JAIST Repository
https://dspace.jaist.ac.jp/

Title	犯罪捜査のための著者の同一性判定
Author(s)	塩永，真直
Citation	
Issue Date	2021-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/17163
Rights	
Description	Supervisor:白井 清昭，先端科学技術研究科，修士（情報科学）

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

修士論文

犯罪捜査のための著者の同一性判定

塩永真直

主指導教員 白井清昭

北陸先端科学技術大学院大学
先端科学技術研究科
（情報科学）

令和３年３月

Abstract

In recent years, cybercrime has been on the rise due to the widespread use of the Internet, the increase in digital contents, and the growing number of users of social media. One of cybercrime is "impersonation" on texts in blogs, microblogs, electronic bulletin boards, e-mails, and chat rooms. In order to prevent crimes caused by impersonation, it is necessary to develop a technique that automatically estimates the author of a text based on its contents and detects impersonation.

There is a long history of research on author estimation by quantitative analysis of stylistic features of texts. There are two major tasks in author estimation: author classification and author identity classification. Author classification is a task to classify an author of a given text into several author candidates. On the other hand, author identity classification is a task to determine whether two texts are written by the same author. Although there are many previous studies on author classification, a few studies are carried out on author identity classification for Japanese documents. In particular, no research has applied supervised machine learning, which has been successful in the field of natural language processing in recent years. Therefore, author identity classification is ongoing and still an important research topic.

This study proposes a method for author identity classification of Japanese texts toward automatic detection of impersonation. Texts on blogs are used to develop our proposed system. When a text written by an unknown author (uncertain text) and a set of texts written by a known author (reference text), we aim at automatically determining whether they are written by the same author or not. We train such a model by supervised machine learning. In addition, it is supposed that most of the texts are written by the same author when author identity classification is performed for impersonation detection. Therefore, we propose a model optimized for impersonation detection so that it can accurately detect a small number of text pairs written by different authors.

This research contains two major topics: "construction of the dataset" and "construction of the author identity classification model". The dataset in the first topic is used for training and evaluation of the model.

The dataset used in the proposed method is constructed by the following procedures. First, we collect blog articles from Ameba blog with author IDs. Each retrieved article consists of 300 words or more, and the number of articles per author is 10 to 20. The number of authors is 596, and the number of articles is 10,433. Next, from the retrieved blog articles, we obtain pairs of reference texts and uncertain texts (hereinafter referred to as "instances"). Two types of instances are made: "same author pair" where the authors of the reference and uncertain texts are the same, and "different author pair" where the authors are different. If

the same author pair and different author pair are created by all combinations of authors, the number of different author pairs will be much larger than the same author pairs. Therefore, by suppressing the creation of the different author pairs, we obtain the same number of the same and different author pairs. Finally, we divide the obtained instances into training, development, and test data. More precisely, after dividing the authors in the dataset into three sets, the same and different author pairs are created using the method described above. The training data is used for training the author identity classification model, the development data is used for parameter optimization of the model, and the test data is used for evaluation of the method. The ratio of the training, development, and test data is 8:1:1. In addition, considering the situation of impersonation detection, the number of impersonation instances (different author pairs) is expected to be much smaller than non-impersonation instances (same author pairs). Therefore, in the development and test data, we randomly remove different author pairs so that the ratio of same author pairs to different author pairs becomes 10:1. On the other hand, in the training data, the numbers of two types of instances are kept equal.

The author identity classification model is constructed by the following procedures. First, as a preprocessing of text, we perform morphological analysis on the texts using the morphological analysis engine MeCab. Next, we extract features using results of morphological analysis, i.e. word segmentation and part-of-speech (POS). Four types of features are extracted: uni-gram of words, bi-gram of particles, tri-gram of POSs, and words before the comma. These features might be effective in author identity classification because they represent the author's writing style and habits, not contents of the text. Next, we create a "document vector" that represents a blog article as a feature vector. Values of the feature vector (feature weights) are set to the relative frequency of the feature in a text, which considers difference of length of the documents. Next, we create an "instance vector", which is a vector of an instance, in other words, pairs of uncertain and reference texts. The document vectors of uncertain texts and reference texts are made by the above procedure. A set of reference texts is represented as a single vector by averaging vectors of individual reference texts. The vectors of uncertain and reference texts are combined to form an instance vector in three ways: difference, sum, and concatenation of two vectors. Next, we learn two models for author identity classification from the training data. The first model is a classifier obtained by Random Forest, a machine learning algorithm that has been reported to be useful for the author estimation task in previous studies. It is denoted as "Model R" hereafter. The second model is a decision system that chooses the same author pair when the reliability of the classification by Random Forest is less

than a pre-defined threshold T, otherwise chooses the class classified by Random Forest as is. It is a bias model preferring the same author pair. The threshold T is optimized using the development data. Hereafter, it is denoted as "Model B". A classifier trained from the totally balanced dataset tends to wrongly classify a same author pair as a different author pair when it is applied to imbalance data where the number of the different author pair is quite a few. Model B is designed to tackle this problem.

In the experiments to evaluate the proposed method, the performance of the author identity classification was evaluated by a 5-fold cross validation using the dataset. We compared six models obtained by combination of classification models (Model R and Model B) and instance vector creation methods (difference, sum, and concatenation of vectors). In addition to the precision, recall, and F-measure of the detection of different author pairs, the specificity, FP rate, and FN rate were used as evaluation criteria. Comparing three methods of creating the instance vectors, the model based on the difference of the vectors had the highest F-measure. The performance of the model based on the sum and concatenation of the vectors were very poor. Comparing Model R and Model B, Model B was higher for most evaluation indices. The bias toward classification into the same author pairs reduced the number of false positives, resulting in the improvement of the F-measure. Finally, the best F-measure, 0.69, was obtained by Model B based on the instance vectors made by the difference of two vectors. The recall and the specificity were 0.76 and 0.95, respectively.

From the above results, we have confirmed the effectiveness of the proposed method in author identity classification toward automatic detection of impersonation.

概要

近年では，インターネットの普及，デジタルコンテンツの増加，ソーシャル・ネットワーク・サービスの利用者数の拡大に伴って，サイバー犯罪が増加している．サイバー犯罪の 1 つに，ブログ，マイクロブログ，電子掲示板，メール，チャット等のテキストを使った「なりすまし」がある．なりすましによる犯罪を防止するためには，文章の内容からテキストの著者を自動的に推定し，なりすましを検知する技術が必要となる．

文章の文体的な特徴を計量分析し，著者を推定する研究は古くから行われている．著者推定には大きく分けて著者分類と著者同一性判定のタスクがある．著者分類はテキストの著者を複数人の著者候補の中から選択する処理である．一方，著者同一性判定は 2 つのテキストが同一著者によって書かれたものかどうかを判定する処理である．なりすまし検知には著者同一性判定が必要であるが，著者分類の先行研究は多いものの，日本語文書を対象とした著者同一性判定に関する先行研究は少ない．特に，近年自然言語処理の分野で成功を収めている教師あり機械学習を適用した研究はない．そのため，確固たる技術は確立されておらず，重要な研究課題となっている．

本研究では，なりすましの自動検知を目的とし，日本語のテキストを対象とした著者同一性判定手法を提案する．具体的には，ブログ上のテキストを分析の対象とし，著者不明の文書（疑問文書）が入力されたとき，著者が明確な文書群（対照文書）と比較し，それらが同一人によって書かれたものかどうかを自動的に判定するモデルを機械学習により学習する．また，なりすまし検知を対象にする文書群は圧倒的に同一人によって書かれた文書が多いことが想定される．そのため，同一人データに偏りのあるデータから少数の別人データを検出するために，なりすまし検知に特化するようにバイアスをかけたモデルを提案する．

本研究は，大きく分けて，著者同一性判定モデルの学習および評価に用いる「データセットの構築」と「著者同一性判定モデルの構築」の 2 つのテーマに取り組む．

提案手法で用いるデータセットは以下の手順で構築する．まず，Ameba ブログから著者 ID とともにブログ記事データを収集する．その際，300 文字以上の記事を対象に，著者 1 人あたりの記事数が 10〜20 記事となるようにする．得られたブログ記事データは，著者数が 596 名，記事数が 10,433 記事であった．次に，収集したブログ記事データから，対照文書と疑問文書の組（以下,「事例」と呼ぶ）を獲得する．事例には，対照文書と疑問文書の著者が同じ事例である「同一人ペア」と，対照文書と疑問文書の著者が異なる事例である「別人ペア」の 2 種類がある．著者全ての組み合わせで同一人ペア・別人ペアを作成すると，別人ペア数が同一人ペア数よりもはるかに多くなる．そのため，別人ペアの組み合わせをランダムに減らすことにより，同じ数の同一人ペアと別人ペアの事例を得る．最後に，得られた事例を訓練データ，開発データ，テストデータに分割する．正確には，著者を 3 つのセットに分割後，前述した方法で同一人ペアと別人ペアを作成する．訓練データは著者同一性判定モデルの学習，開発データはモデルのパラメータ最適

化，テストデータは提案手法の評価にそれぞれ用いる．分割の比率は，訓練データ：開発データ：テストデータの比が 8：1：1 になるようにする．また，実際になりすましを検知する場面を想定すると，なりすましの事例（別人ペア）はなりすましがされていない事例（同一人ペア）よりも圧倒的に少ないと予想される．そのため，開発データとテストデータについては，同一人ペアと別人ペアの比が 10：1 になるように，別人ペアをランダムに間引く．一方，訓練データでは，同一人ペアと別人ペアの数は同数のままとする．

著者同一性判定モデルは以下の手順で構築する．まず，テキストの前処理として，記事データに対して形態素解析エンジン MeCab を用いて形態素解析を行う．次に，形態素解析により分割された単語と品詞情報を用いて素性を抽出する．具体的には，単語の uni-gram，助詞の bi-gram，品詞の tri-gram，読点前の単語の 4 種類の素性を抽出する．これらの素性を用いる理由は，文書の内容に依存せず，著者の書き方のスタイルや癖を表すことから，著者同一性判定に有効であると考えられるためである．次に，1 つの文書（ブログ記事）を素性ベクトルで表現した「文書ベクトル」を作成する．素性ベクトルの値（素性の重み）については，ブログ記事毎に文字数が異なるため，1 つの文書内での相対出現頻度を用いる．次に，事例，すなわち疑問文書と対照文書の組をベクトルで表現した「事例ベクトル」を作成する．前述した方法で，疑問文書のベクトル，対照文書群における個々のベクトルを作成する．対照文書ベクトルについては，対照ベクトル群の個々の文書ベクトルの平均を算出し，1 つのベクトルで表現する．疑問文書のベクトル，対照文書ベクトルの組から，2 つのベクトルの差，2 つのベクトルの和，2 つのベクトルの連結の 3 種類の方法で事例ベクトルを作成する．次に，訓練データから 2 種類の著者同一性判定の分類モデルを学習する．1 つ目は，先行研究により著者推定のタスクでの有用性が報告されている機械学習手法であるランダムフォレストを用いて分類器を構築したモデルである．これをモデル R とする．2 つ目は，ランダムフォレストによる判定の信頼度が閾値 T 未満のときには常に同一人と判定するように同一人判定にバイアスをかけた判定システムを構築する．閾値 T は開発データを用いて最適化する．これをモデル B とする．これは，実際のなりすまし検出の場面では別人ペアの事例が少なく，同一人ペアと別人ペアを同数含む均衡データから学習された分類器は，同一人ペアを別人ペアと誤判定することが多いという問題に対する対策である．

提案手法の評価実験では，分類モデル（モデル R・モデル B）と事例ベクトル作成方法（ベクトルの差・ベクトルの和・ベクトルの連結）の組み合わせによる 6 種類のモデルに対して，データセットを用いた 5 分割交差検定によって著者同一性判定の性能を評価した．評価基準として，別人ペア検出の精度・再現率・F 値に加えて，特異度・FP 率・FN 率などを用いた．

事例ベクトルの作成方法を比較すると，ベクトルの差によって事例ベクトルを作成する方法が一番 F 値が高かった．ベクトルの和と連結によるモデルは F 値が非常に低かった．モデル R とモデル B を比較すると，様々な評価指標について，全

般的にモデルＢの方が高かった．同一人判定にバイアスをかけることで偽陽性の数が減り，これによりＦ値が改善した．結論として，事例ベクトルを対照文書ベクトルと疑問文書ベクトルの差によって作成したときのモデルＢのＦ値が最も高く，その値は0.69であった．このモデルの再現率は0.76，特異度は0.95であった．以上から，なりすましの自動検知を目的とした著者同一性判定において，提案手法の有用性を確認することができた．

目 次

図 目 次

表 目 次

第1章　はじめに

1.1　背景

近年ではインターネットの普及，デジタルコンテンツの増加，ソーシャル・ネットワーキング・サービス（Social Networking Service; SNS）の利用者数の拡大に伴って，サイバー犯罪が増加している [1]．サイバー犯罪とは，一般に，インターネット等の高度情報通信ネットワークを利用した犯罪，コンピュータまたは電磁的記録を対象とした犯罪等，情報技術を利用した犯罪などを指す．サイバー犯罪には，匿名性が高い，痕跡が残りにくい等の特徴があるといわれており，ブログ，マイクロブログ（Twitter など），電子掲示板，メール，チャット等のテキストが犯罪に使われることも散見される．

サイバー犯罪の 1 つに「なりすまし」がある．なりすましとは，インターネット上に投稿されている情報などを無断で利用し，他人のアカウントを作成すること，あるいは他人のアカウントを自分が所有者であるように装うことである．なりすましには，ブログ，マイクロブログ，電子掲示板，メール，チャット等のテキストを使った脅迫・殺害予告や嘘の発言による信用失墜行為などもあり，これらは犯罪に発展する可能性もある．なりすましによる犯罪を防止するためには，文章の内容からテキストの著者を自動的に推定し，なりすましを検知する技術が必要となる．

文章の文体的な特徴を計量分析し，著者を推定する研究は古くから行われている．著者推定には大きく分けて，著者分類と著者同一性判定のタスクがある [2]．著者分類はテキストの著者を複数人の著者候補の中から選択する処理である．一方，著者同一性判定は 2 つのテキストが同一著者によって書かれたものかどうかを判定する処理である．なりすましの検知には著者同一性判定が必要であるが，日本語の文書の著者推定に関する先行研究の多くは著者分類を扱ったものであり，著者同一性判定に関する実験や検証が少ない．そのため，日本語テキストを対象とした著者同一性判定については確固たる技術は確立されておらず，重要な研究課題である．

1.2 目的

上記の研究背景をもとに，本研究ではサイバー犯罪捜査に向けた著者の同一性判定を研究目的とし，これに取り組む．文書を用いたサイバー犯罪の中でも特になりすましに着目し，犯罪捜査におけるなりすまし検知に使用することを想定し，日本語の文書の著者の同一性を自動的に判定する手法を提案する．具体的には，ブログデータを分析の対象とし，著者不明の文書（疑問文書）が入力されたとき，著者が明確な文書（対照文書）と比較し，それらが同一人によって書かれたものかどうかを判定するモデルを提案する．なりすまし検知を対象にする文書群は圧倒的に同一人によって書かれた文書が多いことが想定されることから，別人データと比較して同一人データが多い偏りのあるデータを作成し，これを用いて提案手法の有効性を実験的に評価する．特に，同一人データに偏りのあるデータから少数の別人データを検出するために，なりすまし検知に特化するようにバイアスをかけたモデルを提案する．

1.3 本論文の構成

本論文は 5 章から構成される．第 2 章では，著者推定に関する先行研究を著者分類と著者同一性判定に分けて紹介する．また先行研究に対する本研究の特徴も論じる．第 3 章では，本研究で提案する著者同一性判定モデルを詳述する．第 4 章では，提案手法の評価実験について報告する．最後に，第 5 章では，本論文のまとめと今後の課題について述べる．

