ISSN 1345_0093

愛知大学情報メディアセンター紀要

Vol.35/No.1 2026.JULY

COM

情報メディアセンター利用案内

◇サービス時間〈月～土曜日〉（都合により変更する場合があります。掲示およびホームページをご覧下さい。）

豊橋校舎（2025年9月15日よりCenter.Bldg.2階に移転しました）

区分	Center.Bldg.2F メディアゾーン	121～124教室 12A･12B情報ゼミ室 523教室	SEサービス
講義期間	（月～金） 9：00～19：50 （土） 10:00～16:50※	講義利用のみ	（月～金） 9：00～17：40
講義期間以外	（月～金） 9：00～16：50 （土） 10:00～16:50※		

※オリエンテーション期間、定期試験期間、及び卒論期間の土曜日のみ、記載された時間で開館します。それ以外の土曜日は無人でメディアゾーンを開館します。

名古屋校舎

区分	厚生棟4F メディアゾーン受付	講義棟7F メディアカウンター	SEサービス
講義期間	（月～金） 9：00～20：00 （土） 10：00～17：00	（月～金） 9：00～20：00 （土） 閉室	（月～金） 9：00～18：40
講義期間以外	（月～金） 9：00～17：00	閉室	（月～金） 9：00～17：00

車道校舎

区分	受付（3階）	メディアゾーン	SEサービス
講義期間	（月～金） 9：00～17：30	（月～金） 9：00～19：00	（月～金） 9：00～17：30
講義期間以外	（月～金） 9：00～17：00	（月～金） 9：00～17：00	（月～金） 9：00～17：00

■センター閉室日　／　日曜・祝日・夏期休暇期間（8月土曜日）・年末年始・創立記念日（11/15）・入試期間ただし，祝日授業日は開室

はじめに

情報メディアセンター所長　岩田　員典

本年度も愛知大学情報メディアセンター紀要COM50号を無事発刊できました。お忙しい中投稿していただいた執筆者の方々をはじめ、編集委員や発刊に関わっていただいた方々にお礼を申し上げます。今号は全５件となっております。多くの方々にご愛読いただければ幸いです。

さて、前号に引き続きAIの話題となりますが、今号でもAIに関する内容を含むものが３件あります。１件は私の論文で生成AIを利用して、疑似レポートの作成を行い、それを分析してみるというものです。『生成AI協働型の「データ分析論」における教育実践』では地域政策学部のデータ分析論において生成AIを協働させるという実践報告になっており、非常に興味深い内容となっていますので、是非ご一読ください。また、タイトルからはAIとの関連が分かりづらいのですが、『Moodle4.5および5.xバージョンにおける主要機能の進化に関する報告』では、MoodleにAI 技術が統合されており、今後どのような機能強化が行われるかについて言及されています。このように、AI関連の技術は生成AIの登場・普及により全世界に大きな衝撃と変革をもたらしています。日本ではまだ影響は少ないのですが、生成AIの台頭により、アメリカでは多くの仕事が代替できるようになったため、経験が少ない新卒者や代替が効く人員は不要だという状況が生じているようです。そのため、コンピュータサイエンス系の学部を卒業しても就職先がないという事態が起きています[1]。しかし、これは経験豊富な人材がきちんと生成AIを利用できるという前提があるので、「経験が無い人材を雇わない= 育てない」という構図は社会全体として将来的には非常に危ういのではないかと危惧しております。そのため、大学においては生成AIなどの仕組みから利用方法について教育をし、既にある程度の経験がある人材として社会に送り出すことが肝要になってくると考えております。

また、本学では、2027年度には社会情報学部の設置が決定しており、このような社会の変革に対応した教育をこの学部を中心に進めていけるのではないかと思っております。このように、生成AIをはじめとしたAIやコンピュータ関連技術は日々進化していきますので、それらの進歩に関した情報をCOMを通じて皆様に提供していく所存です。

---

[1] https://www.businessinsider.com/highest-unemployment-rates-college-majors-2026-2
ただし、理工系は大学院に進学をしないと就職は不利なので、その点は考慮する必要がある。

目　　次

はじめに …… 情報メディアセンター所長：岩田　員典

論文

動画教材における時刻連動型自動採点システムの設計と実装
…… 岩田　員典 …… 1

レポートデータに対する文章類似度指標と可視化手法の比較
― 人工レポートデータを用いて― …… 岩田　員典 ……15

GUI の自動操作とスクリーンショットに基づくGUI 採点支援機能を
有したPython プログラミング実習用システムの開発 …… 毛利　元昭 ……33

情報教育実践報告

生成AI 協働型の「データ分析論」における教育実践
― 地域政策学部の学生を対象に― …… 蒋　　湧 ……61

研究ノート

Moodle 4.5 および 5.x バージョンにおける主要機能の進化に関する報告
…… 森野　誠之 ……77

センターだより

ICT 委員会　会議報告……85

情報メディアセンター主催行事……86

Moodle（LMS）運営業務報告（2025年度）……87

ICT 委員会構成員……90

情報メディアセンター沿革・歴代所長……91

編集後記……92

原稿募集要項

執筆要項

論文

動画教材における時刻連動型自動採点システムの設計と実装

岩田 員典（愛知大学 経営学部）

要旨

本稿では，任意の動画教材に対して，出題者が指定した再生時刻に設問を設定し，自動採点と再視聴制御を実現する「時刻連動型自動採点システム」を設計・実装する．本システムは，動画に対する設問を JSON 形式で分離管理し，管理者が設定ファイルを生成してサーバに配置することで，動画を編集することなく設問を設定することができる．また，学習者はブラウザ上で動画を視聴しながら出題・採点を受けられる．学習者が合格点に到達できない場合，学習者は「その範囲の動画を再視聴して再解答」または「問題のみ再解答」を選択することができる．さらに，本手法は教員の採点負担を大幅に軽減し，動画教材と連動した設問による評価の枠組みへと拡張するものである．

キーワード： 動画教材，課題設定，自動採点，時刻連動型

1. はじめに

大学教育におけるオンデマンド学習の普及に伴い，動画教材を用いた学習支援が急速に拡大している．しかし多くの学習管理システム（LMS）では，動画は単なる「視聴コンテンツ」として位置付けられ，学習者の理解度をその場で評価する仕組みが不十分である．また，基本的に出題は動画視聴後と固定的であるため，動画内容と学習行動を柔軟に連動させることが難しい．

さらに，近年の大規模授業では，学習者数の増加に伴って教員の採点負担が急増しており，学習成果のフィードバックが遅延する問題が指摘されている．特に知識理解型の科目では，単純選択問題や短答問題の自動化による負担軽減が強く求められている．

これらの課題に対し，本研究では，動画再生中の任意の時刻に設問を提示し，解答データを即時採点・再視聴制御する「時刻連動型自動採点システム」を設計・実装した．本システムの特徴は次の四点にまとめられる．

1. 動画と設問を JSON 形式で分離管理し，任意の動画に後付けで設問を設定できる．
2. 採点と再視聴を自動化し，教員の採点・管理負担を軽減する．
3. 学習者の誤答に応じた再学習支援を

実現し，自己調整学習を促進する．

4. 解答に対する即時フィードバックを可能とし，学習者の理解保持を支援する．

本稿ではまず，システム全体の構成を示し，次に採点アルゴリズムの仕組みを詳細に解説する．続いて，教育的意義および既存研究との比較を通じて，本手法の新規性と有効性を検証する．

2. システム構成

提案システムは，管理者（教員）フェーズと利用者（学習者）フェーズに分かれ，JSON 形式の設定ファイルを中心に連携する．図 1 に連携の全体構成を示す．また，サーバに配置されるファイルは図 2 のような構造となる．それぞれのファイルについては後述する．

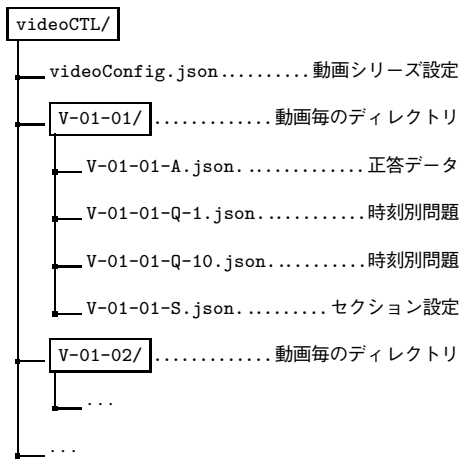


図 2: サーバに配置するファイルのディレクトリ構造

3. 管理者フェーズ

本章では図 1 の管理者フェーズについて説明する．

3.1. 動画シリーズの設定（図 1 の (1)）

管理者は (1) でどのビデオを出題するかの全体設定を行う．この設定は `videoConfig.json` として，図 2 で示したように `videoCTL` ディレクトリの直下に置く．

