learning)** : 入力データとその正解のラベルを用いてモデルを構築する方法。
2. **教師なし学習 (Unsupervised Learning)** : データの構造やパターンを学習する方法で、正解のラベルを必要としない。
3. **強化学習 (Reinforcement Learning)** : エージェントが環境からの報酬をもとに最適な行動を学習する方法。

ラベルとは、機械学習における教師あり学習の文脈で、各データが属するクラスやカテゴリを示すものである。ラベルは、モデルが正しい予測を行うための基準として機能する。

本研究では、SAT 問題が解けるかどうかという明確な分類するためのラベルが存在するため、**教師あり学習**が最も適している。特に、特徴とラベルの関係性をモデルが学習することで、問題の特性に基づいた高精度な分類が可能となる。

各 SAT 問題に対して以下のようなラベルを付与する。

- TIMEOUT : 長時間を要する問題 (クラス 0)
- NON-TIMEOUT : 短時間で解決可能な問題 (クラス 1)

このラベルは、前述の SAT ソルバーの実行結果に基づいて設定する。モデルは、ラベルを用いて特徴量と分類結果の関係を学習する。具体的には、特徴量 (例: 変数の数、句の数、標準偏差など) から、問題が短時間で解けるかどうかを予測する能力を獲得する。

3.5 アルゴリズムの選択

本研究に使われるデータの総数は 2271 組であるため、少量のデータを用いて SAT 問題を効率的に分類することを考慮し、以下の機械学習アルゴリズムを採用する。

1. **SVM (Support Vector Machine : サポートベクトルマシン)** [13] : SVMは、データを高次元空間にマッピングし、異なるクラスを分離するための最適な境界を見つけるアルゴリズムである。この最適な境界とは、クラス間のマージン、すなわち分類境界とデータ点の間隔を最大化する直線や平面を指す。このアプローチにより、SVMは線形分離が可能なデータだけでなく、非線形なデータにも対応することができる。

2. **ランダムフォレスト (Random Forest)** [4] : ランダムフォレストは、複数の決定木 (Decision Tree) を構築し、それぞれの結果を統合することで分類や回帰を行うアンサンブル学習手法である。アンサンブル学習とは、複数のモデルを組み合わせて1つの最終的な予測を行う手法であり、単一のモデルよりも安定した性能を発揮する。

ランダムフォレストはデータを条件に基づいて繰り返し分割し、最終的に予測結果を導き出すアルゴリズムである。ただし、単一の決定木では過学習のリスクが高いため、ランダムフォレストでは以下の工夫が施されている。

- **ブートストラップ法 (Bootstrap Method)** [7] : 元のデータセットからランダムにサブセットを作成し、それぞれのサブセットで独立した決定木を学習する。この手法により、各木が異なるデータに基づいて学習するため、モデル全体の多様性が向上する。
- **特徴のランダムな選択** : 各決定木が分岐を決定する際に使用する特徴量をランダムに選択する。これにより、特定の特徴量に依存することを防ぎ、モデルの汎化性能が向上する。

この二つのアルゴリズムは、少量のデータでも高い分類精度を発揮するため、SAT 問題の分類精度を向上させることが期待できる。

本研究では、これらの手法を用いることで、SAT 問題から特徴を抽出し、各アルゴリズムを使用し、SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類するモデルを構築し、分類性能を評価する。そして、モデルの精度を比較し、最適な分類手法を選択する。

第4章 実験結果と考察

本章では、前章で述べた手法を基に、SAT ソルバー CaDiCaL を使用して CNF 形式の SAT 問題を実行した結果について説明する。また、CNF から取得した特徴を用いた SVM とランダムフォレストによるモデル構築と評価について述べる。

4.1 取得したデータについて

実験で使ったデータセットには、不均衡なラベル分布が確認された。以下の図 4.1 に示すように、TIMEOUT（長時間を要する問題）と NON-TIMEOUT（短時間での解決が可能な問題）の割合がそれぞれ 66.1% と 33.9% と大きく偏っている。