第2章　関連研究

2.1　著者分類に関する研究

本節では，日本語の文章に対して，テキストの著者を複数人の著者候補の中から推定する著者分類に関する研究についてまとめる．著者分類に関する研究には，分類に有効な素性の種類の分析に焦点を当てたものや機械学習手法や多変量解析手法等の分類法を探求するものなどがある．

有効な素性の分析に関する研究

松浦と金田は，文学作品に対して形態素解析のような前処理を必要としない文字単位の n-gram 分布を用いた著者分類手法を提案している [3]．n-gram とは，隣接する n 個の要素の列で，要素間の共起関係を表すものである．n の値が 1 のものを uni-gram，2 のものを bi-gram，3 のものを tri-gram という．文字 n-gram は隣接する n 個の文字の列である．彼らの手法では，個々の文書について，それに出現する文字の tri-gram 出現頻度の分布を求める．これを tri-gram 分布と呼ぶ．各文章の tri-gram 分布間の距離を算出することで文章間の類似度を求める．芥川龍之介の短編小説 10 作品と菊池寛の短編小説 8 作品を用いて 2 人の作家の分類を行ったところ，2 万 5000 文字以上の文章では著者の分類が可能であったと報告している．

さらに，松浦と金田は，上記と同様に文字の n-gram 分布を用いた分類手法において，分類対象の著者数を 8 人に増やした検証を行っている [4]．文書に対して文字の n-gram（n=1〜10）分布を求め，文字 n-gram 分布の非類似度に基づいて著者分類を行った．その結果，3 万字程度の長さの文章に対しては文字の bi-gram，文字の tri-gram が著者分類に有効であることを示した．

金は，文学作品に対する著者分類として，単語の長さを素性とする著者分類手法を提案し，それを用いて 3 人の作家の作品を分類している [5]．金の手法では，個々の文章に対し，品詞毎（名詞，動詞，形容詞，形容動詞，助詞，助動詞，副詞）に単語の長さ（文字数，1 文字〜6 文字）の出現頻度分布を求め，その分布の群内（同じ著者）の距離と群間（異なる著者）の距離を比較することで著者を分類した．井上靖，三島由紀夫，中島敦の 3 人の作家の合計 21 編の文章を用いて分析を行った結果，助詞の長さには書き手の個性が出にくく，名詞の長さには文章の内容への依存性が高いため著者の特徴が明確には現れないが，動詞の長さには比較的著者の特徴が現れやすいと報告している．

上阪と村上は，品詞の出現率によって 2 人の作家の文学作品の文体を比較している [6]．江戸時代前期の俳諧師・浮世草子作家である井原西鶴と，その弟子で俳諧師・浮世草子作家の北条団水の作品を対象に，出現頻度の高い主要 7 品詞（名詞，助詞，動詞，助動詞，形容詞，副詞，連体詞）について，その出現率を比較した．まず，初期の西鶴浮世草子と団水浮世草子をウェルチの t 検定と主成分分析を用いて比較分析を行った結果，助詞と形容詞の出現頻度に違いがみられた．また，西鶴遺稿集と団水浮世草子についても同様の比較分析を行った結果，西鶴は団水と比較して助詞を多く使うという特徴がみられた．これらの結果から，著者分類に品詞の出現率を素性として用いることの有用性が示されている．

金は，3 人の作家の文学作品を対象に，助詞の出現頻度を手がかりとした著者分類手法を提案している [7]．実験の前段階として，品詞の出現頻度を調査し，助詞の使用率（35〜45%）が最も高く，次に名詞（25〜30%），動詞（15〜20%）の順となり，これらの 3 品詞で全単語の約 80%を占めることを示した．そのため，最も出現頻度の高い助詞の計量分析を行うこととした．金の手法では，特に出現頻度の高い 23 種類の助詞の uni-gram の出現率を求め，それらについて群内（同じ著者）の距離と群間（異なる著者）の距離を比較することで著者を分類した．井上靖，三島由紀夫，中島敦の 3 人の作家の合計 28 編の文章を用いた実験の結果，全ての文章について著者を正しく分類でき，3 クラス分類における正解率は 100%であったと報告している．このことから，助詞の uni-gram は，動詞の長さの分布，読点の打ち方に関する情報と比較して，有用性が高い素性であることが示されている．

さらに，金は，文学作品と比較して文字数が少なく文体的特徴が現れにくいと考えられる日記文に対して，同様に助詞の uni-gram を手がかりとした著者分類手法の有用性を検証した [8]．6 人の著者の各 10 編の日記文を用いて実験を行った結果，約 85%の正解率で著者を分類できたと報告している．これらの結果から，日記のような短い文章の著者分類についても助詞の uni-gram を素性として用いることの有用性が示されている．

金は，日記と作文を対象に，助詞の n-gram の出現頻度を素性とした著者分類手法を提案している [9]．提案手法では，助詞の uni-gram・bi-gram・tri-gram の各出現頻度を求め，距離法による判別分析を行う．なお，助詞の bi-gram・tri-gram では，テキストから助詞を抽出し，ある助詞とその次に出現する助詞を連結したものを bi-gram，さらに 3 番目に出現する助詞を加えて連結したものを tri-gram と定義している．通常の n-gram は隣接した要素の列を指すが，ここでの助詞 bi-gram，tri-gram では助詞は必ずしも隣接していない．6 人の著者の各 10 編の日記文，11 人の著者の各 10 編の作文を用いて分析を行った結果，日記では bi-gram，作文では tri-gram がそれぞれ著者分類に最も有効であった．さらに，助詞の細分類情報（副助詞，格助詞など）を含めた助詞 n-gram を用いたときの方が分類の正解率が高くなったと報告している．

金は，文学作品・作文・日記を対象に，文節内の単語 n-gram を用いた著者分類

手法を提案している [10]．提案手法では，文章を文節に区切り，文節内で隣接する n 個の単語情報を結合して 1 つの素性とする．単語情報とは，単語の表層形，品詞，品詞の細分類などであり，表 2.1 に示す A〜D の 4 つのパターンを定義する．「パターンの例」の列に示したのは表 2.1 における文節 A（括弧内は品詞と品詞細分類，/は単語の区切り）から抽出されるパターンである．また，パターン B とパターン D において連結される「単語の表層形」は助詞と記号のみに限定される．

表 2.1: 金 [10] が提案した文節パターン

	定義	パターンの例
パターン A	品詞を連結	名詞_助詞
パターン B	品詞と単語の表層形を連結	名詞_の
パターン C	品詞細分類を連結	一般_連体化
パターン D	品詞細分類と単語の表層形を連結	一般_の

文節 A: 文章 (名詞-一般) / の (助詞-連体化)

分析対象のデータとして，10 人の著者の各 20 編の小説，11 人の著者の各 10 編の作文，6 人の著者の各 10 編の日記文を用いた．分類のための機械学習アルゴリズムとしてランダムフォレストを用いた．パターン A〜D の 4 種類の素性の有用性を比較した結果，小説ではパターン B を素性とした分類モデルの正解率が最も高くなった．一方，作文・日記文ではいずれもパターン C を素性とした分類モデルの正解率が高いと報告している．また，著者分類において，書き手の特徴が顕著に現れたと報告されている [11] 品詞の tri-gram との比較も行った．その結果，文節パターン C は，品詞 tri-gram とほぼ同等の正解率が得られたと報告している．

金は，文学作品を対象に，読点の情報を素性として用いた著者分類手法を提案している [12]．読点は日本語固有の特徴であり，読点の打ち方には明確な規則が存在しないため，書き手ごとに特徴が現れやすいといわれている．金の手法では，各文章の読点前の文字，読点を打つ間隔，読点前の品詞の頻度分布を求め，主成分分析を行った．井上靖，三島由紀夫，中島敦の 3 人の作家の文章を分析した結果，読点前の文字や読点前の品詞に書き手の特徴が現れたと報告している．

吉田らは，14 名の作家の文学作品に対して，複数の素性を組み合わせて用いて著者分類を行っている [13]．使用した素性は，文字の tri-gram，形態素の tri-gram，読点前の文字，行別ひらがな出現数，品詞の出現数，文頭における各品詞の出現数，文末における各品詞の出現数，1 文あたりの文字数，1 文あたりの漢字数，1 文あたりの読点数である．これらの素性の出現頻度分布の非類似度に基づき著者分類を行う．まず，各素性を単独で用いた後，各素性の重みを変化させながら素性を全て組み合わせて著者分類を行った．それらの正解率を求めるとともに，安定して高い正解率を与える各素性の重みを算出した．その結果，素性を全て組み合わせて用いた場合，各素性を単独で用いるのと比較して正解率が 10%程度高く

なった．また，特に文字の tri-gram，形態素の tri-gram，読点前の文字，行別ひらがな出現数の重みが高い値となり，これらの素性が著者分類に有効であったと報告している．

分類方法に関する研究

金は，文学作品・作文・日記を対象に，単語の品詞情報の出現頻度を素性とし，機械学習によって著者を分類するモデルを学習する手法を提案している [14]．分析対象のデータとして，10 人の著者の各 20 編の小説，11 人の著者の各 10 編の作文，6 人の著者の各 10 編の日記文を用いた．機械学習手法として，k 近傍法，サポートベクターマシン（Support Vector Machine; SVM），バギング（Bagging），ブースティング（boosting），ランダムフォレスト（Random Forests）を用い，これらによって得られた分類モデルの正解率を比較した．その結果，ランダムフォレストが最も正解率が高かったと報告している．また，3 種類のデータのうち，文学作品が最も正解率が高く，次に作文，日記文の順であり，この順位は，文章の平均の長さと対応していることがわかった．

三品と松田は，文学作品とブログの文章を対象とした著者分類タスクにおいて，複数の機械学習アルゴリズムを比較した [15]．素性として，読点前の文字，単語の長さ，助詞の uni-gram，品詞の n-gram を用いた．また，分析対象のデータとして，10 人の著者の各 20 編の小説，5 人の著者の各 10 編のブログ記事を用いた．機械学習手法として，バギング（Bagging），AdaBoost，ランダムフォレスト（Random Forests），MART 法（Multiple Additive Regression Trees）を選択し，これらの分類性能を F 値で評価し，比較した．その結果，均衡データを用いた多クラス分類ではランダムフォレストが最も有効であったと報告している．さらに，ブログ記事データでの分類ではランダムフォレストが他の分類法と比較して圧倒的に F 値が高かった．このことから，文字数が少ない文章（ブログ記事）での著者分類におけるランダムフォレストの有用性が示された．また，素性については，単語の長さ，助詞の uni-gram よりも，読点前の文字，品詞の n-gram の方が有効であったと報告している．

2.2　著者同一性判定に関する研究

本節では，2 つのテキストが同一著者によって書かれたものかどうかを判定する著者同一性判定に関する研究を紹介する．紹介する研究には剽窃の検出も含み，厳密には著者の同一性を判定するものではないが，関連研究として紹介する．

財津と金は，日本語のブログ記事を対象とし，主成分分析・対応分析・多次元尺度法・階層的クラスター分析の多変量解析手法を複合的に用いた著者の同一性判定手法を提案した [17, 18]．具体的には，同一人ユニット（疑問文書，疑問文書

と同じ著者の対照文書，他の 4 名の著者の対照文書の 6 つの文書で構成），別人ユニット（疑問文書，疑問文書と異なる 5 名著者の対照文書の 6 つの文書で構成）を作り，それらを多変量解析を用いて 2 次元上に圧縮して表示したときの疑問文書・対照文書の位置関係によりスコアリングを行い，スコアの分布を同一人・別人で比較する．また，階層的クラスター分析では，テンドログラムでの疑問文書と対照文書の位置関係によってスコアリングを行う．スコアリングは，疑問文書が対照文書に包含されていれば 2 点，近接していれば 1 点，混在していれば 0 点，分離していれば − 1 点のように決める．そのため，ユニット単位でみると，5 クラスの著者分類を行っているとも考えられる．素性としては，非自立語の使用率，品詞の bi-gram，助詞の bi-gram，読点の打ち方を用いた．実験では，100 人の著者のブログ記事を 4000 文字ずつ取得し，冒頭 1000 文字程度を疑問文書として分割し，残りの 3000 文字程度を対照文書としたデータを作成し，提案手法によって著者の同一性を判定した．その結果，感度・特異度ともに高い値が得られたと報告している．

Stamatatos は，英語の文書を対象とした単独剽窃検出の問題に対し，自動的に文体の矛盾を検出して文書が剽窃されたものかどうかを判定する手法を提案した [16]．剽窃分析は主に外部剽窃検出と単独剽窃検出の 2 種類の方法がある．外部剽窃検出は，判定対象の文書とコーパスとを比較して剽窃の有無を判定する問題である．一方，単独剽窃検出は，判定対象の文書と比較するコーパスが与えられないという条件の下で剽窃の有無を判定する問題である．ここでのコーパスとは剽窃の対象となっている可能性のある文書の集合を指す．提案手法では，あらかじめ決められた長さの文字列を「sliding window」と定義し，それを文書に対して一定の間隔でずらしながら window 内テキストを抽出し，それらと文書全体テキストを比較する．具体的には，window 内のテキスト，文書全体のテキストの文字の tri-gram の出現頻度の分布を求め，その距離を算出する．対象文書に剽窃が含まれているかどうかを判定する際には，それぞれの window 内テキストと文書全体のテキストとの距離の標準偏差（ばらつき）が閾値より低ければ剽窃ではない文書と判定し，閾値以上であれば剽窃である文書と判定する．実験では，sliding window の文字の長さは 1000 文字，window をずらしていく間隔は 200 文字に設定した．分析対象データとして，過去のコンペティションで使用された IPAT-DC と IPAT-CC コーパスを用いた．文書単位の剽窃検出では，70%以上の剽窃されていない文書を正しく判定できたが，剽窃された文書のうち約 1/3 は剽窃されていないと誤って判定した．また，剽窃箇所単位の剽窃検出では，再現率（Recall）は 50%近くあり，約半分の剽窃箇所を検出できたが，精度（Precision）は低く誤検出が多かったと報告している．

2.3　本研究の特色

日本語文書を対象に著者を解析する先行研究の多くが著者分類問題を取り扱っており，著者同一性判定問題に関する先行研究はほとんど存在しない．また，著者同一性判定問題を取り扱っている先行研究でも，機械学習の手法は採用されていない．そのため，本研究では，日本語の文書に対して，機械学習手法を用いた著者同一性判定手法を提案する．サイバー犯罪におけるなりすましの検知に適用することを想定し，これに適した手法を探求する．

第3章　提案手法

3.1　概要

ある著者が過去に書いた文書を「対照文書群」と定義する．これは複数 (N 件) の文書から構成されることを仮定する．以降，簡単のため，特に断りのない限り，対照文書群を単に「対照文書」と記す．一方，著者がわからない 1 つの文書を「疑問文書」と定義する．本研究の目的は，対照文書と疑問文書が与えられたとき，対照文書と疑問文書の著者が同一であるか否かを判定することである．

本章では，提案手法を「データセットの構築」と「著者同一性判定モデル」の 2 つに分けて説明する．前者は 3.2 節で，後者は 3.3 節で説明する．本節ではそれぞれの概要を述べる．

「データセットの構築」では，著者同一性判定モデルの学習および評価に用いるデータセットの構築について述べる．処理の流れを図 3.1 に示す．ブログ記事を収集し，その記事データから疑問文書と対照文書の組である事例を獲得する．後述するように，同一人ペアと別人ペアの 2 種類の事例を獲得する．さらに，事例を訓練データ，開発データ，テストデータの 3 つのセットに分割する．これらのデータは,「著者同一性判定モデル」で構築するモデルの学習，パラメータの最適化，モデルの評価に用いる．