リスト 1 に `videoConfig.json` の例を示す．`VIDEO01` など最上位のキーはコンテンツ管理に利用し，これを基に表示順を並べ替えてユーザに提示される．このキーに対応した値（オブジェクト）がひとまとまりのコンテンツとして扱われる．また，2 階層目のキー `Name` は動画のタイトルを表し，`videos` は更にネストして 3 階層目で動画の ID とサブタイトルを指定する．なお，サブタイトルに `null` を指定した場合は何も表示されない．

リスト 1: (1) 動画シリーズの設定

```
{
  "VIDEO02": {
    "Name": "ネットワークアクセスの基礎 02",
    "videos": {
      "V-02-01": null,
      "V-02-02": "その2"}
  },
  "VIDEO01": {
    "Name": "ネットワークアクセスの基礎 01",
    "videos": {
      "V-01-01": "その1",
      "V-01-02": null}
  }
}
```

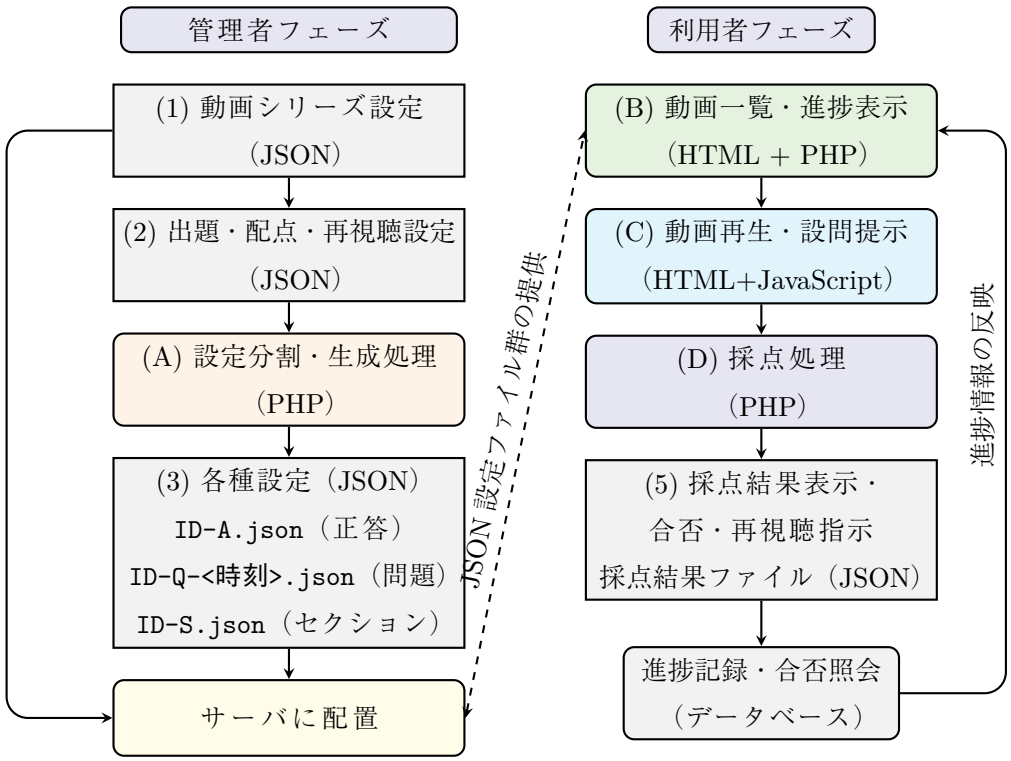


図 1: 動画教材における時刻連動型自動採点システムの全体構成

リスト 1 の JSON に基づいて作成された動画の一覧表示を図 3 に示す．ここでは，VIDEO01 がコンテンツ「ネットワークアクセスの基礎 01」を取り扱っており，コンテンツを構成するビデオが ID として V-01-01 と V-01-02 を持つことを表している．

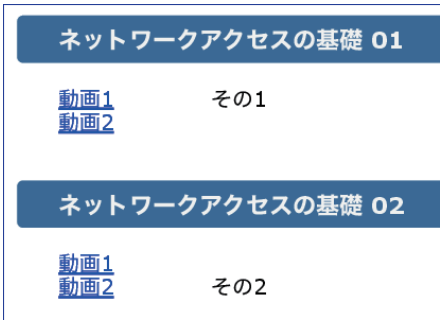


図 3: リスト 1 による動画一覧

3.2. 出題・配点・再視聴設定（図 1 の (2) と (A)）

(2) の出題・配点・再視聴設定では，設問内容，出題時刻，配点および再視聴方式を JSON ファイルとして定義する．リスト 2 はその JSON ファイルの記述例である．ここで，キー id に対応する値は (1) で指定した videos 内のキーと一致している必要がある．これらキーと値についての概要を表 1 に示す．

本 JSON ファイルを (A) の処理にかけることで，サーバに設置する正答，問題，セクションに関するファイルが生成される．

リスト 2: (2) 出題・配点・再視聴設定

```
{
  "id": "V-01-01",
  "play-mode": 7,
  "shuffle": true,
  "passing-percentage": 100,
  "score-proportion": 10,
  "sections": {
    "1": {
      "start": 0,
      "passing-percentage": 60,
      "questions": [
        [
          {
            "question": "A を
                ↪ 選んでね",
            "answer": ["A"],
            "dummies": ["B", "C",
                ↪ "D"],
            "score-proportion": 10
          },
          {
            "question": "M, N を
                ↪ 選んでね",
            "answer": ["M", "N"],
            "dummies": ["I", "L"],
            "feedback":
                ↪ "NとNを選ぼう. ",
            "score-proportion": 10
          }
        ],
        [
          {
            "question": "B と C
                ↪ を選んでね",
            "answer": ["B", "C"],
            "dummies": ["A", "D"],
            "feedback":
                ↪ "BとCを選ぼう. ",
            "score-proportion": 20
          }
        ]
      ]
    }
  }
}
```

3.2.1. セクションの設定

表 1 で記載した sections の詳細について説明する．

キー sections は値としてオブジェクトを持ち，オブジェクト内のプロパティが各セクションになる．各プロパティのキーは正の整数で表し，この値が動画における問題の出題時刻（秒）となる．各値は当該時刻に提示する問題群の設定（開始時刻 start，合格基準 passing-percentage の上書き，問題集合 questions など）を保持する．questions は 二次元配列で，一次元目が問題番号、二次元目がその問題番号に属する候補問題の集合を表す．要素が 1 つの場合はその問題が必ず出題され，2 つ以上ならランダムに 1 つが選ばれる．

つまり，キーで示された値のタイミングで問題が出題される．そして，プロパティが持つ値の部分が問題の詳細設定となり，オブジェクトとして記述する．この問題の詳細設定については表 2 に概要を示す．

リスト 2 では，sections が持つプロパティは一つであり，そのキーは「1」となっている．そして，その値としてオブジェクト {"start": 0, ...} を持つ．そのため動画内の時刻「1」（秒）になるとオブジェクト内の問題が出題される．

ここで，表 2 の sections の説明だけでは questions について理解しづら

表 1: 出題・配点・再視聴設定の JSON ファイルにおけるキーと値

キー	値の概要
id	問題の ID．リスト 1 の ID と一致する必要がある．
play-mode	合格点に達しなかった場合の再受験方法を制御する ビットマスク．3bit で表現． bit0（値 1）： 先頭から動画を再生して再受験 bit1（値 2）： 誤答したセクションのみを再生して再受験 bit2（値 4）： 動画を再生せず，再受験のみ
shuffle	各セクションで問題をシャッフルして出題するかどうか
passing-percentage	セクションでの合格基準．値は %
score-proportion	各問題の得点．セクション内で合計をとり 比率に合わせて各問題の得点を決める．
sections	各問題を設定． 値はオブジェクトとなり，各問題設定を持つ． この各オブジェクトのキーが問題の出題時刻となる． 詳細は 3.2.1 項にて説明．

いので，リスト 2 を用いて説明をする．この例では questions の一次元目は 2 つ値を持っており，二次元目はそれぞれ 2 つと 1 つのオブジェクトを持つ．より具体的には questions[0][0]，questions[0][1]，questions[1][0] と 3 つの設問を持つことになる．ここで，questions[0] がもつ 2 つの設問のうち，いずれか 1 つが出題される．つまり，問題番号「0」は 2 つの問題からどちらか一方が出題される．一方で，問題番号「1」は一問しかないため，必ずこの問題が出題される．

questions の各要素（二次元目）は表 3 で示すオブジェクトとなる．ここで，解答方式がラジオボタンになるかチェックボックスになるかは answer の値である配列の要素数によって自動で判断される．

このような JSON ファイルを (A) の設定分割・生成処理にかけることで，(3) の各種設定が作成される．

3.3. 各種設定ファイル（図 1 の (3)）

(3) の各種設定は以下の (3.3) から (3.3) に示すの 3 種類のファイルに分けられる．ここで (3.3) と (3.3) は各動画に対して一つずつ作成されるのに対して，(3.3) は設問の数だけ作成される．そのた