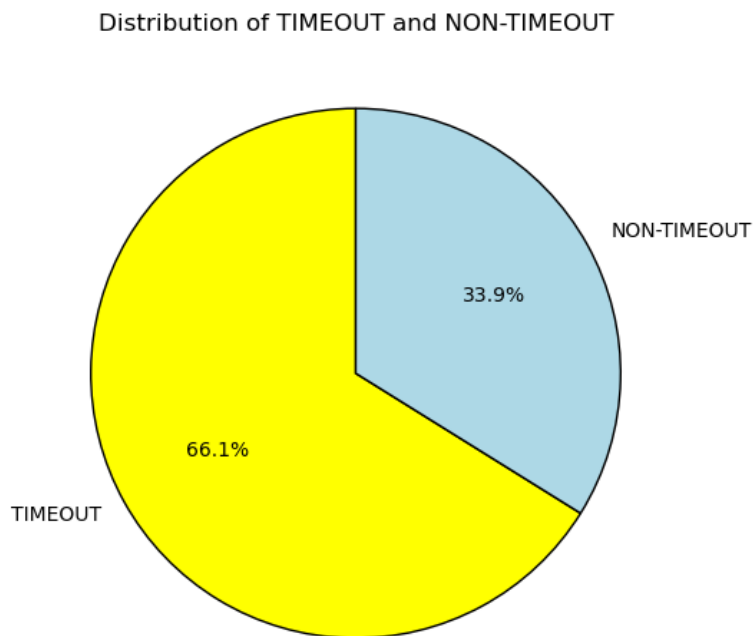


図 4.1 : TIMEOUT と NON-TIMEOUT の分布

この不均衡なデータセットは、モデルの学習において、少数派クラス（この場合はNON-TIMEOUT）の予測精度を低下させる可能性がある。そのため、データのバランスを改善する必要がある。

不均衡の問題を解決するために、本研究ではSMOTE (Synthetic Minority Over-Sampling Technique) [5] を用いてオーバーサンプリングを行った。

オーバーサンプリング

オーバーサンプリングとは、データセット内の少数派クラスのサンプル数を増加させることで、クラス間の分布を均等に近づける手法である。これにより、少数派クラスが無視される問題を緩和できる。

本論文では、オーバーサンプリング手法の一つであるSMOTEを採用する。SMOTE (Synthetic Minority Oversampling Technique) は、少数派クラスのデータを用いて新しい合成データを生成する手法である。具体的には、少数派クラスのデータポイントに対して近傍のデータポイントをランダムに選び、その間を補間することで新しいデータを作成する。この方法は、単純なデータ複製に比べて多様なサンプルを生成できる点が特徴である。

SMOTE を適用した結果、TIMEOUT と NON-TIMEOUT の割合が均衡し、それぞれ 50.0%となった（図 4.2 参照）。

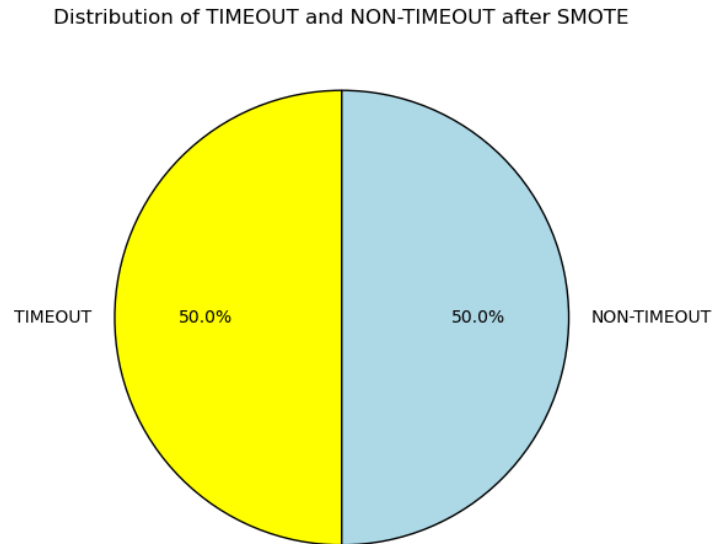


図 4.2：SMOTE 適用後の TIMEOUT と NON-TIMEOUT の分布

このように、SMOTE を用いたオーバーサンプリングにより、不均衡なデータセットを均等化することができた。この前処理により、機械学習モデルがデータ全体をバランス良く学習できる環境を構築した。

4.2 モデルの構築と評価

本研究では、前処理済みのデータを用いて、SVM とランダムフォレストの 2 つの手法で SAT 問題を NON-TIMEOUT（短時間での解決が可能な問題）と TIMEOUT（時間がかかる問題）に分類するモデルを構築した。それぞれのモデル構築プロセスと評価結果を以下に示す。