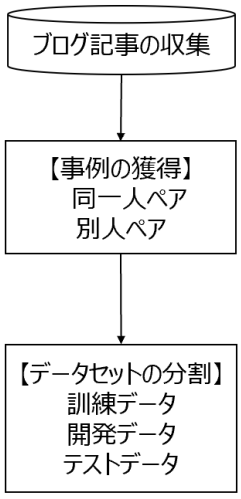


図 3.1: データセット構築手続きの概要

「著者同一性判定モデル」では，著者同一性判定モデルの構築方法について述べる．処理の流れを図 3.2 に示す．ブログ記事データに対して形態素解析による素性抽出を行い，1 つの文書（ブログ記事）を素性ベクトルで表現した文書ベクトルを作成する．文書ベクトルの組から事例ベクトルを作成する．事例ベクトルは，ベクトルの差，ベクトルの和，ベクトルの連結の 3 種類の方法で作成する．事例ベクトルを用いて，ランダムフォレストによる分類モデル，同一人判定にバイアスをかけたモデルを構築する．

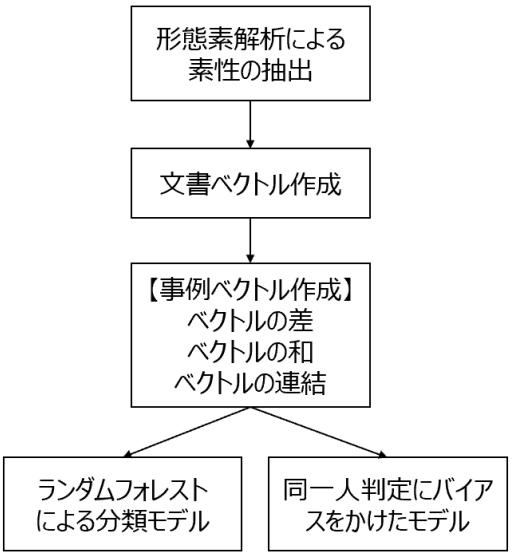


図 3.2: 著者同一性判定モデルの学習

3.2 データセットの構築

本節では，著者の同一性判定モデルの学習および評価に用いるデータセットの構築手法について述べる．

3.2.1 ブログ記事の収集

本研究では，ブログデータを分析の対象とした．その理由は以下の通りである．ブログでは，Twitter などのマイクロブログのように文字数の制限がなく，ユーザは書きたいことを自由に記載できるため，テキストに著者の特徴が比較的現れやすいと考えられる．また，ブログサイトではアカウント（著者）毎にページが設定され，そこに記事が掲載されているため，ブログ記事に対する著者を確実に同定できる．これらに加えて，大量のデータを獲得することも容易である．

Ameba ブログ [19] からデータの収集を行う．Ameba ブログでは，ブロガーは自身のブログページに対して予め運営者が決めたブログのジャンルを設定すること

ができる．さらに，Ameba ブログは各ジャンルに対してのブログページのランキングを閲覧することができる．実際にテキストのなりすまし検知を行うことを想定すると，文書の内容の偏りをなくし，幅広いジャンルの文書を分析する必要がある．そのため，様々なジャンルのランキング上位 100 名の著者について，各著者の最新記事を 20 件取得する．

記事の取得を行う際に，著者 ID も記事データと併せて取得し，著者 ID から記事の著者が同定できるようにする．ダウンロードした記事の HTML ファイルから，記事のタイトルや日付等は除き，記事の本文のみを取得するようにクリーニングを行う．また，極端に文字数が少ないデータの混在を防ぐため，記事の文字数が 300 文字未満のものは除外する．ただし，記事を除外した結果，著者 1 人あたりの記事数が 10 記事に満たないものは使用しない．そのため，著者 1 人あたりの記事数は最大で 20 記事，最小で 10 記事となる．

上記の方法で，2020 年 3 月にブログ記事データを収集した．得られたブログ記事データの統計情報を表 3.1 に示す．著者数は 596 名，記事数は 10,433 記事となった．

表 3.1: 収集したブログ記事データの統計情報

著者数	596
記事数	10,433
記事の文字数の平均	1,269
記事の文字数の中央値	941
記事の最小文字数	300
記事の最大文字数	18,031

収集したブログ記事データの 1 記事あたりの文字数の分布を図 3.3 に示す．ただし，最大文字数（18,031 文字）の 1 記事は例外的に文字数が他の記事と比べて突出して多いため，グラフから除いた．1 記事あたりの文字数は，500〜1,000 文字の記事が最も多く，次いで 1,000〜1,500 文字の記事が多かった．

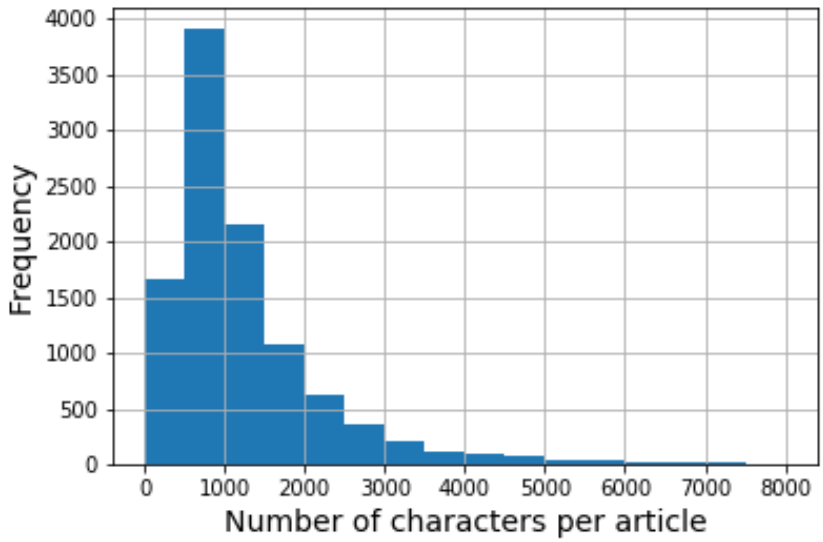


図 3.3: 1 記事あたりの文字数の分布

次に，著者 1 人あたりの記事数の分布を図 3.4 に示す．20 記事得られた著者が最も多く，次いで 19 記事得られた著者が多かった．約半数の著者は 20 記事もしくは 19 記事得ることができた．

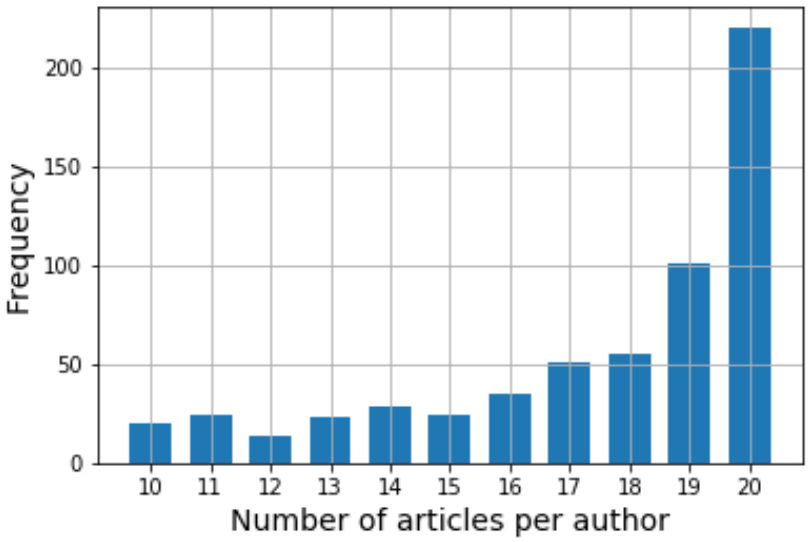


図 3.4: 著者 1 人あたりの記事数の分布

3.2.2 事例の獲得

本項では，先に収集したブログ記事データから，事例を獲得する方法について述べる．本研究における事例とは，対照文書と疑問文書の組を指す．事例には以下の 2 種類がある．

同一人ペア 対照文書と疑問文書の著者が同じ事例

別人ペア 対照文書と疑問文書の著者が異なる事例

事例を作成する最も簡単な方法は，収集した著者別のブログ記事を用いて，全ての著者の組について対照文書と疑問文書の組を作ることである．しかしながら，著者の全ての組み合わせで同一人ペア・別人ペアを作成すると，別人ペアの数が同一人ペアの数よりもはるかに多くなる．そのため，下記の方法でそれぞれの事例を獲得し，同じ数の同一人ペアと別人ペアを作成する．

- 同一人ペアの作成

  疑問文書をある著者 A の 1 つのブログ記事とし，対照文書を著者 A が書いた疑問文書以外の全てのブログ記事とする．

- 別人ペアの作成

  作成した著者 A の同一人ペアについて，対照文書はそのまま（著者 A が書いた同一人ペアでの疑問文書以外の全てのブログ記事）とし，疑問文書を著者 A 以外の著者 1 名が書いた 1 つのブログ記事と入れ換えて，著者が異なる疑問文書と対照文書の組を得る．著者の偏りを避けるため，新たに疑問文書とするブログ記事の著者は著者 A 以外からランダムに選択する．この操作を全ての同一人ペアについて行い，同数の別人ペアを得る．

3.2.3 データセットの分割

ブログデータから得られた事例を訓練データ，開発データ，テストデータに分割する．正確には，まず著者を 3 つのセットに分割後，前項で述べた手法で同一人ペアと別人ペアを作成する．訓練データは著者同一性判定モデルの学習に用いる．テストデータは提案手法の評価に用いる．開発データは，3.3.6 項で後述するモデルのパラメータの最適化に用いる．

まず，図 3.5 に示すように，著者を訓練データ，開発データ，テストデータ用にランダムに分割する．著者の分割は著者 ID をもとにランダムに行う．また，分割の比率は，訓練データ：開発データ：テストデータの比が 8:1:1 となるようにする．

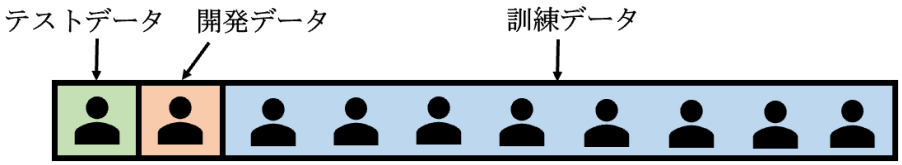


図 3.5: 著者の分割

次に，訓練データ，開発データ，テストデータを作成する．これらは事例（同一人ペアと別人ペア）の集合である．訓練・開発・テスト用に分けられた著者のデータから，前項で述べた方法で同一人ペアを作成する．次に，同一人ペアにおける疑問文書を別の著者の疑問文書と置き換えることで別人ペアを作成する．この際，訓練データ，開発データ，テストデータのそれぞれの作成において，置き換える疑問文書の著者は，それぞれのデータ用に割り当てられた著者の中から選択する．すなわち，実験に使用する著者は，訓練データ，開発データ，テストデータで完全に重なりがないようにする．以上の手続きで，同一人ペアと別人ペアを同数含む訓練データ，開発データ，テストデータが作成される．

さらに，開発データとテストデータについては，別人ペアの数を減らす．実際になりすましを検出する場面を想定すると，なりすましの事例（別人ペア）はなりすましがされていない事例（同一人ペア）よりも圧倒的に少ないと予想されるためである．具体的には，ある著者の同一人ペアを元に作成された別人ペアについて，同一人ペアと別人ペアの比が 10:1 になるように，別人ペアをランダムに間引く．一方，訓練データでは，同一人ペアと別人ペアの数は同数のままとする．分類クラスの偏りが大きいデータから著者同一性判定モデルを学習すると，データ数の大きいクラス（この場合は同一人ペア）をほぼ常に選択するような品質の悪い分類モデルが学習されることが多いためである．

3.3 著者同一性判定モデル

本節は著者の同一性を判定するモデルを機械学習する方法について述べる．まず，テキストの前処理として形態素解析を行う．その詳細は 3.3.1 項で述べる．次に，前処理済みのテキストから機械学習のための素性を抽出する．詳細は 3.3.2 項で述べる．次に，事例を素性ベクトルで表現する．まず最初にブログ記事に対する素性ベクトル（文書ベクトル）を作成し（3.3.3 項），次に疑問文書と対照文書の組に対する素性ベクトル（事例ベクトル）を作成する（3.3.4 項）．最後に，素性ベクトルで表現された訓練データを元に，著者の同一性を判定する 2 つのモデルを学習する．2 つのモデルの詳細はそれぞれ 3.3.5 項，3.3.6 項で述べる．

3.3.1 形態素解析

記事データに対して形態素解析を行い，意味を持つ最小単位である形態素に分割し，品詞情報を付与する．日本語では，形態素は単語とほぼ同じ概念を指すことが多いため，本論文でも形態素は単に単語と表記する．形態素解析にはオープンソースの形態素解析エンジンである MeCab[20] を用いる．MeCab の辞書には IPA 辞書を用いる．したがって，分割した単語に付与される品詞情報は IPA 品詞体系に従う． IPA 品詞体系では，単語の品詞は，主に連体詞・接頭詞・名詞・動詞・形容詞・副詞・接続詞・助詞・助動詞・感動詞・記号・フィラーに分類される．MeCab が出力するフォーマットは，表層形に続き，品詞・品詞細分類 1・品詞細分類 2・品詞細分類 3・活用型・活用形・原形・読み・発音が「,（カンマ）」区切りで出力される．なお，値がないものは「*（アスタリスク）」が入る．MeCab が出力する 1 つの単語に対する品詞の例を表 3.2 に示す．他の品詞の例については付録 A に示す．

表 3.2: 動詞「思う」の MeCab の品詞情報

表層形	思う
品詞	動詞
品詞細分類 1	自立
品詞細分類 2	*
品詞細分類 3	*
活用型	五段・ワ行促音便
活用形	基本形
原形	思う
読み	オモウ
発音	オモウ

本研究で収集したブログの全記事データにおける品詞の出現頻度の割合を表 3.3 に示す.「その他」の列は比較的出現頻度の低い接頭詞・記号・フィラーなどをまとめたものである.「その他」を除くと，出現頻度が高い品詞は，名詞，助詞，動詞，助動詞の順であった．先行研究での文学作品の品詞の出現頻度の調査では，助詞，名詞，動詞の順に高かったと報告されている [7]，本研究のデータとは名詞と助詞の順位が逆になっており，ブログ記事と文学作品とでは品詞の使われ方の傾向が異なることがわかる．

表 3.3: ブログ記事データにおける品詞の出現頻度の割合

品詞	出現頻度（%）
名詞	34.7
助詞	23.3
動詞	11.5
助動詞	8.2
副詞	2.0
形容詞	1.5
形容動詞	1.1
連体詞	0.7
接続詞	0.5
感動詞	0.3
その他	16.1

3.3.2 素性の抽出

1 つの文書 (ブログ記事) から機械学習のための素性を抽出する．本研究で取り組む問題は，疑問文書と対照文書の組（すなわちテキスト）に対してそれらの著者が同一か否かを判定することであるが，これはテキストをあらかじめ定められた分類カテゴリのいずれかに分類する「文書分類」と呼ばれるタスクのひとつである．文書分類の多くの先行研究では自立語が素性として使われる．なぜなら，文書分類では一般に文書の内容によってカテゴリに分類するため，内容を表す自立語が文書分類の有力な素性となるためである．しかし，著者の同一性判定では，文書の内容が一致するかではなく，著者の文書の書き方のスタイルや癖が一致しているかを考慮する必要がある．そのため，本研究では主に非自立語の情報や自立語の品詞情報を素性とする．これらは文書の内容に依存せず，著者の書き方のスタイルを表すことから，著者同一性判定に有効と考えられるためである．

MeCab を用いた形態素解析により単語に分割され品詞情報を付与されたものから，単語の uni-gram，助詞の bi-gram，品詞の tri-gram，読点前の単語の 4 種類の素性をそれぞれ抽出する．これらの素性の抽出には，MeCab の出力情報「表層形」「品詞」「品詞細分類 1」を用いる．

以下，本研究で抽出する 4 種類の素性とその抽出方法を説明する．また，素性抽出の具体例として，図 3.6 の文からどのような素性が抽出されるのかを示す．この文は，北大路魯山人（きたおおじ ろさんじん）の作品「猪の味」の冒頭の文である．青空文庫 [21] から引用した．図 3.6 には MeCab による形態素解析の結果（単語分割ならびに各単語の表層形，品詞，品詞細分類 1）も示す．