表 2: sections の詳細設定

キー	値の概要
start	このセクションの動画の開始時刻を表す． この時刻からキーで示した時刻までがセクションとなる．
passing-percentage	セクションの合格基準．全体設定をここで上書きができる． 指定がない場合は全体設定を利用する．
questions	問題の設定をする．値は二次元配列となる． ここで二次元配列となっているのは，一次元目が 問題番号を表し，二次元目はその問題番号における問題の 候補を表すからである． なお，二次元目のオブジェクトが一つしかない場合は その問題が出題される．また，配列の要素が設問となり， 各設問はオブジェクトとして記述する．

表 3: question の詳細設定

キー	値の概要
question	問題文を記載する．
answer	配列として正答を記載する．正答が一つの場合は ラジオボタンに，複数ある場合はチェックボックスになる．
dummies	正答以外として表示するダミー選択肢
feedback	間違った場合に表示されるメッセージ．
score-proportion	得点比率．全体設定をここで上書きができる．

め，(3.3) は ID に出題時刻を加えたファイル名となる．リスト 2 から作成された 3 種類の JSON ファイルをそれぞれリスト 3，4，5 に示す．

(I) ID-A.json
　正答データ：リスト 3

(II) ID-Q-<time>.json
　時刻別問題：リスト 4

(III) ID-S.json
　セクション設定：リスト 5

以下で，これらの詳細について説明する．

(I) ID-A.json（正答データ）
　questions の二次元目として複数配置されていた問題も，本ファイ

ル内では同じネストレベルでプロパティとして存在する．したがって，ID そのままでは区別できなくなるため，これまでの ID の基本部分（ID-A.json などの ID）に出題時刻，配列における一次元目の位置と二次元目の位置を加えたものを，プロパティのキーとする．これが本ファイルにおける問題の ID となる．つまり，ID-出題時刻-一次元目-二次元目 のような表現になる．これをキーとして設定し，値はオブジェクトになる．

たとえば，ID の基本部分が V-01-01 であるとすると，出題時刻が「1」で，問題一覧の配列における場所が [0][1] の場合に，V-01-01-1-0-1 となる（リスト 3の 2 つ目のプロパティ）．

正答はキー answer に対する値として保存されるが，正答をそのまま保存するのではなく md5 によるハッシュ値が保存される．このようにすることで，ID-A.json をユーザが手に入れたとしても即座には正答が分からないようになっている．

(II) ID-Q-<time>.json（時刻別問題）

ここで time はその問題を出題する時刻（単位秒）を表す．たとえば，ID-Q-300.json の場合に，動画開始から 300 秒後に出題する問題が保存されている．

本ファイルは表 3 に基づいた問題に関する情報を持つが，ファイルの内容が出題時にフロントエンドに渡されるため，答えは含んでいない．したがって，選択肢は answer と dummies を合わせて candidates として保持されている．type は答えが複数ある場合は「checkbox」に，単一の場合は「radio」となる．また，id は上記 (3.3) に合わせた値になる．

1. ID-S.json（セクション設定）

   キー timing で指定されている値がセクション（時間帯）を表す．セクションは二次元配列になっており，一次元目は各セクションを差し要素数はセクションの数になる．二次元目は各セクションの開始時刻と終了時刻を保持するため要素数は 2 となる．また，キー q-base-id は上記 (3.3) の<time>より前の部分を値として持つ．

   なお，その他のキー id，shuffle，play-mode，passing-percentage はいずれも表 1 と同じである．

リスト 3: V-01-01-A.json の例

```
{
 "V-01-01-1-0-0": {
  "feedback": "",
  "score-proportion": 10,
  "answer": [
   "7fc56270e7a70fa81a5935b72eacbe29"
  ]},
 "V-01-01-1-0-1": {
  "feedback": "MとNを選ぼう. ",
  "score-proportion": 10,
  "answer": [
   "69691c7bdcc3ce6d5d8a1361f22d04ac",
   "8d9c307cb7f3c4a32822a51922d1ceaa"
```

```
  ]},


 "V-01-01-1-1-0": {
  "feedback": "BとCを選ぼう. ",
  "score-proportion": 10,
  "answer": [
   "9d5ed678fe57bcca610140957afab571",
   "0d61f8370cad1d412f80b84d143e1257"
  ]
 }
}
```

リスト 4: V-01-01-Q-1.json の例

```
{
  "shuffle": true,
  "passing-percentage": 60,
  "start": 0,
  "end": 1,
  "questions": [
    [
      {
        "question": "A を 選んでね",
        "id": "V-01-01-1-0-0",
        "type": "radio",
        "candidates": [
          "A",
          "B",
          "C",
          "D"
        ]
      },
      {
        "question": "M, N を
            ↪ 選んでね",
        "id": "V-01-01-1-0-1",
        "type": "checkbox",
        "candidates": [
          "I",
          "L",
          "M",
          "N"
        ]
      }
    ],
    [
      {
        "question": "B と C
            ↪ を選んでね",
        "id": "V-01-01-1-1-0",
        "type": "checkbox",
        "candidates": [
          "A",
          "B",
          "C",
          "D"
        ]
      }
    ]
 ]
}
```

リスト 5: V-01-01-S.json の例

```
{
  "timing": [
    [
      0,
      1
    ]
  ],
  "shuffle": true,
  "play-mode": 7,
  "passing-percentage": 100,
  "id": "V-01-01",
  "q-base-id": "V-01-01-Q-"
}
```

4. 利用者フェーズ

本章では図 1 の利用者フェーズについて説明する.

4.1. 動画の選択と設問の表示（図 1 の (B) と (C)）

図 1 の利用者フェーズ (B) において，図 3 のような動画一覧は (1) の「動画シリーズの設定」で用意された `videoConfig.json` により作成される．利用者はこの一覧から受講したい動画を選択する.

(C) において，動画が選択されると動画の ID に合わせて (3.3) の `ID-S.json` が読み込まれる．これにより，利用者が動画を視聴すると，`ID-S.json` で指定されたタイミングに応じて (3.3) の `ID-Q-<time>.json` が読み込まれ，そこに記載された問題が出題される．このように，問題はあらかじめ読み込まれているわけではなく，出題のタイミン

グで適宜ダウンロードされる．そのため `ID-Q-<time>.json` のように時刻がファイル名に含まれている．

ここでは例として，`V-01-01-Q-1` の `V-01-01-1-0-0` （設問 1 の候補のうちの 1 つ目）と `V-01-01-1-1-0` （設問 2）が出題されている状態を図 4 と図 5 に示す．なお，設問 1 は間違いとなるように解答を選び，設問 2 は正解となるように解答を選んでいる．



図 4: 問題への解答例 1



図 5: 問題への解答例 2

4.2. 解答の送信と採点処理（図 1の (D) と (5)）

それぞれの問題の解答を選択すると図 6 のように採点前の確認画面になる．ここで，右にある「変更」ボタンを押すと，選択済みの解答を変更することができる．下部にある「採点」ボタンを押すと解答がサーバに送信され，(3.3) の `ID-A.json` に保存されている値と照合することで即座に採点され，その結果が表示される．なお，採点の仕組みについては 5.1 項で説明する．

採点した結果，合格点に達しなかった場合は図 7 のように，問題のみを再受験するか，動画を見て問題を再受験するか，再受験無しで次のセクションへ移動するかを選ぶことができる[注 1)]．一方で，合格した場合は図 8 のように次のセクションへ進むボタンだけが表示される[注 2)]．



図 6: 採点前の確認



図 7: 合格点に達しなかった場合



図 8: 合格した場合

また，採点情報はデータベースに蓄積することを想定しており，教員が後から個別進捗を確認できる．

なお，本システムは本学の情報リテラシー入門・応用で利用している HITs[1)] に組み込む予定である．

5. 採点アルゴリズムと教員負担の軽減

5.1. 採点アルゴリズム

本節では図 1 の (D) と (5) の部分について説明する．

本システムの採点アルゴリズムは，単純な構成を採用しており，以下の 3 段階で処理される．

(1) 入力データ受理と識別
学習者の入力は JSON 形式で送信され，動画 ID・設問 ID・解答値をキーとして受理される．同時にユーザ情報（学籍番号など）を付与し，採点ログの一意性を担保する．

(2) 正答照合と部分点処理
サーバ側では `ID-A.json` を参照し，設問ごとにハッシュ照合を行う．

(3) 合否判定と再視聴指示
総得点率 r を算出し，合格閾値 T を超えれば「合格」とする．誤答があった場合は `ID-S.json` に基づき，「該当問題の再解答」または「該当セクションの再視聴」モードを選択する．この動作はクライアント側の JavaScript で制御され，動画のシーク操作やボタン制限によって再視聴が強制される．