4.2.1 評価指標

モデルの性能を評価するため、以下の指標を使用した。

1. **混同行列 (Confusion Matrix)** : 正解データとモデルの予測結果を比較し、真陽性 (True Positive, TP)、真陰性 (True Negative, TN)、偽陽性 (False Positive, FP)、偽陰性 (False Negative, FN) の4つの値を示す。以下の形式で表される。

$$\begin{bmatrix} TN & FP \\ FN & TP \end{bmatrix} \quad (4.1)$$

- 真陽性 (True Positive, TP) : 実際に正のクラスに属するデータをモデルが正と予測した場合。ここで正のクラスとは、NON-TIMEOUT を指す。
- 真陰性 (True Negative, TN) : 実際に負のクラスに属するデータをモデルが負と予測した場合。ここで負のクラスとは、TIMEOUT を指す。
- 偽陽性 (False Positive, FP) : 実際には負のクラスに属するデータをモデルが正と誤って予測した場合。
- 偽陰性 (False Negative, FN) : 実際には正のクラスに属するデータをモデルが負と誤って予測した場合。

2. **精度 (precision)** : モデルが正と予測したデータのうち、実際に正であるデータの割合。

$$\text{precision} = \frac{TP}{TP + FP} \quad (4.2)$$

偽陽性 (FP) が多い場合に精度が低下するため、正しい分類の信頼性を評価する際に役立つ。

3. **再現率 (recall)** : 実際に正であるデータのうち、モデルが正と予測した割合。

$$\text{recall} = \frac{TP}{TP + FN} \quad (4.3)$$

偽陰性 (FN) が多い場合に再現率が低下するため、見逃しを最小限に抑えたい場合に重要。

4. **f1 スコア (f1-score)** : 精度と再現率の調和平均であり、全体的なバランスを示す指標。

$$\text{f1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4.4)$$

この指標は、モデルが「正確性」と「見逃しの少なさ」を同時に達成しているかを評価する。

5. **ROC 曲線と AUC スコア** [3] : モデルの閾値設定に依存せず、分類性能を視覚的に比較できる。AUC スコアが高いほど、モデルが全体的に優れていることを示し、異なるモデル間の比較に役立つ。

- **ROC 曲線 (Receiver Operating Characteristic Curve)** [3] : モデルの分類性能を視覚的に示す曲線であり、横軸に偽陽性率 (False Positive Rate, FPR)、縦軸に真陽性率 (True Positive Rate, TPR) をプロットしたもの。

- **AUC (Area Under the Curve) スコア** : ROC 曲線の下を面積を数値化した指標であり、値が 1 に近いほど分類性能が高いことを示す。

6. **正確度 (accuracy)** : 全体の予測がどれだけ正確であるかを示す指標である。

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.5)$$

7. **マクロ平均 (macro avg)** : 各クラスの指標を均等に計算した平均値。クラス間の不均衡に影響されない。この値が大きくなるほど、クラス間の不均衡に関係なく、全クラスでバランスよく性能を発揮していることを示す。

8. **重み付き平均 (weighted avg)** : 各クラスのサポート数 (データ数) に基づいて加重平均を計算した値。クラス不均衡の影響を反映する。この値が大きくなるほど、現実のデータ分布に対応して全体的な性能が高いことを示し、応用可能なモデルであることを意味する。

9. **サポート (support)** : 各クラスに含まれる実際のデータ数。評価指標を解釈する際の重要な基準となる。

以上の指標を用いることで、SVM やランダムフォレストで構築した分類モデルの性能を評価した。

4.2.2 SVM のモデル

SVM (Support Vector Machine) は前述の通りに、データを高次元空間にマッピングし、異なるクラスを分離するための最適な境界を見つけるアルゴリズムである。本研究では、全てのデータの内、8 割を訓練データ、2 割を検証データに使用し、CNF 形式の特徴を用い

て、SAT 問題が短時間で解けるか (NON-TIMEOUT) または時間がかかるか (TIMEOUT) を分類するモデルを構築した。

SVM モデルの混同行列の結果を以下の式に示す。

$$\begin{bmatrix} 91 & 69 \\ 86 & 209 \end{bmatrix} \quad (4.6)$$

SVM モデルの評価結果を以下の表 4.1 と図 4.3 SVM の ROC 曲線と AUC スコアに示す。

表 4.1 : SVM モデルの評価結果

class	precision	recall	f1-score	support
TIMEOUT (0)	0.51	0.57	0.54	160
NON-TIMEOUT (1)	0.75	0.71	0.73	295
accuracy	0.66 (455 件)			
macro avg	0.63	0.64	0.63	455
weighted avg	0.67	0.66	0.66	455