猪の美味さを初めてはっきり味わい知ったのは、私が十ぐらいの時のことであった。

ID	表層形	品詞	品詞細分類 1
1	猪	名詞	一般
2	の	助詞	連体化
3	美味	形容詞	自立
4	さ	名詞	接尾
5	を	助詞	格助詞
6	初めて	副詞	一般
7	はっきり	副詞	助詞類接続
8	味わい	動詞	自立
9	知っ	動詞	自立
10	た	助動詞	*
11	の	名詞	非自立
12	は	助詞	係助詞
13	、	記号	読点

ID	表層形	品詞	品詞細分類 1
14	私	名詞	代名詞
15	が	助詞	格助詞
16	十	名詞	数
17	ぐらい	助詞	副助詞
18	の	助詞	連体化
19	時	名詞	非自立
20	の	助詞	連体化
21	こと	名詞	非自立
22	で	助動詞	*
23	あっ	助動詞	*
24	た	助動詞	*
25	。	記号	句点

図 3.6: 素性抽出の説明に用いる例文

単語の uni-gram

先行研究により，品詞の出現率，助詞の uni-gram が著者推定に有効であるとされている [6, 7, 8]．そこで，単語もしくは品詞のみの情報ではなく，単語と品詞の組を素性とする．本研究はこれを「単語の uni-gram」と呼ぶ．通常，単語の uni-gram は単語の表層形もしくは基本形を指すが，ここでは，簡単のため，単語と品詞の組を「単語の uni-gram」と呼ぶことに注意していただきたい．また，単語単独で意味を有する自立語（名詞・動詞・形容詞・形容動詞）と記号以外の品詞を抽出の対象とする．自立語を除外するのは，自立語はテキストの意味内容を表すため，書き手のスタイルを判定の手がかりとする著者同一性判定のときには悪影響を与えると考えたためである．

また，先行研究により，品詞情報だけでなく，品詞の細分類情報を付随させた方が分類精度が高くなることが報告されていることから，単語の表層形・品詞・品詞細分類 1 を「_（アンダーバー）」で連結させたものを素性とする．例えば,「に_助詞_格助詞」「な_助動詞_*」「ありがとう_感動詞_*」がそれぞれ 1 つの素性となる．なお，助動詞や感動詞などのように品詞細分類 1 が存在しない品詞については「*（アスタリスク）」とする．品詞細分類 1 の情報も用いることによって，同じ接頭詞「お」であっても接続の違いにより「お_接頭詞_名詞接続」「お_接頭詞_形容詞接続」といったように違う素性として扱う．

図 3.6 の文から抽出した単語 uni-gram 素性を表 3.4 に示す.「抽出元単語」の列は，素性を抽出する元となる単語の図 3.6 における ID である．

表 3.4: 単語 uni-gram の素性の抽出例

素性	抽出元単語
の_助詞_連体化	2
を_助詞_格助詞	5
初めて_副詞_一般	6
はっきり_副詞_助詞類接続	7
た_助動詞_*	10
は_助詞_係助詞	12
が_助詞_格助詞	15
ぐらい_助詞_副助詞	17
の_助詞_連体化	18
の_助詞_連体化	20
で_助動詞_*	22
あっ_助動詞_*	23
た_助動詞_*	24

助詞の bi-gram

先行研究により，日記文のような比較的短い文章に対しては，助詞の n-gram の中でも特に bi-gram が著者推定に有効であるとされている [9]，このことから，助詞の bi-gram を素性の 1 つとして用いる．通常の bi-gram と異なり，隣接する 2 つの助詞の並びではなく，まず，テキストから助詞を抽出し，ある助詞とその次に出現する助詞を連結させたものを助詞の bi-gram と定義する．さらに，助詞の細分類情報（副助詞，格助詞など）を含めた助詞 n-gram を用いたときの方が著者分類の正解率が向上したと報告されていることから，助詞の表層形・品詞細分類 1 を「_（アンダーバー）」で連結し，さらに隣接する 2 つを「-（ハイフン）」で連結したものを素性とする．例えば,「から_格助詞-の_連体化」「は_係助詞-て_接続助詞」がそれぞれ 1 つの素性となる．

図 3.6 の文から抽出した助詞 bi-gram 素性を表 3.5 に示す.「抽出元単語」の列は，素性を抽出する元となる単語の図 3.6 における ID である．また，表 3.5 の枠内では，助詞 bi-gram の素性を抽出した助詞の位置を下線で示している．繰り返しになるが，通常の単語 bi-gram のように隣接する助詞を抽出するのではなく，離れた場所に出現する 2 つの助詞を連結したものを素性としていることに注意していただきたい．

表 3.5: 助詞 bi-gram の素性の抽出例

素性	抽出元単語
の_連体化-を_格助詞	2,5
を_格助詞-は_係助詞	5,12
は_係助詞-が_格助詞	12,15
が_格助詞-ぐらい_副助詞	15,17
ぐらい_副助詞-の_連体化	17,20
の_連体化-の_連体化	18,20

猪の美味さを初めてはっきり味わい知ったのは、私が十ぐらい の時のことであった。

品詞の tri-gram

先行研究により，品詞の n-gram の中でも特に品詞の tri-gram が著者推定に有効であることが示されている [11] ことから，品詞の tri-gram を素性の 1 つとして用いる．隣接する三つの品詞情報を「-（ハイフン）」で連結したものを素性とする．例えば,「助詞-副詞-動詞」「動詞-助動詞-名詞」がそれぞれ 1 つの素性となる．

図 3.6 の文から抽出した品詞 tri-gram 素性を表 3.6 に示す.「抽出元単語」の列は,素性を抽出する元となる単語の図 3.6 における ID である．

表 3.6: 品詞 tri-gram の素性の抽出例

素性	抽出元単語	素性	抽出元単語
名詞-助詞-形容詞	1,2,3	助詞-記号-名詞	12,13,14
助詞-形容詞-名詞	2,3,4	記号-名詞-助詞	13,14,15
形容詞-名詞-助詞	3,4,5	名詞-助詞-名詞	14,15,16
名詞-助詞-副詞	4,5,6	助詞-名詞-助詞	15,16,17
助詞-副詞-副詞	5,6,7	名詞-助詞-助詞	16,17,18
副詞-副詞-動詞	6,7,8	助詞-助詞-名詞	17,18,19
副詞-動詞-動詞	7,8,9	助詞-名詞-助詞	18,19,20
動詞-動詞-助動詞	8,9,10	名詞-助詞-名詞	19,20,21
動詞-助動詞-名詞	9,10,11	助詞-名詞-助動詞	20,21,22
助動詞-名詞-助詞	10,11,12	名詞-助動詞-助動詞	21,22,23
名詞-助詞-記号	11,12,13	助動詞-助動詞-助動詞	22,23,24

読点前の単語

先行研究により，読点前の文字や品詞に書き手の特徴が現れることが報告されている [12] ことから，読点前の単語を素性の 1 つとして用いる．先ほど述べた単語 uni-gram の素性と同様に，単語の表層形・品詞・品詞細分類 1 を「_（アンダーバー）」でそれぞれ連結し，さらに末尾に「-（ハイフン）読点」を付けたものを素性とする．ただし，単語単独で意味を有する自立語（名詞・動詞・形容詞・形容動詞）および記号については，単語の表層形は除外し，品詞・品詞細分類 1 に「-読点」という記号を連結したものを素性とする．自立語について単語の表層形を除外したのは，表層形はテキストの意味内容を表すため，書き手のスタイルを判定の手がかりとする著者同一性判定のときには使わない方がよいと考えたためである．例えば,「は_助詞_係助詞-読点」「名詞_副詞可能-読点」がそれぞれ 1 つの素性となる．

図 3.6 の文，および同じ作品の冒頭部分から抽出した読点前の単語の素性を表 3.7 に示す．枠内は抽出対象のテキストで，下線は素性を抽出した単語を表す．また,「抽出箇所」の列は枠内の下線の番号に対応している．抽出箇所（2）では，読点前の単語は「当時」であり，これは名詞なので，素性に単語の表層形を含まないことに注意していただきたい．

表 3.7: 読点前の単語の素性の抽出例

素性	抽出箇所
は_助詞_係助詞-読点	(1)
名詞_副詞可能-読点	(2)
が_助詞_接続助詞-読点	(3)
て_助詞_接続助詞-読点	(4)

猪の美味さを初めてはっきり味わい知ったのは,(1) 私が十ぐらいの時のことであった．当時,(2) 私は京都に住んでいたが,(3) 京都堀川の中立売に代々野獣を商っている老舗があって,(4) 私はその店へよく猪の肉を買いにやらされた．

データセットからの素性抽出の結果

3.2 節で述べたデータセットから素性を抽出した．ただし，過学習を避けるため，出現頻度の少ない素性は除外した．具体的には，ブログデータ全体において 1 回しか出現しない素性は全て削除した．結果として，4 つの素性の異り数は 11,437 種類となった．素性数の内訳を表 3.8 に示す．4 つの素性の中では，助詞の bi-gram が最も素性の種類が多く，読点前の単語が最も少なかった．

表 3.8: 素性の異り数

素性	異り数
単語の uni-gram	2,292
助詞の bi-gram	7,296
品詞の tri-gram	1,397
読点前の単語	452
合計	11,437

各素性が出現する記事数を調査した．図 3.7 は，横軸 X は記事数，縦軸 Y は X 個の記事に出現する素性の数（異り数）を表す．グラフの左端は 1 記事のみに出現する素性数を表し，右端は全ての記事に出現する素性数を表す．大部分の素性がグラフの左側に偏っており，これは多くの記事に出現する素性が少ないことを表している．そのため，出現する素性は記事によってばらつきがあることがわかった．

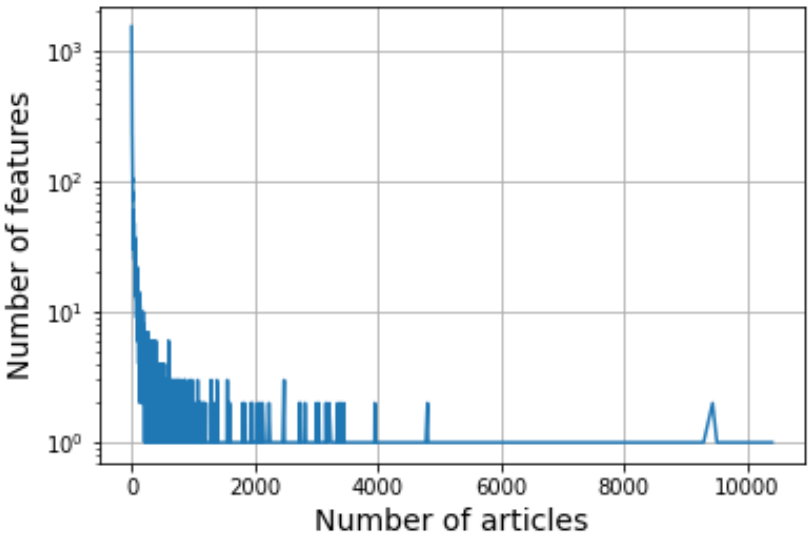


図 3.7: 出現記事数に対する素性数の分布

ある素性がある著者によって書かれたテキストに出現したとき,「素性 f が著者 a に使用される」と表現する．各素性について，それを使用する著者数を調査した．図 3.8 は，横軸 X は著者数，縦軸 Y は X 人の著者に使用された素性数（異り数）を表す．グラフの左端は 1 人の著者のみに使用される素性数を表し，右端は全ての著者に使用される素性数を表す．1 人の著者にのみ使用される素性は 124 個と比較的少なかったものの，大部分の素性がグラフの左側に偏っており，これは多くの著者に使用される素性が少ないことを表している．そのため，ある程度著者によって素性の使い方に差があることがわかった．

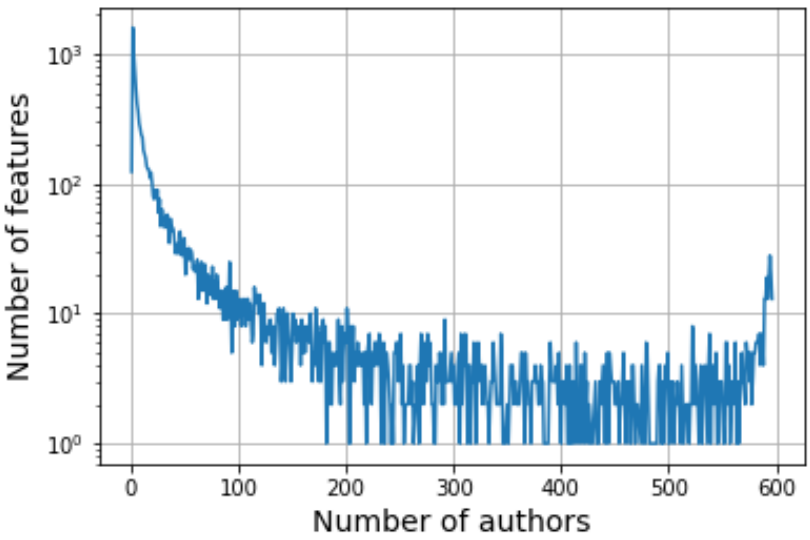


図 3.8: 使用著者数に対する素性数の分布

3.3.3 文書ベクトル

1 つの文書（ブログ記事）を素性ベクトルで表現したものを「文書ベクトル」と定義する．ベクトルの次元は 3.3.2 項で説明した素性に対応させる．素性ベクトルの値 (素性の重み) については，ブログ記事毎に文字数が異なるため，1 つの文書内での相対出現頻度を用いる．ただし，素性の種類によって算出方法が若干異なる．

単語 uni-gram・読点前の単語

単語 uni-gram・読点前の単語については，式（3.1）のとおり，素性の出現頻度のブログ記事の全形態素数に対する割合に 100 を乗じたものを素性の重みとする．

$$\text{素性の重み} = \frac{\text{素性の頻度}}{\text{ブログ記事中の全形態素数}} \times 100 \tag{3.1}$$

助詞の bi-gram

助詞の bi-gram については，式（3.2）のとおり，素性の出現頻度の全ての助詞 bi-gram 素性の出現頻度の総和に対する割合に 100 を乗じたものを素性の重みとする．

$$\text{素性の重み} = \frac{\text{素性の頻度}}{\text{ブログ記事中の全ての助詞 bi-gram の頻度の和}} \times 100 \tag{3.2}$$

品詞の tri-gram

品詞の tri-gram については，式（3.3）のとおり，素性の出現頻度の全ての品詞 tri-gram 素性の出現頻度の総和に対する割合に 100 を乗じたものを素性の重みとする．

$$\text{素性の重み} = \frac{\text{素性の頻度}}{\text{ブログ記事中の全ての品詞 tri-gram の頻度の和}} \times 100 \qquad (3.3)$$

なお，対象ブログ記事に出現しない素性については，素性ベクトルの値（素性の重み）は 0 とする．

3.3.4 事例ベクトル

事例，すなわち疑問文書と対照文書の組をベクトルで表現する．これを「事例ベクトル」と定義する．事例ベクトル作成の手順を述べる．まず，疑問文書のベクトル，ならびに対照文書群における個々の文書ベクトルを 3.3.3 項で説明した方法で作成する．次に，対照文書群については，個々の文書ベクトルの平均を算出し，1 つのベクトルで表現する．これを「対照文書ベクトル」と呼ぶ．最後に，疑問文書ベクトルと対照文書ベクトルの組から，以下の 3 種類の方法で事例ベクトルを作成する．