5.2. 教員負担の軽減

従来，教員は学生のレポート・小テスト採点に多大な労力を要していた．本システムでは，HITs[1)] と同様に採点が全自動で行われるため，採点作業時間は大幅に削減できる．

さらに，動画を直接編集せずに JSON ファイルを用意するだけで動画にあった問題が作成できるため，それぞれを独立して準備すればよい．したがって，動画に対して柔軟に問題設定ができるだけでなく，動画を活用した課題の設定が容易になる．

また，採点結果と再視聴ログがデータベースに蓄積されることで，個別の理解度把握が容易になり，教員は学習支援や教材改善に注力できるようになる．

不正防止策として，解答選択肢のシャッフルおよびシーク操作の制限を実装している．一方で，再生速度の制限は学習者の主体性を阻害する可能性があるため採用していない．また，ユーザ管理や IP アドレスによるアクセス管理は連携先のシステム（HITs）で実施している．

6. 教育的意義および既存研究との比較

本システムは「学習の途中で理解を確認し，誤解を修正する」点に特徴がある．

従来の一方向的動画教材とは異なり，学習者の応答に基づいて動画の流れが変化するため，能動的学習（Active Learning）の促進が期待できる．

Wachtler ら（2016）[2] や Rice ら（2019）[3] は，出題タイミングが学習成果に与える影響を分析し，設問を挿入することが理解保持に有効であると報告している．本研究はこれを実際に運用可能なシステムとして具体化した．

佐藤ら（2015）[4] は講義動画にクイズを挿入した場合に ARCS モデルに基づく学習意欲のどの側面が向上するかを実験的に検討している．その結果，クイズ挿入によって「自信」と「満足感」が有意に向上したが，「注意」と「関連性」では有意な向上は見られなかったと述べられている．したがって，動画に設問を埋め込むことによって受講者の「自信」と「満足感」の側面を向上させる効果があることが示唆している．

Davies（2022）[5] は PlayPosit（現 WeVideo[6]）というツールを，YouTube 等の任意動画に時間埋め込み型のクイズやリフレクティブ・ポーズなどを追加する実践報告をしている．本稿で述べている方針と類似しているだけでなく，GUI でクイズなど設定が可能であるため使いやすい．ただし，有料であるため費用的な面から導入が難しい場合が想定される．

また，Bloom の完全習得学習（Mastery Learning）[7, 8]，や Zimmerman の自己調整学習理論[9] に基づき，「誤答後の即時再学習」を自動的に誘発する仕組みを実現した．

加えて，Office 自動採点システム HITs[1] を基盤に，動画・時刻連動という新しいメディア次元を導入した点も特徴的である．

さらには，Romero ら（2020）[10] が指摘するラーニングアナリティクスへの発展可能性を内包しており，本システムは学習データ活用のプラットフォームとしても期待できる．

7. 本手法の新規性と有効性

提案手法は，技術的・教育的両側面で以下のような新規性を有する．

7.1. 技術的観点

1. 動画と設問を完全に分離した構造により，既存動画への後付け設問設定を可能とした．
2. LMS を介さず，軽量な PHP・JSON 構成で運用できる．
3. 正答照合・部分点処理・再視聴制御を自動化し，採点作業の効率化を実現．
4. 学習ログを標準化形式で保存し，教員側での学習分析が容易．

7.2. 教育的観点

1. 誤答時の即時再視聴により，メタ認知的学習を支援.
2. 出題時刻制御により，動画内容と理解確認を緊密に連動.
3. 自動採点により，教員の負担を軽減し，学習者への即時フィードバックを実現.
4. 進捗データの可視化により，学習者のモチベーション維持を支援.

現状は自動採点の信頼性を担保するため選択式問題を中心とした設計を採用しているが，今後は大規模言語モデル（Large Language Model: LLM）を組み込むことで短文記述式への拡張が期待される．これにより，より高次の学習目標に対応する評価形式の実現を目指す.

8. おわりに

8.1. まとめ

本稿では，任意の動画教材に対して，時刻連動型の自動採点・再視聴制御を実現するシステムを設計・実装した．本システムは既存動画を改変することなく，理解確認と即時フィードバックを統合した学習支援を可能にし，教員の教材作成および採点負担を大幅に軽減する.

8.2. 今後の課題

本システムは現在運用テスト段階にあり，実授業環境における学習効果測定は今後の研究課題である．特に，理解度向上への寄与，再学習行動の変容，視聴ログを活用した行動分析などについて，授業実践を通じて詳細に検証する予定である.

また，現在は JSON ファイルを各種エディタで作成する必要があるが，それが不慣れな管理者向けに Web ベースの設問用 JSON ファイル作成ツールも提供する予定である.

さらに，将来的には LLM による記述式採点支援，および LTI 標準に準拠した Moodle 等との連携を検討している.

謝辞

本稿を執筆するにあたり本学国際コミュニケーション学部教授 梅垣敦紀先生に作成していただいた upLaTeX のクラスファイル [11] を利用した。ここに感謝の意を表す.

注

注 1）この表示は `play-mode` により変更可能である.

注 2）合格基準が満点でなく，得点が満点でない場合は，再度問題を解くことができる.

参考文献

1） 岩田 員典, 松井 吉光, 「情報リテラシーのための統合システム構築と運用 –Microsoft Excel・Word 自動採点システムを含む–」, 愛知大学情報メディアセンター紀要「COM」, Vol. 31, No. 1, pp. 1–10, 2022, https://saturn.aichi-u.ac.jp/img/center/pdf/com46-2.pdf.

2） Wachtler, J., Hubmann, M., Zöhrer, H., & E. Martin, “An analysis of the use and effect of questions in interactive learning-videos”. Smart Learn. Environ, Vol. 3, No. 13, 2016, https://doi.org/10.1186/s40561-016-0033-3.

3） Rice, P., Beeson, P. & Blackmore-Wright, J. “Evaluating the Impact of a Quiz Question within an Educational Video.” TechTrends 63, pp. 522–532, 2019, https://doi.org/10.1007/s11528-019-00374-6.

4） 佐藤 満明, 柄本 健太郎, 向後 千春, 「講義動画中におけるクイズの提示が受講者の学習意欲に及ぼす効果」, 日本教育工学会論文誌, 39 巻, Suppl 号, pp. 77–80, 2015 https://www.jstage.jst.go.jp/article/jjet/39/Suppl/39_S39051/_article/-char/ja/

5） Davies R., “Interactive Videos with PlayPosit” The Language Teacher, 46(3), 2022. https://jalt-publications.org/articles/27737-interactive-videos-playposit.

6） WeVideo, https://www.wevideo.com.（2025 年 11 月 19 日アクセス）.

7） Bloom, B. S., “Learning for Mastery”. Evaluation Comment (UCLA—CSIEP), Vol. 1, No. 2, pp. 1-12, 1968

8） Akpan, B. “Mastery Learning—Benjamin Bloom”. In: Akpan, B., Kennedy, T.J. (eds) Science Education in Theory and Practice. Springer Texts in Education. Springer, Cham. 2020, pp. 149-- 162https://doi.org/10.1007/978-3-030-43620-9_11

9） Zimmerman, B. J., “Becoming a Self-Regulated Learner: An Overview,” *Theory Into Practice*, 41(2), pp. 64–70, 2002, https://www.tandfonline.com/doi/abs/10.1207/s15430421tip4102_2.

10） Romero, C., Ventura, S., “Educational data mining and learning analytics: An updated survey,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(3), 2020,

https://arxiv.org/abs/2402.07956.

11 ） 梅垣 敦紀, 「愛知大学情報メディアセンター紀要「COM」への upLaTeX での投稿」, 愛知大学情報メディアセンター紀要「COM」, Vol. 34, No. 1, pp. 1–10, 2025, https://saturn.aichi-u.ac.jp/img/center/pdf/com49-2.pdf.