表 4.1 と図 4.3 に基づき、SVM モデルの評価は以下である。

精度 (precision)

- **TIMEOUT クラス (0)** : モデルが TIMEOUT と予測したデータのうち、正解率は 51% であった。この結果は、TIMEOUT クラスにおいて偽陽性 (FP) の発生が多い可能性を示唆する。ここで、偽陽性 (FP, False Positive) は、実際にはクラス 0 (TIMEOUT) であるデータをモデルがクラス 1 (NON-TIMEOUT) と誤って予測した場合を指す。
- **NON-TIMEOUT クラス (1)** : モデルが NON-TIMEOUT と予測したデータの正解率は 75% であり、NON-TIMEOUT クラスでの予測性能は比較的高いことが確認された。

再現率 (recall)

- **TIMEOUT クラス (0)** : 実際に TIMEOUT であるデータのうち、57% を正しく予測した。この値は、TIMEOUT クラスにおける偽陰性 (FN) の影響を反映している。ここで、偽陰性 (FN, False Negative) は、実際にはクラス 1 (NON-TIMEOUT) であるデータをモデルがクラス 0 (TIMEOUT) と誤って予測した場合を指す。

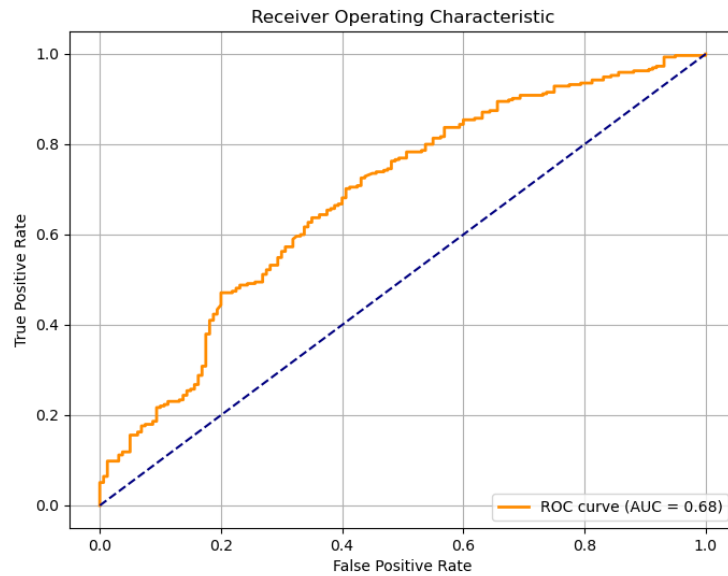


図 4.3：SVM の ROC 曲線と AUC スコア

- **NON-TIMEOUT クラス (1)**：実際に NON-TIMEOUT であるデータのうち、71% を正しく予測した。NON-TIMEOUT クラスでの再現率は比較的良好であった。

f1 スコア

- **TIMEOUT クラス (0)**：精度と再現率を均衡的に評価した結果、TIMEOUT クラスの F1 スコアは 0.54 であった。この値は、TIMEOUT クラスにおける予測性能が限定的であることを示している。
- **NON-TIMEOUT クラス (1)**：NON-TIMEOUT クラスの F1 スコアは 0.73 であり、精度と再現率の両面で高い性能を達成している。

正確度 (accuracy)

全体の正確度は 66% であり、モデルは 455 件のデータ中、約 66% を正しく予測した。ただし、accuracy はクラス間の不均衡を考慮しない指標であるため、クラスごとの性能評価を補足するためには他の指標も必要である。

マクロ平均 (macro avg)

マクロ平均の精度は0.63、再現率は0.64であり、全てのクラスを均等に評価した場合の性能を示している。この結果は、クラス間の性能差が存在することを反映している。

重み付き平均 (weighted avg)

重み付き平均の精度は0.67、再現率は0.66であり、データ分布に基づいた全体的な性能を示している。特に NON-TIMEOUT クラスが全体の性能に与える影響が大きいことが確認された。この理由として、NON-TIMEOUT クラスの support 数（データ数）が TIMEOUT クラスよりも多いことが挙げられる。具体的には、455 件のデータ中、NON-TIMEOUT クラスのデータは 295 件を占めており、全体の約 65%に相当する。重み付き平均は各クラスの support 数を重みとして計算されるため、多数派クラスである NON-TIMEOUT クラスが指標全体に与える影響が大きくなる。