ベクトルの差

疑問文書のベクトルと対照文書のベクトルの各値（素性の重み）の差をとる．具体的には，事例ベクトル $\vec{v}$ を式（3.4）の通りに作成する．$\vec{v_q}$ は疑問文書ベクトル，$\vec{v_r}$ は対照文書ベクトルを表す．

$$\vec{v} = \vec{v_r} - \vec{v_q} \qquad (3.4)$$

ベクトルの和

疑問文書のベクトルと対照文書のベクトルの各値（素性の重み）の和をとる．具体的には，事例ベクトル $\vec{v}$ を式（3.5）の通りに作成する．先ほどと同様に，$\vec{v_q}$ は疑問文書ベクトル，$\vec{v_r}$ は対照文書ベクトルを表す．

$$\vec{v} = \vec{v_r} + \vec{v_q} \qquad (3.5)$$

ベクトルの連結

疑問文書のベクトルと対照文書のベクトルを連結したベクトルを作成する．具体的には，事例ベクトル $\vec{v}$ を式（3.6）の通りに作成する．v_{qk} は疑問文書ベクトルの k 番目の要素，v_{rk} は対照文書ベクトルの k 番目の要素，n は素性数を表す．事例ベクトルの次元数は $2 \times n$ となる．

$$\vec{v} = \begin{pmatrix} v_{r1} \\ v_{r2} \\ \vdots \\ v_{rn} \\ v_{q1} \\ v_{q2} \\ \vdots \\ v_{qn} \end{pmatrix} \tag{3.6}$$

3.3.5　ランダムフォレストによる分類モデル

本項では，ランダムフォレストによる著者同一性判定の分類モデルについて述べる．以降，これを「モデル R」と記す．モデル R は，疑問文書と対照文書の組に対して，それらの著者が同一人か別人かを判定する分類器である．

3.2 節で作成した訓練データ，開発データ，テストデータのうち，訓練データを用いて分類器（モデル R）を学習する．なお，開発データは使用しない．テストデータはモデル R による著者同一性判定の性能評価のために用いる．

訓練データの作成に用いるとして割り当てられた著者の各文書ベクトルを組み合わせ，事例（同一人ペア，別人ペア）のベクトルを作成する．次に，こうして作成した訓練データの事例ベクトルの集合から分類器を機械学習する．分類器を構築する際の機械学習手法には，先行研究により有用性が報告されている [14] ランダムフォレスト [22] を用いる．ランダムフォレストは，多数の決定木を使った集団学習アルゴリズムである．訓練データとそれに対応する素性をいずれもランダムに抽出し，1 つの決定木を学習する．同様の操作を繰り返し，複数の決定木を学習し，最終的に複数の決定木から構成される分類器を構築する．学習されたランダムフォレストを未知のデータに適用する際は，判定対象のデータを複数の決定木に入力して，それらが選択する分類クラスの多数決により最終的な出力を決める．

3.3.6　同一人判定にバイアスをかけたモデル

本項では，モデル R の問題点を踏まえた上で，その対策として，同一人判定にバイアスをかけたモデルについて述べる．以降，このモデルを「モデル B」と記す．

3.2.3 項で述べたように，本研究で用いるデータセットでは，訓練データは均衡データ（同一人ペアの数＝別人ペアの数）であるのに対し，テストデータでは実際のなりすましの場面を想定して同一人ペアの数が圧倒的に多い．このような状況では，均衡データから学習された分類器は同一人を別人と誤判定する可能性が高いことが予想される．

この問題に対する対策として，判定の信頼度を利用して同一人判定にバイアスをかけるモデルを提案する．具体的には，ランダムフォレストによる判定の信頼度を利用する．ランダムフォレストは未知のデータを判別する際，判別結果とともに各クラスごとの予測の信頼度も合わせて出力することができる．信頼度は 0〜1 までの数値であり，0 に近いときは信頼度が低く，1 に近いと信頼度が高いことを表す．まず，モデル R によって未知の事例に対し同一人か別人かを判定する．モデル R が「別人」と判定しても，判定の信頼度が閾値 T 未満のときには「同一人」と判定する．すなわち，別人との判定が信頼できないときは，同一人であると判定を変える．別人と判定するときは判定の信頼度が閾値 T 以上を満たしているときのみとする．このように，モデル B では，別人と判定する基準を厳しくし，同一人を別人と誤って判定することを減らすことを狙っている．モデル B による未知の事例に対する判定のフローを図 3.9 に示す．

上記の閾値 T は 3.2.3 項で述べた開発データを用いて最適化する．具体的には，閾値 T を変動させて，モデル R によって開発データの著者同一性判定を行う．なりすまし検出（別人ペアの検出）の F 値を測り，それが最大となる T を選択する．閾値 T の最適化の詳細については 4.3.1 項で報告する．

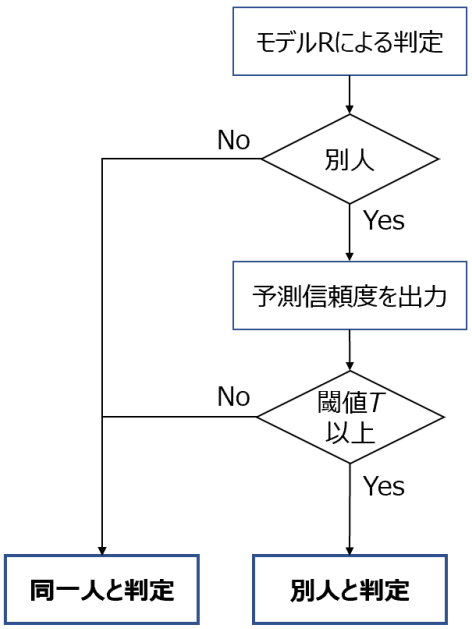


図 3.9: モデル B による判定のフロー

第4章　評価実験

本章では，著者の同一性を判定する提案手法の評価実験について述べる．まず，4.1 節で実験の手順について説明し，4.2 節で評価基準について述べる．4.3 節では，3.3.6 項で述べたモデル R における閾値 T の最適化の結果を報告する．最後に，4.4 節で実験結果を詳細に報告し，提案手法の有効性について考察する．

4.1　実験手順

本論文では，分類モデル（モデル R・モデル B）と事例ベクトル作成方法（ベクトルの差・ベクトルの和・ベクトルの連結）の組み合わせにより，6 種類のモデルを提案した．モデル R はランダムフォレストによる分類モデル（3.3.5 項），モデル B は同一人判定にバイアスをかけた分類モデル（3.3.6 項）である．一方，事例ベクトル作成方法は，3.3.4 項で述べたように，疑問文書のベクトルと対照文書のベクトルを組み合わせる 3 つの方法を示している．これらのモデルおよび略号を表 4.1 にまとめる．

表 4.1: 著者同一性判定モデルの一覧

略号	分類モデル	事例ベクトル作成方法
R-dif	モデル R	ベクトルの差
R-sum	モデル R	ベクトルの和
R-con	モデル R	ベクトルの連結
B-dif	モデル B	ベクトルの差
B-sum	モデル B	ベクトルの和
B-con	モデル B	ベクトルの連結

実験では，5 分割交差検定によって 6 つのモデルを評価する．図 4.1 は交差検定におけるデータの分割方法を示している．まず，データセットにおける著者の集合をランダムに 10 個の部分集合に分割する．1 回目の試行では，1 番目の部分集合をテストデータ，2 番目の部分集合を開発データとし，残りを訓練データとする．2 回目の試行では，3 番目の部分集合をテストデータ，4 番目の部分集合を開発データとし，残りを訓練データとする．以下，同様に 3 回目から 5 回目の試行を行う．それぞれの試行では，分割された著者から 3.2.3 項で述べた方法で事例を作成する．

このとき，訓練データ・開発データ・テストデータ間に同じ著者によって書かれたテキストは含まれない．そして，モデルＲは訓練データを用いて，モデルＢは訓練データと開発データを用いて学習する．最後に，学習されたモデルをテストデータに適用し，その著者同一性判定の性能を評価する．

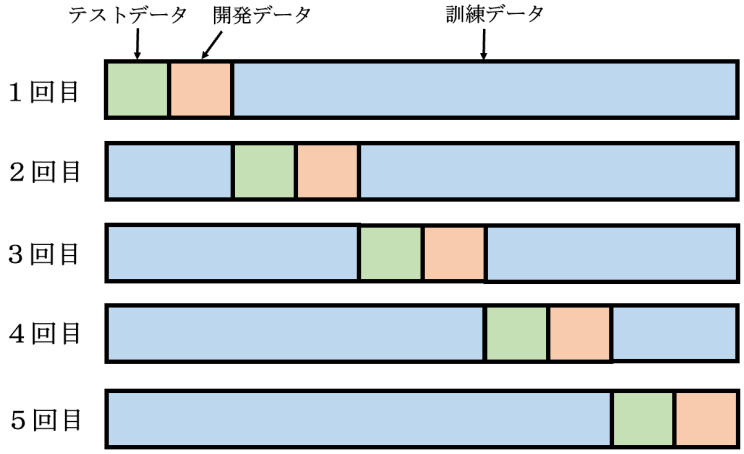


図 4.1: 5 分割交差検定によるデータの分割

交差検定の各試行におけるデータ毎の著者数を表 4.2 に示す．

表 4.2: データセットにおける著者数

	１回目	２回目	３回目	４回目	５回目
訓練データ	476	476	476	478	478
開発データ	60	60	60	59	59
テストデータ	60	60	60	59	59

交差検定の各試行におけるデータ毎の事例数，すなわち別人ペアと同一人ペアの数を表 4.3 に示す．既に述べたように，訓練データでは別人ペアと同一人ペアが同じ数になるように，開発データとテストデータでは両者の比がおよそ 1:10 になるように作成されている．

表 4.3: データセットにおける事例数

		１回目	２回目	３回目	４回目	５回目
訓練データ	別人	8355	8277	8331	8369	8400
訓練データ	同一人	8355	8277	8331	8369	8400
開発データ	別人	120	120	120	118	118
開発データ	同一人	1026	1058	1065	1068	987
テストデータ	別人	120	120	120	118	118
テストデータ	同一人	1052	1098	1037	996	1046

4.2 評価基準

本節では，著者同一性判定モデルの評価基準について述べる．文書の組（対照文書と疑問文書）が別人によって書かれたものか同一人によって書かれたものかを判定する際，その判定結果には以下の 4 つのケースが考えられる．なお，本研究では，なりすまし検知を想定しているため別人ペアを正例，同一人ペアを負例とする．

True Positive; TP
別人ペア（正例）を正しく別人ペア（正例）と判定

True Negative; TN
同一人ペア（負例）を正しく同一人ペア（負例）と判定

False Positive; FP
同一人ペア（負例）を誤って別人ペア（正例）と判定

False Negative; FN
別人ペア（正例）を誤って同一人ペア（負例）と判定

混同行列（Confusion Matrix）は，表 4.4 に示すように，上記の TP，TN，FP，FN の関係を表す行列である．

表 4.4: 混同行列

		[モデルの予測結果] 正例（別人）	負例（同一人）
[正解]	正例（別人）	TP(True Positive)	FN(False Negative)
	負例（同一人）	FP(False Positive)	TN(True Negative)

本実験では，著者同一判定の評価指標として，精度（Precision），再現率（Recall），F 値（F-measure），正解率（Accuracy），特異度（Specificity），FP 率（False Positive Rate; FPR），FN 率（False Negative Rate; FNR）を用いる．それぞれの定義を以下に示す．

精度（Precision）

精度（Precision）は，判定システムがテストデータに対して別人ペア（正例）と判定したもののうち，正しく別人ペア（正例）と判定した割合であり，式（4.1）より算出される．

$$\text{精度}（Precision）= \frac{TP}{TP + FP} \tag{4.1}$$

再現率（Recall）

再現率（Recall）は，テストデータ全体における別人ペア（正例）のうち，判定システムが正しく別人ペア（正例）と判定した割合であり，式（4.2）より算出される．

$$再現率（Recall）= \frac{TP}{TP + FN} \tag{4.2}$$

なりすまし検知における再現率は，なりすましの犯人群を正しく犯人群であると判定できた割合となる．なお，再現率は，感度（Sensitivity）や TP 率（True Positive Rate; TPR）とも呼ばれる．

F 値（F-measure）

F 値（F-measure）は，精度（Precision）と再現率（Recall）の調和平均であり，式（4.3）より算出される．

$$F 値（F\text{-}measure）= \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4.3}$$

正解率（Accuracy）

正解率（Accuracy）は，テストデータ全体のデータのうち，判定システムによる別人・同一人の判定が正解と一致した割合であり，式（4.4）より算出される．

$$正解率（Accuracy）= \frac{TP + TN}{TP + FP + TN + FN} \tag{4.4}$$

特異度（Specificity）

特異度（Specificity）は，テストデータ全体における同一人ペア（負例）のうち，判定システムが正しく同一人ペア（負例）と判定した割合であり，式（4.5）より算出される．

$$特異度（Specificity）= \frac{TN}{FP + TN} \tag{4.5}$$

なりすまし検知における特異度は，なりすましの無実群を正しく無実群であると判定できた割合となる．なお，特異度は TN 率（True Negative Rate; TNR）とも呼ばれる．

FP 率（False Positive Rate; FPR）

FP 率（False Positive Rate）は，テストデータ全体における同一人ペア（負例）のうち，判定システムが誤って別人ペア（正例）と判定した割合であり，式（4.6）より算出される．

$$FP\text{率}（False\ Positive\ Rate; FPR）= \frac{FP}{FP + TN} \tag{4.6}$$

なりすまし検知における FP 率は，なりすまされていないのに，なりすまされていると誤って検知してしまう割合となり，えん罪につながる可能性の高さを示しているといえる．

FN 率（False Negative Rate; FNR）

FN 率（False Negative Rate）は，テストデータ全体における別人ペア（正例）のうち，判定システムが誤って同一人ペア（負例）と判定した割合であり，式（4.7）より算出される．

$$FN\text{率}（False\ Negative\ Rate; FNR）= \frac{FN}{TP + FN} \tag{4.7}$$

なりすまし検知における FN 率は，なりすまされているのに検知せず，なりすまされていないと誤って判定してしまう割合となり，犯人の取り逃がしにつながる可能性の高さを示しているといえる．

4.3　閾値Tの最適化

同一人判定にバイアスをかけたモデルBでは，3.3.6項で述べたように，判定の信頼度が閾値T以下のときには同一人と判定する．本節では，閾値Tの最適化の結果を報告する．閾値Tを0〜1までの間で0.01ごとに変化させ，モデルBによって開発データの著者の同一性判定を行い，精度・再現率・F値を計測し，F値が最大となる値を最適な閾値として選択する．

また，信頼度の閾値Tを変化させたときのROC曲線を描画し，ROC曲線下面積（AUC）を求める．これらの定義を以下に示す．

ROC曲線（Receiver Operating Characteristic Curve）

ROC曲線は，横軸にFP率（False Positive Rate; FPR），縦軸に再現率（Recall）をとり，閾値Tを変化させたときの両者の変化を示す曲線である．

ROC曲線下面積（Area Under the ROC Curve; AUC）

ROC曲線下面積は，ROC曲線の下の面積であり，0〜1の値をとる．この値が大きいほど判定システムの性能が高いことを示す．全ての事例に対して正解する完璧なシステムのとき，AUCは最大値1となる．

4.3.1　各モデルBにおける閾値Tの決定

表4.5は，モデルB-dif，B-sum，B-conのそれぞれについて，開発データにおいてF値が最大となった閾値TとそのときのF値を示した表である．交差検定の5回の試行のそれぞれの結果，およびその平均を示す．最終的に最適化された閾値は「平均」の行に記載されている値である．

表 4.5: 閾値 T の最適化の結果

モデル	交差検定	閾値 T	F 値
B-dif	1 回目	0.57	0.664
	2 回目	0.58/0.59	0.685
	3 回目	0.58/0.59	0.718
	4 回目	0.56	0.659
	5 回目	0.6	0.643
	平均	0.578	0.674
B-sum	1 回目	0.68	0.308
	2 回目	0.69	0.299
	3 回目	0.69	0.376
	4 回目	0.67	0.309
	5 回目	0.65	0.307
	平均	0.676	0.320
B-con	1 回目	0.57	0.362
	2 回目	0.58/0.59	0.436
	3 回目	0.61	0.466
	4 回目	0.58/0.59	0.431
	5 回目	0.61	0.426
	平均	0.59	0.424