Title: **Design and Implementation of a Time-Synchronized Automatic Grading System for Video-Based Instructional Materials**
Author: **IWATA, Kazunori**

論文

レポートデータに対する文章類似度指標と可視化手法の比較分析の検討 –人工レポートデータを用いて–

岩田 員典（愛知大学 経営学部）

要旨

本研究では，大学のレポート課題のように「大きなテーマは共通だが内容や論点が異なる」文章集合を対象とし，複数のテキスト埋め込みモデルによる類似度計算および可視化手法の振る舞いを比較する．実際の学生レポートのみを用いた評価では，教師ラベルの粒度やデータ数の制約によりモデル間の差が明確に現れにくいという問題がある．そこで本研究では，大テーマを共通としつつ，執筆方針や論点構成の異なるレポートを擬似的に大量生成する人工レポートデータ生成手法を構築し，その上で TF–IDF，fastText，Sentence-BERT (SBERT)，E5，OpenAI Embeddings の 5 手法を比較する．さらに，t-SNE，UMAP による次元削減結果を可視化し，各モデルがレポートの構造をどの程度クラスタとして再現できるかを検討する．実験の結果，SBERT，E5，OpenAI Embeddings は人工レポートの執筆戦略ごとのクラスタをほぼ完全に分離できる一方で，TF–IDF と fastText はクラスタ構造を十分に再現できないことが定量評価・可視化の両面から確認された．これにより，レポート自動分類や類似レポート検出においては，従来広く用いられてきた BoW/単語平均型の埋め込みよりも，文レベル意味ベクトルモデルを採用することの有効性が示された．

キーワード： LLM，疑似レポート作成，レポート分類

1. はじめに

大学教育において，レポート課題は学習者の理解度や思考の深さを把握するための重要な評価手段である．しかし，提出されるレポートの数は多く，教員がすべてを精読し，内容の類似性や論点の偏りを把握することは容易ではない．このため，レポートの自動分類や類似レポート検出に向けた自然言語処理技術（Natural Language Processing: NLP）の活用が求められている．

従来，文章の類似度計算には TF–IDF による Bag-of-Words (BoW) 表現[1]や，word2vec[2, 3]の拡張である fastText[4]などの単語分散表現を平均した文ベクトルが広く用いられてきた．しかし，これらの手法は単語の出現頻度や局所的な共起に依存しており，長文レポートのように語彙が多様で構造が複雑な文章に対し

ては，執筆方針や論点の違いを十分に捉えられないことが指摘されている．

一方，近年は Transformer ベースの大規模言語モデル（Large Language Model: LLM）[5] を用いた文レベルの埋め込みモデルが多数提案されている．Sentence-BERT (SBERT) [6] や E5 [7] などは，類似文対や検索タスクを用いた対照学習により，意味的に類似した文を近傍に写像することを目的として設計されている．これらのモデルは，英語を中心としたベンチマークにおいて従来手法を大きく上回る性能を示しているが，日本語のレポートデータに対してどの程度有効であるかについては，体系的な検証が十分ではない．

さらに，教育実践で得られるレポートデータは，授業や年度によってテーマや難易度が異なり，事前に明確なラベルを付与することが難しいという課題がある．そのため，実データのみを用いて埋め込みモデルを比較しようとすると，「そもそも教師ラベルの粒度が粗すぎて差が出ない」「レポートの質がばらつきすぎて評価が安定しない」といった問題に直面する．

本研究では，これらの問題に対して次のアプローチをとる．まず，大テーマ（例：“生成 AI と社会”）を共通としつつ，複数の異なる執筆戦略（サブテーマや論点構成のパターン）を持つレポートを大規模に生成する人工レポートデータセットを構築する．このとき，文体や語彙をある程度ランダム化しつつ，戦略ごとの一貫性（クラスタ構造）が明確になるように設計する．次に，この人工レポートデータに対して 5 種類の埋め込みモデルを適用し，ペアワイズ類似度に基づく分類性能と，t-SNE [8]，UMAP [9] による可視化結果を用いて，各モデルの振る舞いを比較する．

2. 問題設定

本研究では，以下のようなレポート集合を対象とする．

- すべてのレポートは同一の大テーマ T（例：生成 AI と社会的影響）について書かれている．
- それぞれのレポートは，K 種類の執筆戦略のいずれかに基づき，主にどの観点から議論を展開するかが異なる（例：教育，産業，倫理，技術，利用者体験など）．
- 各レポートは数段落から構成され，導入・本論・結論の構造を持つ．
- 教員にとっては，「同じ戦略に属するレポート同士は類似度が高い」「異なる戦略のレポート同士は類似度が低い」ことが望ましい．

このとき，テキスト埋め込みモデル $f(\cdot)$ を用いて得られるベクトル表現 $\mathbf{x}_i =$

$f(d_i)$ に対し，コサイン類似度

$$\mathrm{sim}_{\cos}(d_i, d_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \, \|\mathbf{x}_j\|}$$

が，執筆戦略ラベル y_i, y_j と整合的になることが望ましい．

しかし，実際の授業では，y_i を厳密に定義してラベル付けすることが難しい．そこで本研究では，レポートそのものを人工的に生成し，K 個の戦略ラベルを制御可能な形で与えることで，埋め込みモデルの性能を定量的に比較可能な環境を構築する．

3. 提案する人工レポート生成モデル

本章では，生成 AI を利用して人工レポートを作成する手法について述べる．本研究では，LLM の GPT-4 系モデル[10]を用いて多様な観点・構造を持つレポートを自動生成した．GPT 系モデルは Few-shot / Instruction-following に基づく高品質な長文生成が可能であり[11]，入力テンプレートやプロンプト設計に応じて文章構造・論点の多様化を制御できる点から，本研究の人工レポートデータ生成に適している．

提案手法は大きく次の 3 要素から構成される．

(1) **グローバルなテンプレート**：導入・結論の共通文を固定し，その間に生成される本文部分だけを変化させることで，レポート構造を一定に保ちながら内容の多様性を確保する．

(2) **執筆戦略の設計**：GPT-4 に対して複数の「執筆方針」（例：社会・文化的観点，技術的観点，制度設計的観点など）を指定し，レポート生成時の論点・内容展開が異なるように制御した．この手法は，LLM によるレポート生成研究や構造化プロンプト設計に関する先行研究[12]を参考にしている．

(3) **多様性付与のための摂動パラメータ**：temperature や randomness パラメータを変動させることで，文体・語彙選択・論点展開に揺らぎ（stochasticity）を付与し，分類性能評価に利用できる多様な人工レポート群を生成した．これにより，埋め込みモデル（特に OpenAI Embeddings[13]）に対して，内容類似度の評価を行うベンチマークデータセットを人工的に構成できる．

以上の 3 要素を組み合わせることで，多様性と一貫性を併せ持つ人工レポートデータセットを効率的に生成し，文書埋め込みモデルの評価実験に利用した．

3.1. 基本テンプレート

すべてのレポートは，共通の導入文と共通の結論文を持つテンプレートに基づ

く．導入では，テーマ T に関する一般的背景を説明し，結論では，AI 技術の可能性と課題を総括する．このように導入と結論を共通化することで，本論部分の違い（すなわち執筆戦略の違い）が埋め込み空間にどのように反映されるかを明確に観察できるようにした．さらに，本研究では導入文・結論文の複数の候補を用意し，後述する摂動パラメータに応じてランダムに選択することで，テンプレートそのものにも軽微な多様性を持たせている．

3.2. 執筆戦略の設計

大テーマ T に対して，$K = 10$ 種類の執筆戦略を定義した．各戦略は，主に焦点を当てる観点や議論の流れが異なるように設計している．一例として以下を挙げる．

- Strategy 1: 教育・学習支援への影響を中心とする．
- Strategy 2: 産業構造や雇用への影響に焦点を当てる．
- Strategy 3: 倫理・法制度上の課題を詳細に検討する．
- Strategy 4: モデル構造・性能評価など技術的観点を重視する．
- Strategy 5: クリエイティブ分野（芸術・表現）への影響を論じる．
- Strategy 6–10: 利用者体験，アクセシビリティ，国際格差，リスクコミュニケーションなど，別の観点をそれぞれ中心に据える．

各戦略に対して，生成 AI に与えるプロンプトでは，「どの観点を中心に議論するか」「どのような具体例を含めるか」「他の観点にどの程度触れるか」を自然言語で指定した．これにより，同じ戦略に属するレポート同士は類似した論点構成を持ちつつ，語彙や表現はランダム性を保つようにしている．

3.3. 多様性付与のためのパラメータ化

人工レポートデータが単調になりすぎると，どの埋め込みモデルでも容易にクラスタ分けできてしまい，モデル間の差が見えなくなる．そこで以下の 3 つのパラメータを導入し，多様性を制御した．

- P_1：**特徴観点数**　各レポートで必ず触れるべき話題の数を表す．例えば $P_1 = 12$ とした場合，教育・産業・倫理・技術など複数の観点を本論のどこかで扱うようにプロンプトで指定する．これにより，語彙レベルでは戦略ごとの差が出にくくなり，埋め込みモデルが論点構造そのものに基づいてクラスタを捉えられるかを評価できる．
- P_2：**文体レベルの混合比**　高品質・中

品質・低品質の 3 種類の文体をどの比率で混在させるかを表す．例えば $P_2 = 0.5 : 0.3 : 0.2$ とした場合，50% を論理構成の整った高品質文，30% を冗長だが意味は通る中品質文，20% を語彙や論理がやや破綻した低品質文として生成する．低品質文は，あえて接続詞の誤用や話題の飛躍を含めることで，実際の学生レポートに近いノイズを導入することを意図している．

P_3：**各戦略あたりのレポート数**　各執筆戦略から生成するレポート数であり，本研究では $P_3 = 20$ として 10 戦略から合計 200 本のレポートを生成した．

これらのパラメータの値は，予備実験により，TF–IDF や fastText ではクラスタが十分に分離せず，一方で SBERT や E5，OpenAI Embeddings では戦略ごとのクラスタが明確に現れる設定として選択した．