ROC 曲線と AUC スコア

- **AUC スコアの解釈:** AUC スコアは 0.68 であり、これは SVM モデルがランダムな分類（AUC スコアが 0.5）よりも高い性能を持つことを示している。しかしながら、AUC スコアが 1.0 には達していないため、分類モデルとしての性能には改善の余地がある。
- **ROC 曲線の形状:** ROC 曲線は斜めの基準線（FPR と TPR が等しい場合）よりも上に位置しており、一定の識別能力を示している。ただし、曲線の形状が完全に基準線から離れていないため、特定の閾値では偽陽性率（FPR）が高くなる可能性がある。ここでは、TIMEOUT を NON-TIMEOUT と予測し、分類が誤る可能性がある。

総合的な評価

以上の結果から、SVM モデルの性能について以下のことが明らかになった。

- **強み:** NON-TIMEOUT クラスにおいて、精度、再現率ともに良好な性能を示している。また、ROC 曲線が基準線よりも上にあることから、一定の分類能力を有していることが確認された。

- **課題:** TIMEOUT クラスでは、精度と再現率が低く、予測性能に課題が残る。AUC スコアが 0.68 と限定的であり、モデル全体の性能向上が必要である。

性能が期待より低い理由の分析

SVM モデルの性能が期待よりも低い理由として、以下の要因が考えられる。

1. データの非線形性：CNF 形式の特徴量は複雑な非線形関係を持つ可能性があり、線形カーネルでは十分に表現できなかった。
2. 特徴量の不足：抽出した特徴量がモデルの学習に十分な情報を提供していなかった可能性がある。
3. クラス間の曖昧さ：TIMEOUT と NON-TIMEOUT の境界が曖昧であり、モデルが正確な分類を行うのが困難だった。

4.2.3 ランダムフォレストモデル

ランダムフォレストは前述の通りに、複数の決定木を用いたアンサンブル学習手法であり、各決定木の予測結果を統合することで、分類または回帰を行うアルゴリズムである。本研究では、CNF 形式の特徴を用いて、SAT 問題が短時間で解けるか（NON-TIMEOUT）または時間がかかるか（TIMEOUT）を分類するモデルを構築した。データセット全体を 8 割の訓練データと 2 割の検証データに分割し、モデルの学習および評価を実施した。

ランダムフォレストモデルの混同行列の結果を以下の式に示す。

$$\begin{bmatrix} 248 & 47 \\ 50 & 256 \end{bmatrix} \quad (4.7)$$

ランダムフォレストモデルの評価結果を以下の表 4.2 および図 4.4 に示す。

表 4.2：ランダムフォレストモデルの評価結果

class	precision	recall	f1-score	support
TIMEOUT (0)	0.83	0.84	0.84	295
NON-TIMEOUT (1)	0.84	0.84	0.84	306
accuracy	0.84 (601 件)			
macro avg	0.84	0.84	0.84	601
weighted avg	0.84	0.84	0.84	601

表 4.2 と図 4.4 に基づき、ランダムフォレストモデルの評価は以下の通りである。

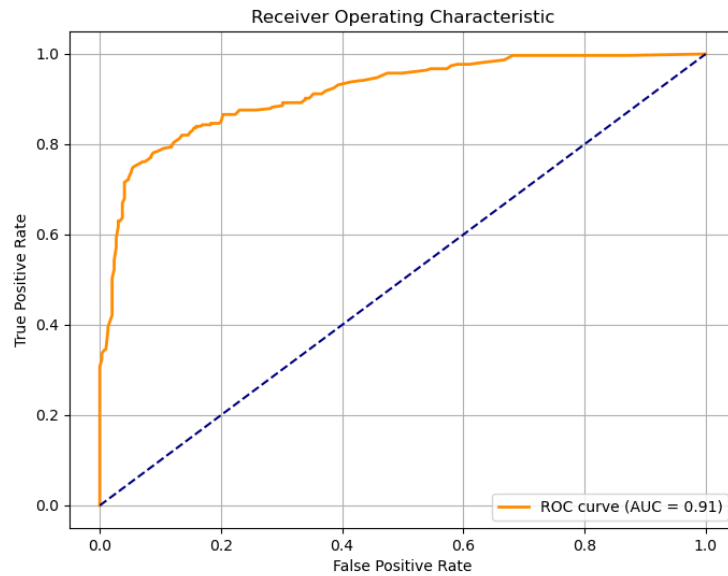


図 4.4：ランダムフォレストの ROC 曲線と AUC スコア

精度 (precision)