最適化後の閾値 T は，モデルによって若干異なり，B-dif では 0.578，B-sum では 0.676，B-con では 0.59 となった．いずれにせよ 0.5 より大きい値であり，別人（正例）と判定する基準（信頼度）を少し厳しく設定したときに F 値が最大となることがわかった．

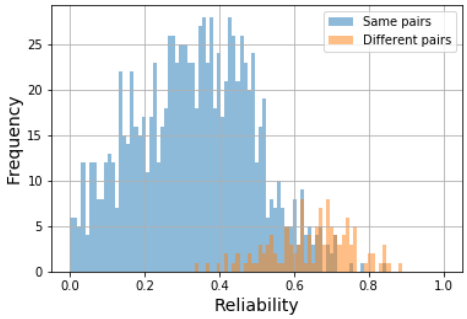
次に，提案モデルの性質を明らかにするための追加実験を行う．まず，モデル R を開発データに適用したとき，正例と負例に対する判定の信頼度の分布を調べる．具体的には，横軸を判定の信頼度，縦軸をその信頼度を持つ正例または負例の数とするグラフを作成する．また，閾値 T を変化させたときの開発データにおけるモデル R の精度，再現率，F 値の変動を調べる．閾値 T の最適化はモデル B のために行うが，ベースとなる判定モデルはモデル R なので，ここではモデル R の性能を測っていることに注意していただきたい．交差検定における 5 回の試行の結果をそれぞれ図 4.2〜図 4.6 に示す．これらの図において，(a-1)，(a-2)，(a-3) はそれぞれモデル R-dif，R-sum，R-con による判定の信頼度の分布を示す．青色の「Same pairs」は同一人ペアの事例の頻度，オレンジ色の「Different pairs」は別人ペアの事例の頻度を表す．一方，(b-1)，(b-2)，(b-3) はそれぞれモデル R-dif，R-sum，R-con による精度（Precision），再現率（Recall），F 値（F-measure）の変動を示す．

信頼度の分布をみると，モデル B-dif については，別人ペア（正例）と同一人ペア（負例）の分布がある程度分かれている一方，モデル B-sum と B-con については，別人ペアと同一人ペアとで信頼度が重なっている部分が大きいことがわかった．このことから，事例ベクトルの作成方法として，ベクトルの和や連結よりも，ベクトルの差を取った方がよいことが窺える．

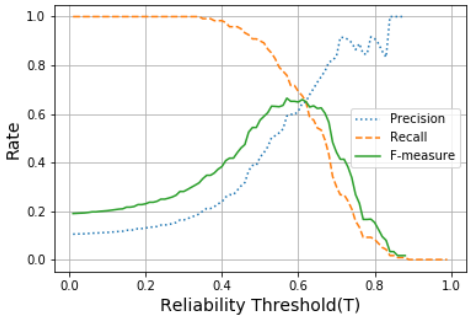
閾値と精度・再現率・F 値の関係のグラフをみると，いずれのグラフも閾値を上げると精度が向上するが再現率は低下する傾向がみられる．一般に精度と再現率はトレードオフの関係があるので，この結果は自然である．いずれのケースでもピークとなる閾値があり，これが表 4.5 に示した閾値の値に対応する．また，グラフの形状を比べると，いずれのモデルも再現率については大きな違いはみられなかった．一方，精度についてはグラフの右側でモデル R-dif とモデル R-sum，R-con に差がみられた．モデル R-sum，R-con では信頼度の閾値を上げていったときに，モデル R-dif と比べて精度が上がりにくくなっている．これは信頼度の分布とも関係しており，モデル R-sum，R-con では別人ペア（正例）と同一人ペア（負例）の分布の重なりが大きいことが原因であるといえる．このことからも，事例ベクトルの作成方法として，ベクトルの和や連結よりも，ベクトルの差を取った方がよいことが窺える．

さらに，モデル R を開発データに適用したときの ROC 曲線と AUC を調べる．モデル R-dif，R-sum，R-con の結果をそれぞれ図 4.7，4.8，4.9 に示す．5 本の曲線は 5 分割交差検定のそれぞれの試行の結果である．ROC 曲線は，全体的にモデル R-dif では左上方に位置し，AUC についてもいずれも 0.9 以上と高い値であった．一方，モデル R-sum と R-con では，R-dif と比較すると ROC 曲線が右下方に位置し，AUC についても 0.7〜0.8 と低下した．この結果からも，事例ベクトルの作成方法として対照文書ベクトルと疑問文書ベクトルの差を取る方式が最も適しているといえる．

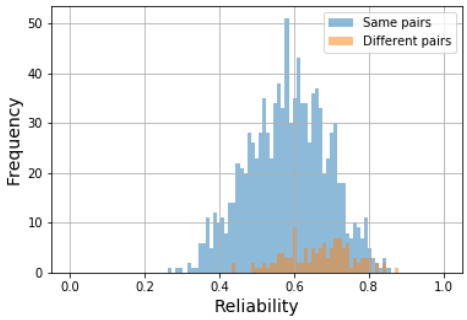
(a-1) 信頼度の分布 (R-dif)



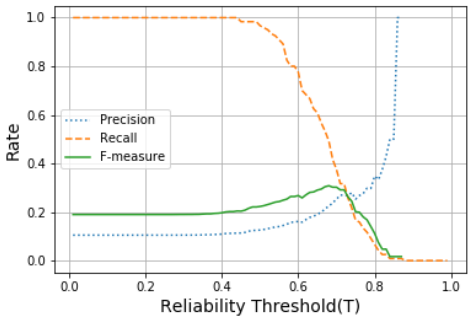
(b-1) 評価指標の変化 (R-dif)



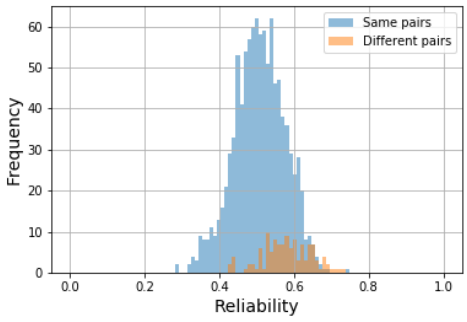
(a-2) 信頼度の分布 (R-sum)



(b-2) 評価指標の変化 (R-sum)



(a-3) 信頼度の分布 (R-con)



(b-3) 評価指標の変化 (R-con)

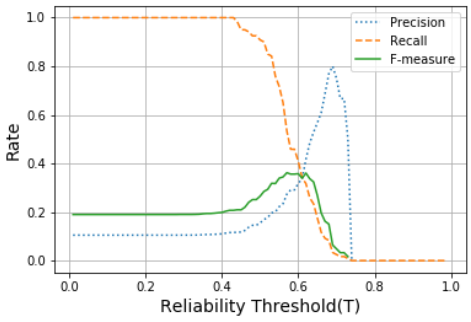
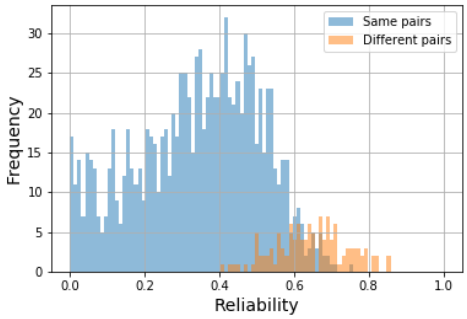
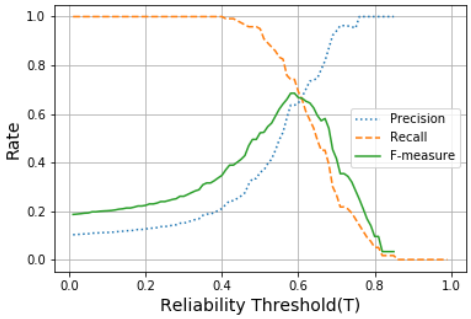


図 4.2: 開発データにおけるモデル R の評価 (交差検定 1 回目)

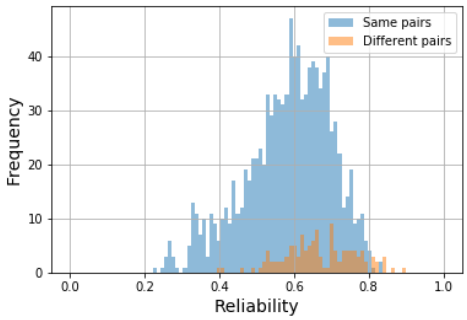
(a-1) 信頼度の分布 (R-dif)



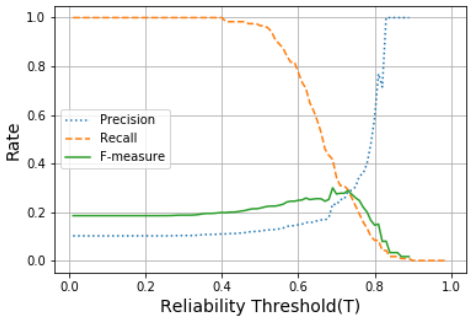
(b-1) 評価指標の変化 (R-dif)



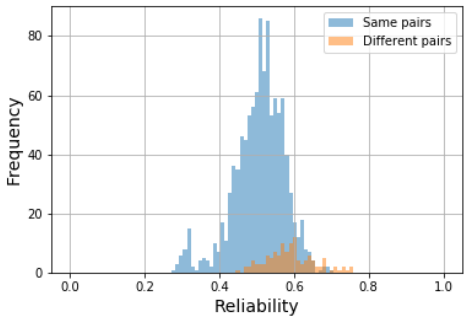
(a-2) 信頼度の分布 (R-sum)



(b-2) 評価指標の変化 (R-sum)



(a-3) 信頼度の分布 (R-con)



(b-3) 評価指標の変化 (R-con)

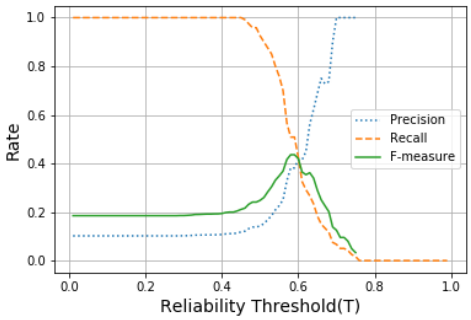
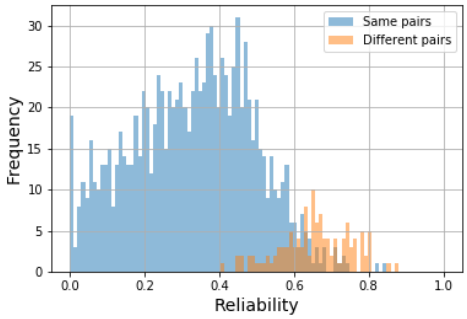
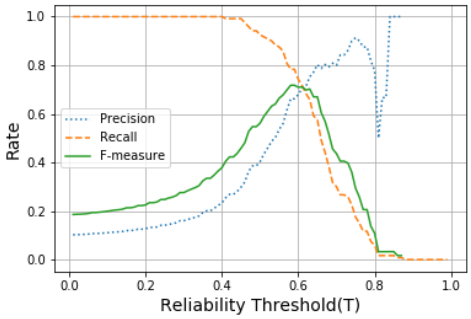


図 4.3: 開発データにおけるモデル R の評価 (交差検定 2 回目)

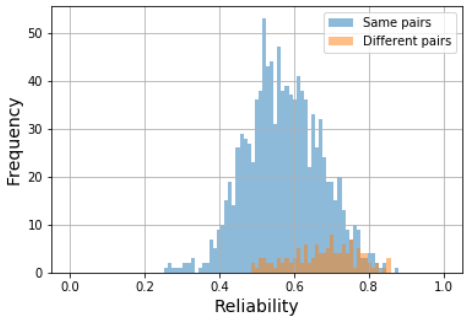
(a-1) 信頼度の分布 (R-dif)



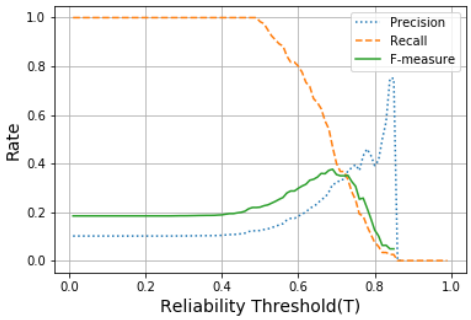
(b-1) 評価指標の変化 (R-dif)



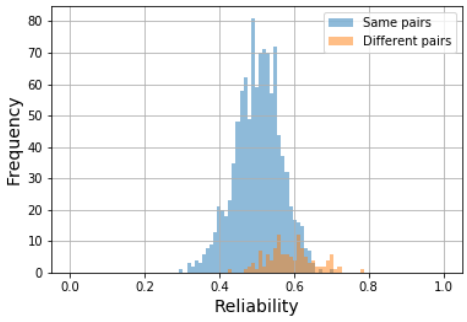
(a-2) 信頼度の分布 (R-sum)



(b-2) 評価指標の変化 (R-sum)



(a-3) 信頼度の分布 (R-con)



(b-3) 評価指標の変化 (R-con)

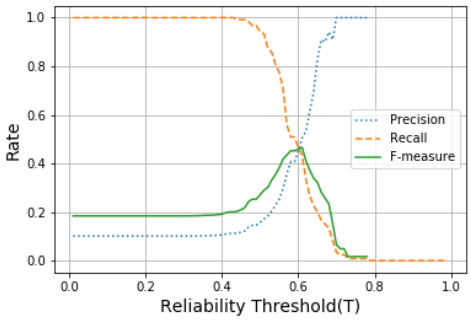
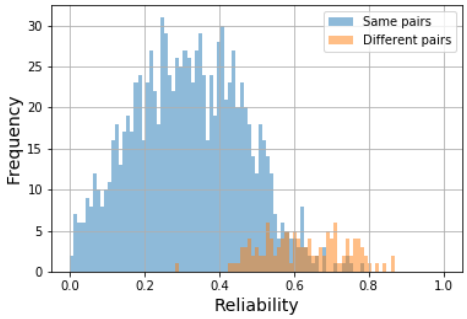
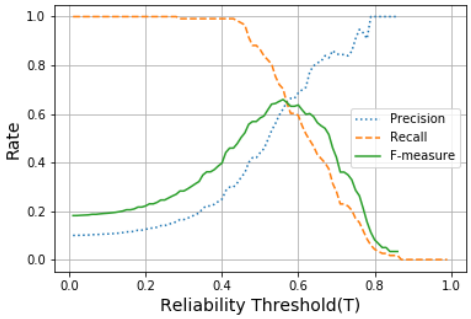


図 4.4: 開発データにおけるモデル R の評価 (交差検定 3 回目)

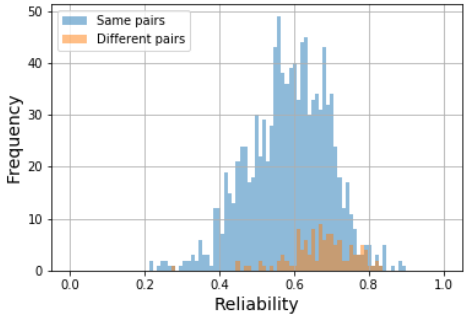
(a-1) 信頼度の分布 (R-dif)



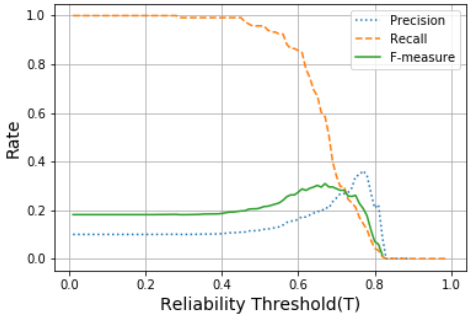
(b-1) 評価指標の変化 (R-dif)