4. 埋め込みモデルと評価設計

本章では，比較対象とする埋め込みモデルと，類似度評価および可視化方法について述べる．

4.1. 比較対象とした埋め込みモデル

本研究では以下の 5 手法を比較対象とした．

TF–IDF　各レポートを形態素解析後に単語単位で分割し，scikit-learn の `TfidfVectorizer` を用いて TF–IDF ベクトルを作成した．これは BoW モデルとして広く用いられるベースラインである．

fastText　Bojanowski らによる fastText[4] の事前学習済み日本語モデルを用い，各単語の分散表現を取得した．文章ベクトルは，出現単語ベクトルの単純平均としている．fastText は文字 n グラムに基づく subword 情報を利用するため，未知語に対しても分散表現を推定できる利点を持つ．

SBERT　Sentence-BERT[6] は，BERT をシャムネットワーク構造で fine-tune した文埋め込みモデルである．本研究では多言語版の事前学習モデルを利用し，レポート全体を 1 ベクトルに写像した．

E5　Wang らによる E5[7] は，大規模なテキストペアデータセットを用いた弱教師あり対照学習により学習された汎用テキスト埋め込みモデルである．検索・クラスタリング・分類など多様なタスクで高い

性能を示している．

OpenAI Embeddings

OpenAI が提供する `text-embedding-3-large` モデルを用い，各レポートの埋め込みを取得した[13]．本モデルは多言語かつ長文への対応を特徴とし，検索・類似度計算・推薦などへの応用を想定して設計されている．

これらの手法は近年のテキスト埋め込み研究において広く利用されており，LLM 時代の embedding の性能比較や設計指針については Morris らによる包括的なサーベイが存在する[14]．

いずれの手法においても，レポート全文から 1 本の固定長ベクトルを得ることを前提とし，コサイン類似度によりレポート間の類似度を計算した．

4.2. 類似度評価指標

文書埋め込みモデルの性能を評価するため，本研究では ROC-AUC（Receiver Operating Characteristic - Area Under Curve），Average Precision（AP：平均適合率），PR 曲線，Mean Reciprocal Rank (MRR)，および Recall@k を用いる．これらの評価指標は，情報検索の分野で標準的に利用されるものであり[1]，ランキング性能や検索精度を多面的に評価できる．

4.2.1. 混同行列と基本指標（TP, FP, TN, FN）

文章類似度判別を含む 2 クラス分類問題では，分類器の出力を混同行列（confusion matrix）に整理することで分類性能を体系的に評価できる．混同行列は，実際のクラスと分類器の予測結果の対応を表す 2×2 の表であり，表 1 に示す 4 つの要素から構成される．

表 1: 混同行列

		予測	
		Positive	Negative
実際	Positive	TP	FN
	Negative	FP	TN

ここで，TP などの各要素は以下の値を表す．

- ***TP*** ：実際に Positive のものを Positive と正しく予測した数
- ***FP*** ：実際は Negative であるにもかかわらず，誤って Positive と予測した数（偽陽性）
- ***TN*** ：実際が Negative のものを Negative と正しく予測した数
- ***FN*** ：実際は Positive であるにもかかわらず，誤って Negative と予測した数（偽陰性）

これら 4 つの値を用いることで，分類器の性質を多面的に評価できる．

4.2.1.1. 具体例

たとえば本研究のように，文章ペアが「内容的に類似している」かどうかを判定する分類器を考える．実際の正例（Similar）が 20 件，負例（Dissimilar）が 80 件存在し，分類器が表 2 のように予測したとする：

表 2: 分類器の予測結果

		予測	
		Positive	Negative
実際	Positive	18	2
	Negative	10	70

このとき，

$$TP = 18,\ FN = 2,\ FP = 10,\ TN = 70$$

となる．

4.2.2. Recall（再現率）

Recall（再現率）は，実際に Positive である事例をどれだけ見逃さずに検出できたかを測る指標であり，次式で定義される．

$$\text{Recall} = \frac{TP}{TP + FN} \tag{1}$$

上記の例では，

$$\text{Recall} = \frac{18}{18 + 2} = 0.90$$

となり，実際に類似している文章ペアの 90% を正しく検出できていることを意味する．文章類似度分析のように，見逃し（False Negative）を避けたい場合に特に重要となる指標である．

4.2.3. Precision（適合率）

Precision（適合率）は，分類器が Positive と判断した事例のうち，どれだけが正しいかを測る指標であり，以下で与えられる．

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

例では，

$$\text{Precision} = \frac{18}{18 + 10} \approx 0.643$$

となり，分類器が「類似している」と判断した文章ペアの約 64% が，実際に類似していたことを示す．これは誤検出（False Positive）を抑える能力を表す指標であり，類似判定を厳密に行いたい場合に重要となる．

4.2.4. Precision と Recall の関係

一般に, Recall を高めると FP が増加し Precision が低下する傾向がある．一方，Precision を高めようとすると Positive 判定が慎重になり FN が増加しやすく，Recall が低下する．したがって両者はト

レードオフの関係にあり，タスクの目的に応じてどちらを重視するかを選択する必要がある．

本研究では，文章レポートの内容的類似性を評価するため，Positive（類似）・Negative（非類似）判定を用いた指標（Recall, Precision）に加え，ROC-AUC，Average Precision（AP），MRR，Recall@k といったランキング指標も併用し，文書埋め込みモデルの多面的な性能比較を行う．

4.2.5. ROC-AUC

ROC 曲線は，判定閾値を変化させたときに得られる縦軸の True Positive Rate（TPR）と横軸の False Positive Rate（FPR）の関係を可視化したものである．TPR と FPR は以下の式で定義される．

$$\mathrm{TPR} = \frac{TP}{TP + FN}, \quad \mathrm{FPR} = \frac{FP}{FP + TN}$$

ここで，TPR は正例クラスに対する再現率（Recall，式 (1)）と等価である．

そして，ROC-AUC は，ROC 曲線の下の面積である．正例ペアのスコアが負例ペアより高いほど ROC-AUC は 1 に近づき，完全識別であれば ROC-AUC=1.0，ランダム識別器の場合は 0.5 となる．したがって，ROC-AUC は閾値に依存しない分類性能指標として広く用いられる．