- **TIMEOUT クラス (0)**：モデルが TIMEOUT と予測したデータのうち、83%が正しい予測であることを示している。これは、TIMEOUT クラスでの誤分類（偽陽性）が少ないことを示唆する。ここで、偽陽性 (FP, False Positive) は、実際にはクラス 0 (TIMEOUT) であるデータをモデルがクラス 1 (NON-TIMEOUT) と誤って予測した場合を指す。
- **NON-TIMEOUT クラス (1)**：モデルが NON-TIMEOUT と予測したデータのうち、84%が正しい予測であることを示している。NON-TIMEOUT クラスでも高い精度を達成している。

再現率 (recall)

- **TIMEOUT クラス (0)**：実際に TIMEOUT であるデータのうち、84%を正しく識別している。これは、TIMEOUT クラスにおける見逃し（偽陰性）が少ないことを示している。
- **NON-TIMEOUT クラス (1)**：実際に NON-TIMEOUT であるデータのうち、84%を正しく識別している。この結果から、NON-TIMEOUT クラスでも高い識別能力を持つことがわかった。

f1 スコア

- **TIMEOUT クラス (0)** : 精度と再現率を均衡的に評価した結果、f1 スコアは 0.84 であった。これは、TIMEOUT クラスでのモデルの性能がバランス良く高いことを示している。
- **NON-TIMEOUT クラス (1)** : NON-TIMEOUT クラスの f1 スコアも 0.84 となり、精度と再現率の両面で優れた性能を発揮していることを示している。

正確度 (Accuracy)

全体の正確度は 84% であり、601 件のデータ中、約 84% を正しく予測した。これは、2 つのクラスを通じて高い予測性能を示している。

マクロ平均 (macro avg)

マクロ平均の精度、再現率、F1 スコアはいずれも 0.84 であり、全てのクラスを均等に評価した場合の性能を示している。この結果は、TIMEOUT クラスと NON-TIMEOUT クラスの両方でバランス良く高い性能を発揮していることを示している。

重み付き平均 (weighted avg)

重み付き平均の精度、再現率、f1 スコアも 0.84 であり、データ分布を考慮しても高い全体的な性能を示している。

総合的な評価

以上の結果から、ランダムフォレストモデルの性能について以下のことが明らかになった。

- **強み**: TIMEOUT クラス、NON-TIMEOUT クラスのいずれにおいても精度、再現率、f1 スコアが高く、全体的にバランスの取れた性能を発揮している。AUC スコアが 0.91 と高いことから、モデルが優れた分類能力を持つことが確認された。
- **課題**: 現時点で大きな課題は見られないが、さらなる性能向上のためにハイパーパラメータの調整や新たな特徴量の追加が必要である。

以上の結果から、ランダムフォレストモデルがSVMモデルよりも優れていることが明らかになった。その理由は以下の観点から得られる。

1. **データの非線形性への適応:**ランダムフォレストは、複数の決定木を組み合わせることで、非線形なデータ分布にも対応可能であり、SVMよりも複雑なパターンを捉える能力を持つ。
2. **クラス間のバランスの向上:**TIMEOUTクラスとNON-TIMEOUTクラスの両方において、ランダムフォレストモデルは精度と再現率が高く、SVMよりもクラス間のバランスが取れている。
3. **特徴量の重要性の活用:**ランダムフォレストは、特徴量の重要度を計算する能力があり、重要な特徴量を効果的に利用することでSVMよりも高い性能を発揮したと考えられる。

今後は、モデルの閾値を調整することで、特定のクラスにおける精度や再現率のさらなる向上を検討する。または、特徴量選択の最適化や追加データの活用により、性能を強化する予定である。

第5章 おわりに

本研究では、SAT 問題の特徴を CNF 形式から抽出し、深層学習の SVM とランダムフォレストのアルゴリズムをそれぞれ適応し、SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」に分類する手法を提案した。