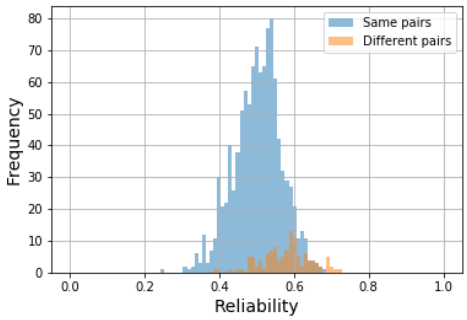
(a-2) 信頼度の分布 (R-sum)



(b-2) 評価指標の変化 (R-sum)



(a-3) 信頼度の分布 (R-con)



(b-3) 評価指標の変化 (R-con)

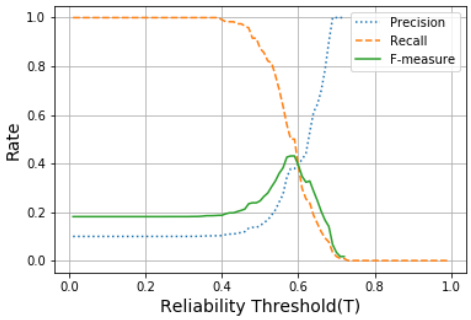
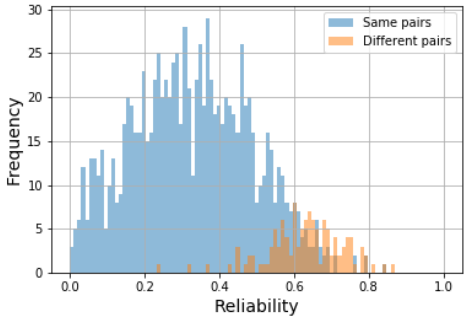
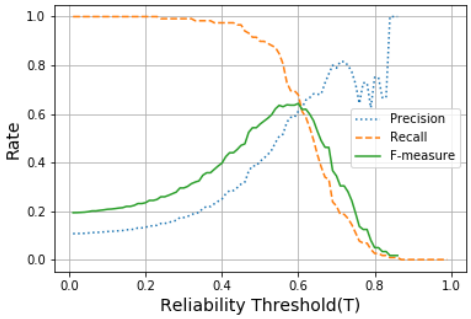


図 4.5: 開発データにおけるモデル R の評価 (交差検定 4 回目)

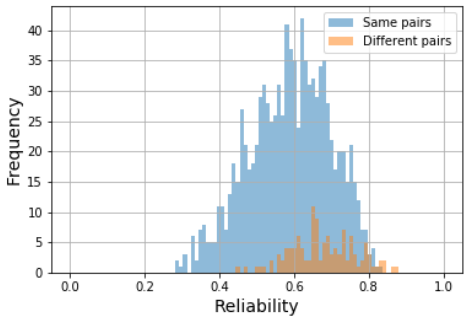
(a-1) 信頼度の分布 (R-dif)



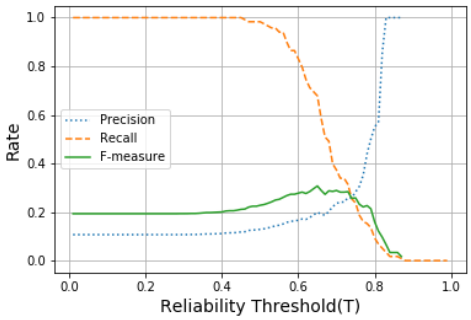
(b-1) 評価指標の変化 (R-dif)



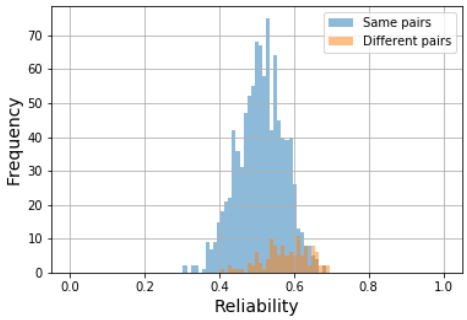
(a-2) 信頼度の分布 (R-sum)



(b-2) 評価指標の変化 (R-sum)



(a-3) 信頼度の分布 (R-con)



(b-3) 評価指標の変化 (R-con)

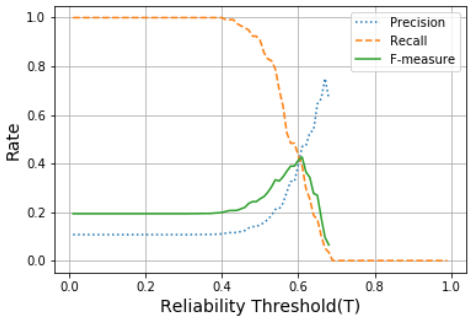


図 4.6: 開発データにおけるモデル R の評価 (交差検定 5 回目)

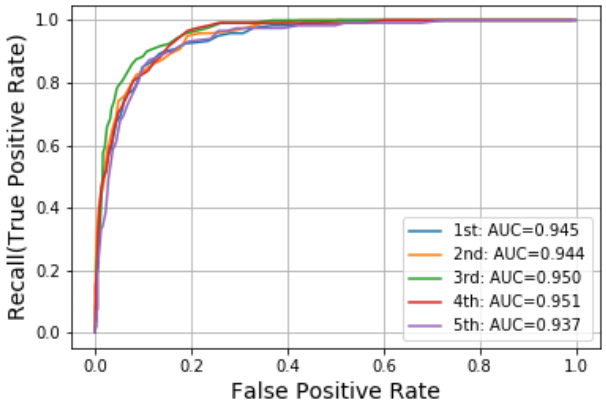


図 4.7: R-dif の開発データにおける ROC 曲線と AUC

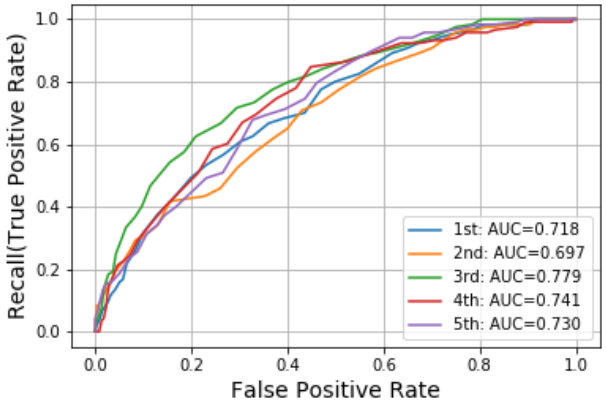


図 4.8: R-sum の開発データにおける ROC 曲線と AUC

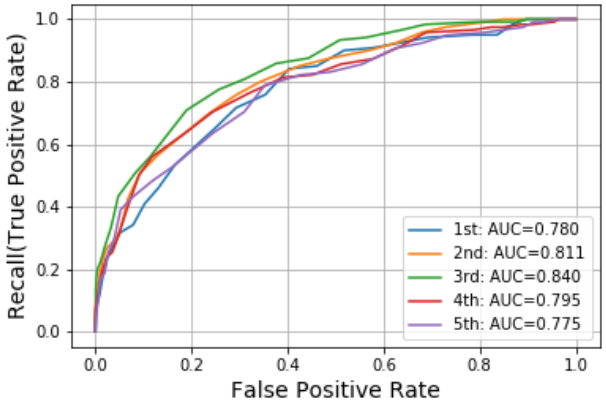


図 4.9: R-con の開発データにおける ROC 曲線と AUC

4.4 実験結果及び考察

表 4.6 は，6 種類の判定モデル R-dif，B-dif，R-sum，B-sum，R-con，B-con について，テストデータを適用させた結果得られた混同行列を示した表である．また，表 4.7 は，同様に 6 種類の判定モデルについて，テストデータを適用させた結果得られた精度，再現率，F 値，正解率，特異度，FP 率，FN 率を示した表である．なお，混同行列については，5 回の交差検定の各回数ごとの結果を示し，精度，再現率，F 値，正解率，特異度，FP 率，FN 率については，5 回の交差検定の各回数ごとの結果と 5 回の平均値を示している．そのため，最終的な各モデルの性能評価は,「平均」の行に記載されている値を用いて行った.

表 4.6: 各モデルの混同行列

			1 回目		2 回目		3 回目		4 回目		5 回目	
R-dif	TP	FN	108	12	115	5	112	8	108	10	113	5
	FP	TN	154	898	160	938	185	852	139	857	160	886
B-dif	TP	FN	91	29	97	23	82	38	96	22	86	32
	FP	TN	49	1003	54	1044	57	980	60	936	34	1012
R-sum	TP	FN	116	4	113	7	116	4	114	4	112	6
	FP	TN	780	272	787	311	825	212	837	159	828	218
B-sum	TP	FN	60	60	46	74	53	67	70	48	73	45
	FP	TN	183	869	155	943	124	913	251	745	318	728
R-con	TP	FN	111	9	112	8	112	8	104	14	113	5
	FP	TN	637	415	691	407	650	387	621	375	650	396
B-con	TP	FN	65	55	67	53	47	73	65	53	44	74
	FP	TN	134	918	151	947	62	975	114	882	65	981

表 4.7: 各モデルの性能評価結果

モデル	交差検定	精度	再現率	F 値	正解率	特異度	FP 率	FN 率
R-dif	1 回目	0.41	0.90	0.57	0.86	0.85	0.15	0.10
	2 回目	0.42	0.96	0.58	0.86	0.85	0.15	0.04
	3 回目	0.38	0.93	0.54	0.83	0.82	0.18	0.07
	4 回目	0.44	0.92	0.59	0.87	0.86	0.14	0.08
	5 回目	0.41	0.96	0.58	0.86	0.85	0.15	0.04
	平均	0.41	0.93	0.57	0.86	0.85	0.15	0.07
B-dif	1 回目	0.65	0.76	0.70	0.93	0.95	0.05	0.24
	2 回目	0.64	0.81	0.72	0.94	0.95	0.05	0.19
	3 回目	0.59	0.68	0.63	0.92	0.95	0.05	0.32
	4 回目	0.62	0.81	0.70	0.93	0.94	0.06	0.19
	5 回目	0.72	0.73	0.72	0.94	0.97	0.03	0.27
	平均	0.64	0.76	0.69	0.93	0.95	0.05	0.24
R-sum	1 回目	0.13	0.97	0.23	0.33	0.26	0.74	0.03
	2 回目	0.13	0.94	0.22	0.35	0.28	0.72	0.06
	3 回目	0.12	0.97	0.22	0.28	0.20	0.80	0.03
	4 回目	0.12	0.97	0.21	0.25	0.16	0.84	0.03
	5 回目	0.12	0.95	0.21	0.28	0.21	0.79	0.05
	平均	0.12	0.96	0.22	0.30	0.22	0.78	0.04
B-sum	1 回目	0.25	0.50	0.33	0.79	0.83	0.17	0.50
	2 回目	0.23	0.38	0.29	0.81	0.86	0.14	0.62
	3 回目	0.30	0.44	0.36	0.83	0.88	0.12	0.56
	4 回目	0.22	0.59	0.32	0.73	0.75	0.25	0.41
	5 回目	0.19	0.62	0.29	0.69	0.70	0.30	0.38
	平均	0.24	0.51	0.32	0.77	0.80	0.20	0.49
R-con	1 回目	0.15	0.93	0.26	0.45	0.39	0.61	0.08
	2 回目	0.14	0.93	0.24	0.43	0.37	0.63	0.07
	3 回目	0.15	0.93	0.25	0.43	0.37	0.63	0.07
	4 回目	0.14	0.88	0.25	0.43	0.38	0.62	0.12
	5 回目	0.15	0.96	0.26	0.44	0.38	0.62	0.04
	平均	0.15	0.93	0.25	0.43	0.38	0.62	0.07
B-con	1 回目	0.33	0.54	0.41	0.84	0.87	0.13	0.46
	2 回目	0.31	0.56	0.40	0.83	0.86	0.14	0.44
	3 回目	0.43	0.39	0.41	0.88	0.94	0.06	0.61
	4 回目	0.36	0.55	0.44	0.85	0.89	0.11	0.45
	5 回目	0.40	0.37	0.39	0.88	0.94	0.06	0.63
	平均	0.37	0.48	0.41	0.86	0.90	0.10	0.52

4.4.1 事例ベクトル作成方法の比較検証

本項では，事例ベクトル作成方法（ベクトルの差，ベクトルの和，ベクトルの結合）の違いについて検証する．表 4.7 に示した各モデルの評価結果から，4.3 節で行った開発データでの検証結果と同様に，テストデータを適用した本実験においても，ベクトルの和，ベクトルの連結を行ったモデルと比較して，ベクトルの差を取ったモデル（R-dif，B-dif）がほぼ全ての評価指標で結果が良かった．F 値を比較すると，R-dif，B-dif の F 値は 0.57，0.69 であるのに対し，R-sum，B-sum では 0.22，0.32，また R-con，B-con では 0.25，0.41 であった．ベクトルの和，ベクトルの連結で事例ベクトルを作成したモデルは，いずれも精度が 0.1〜0.4 程度と低くなっており，その結果 F 値も低くなっている．

以上から，事例ベクトルの作成方法について，ベクトルの差がその他の方法と比べて有効であるといえる．そのため，以降ではベクトルの差によって事例ベクトルを作成するモデル R-dif，B-dif について考察する．

4.4.2 同一人判定へのバイアスの効果の検証

本項では，モデル R-dif とモデル B-dif を比較する．モデル R は通常のランダムフォレストで学習された分類器であるのに対し，モデル B は，3.3.6 項で述べたように，判定の信頼度が十分高くないときには分類器が別人ペアと判定しても同一人ペアに判定を変更する手法である．つまり，同一人判定にバイアスをかけている．2 つのモデルを比較することで同一人判定へのバイアスの効果を検証する．表 4.6 の混合行列をみると，モデル R-dif と比較してモデル B-dif では，FP（同一人ペアを誤って別人ペアと判定）数が 1/2〜1/5 程度減少した．また，TN（同一人ペアを正しく同一人ペアと判定）の数も，B-dif は R-dif と比べて増加した．同一人判定へのバイアスは，同一人ペアの数が別人ペアの数よりはるかに多い不均衡データにおいて，FP を減らしたり TN を増やすための手法であり，その狙い通りの結果が得られている．一方，FN（別人ペアを誤って同一人ペアと判定）数は 2〜5 倍程度増加した．表 4.7 をみると，モデル R-dif と比較してモデル B-dif では，精度，F 値，正解率，特異度が向上した．さらに，FP 率の低下がみられた．これらは，FP 数の大幅な減少によるものである．一方，モデル R-dif と比べてモデル B-dif では，再現率が低下し，FN 率が上昇した．これらは，FN の増加によるものである．

著者同一性判定モデルの性能評価を行う上で重要な指標である F 値については，R-dif では 0.57 であるのに対して，B-dif では 0.69 であった．他の指標についても，精度 0.64，再現率 0.76，特異度 0.95 と比較的高い水準で安定している．なお，正解率については，モデル B-dif は 0.93 と高い値になったものの，テストデータの同一人（負例）・別人（正例）ペアのデータ数に偏りがあるため，正解率は高い値が得られやすい指標であることに留意する必要がある．

次に，FN 率，FP 率に着目する．R-dif と比較して B-dif では，FN 率が 0.07 から 0.24 に増加した．一方，FP 率については 0.15 から 0.05 に減少し，B-dif では低く抑えられているといえる．そのため，B-dif では，なりすましを検知できずに犯人を取り逃がす割合は増えるが，なりすましを誤検知してしまう割合を減らすことができる．つまり，誤ってなりすましを検知することでなりすましをしていない人が疑われてしまうケースを防ぎ，えん罪を抑止する効果があるといえる．えん罪は社会的な問題となっているため，著者同一性判定システムの運用において，なりすましの犯人の取り逃がしより，なりすましのえん罪につながる割合を低くすることも重要であると考える．このような観点からみると，R-dif よりも B-dif の方がよいといえる．