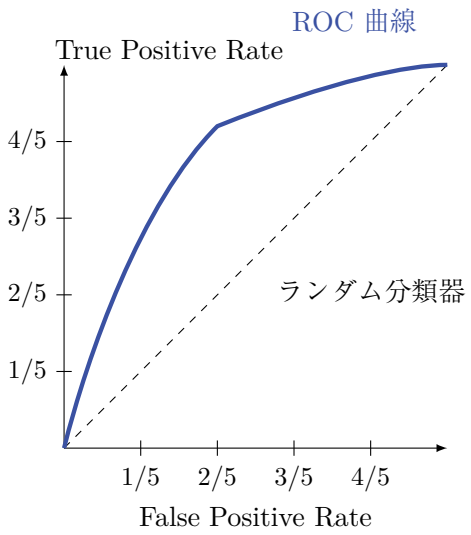


図 1: ROC 曲線の例（正例スコア＞負例スコアの場合）

4.2.5.1. ROC-AUC の例

正例スコアが

$$[0.90, 0.85, 0.80],$$

負例スコアが

$$[0.50, 0.40, 0.30]$$

であれば，どの閾値を選んでも 正例 > 負例 の関係が崩れないため，ROC-AUC = 1.0 になる．

逆に，両者の分布が重なっていれば ROC-AUC は 0.5 前後となり，区別能力の低さを示す．

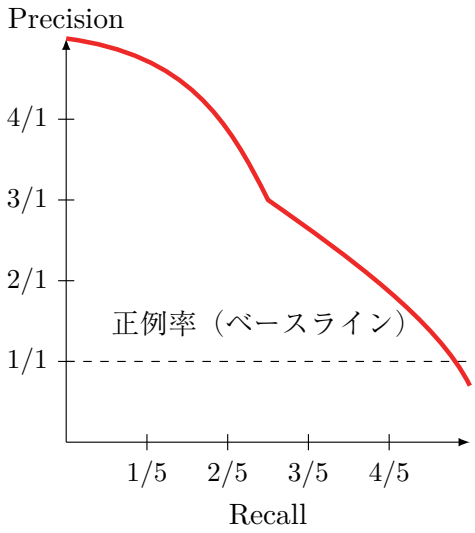


図 2: PR 曲線の例（上位に正例が集中するケース）

4.2.6. Average Precision (AP)

AP は Precision-Recall 曲線（PR 曲線）の下の面積（AUPRC）に等しく，類似度ランキングの上位に正例を多く含められるほど値が大きくなる．

AP は以下で定義される：

$$\mathrm{AP} = \sum_{k=1}^{n} (R_k - R_{k-1}) P_k,$$

ここで，P_k は Precision，R_k は Recall である．

4.2.6.1. AP の例

次のように類似度スコアで並べたとする：

順位	スコア	正例/負例
1	0.95	正例
2	0.90	正例
3	0.60	負例
4	0.55	正例
5	0.20	負例

正例は 3 件のため，Recall は

$$1/3,\ 2/3,\ 1$$

と増加する．Precision は

$$1,\ 1,\ 3/4$$

となり，これらから AP $\approx$ 0.92 が得られる．正例が上位に集中するほど AP が高くなる．

4.2.7. Mean Reciprocal Rank (MRR)

MRR は「最初に正例が出現する順位」に基づく評価指標である．

類似度ランキングを

$$d_{i,(1)}, d_{i,(2)}, d_{i,(3)}, \ldots$$

とするとき，最初の正例の順位を

$$r_i = \min\{k \mid y_{d_{i,(k)}} = y_i\}$$

とし，逆数

$$\mathrm{RR}_i = \frac{1}{r_i}$$

を計算する．これを全クエリで平均したものが MRR である．

4.2.7.1. MRR の例

順位	ラベル
1	3
2	2
3	7
4	5（正例）
5	1

クエリのラベルが 5 なら最初の正例は 4 位であるため，

$$\mathrm{RR} = \frac{1}{4} = 0.25.$$

1 位に正例がくれば RR=1 であり，MRR は「どれだけ上位に正例を置けたか」を測る．

4.2.8. Recall@k の定義

類似度にもとづくランキング評価では，あるクエリ文書 i に対して類似候補集合を類似度の高い順に並べ，上位 k 件に同一クラス（あるいは正例）文書が含まれるかどうかを測定する Recall@k が広く用いられる [1)]．

いま，文書集合 $\mathcal{D} = \{1, \ldots, N\}$ のうち，文書 i と同じクラスに属する正例集合を $\mathcal{P}(i)$ とする．ここで，$i \notin \mathcal{P}(i)$ である．また，文書 i に対して計算した類似度スコアを降順に並べたランキング列を

$$r_i(1), r_i(2), \ldots, r_i(N)$$

と書く．ただし $r_i(1)$ は文書 i に最も類似した文書を表す．このとき Recall@$k(i)$ は

$$\mathrm{Recall@}k(i) = \begin{cases} 1 & \text{if } \exists j \le k \text{ s.t. } r_i(j) \in \mathcal{P}(i), \\ 0 & \text{otherwise} \end{cases}$$

で与えられる．

すなわち，文書 i のランキング上位 k 件の中に「同じクラスの文書が 1 件でも存在するか」を判定する指標である．全データ集合での Recall@k は，すべての文書について平均をとった

$$\mathrm{Recall@}k = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Recall@}k(i)$$

で定義される．

直感的には，ユーザが求める「関連文書（正例）」をランキングの上位 k 件以内で取り出せている割合を意味し，情報検索やレコメンダシステム，文書類似度評価における基本的な性能指標である．

4.2.8.1. Recall@k の例

先ほどの例では，

$$\text{Recall@1} = 0, \quad \text{Recall@5} = 1.$$

上位 1 件では正例がないが，上位 5 件には正例が存在するためである．

Recall@1 $\leq$ Recall@5 $\leq$ Recall@10 となり，k を大きくするほど Recall は増加する．

4.2.9. 指標の位置づけ

以上 4 つの指標には次の特徴がある：

- ROC-AUC：スコア分布の分離度を評価（閾値に依存しない）
- AP：ランキング上位の正例密度を強く評価
- MRR：最初の正例の位置（序盤の正解率）を評価
- Recall@k：特定の上位幅での正解率を評価

本研究では，「同一戦略レポートの検索性能」と「類似度ランキングの品質」を多角的に評価するため，これらすべてを採用した．

4.3. 次元削減と可視化

埋め込み空間におけるクラスタ構造を直感的に理解するため，各モデルのベクトルに対して次元削減を行い，2 次元散布図として可視化した．可視化には非線形手法である t-SNE[8] と UMAP[9] を用いた．

t-SNE は局所的な隣接関係の保持に優れる一方で，大域的な構造が歪むことが知られている[8]．UMAP は多様体学習に基づく次元削減手法であり，t-SNE と同程度の可視化品質を保ちつつ，大域構造の保持と計算効率に優れると報告されている[9]．本研究では，両者の長所・短所を踏まえつつ，クラスタの分離度合いと形状を比較した．

5. 実験結果

本章では，$K = 10$ 戦略，$P_3 = 20$ により生成した合計 200 本の人工レポートを対象とした実験結果を示す．

5.1. 数値評価

表 3 に，各埋め込みモデルの ROC-AUC，Average Precision，MRR，Recall@1/5/10 の結果を示す．ここで TF–IDF および fastText は BoW/単語平均型のベースラインであり，SBERT，E5，OpenAI Embeddings は文レベル埋め込みモデルである．

表 3 から，TF–IDF は ROC-AUC が約 0.59，fastText でも約 0.71 に留まり，ランキング指標も 3 手法に比べて大きく劣ることが分かる．特に TF–IDF の

表 3: 人工レポートデータに対する各埋め込みモデルの類似度評価結果

Method	ROC-AUC	AP	MRR	R@1	R@5	R@10
TF–IDF	0.5938	0.1469	0.5189	0.3833	0.6792	0.8167
fastText	0.7112	0.2077	0.7204	0.6083	0.8750	0.9500
SBERT	0.9998	0.9977	1.0000	1.0000	1.0000	1.0000
E5	0.9947	0.9593	1.0000	1.0000	1.0000	1.0000
OpenAI Emb.	0.9998	0.9982	1.0000	1.0000	1.0000	1.0000

Recall@1 は 0.38 であり，クエリと同一戦略のレポートを最上位に推薦できる割合は 4 割未満である．

一方，SBERT，E5，OpenAI Embeddings の 3 手法は ROC-AUC, AP, MRR, Recall@1/5/10 のいずれにおいても 1.0 近い値を示し，本実験条件においてはほぼ完全に執筆戦略を識別できている．特に SBERT と OpenAI Embeddings は，すべてのランキング指標で 1.0 を記録し，任意のレポートをクエリとしたときに必ず同一戦略のレポートを最上位に位置付けている．

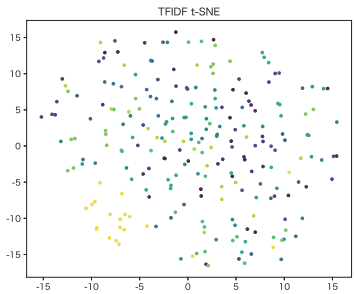


図 3: TF–IDF の t-SNE

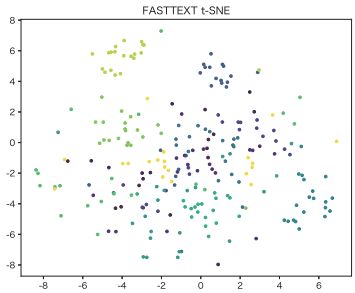


図 4: fastText の t-SNE

5.2. 可視化結果

図 3 〜 12 に，TF–IDF，fastText, E5, SBERT, OpenAI Embeddings の t-SNE および UMAP による 2 次元可視化結果の例をそれぞれ示す．なお，プロットの色は執筆戦略ラベルを表す．