SAT 問題を効率的に分類するために、「問題の規模を表す基本的な特徴」、「節と変数の分布に関する特徴」、「問題の制約密度に関する特徴」、以上 3 つの観点から、SAT ソルバーの計算時間を影響する要因を数値化して CNF から抽出し、CaDiCaL を用いて SAT Competition の過去問題（計 2271 問）の計算時間を計測した。その後、SAT 問題の実行結果をラベルとして、それぞれ SVM とランダムフォレストのアルゴリズムで学習し、SAT 問題を「短時間での解決が可能な問題」と「時間がかかる問題」、二つのクラスに分類するモデルを構築した。本研究の成果は、将来の SAT ソルバーの効率的な活用や量子計算の導入に向けた実用的な基盤を提供することが期待できる。

その結果、ランダムフォレストモデルの精度、再現率、F1 スコア、AUC スコア (0.91) は、SVM モデル (AUC スコア: 0.68) よりも顕著に高く、全体的な分類能力に優れているモデルを構築した。

今後は、モデルの閾値を調整し、特定のクラスにおける精度や再現率のさらなる向上を検討する。または、特徴の選択の最適化や追加データの活用により、ニューラルネットワークの活用を検討する。

謝辞

本論文の執筆にあたり熱心なご指導をいただいた立命館大学情報理工学部次世代コンピューティング研究室の山下茂教授に深く感謝いたします。次に、本論文の添削を行い、さまざまな指摘をしてくれた同期に感謝いたします。そして、楽しい日々を一緒に過ごさせて、快適な勉学環境を提供してくれた研究室の方々に感謝いたします。

また、遠く中国にいる、いつも金銭的、精神的に私を支えてくれた、人生の道を歩んでいる中で、常に私に助言や理解を示してくれる両親を深く感謝いたします。いつもありがとうございます。お疲れ様です。さらに、日本にいる友人や恋人の付き添いに感謝いたします。そして、今までご指導をいただいた家族、友人、先生を深く感謝いたします。気づきが遅くなり、すみませんが、たゆまぬ指導を感謝いたします。

四年前、立命館大学をお選びいただき、ここまで頑張ってきた私を感謝いたします。これからよろしく。最後に、将来のいつか、今はどんな生活を送っても、どんな感情を抱いても、この謝辞を読んでいる誰か（私）へ、学部時代でここまで頑張った自分、大きく変わって、とんでもないほど成長した自分を忘れるな！自信を持って、頑張りを続けて、走り続けてください！迷ったら、LiSA の「dawn」を聴きなさい！

参考文献

- [1] Gilles Audemard and Laurent Simon. On the glucose sat solver. *International Journal on Artificial Intelligence Tools*, 27(01):1840001, 2018.
- [2] Sami Boulebnane and Ashley Montanaro. Solving boolean satisfiability problems with the quantum approximate optimization algorithm. *PRX Quantum*, 5:030348, Sep 2024.
- [3] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.
- [4] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001.
- [5] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.
- [6] ABKFM Fleury and Maximilian Heisinger. Cadical, kissat, paracooba, plingeling and treengeling entering the sat competition 2020. *SAT COMPETITION*, 2020:50, 2020.
- [7] Tim Hesterberg. Bootstrap. *WIREs Computational Statistics*, 3(6):497–526, 2011.
- [8] Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.
- [9] Jiawei Liu, Wenyi Xiao, Hongtao Cheng, and Chuan Shi. Graph neural network based time estimator for sat solver. *International Journal of Machine Learning and Cybernetics*, Aug 2024.
- [10] Edward H. Livingston. The mean and standard deviation: what does it all mean? *Journal of Surgical Research*, 119(2):117–123, 2004.

- [11] Joao Marques-Silva, Inês Lynce, and Sharad Malik. Conflict-driven clause learning sat solvers. In *Handbook of satisfiability*, pages 133–182. ios Press, 2021.
- [12] David Mitchell, Bart Selman, and Hector Levesque. Hard and easy distributions of sat problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, AAAI’92, page 459–465. AAAI Press, 1992.
- [13] William S. Noble. What is a support vector machine? *Nature Biotechnology*, 24(12):1565–1567, Dec 2006.
- [14] Thomas J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC ’78, page 216–226, New York, NY, USA, 1978. Association for Computing Machinery.
- [15] Niklas Sorensson and Niklas Een. Minisat v1. 13-a sat solver with conflict-clause minimization. *SAT*, 2005(53):1–2, 2005.
- [16] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2021.