前項と本項での考察をまとめると，なりすまし検知における著者同一性判定においては，モデル B-dif，つまり，事例ベクトルの作成方法にベクトルの差を用い，なりすまし検知に特化するようにバイアスをかけたモデルが有効であることがわかった．

4.4.3 判定失敗事例の検証

著者同一性判定に失敗しやすい事例の傾向を知るため，テストデータをモデル R-dif に適用させ，5 分割交差検定により判定を行った際に判定に失敗した同一人ペアについて調査した．

判定に失敗した同一人ペアの疑問文書の文字数の分布を図 4.10 に示す．1 記事あたりの文字数の分布（図 3.3 参照）とグラフの傾向を比較すると，0〜500 文字（実際は 300〜500 文字）の記事の割合が多くなっているため，疑問文書の文字数が少なくなると判定に失敗しやすい傾向がみられることがわかった．

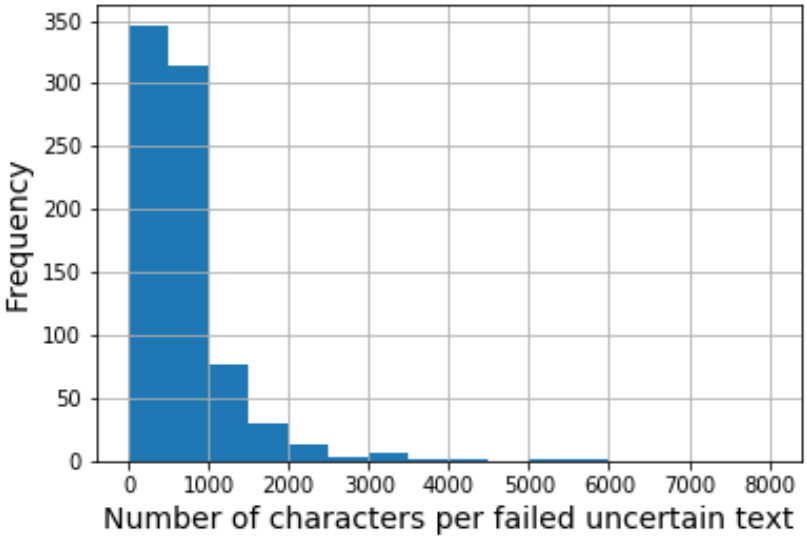


図 4.10: 判定に失敗した同一人ペアの疑問文書の文字数分布

本研究で用いたブログデータは著者 1 人あたり 10〜20 記事とばらつきがあり，著者 1 人あたりの記事数は対照文書群の作成に影響する．そこで，判定に失敗した同一人ペアの著者の記事数の分布を図 4.11 に示す．データセット全体での著者 1 人あたりの記事数の分布（図 3.4 参照）と比較すると，19 記事以下の割合が増加している．対照文書群の記事数が減少すると判定に失敗しやすい傾向にあることがわかった．

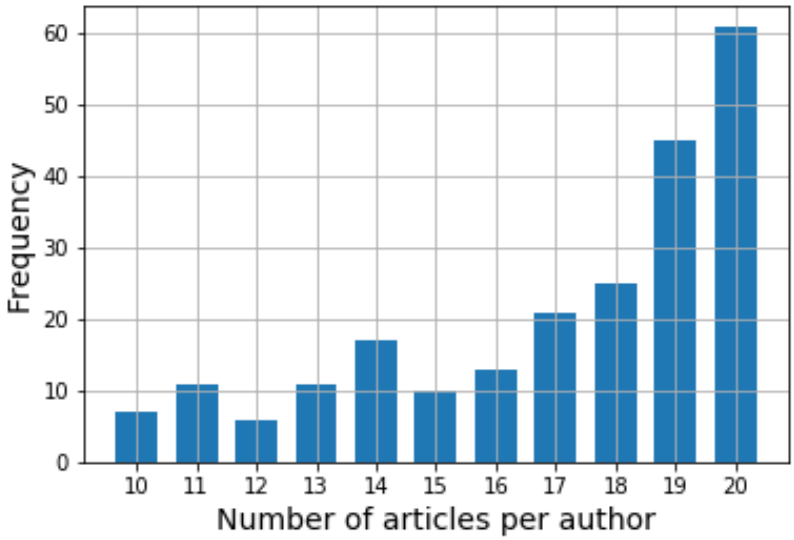


図 4.11: 判定に失敗した同一人ペアの著者 1 人あたりの記事数分布

同じ著者の同一人ペアが判定に失敗した回数の分布を図 4.12 に示す．例えば，横軸が 1 のときの頻度は，1 回しか判定に失敗しなかった著者の人数を表している．テストデータの著者数は交差検定の 5 回の試行の合計で 298 人であり，そのうち 1 回以上判定に失敗した著者数は 228 人であった．多くの著者が数回失敗する程度であったが，同じ著者が 10 回以上繰り返し失敗していることもあり，その著者については個別に記事の調査をする必要があると考えられた．

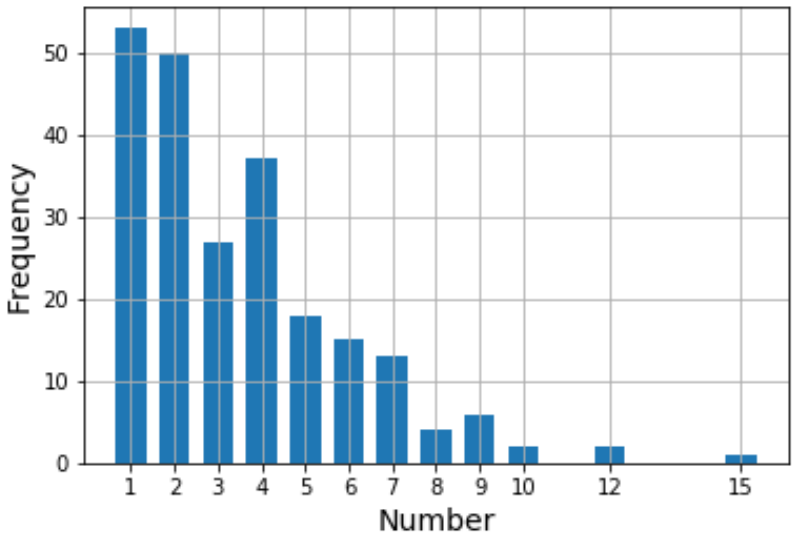


図 4.12: 同じ著者の同一人ペアが判定に失敗した回数の分布

第5章　おわりに

5.1　まとめ

本研究では，なりすましの自動検知を目的とし，日本語のテキストを対象とした著者同一性判定手法を提案した．まず，ブログ記事を著者 ID とともに収集した．得られたブログ記事のテキストに対して形態素解析を行い，4 種類の素性（単語の uni-gram・助詞の bi-gram・品詞の tri-gram・読点前の単語）を抽出し，記事ごとに文書ベクトルを作成した．次に，著者を訓練データ・開発データ・テストデータに分割した．それぞれのデータ内で疑問文書と対照文書の著者が異なる別人ペアと，両者の著者が同じである同一人ペアを作成した．疑問文書と対照文書のベクトルを組み合わせて事例ベクトルを作成する際に，ベクトルの差・ベクトルの和・ベクトルの連結といった 3 つの方法を考案した．訓練データの事例ベクトルを用いてランダムフォレストによって別人ペアと同一人ペアを識別する分類器を学習し，これをモデル R とした．さらに，ランダムフォレストによる判定の信頼度が閾値 T 未満のときには常に同一人と判定するように同一人判定にバイアスをかけた判定システムを構築し，これをモデル B とした．2 種類のモデル（モデル R・モデル B）と 3 種類の事例ベクトル作成方法（ベクトルの差・ベクトルの和・ベクトルの連結）を組み合わせ，合計 6 種類のモデルの性能をテストデータを用いて評価した．

実験では，実際になりすまし検知を行うことを想定し，開発データとテストデータでは別人ペアが同一人ペアより圧倒的に少なくなるように設定した．評価基準として，別人ペア検出の精度・再現率・F 値に加えて，特異度・FP 率・FN 率などを用いた．その結果，事例ベクトルを対照文書と疑問文書の差によって作成したときのモデル B の F 値が最も高く，その値は 0.69 であった．このモデルの再現率は 0.76，特異度は 0.95 であった．

以上から，なりすましの自動検知を目的とした著者同一性判定において，提案手法の有用性を確認することができた．

5.2 今後の課題

本研究で提案したモデルでは，なりすまし（別人ペア）の検出の F 値は 0.69 であり，これを更に改善することが今後の主な課題となる．具体的な方法を以下に述べる．

- 素性の重みの検討

  本研究では，事例ベクトルにおける素性の重みは，著者が書いた文書における出現頻度が反映されているが，著者による素性の使い方の違いは反映されていない．つまり，特定の著者の文書にだけよく出現し，他の著者の文書にはあまり出現しない素性は，著者の同一性判定に有効であると考えられるが，現状の素性の重みには，著者の違いは考慮されていない．この問題に対する解決策として，例えば TF-IDF による重み付けが考えられる．TF-IDF はある著者によって多く使われ，かつ他の著者によってあまり用いられていない素性に対して大きな値を取る．そのため，著者の特徴的な文体が事例ベクトルに反映されると期待できる．

- 新たな素性の検討

  本研究では，機械学習の素性として単語の uni-gram，助詞の bigram，品詞の trigram，読点前の単語の 4 つを用いた．これらの素性はいずれも品詞に着目した素性である．一方，文自体の構造に着目し，作家の文体の類似度を算出した研究が行われている [23, 24]．そこで，新たな素性として文の構文に着目した素性を導入したい．具体的には，形態素解析と構文解析により文の構文木を作成する．構文木は，文の構文的な構造や単語間の係り受け関係を木構造で表したものである．ただし，文章の内容による影響を排除するために，単語単独で意味を有する自立語（名詞・動詞・形容詞・形容動詞など）については品詞毎に割り当てた記号に置き換える．変換された構文木の木構造の一部を素性とするような方法が考えられる．

今後，以上で述べた対応策を実現する方法を検討し，より精度の高い判定モデルの構築に取り組みたい．

謝辞

本研究を進めるにあたり，数多くのご指導およびご助言いただきました主指導教員である白井清昭准教授に深謝いたします．また，副指導教員である長谷川忍准教授，研究室のメンバー，職場の皆様に心より感謝いたします．

参考文献

[1] 警察庁, https://www.npa.go.jp/, accessed January 11, 2021.

[2] Sara El Manar El Bouanani, Ismail Kassou. Article: Authorship Analysis Studies: A Survey. International Journal of Computer Applications, vol.86, no.12, pp.22–29, 2014.

[3] 松浦司, 金田康正. n-gram 分布を用いた近代日本語小説文の著者推定. 情報処理学会研究報告自然言語処理 (NL), vol.1999, no.95, pp.31–38, 1999.

[4] 松浦司, 金田康正. 近代日本小説家 8 人による文章の n-gram 分布を用いた著者判別者. 情報処理学会研究報告自然言語処理 (NL), vol.2000, no.53, pp.1–8, 2000.

[5] 金明哲. 日本語における単語の長さの分布と文章の著者. 社会情報, vol.5, no.2, pp.13–21, 1996.

[6] 上阪彩香, 村上征勝. 西鶴遺稿集の著者に関する統計分析-北条団水の浮世草子との文体比較-. じんもんこん 2014 論文集, vol.2014, no.3, pp.113–118, 2014.

[7] 金明哲. 助詞の分布における書き手の特徴に関する計量分析. 社会情報, vol.11, no.2, pp.15–23, 2002.

[8] 金明哲. 助詞の分布に基づいた日記の書き手の識別. 計量国語学, vol.20, no.8, pp.357–367, 1997.

[9] 金明哲. 助詞の n-gram モデルに基づいた書き手の識別. 計量国語学, vol.23, no.5, pp.225–240, 2002.

[10] 金明哲. 文節パターンに基づいた文章の書き手の識別. 行動計量学, vol.40, no.1, pp.17–28, 2013.

[11] 金明哲. 品詞のマルコフ遷移の情報を用いた書き手の同定. 日本行動計量学会第 32 回全国大会講演論文集, pp.384–385, 2004.

[12] 金明哲. 読点の打ち方と文章の分類. 計量国語学, vol.19, no.7, pp.317–330, 1994.

[13] 吉田篤弘, 延澤志保, 平石智宣, 斎藤博昭. 著者判別に有効な特徴量の推定. 情報処理学会研究報告情報学基礎 (FI), vol.2001, no.86, pp.83–90, 2001.

[14] 金明哲, 村上征勝. ランダムフォレスト法による文章の書き手の同定. 統計数理, vol.55, no.2, pp.255–268, 2007.

[15] 三品光平, 松田眞一. 文章の書き手の同定における分類法の精度比較. アカデミア. 情報理工学編 : 南山大学紀要 , vol.13, pp.35–46, 2013.

[16] Efstathios Stamatatos. Intrinsic plagiarism detection using character n-gram profiles. CEUR Workshop Proceedings, vol.502, pp.38–46 , 2009.

[17] 財津亘, 金明哲. テキストマイニングを用いた筆者識別へのスコアリング導入—文字数やテキスト数, 文体的特徴が得点分布に及ぼす影響—. 日本法科学技術学会誌, vol.22, no.2, pp.91–108, 2017.

[18] 財津亘, 金明哲. テキストマイニングによる筆者識別の正確性ならびに判定手続きの標準化. 行動計量学, vol.45, no.1, pp.39–47, 2018.

[19] Ameba ブログ. https://www.ameba.jp/, accessed January 11, 2021.

[20] MeCab. https://taku910.github.io/mecab/, accessed January 11, 2021.

[21] 青空文庫. https://www.aozora.gr.jp/, accessed January 11, 2021.

[22] Breiman Leo. Random Forests. Machine learning, vol.45, no.1, pp.5–32, 2001.

[23] 金川絵利子, 岡留剛. カーネル法による構文に着目した作家の文体の特徴づけと類似性分析. 人工知能学会論文誌, vol.32, no.3, pp.1–14, 2017.

[24] 小泉知夏, 菅原俊治. 係り受け関係の類似性に着目した小説の著者推定. 研究報告知能システム（ICS）, vol.186, no.7, pp.1–8, 2017.

付 録A　MeCabの品詞情報

形態素解析ツール MeCab によって出力される品詞情報の例を以下に示す.

表 A.1: MeCab の品詞情報 (連体詞，接頭詞，名詞，形容動詞)

表層形	いろんな	お	私	あからさま
品詞	連体詞	接頭詞	名詞	名詞
品詞細分類 1	*	名詞接続	代名詞	形容動詞語幹
品詞細分類 2	*	*	一般	*
品詞細分類 3	*	*	*	*
活用型	*	*	*	*
活用形	*	*	*	*
原形	いろんな	お	私	あからさま
読み	イロンナ	オ	ワタシ	アカラサマ
発音	イロンナ	オ	ワタシ	アカラサマ

表 A.2: MeCab の品詞情報 (形容詞，副詞，接続詞，助詞)

表層形	大きい	とても	しかし	が
品詞	形容詞	副詞	接続詞	助詞
品詞細分類 1	自立	助詞類接続	*	格助詞
品詞細分類 2	*	*	*	一般
品詞細分類 3	*	*	*	*
活用型	形容詞・イ段	*	*	*
活用形	基本形	*	*	*
原形	大きい	とても	しかし	が
読み	オオキイ	トテモ	シカシ	ガ
発音	オーキイ	トテモ	シカシ	ガ

表 A.3: MeCab の品詞情報 (助動詞，感動詞，記号，フィラー)

表層形	ない	ああ	、	あー
品詞	助動詞	感動詞	記号	フィラー
品詞細分類 1	*	*	読点	*
品詞細分類 2	*	*	*	*
品詞細分類 3	*	*	*	*
活用型	特殊・ナイ	*	*	*
活用形	基本形	*	*	*
原形	ない	ああ	、	あー
読み	ナイ	アア	、	アー
発音	ナイ	アー	、	アー