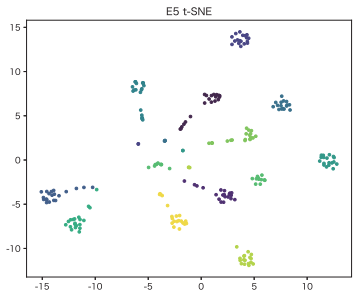


図 5: E5 の t-SNE

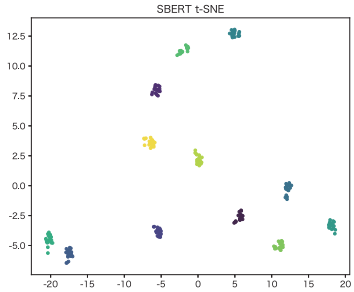


図 6: SBERT の t-SNE

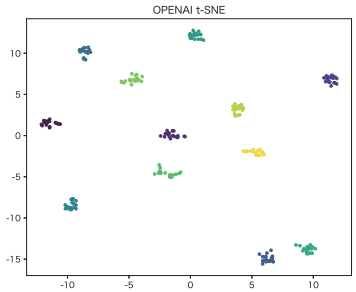


図 7: OpenAI Embeddings の t-SNE

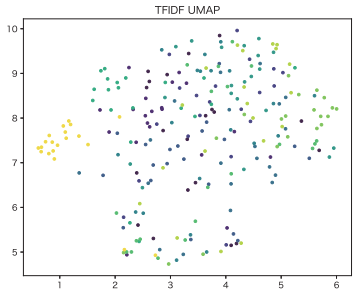


図 8: TF–IDF の UMAP

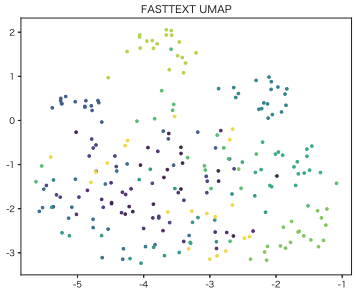


図 9: fastText の UMAP

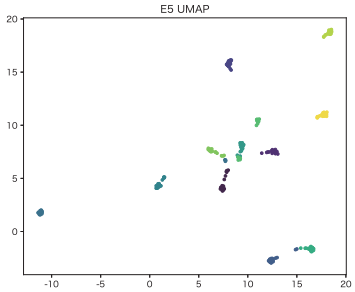


図 10: E5 の UMAP

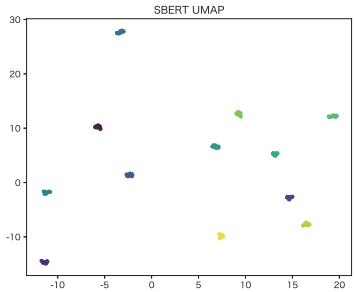


図 11: SBERT の UMAP

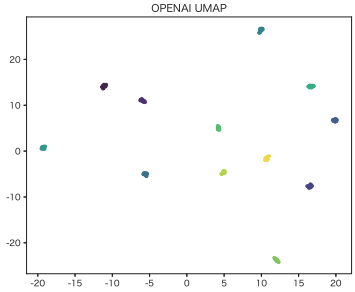


図 12: OpenAI Embeddings の UMAP

SBERT，E5，OpenAI Embeddings のいずれにおいても，t-SNE・UMAP ともに 10 個のクラスタがほぼ完全に分離している．同一戦略に属する点は高密度の塊を形成し，異なる戦略のクラスタとの重なりはほとんど見られない．

これに対して，fastText と TF–IDF では挙動が大きく異なる．fastText の t-

SNE/UMAP では，一部の戦略に対応する薄いクラスタが見られるものの，多くの点が混在しており，明確な 10 クラスタ構造は確認できない．

TF–IDF の可視化では，特定の戦略に対応する小さなクラスタが現れる場合もあるが，全体としては戦略ラベルがほぼランダムに散在している．これは，本研究の人工レポートデータにおいて全戦略で共通の語彙群を意図的に混在させている設計と一致している．

6. 考察

本章では，上記の結果に基づき，BoW/単語平均型埋め込みと文レベル埋め込みモデルの違いについて考察する．

6.1. BoW/単語平均型埋め込みの限界

TF–IDF は単語の出現頻度に基づくベクトル表現であり，単語の順序や長距離の依存関係を考慮しない．本研究の人工レポートデータでは，すべてのレポートが共通導入・共通結論を持ち，さらに本論でも共通語彙が意図的に混在するように設計されている．このため，戦略の違いは主に「どの観点をどの順番でどの程度の深さで論じるか」という談話構造の差として現れる．しかし TF–IDF では，このような構造的な違いが十分に表現されず，単語分布の些細な差に基づいて類似度が決まってしまう．

fastText は subword 情報を利用して単語分散表現を獲得する点で TF–IDF よりも強力であるが，文章レベルの表現としては単語ベクトルの平均に過ぎない．長文レポートでは多様な話題が混在するため，平均操作によって各話題に対応するベクトルが相殺され，結果として文書間の差が縮小してしまう．本研究の結果でも，fastText は TF–IDF よりは高い性能を示したものの，説明したいクラスタ構造を完全に再現するには至らなかった．

6.2. 文レベル埋め込みモデルの優位性

SBERT は，類似文判別タスクに基づき文ペアを近傍に写像するように学習されたモデルである [6]．E5 は大規模な検索ログ等から構成された CCPairs コーパスを用い，クエリと関連文書の埋め込みが近くなるように弱教師あり対照学習を行っている [7]．これらのモデルは，単語レベルではなく文レベルの意味関係を直接的に最適化しているため，「教育的観点からの批判的検討」「技術的詳細の提示」など，執筆戦略を特徴づける高次の意味構造を捉えることができる．

OpenAI Embeddings は，より大規模なデータとモデル容量に基づき学習されており，多言語・長文に対しても安定した意味表現を提供するよう設計されている．本研究の実験では，SBERT および

E5 とほぼ同等，あるいはそれ以上の性能を示し，人工レポートの執筆戦略を完全に復元することが確認された．

これらの結果は，レポート自動分類や類似レポート推薦といった教育データ分析において，BoW/単語平均型の従来ベースラインに比べて，文レベル埋め込みモデルを採用することの実用上の利点を示している．特に，本研究のように語彙レベルでクラス差が小さい設定では，両者の差が顕著に現れる．

6.3. 人工レポートデータを用いた評価の意義と限界

本研究の特徴は，実際の学生レポートではなく，生成 AI による人工レポートを用いて埋め込みモデルを比較した点にある．人工レポートデータを用いることで，下記のような利点が得られる．

(1) 戦略ラベルを厳密に制御できる，
(2) データ数やノイズレベルをパラメータとして調整できる，
(3) 再現性の高い実験が可能になる，

一方で，人工レポートはあくまで生成モデルが作り出した文章であり，実際の学生レポートに見られる誤字脱字や独特の表現，知識の偏りなどを完全には再現していないという限界がある．また，本研究で用いた生成プロンプトやモデルは，埋め込みモデルとは独立であるものの，同じ大規模言語モデルの系統に属しているため，ある程度のバイアスが共有されている可能性も否定できない．

したがって，人工レポートデータに基づく評価結果をそのまま実践現場に適用することは慎重であるべきだが，埋め込みモデルの挙動を比較するためのテストベッドとしては有用であると考えられる．今後は，人工レポートと実際の学生レポートを組み合わせたハイブリッドな評価環境の構築が課題である．

7. まとめ

本研究では，大テーマが共通で執筆方針が異なるレポート集合を想定し，生成 AI を用いて人工レポートデータを構築した上で，5 種類の埋め込みモデル (TF–IDF, fastText, SBERT, E5, OpenAI Embeddings) の類似度計算および可視化の挙動を比較した．

実験の結果，BoW/単語平均型の TF–IDF と fastText は，執筆戦略に対応するクラスタ構造を十分に再現できず，ROC-AUC やランキング指標でも性能が限定的であることが示された．一方で，SBERT，E5，OpenAI Embeddings の 3 手法は，人工レポートの戦略ラベルをほぼ完全に区別し，t-SNE・UMAP による 2 次元可視化でも明確な 10 クラスタ構造を形成した．

これらの結果から，大学レポートの自動分類・類似レポート検索といった教育応用においては，BoW/単語平均型の従来手法ではなく，埋め込みベクトルモデルを用いることが有効であると結論づけられる．今後の課題として，実際の学生レポートデータへの適用と，教員による評価結果との比較，および人工レポートデータ生成プロセスそのものの検証と改良が挙げられる．

謝辞

本稿を執筆するにあたり本学国際コミュニケーション学部教授 梅垣敦紀先生に作成していただいた upLaTeX のクラスファイル[15]を利用した．ここに感謝の意を表す．

参考文献

1） Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to Information Retrieval. https://nlp.stanford.edu/IR-book/, 2008.

2） Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. https://arxiv.org/abs/1301.3781, 2013.

3） Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, 2013.

4） Bojanowski Piotr, Grave Edouard, Joulin Armand, and Mikolov Tomas. Enriching Word Vectors with Subword Information. https://cir.nii.ac.jp/crid/1363388843924591872, 12 2017.

5） Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. https://arxiv.org/abs/1706.03762, 2017.

6） Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. https://arxiv.org/abs/1908.10084, 2019.

7） Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text Embeddings by Weakly-Supervised Contrastive Pre-training. https://arxiv.org/abs/2212.03533, 2024.

8） Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. http://jmlr.org/papers/v9/vandermaaten08a.html, 2008.

9） Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. https://arxiv.org/abs/1802.03426, 2020.

10） OpenAI. GPT-4 Technical Report. https://arxiv.org/abs/2303.08774, 2023.

11） Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, et al. Language Models are Few-Shot Learners. https://arxiv.org/abs/2005.14165, 2020.

12） Mercy Ranjit, Gopinath Ganapathy, Ranjit Manuel, and Tanuja Ganu. Retrieval Augmented Report Generation using OpenAI GPT models. https://arxiv.org/abs/2305.03660, 2023.

13） OpenAI. OpenAI Text Embeddings. https://platform.openai.com/docs/guides/embeddings, 2024.

14） John X. Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander M. Rush. When Text Embedding Meets Large Language Model. https://arxiv.org/abs/2412.09165, 2025.

15） 梅垣敦紀. 愛知大学情報メディアセンター紀要「COM」への upLaTeX での投稿. https://saturn.aichi-u.ac.jp/img/center/pdf/com49-2.pdf, 2025.

論文

GUI の自動操作とスクリーンショットに基づく GUI 採点支援機能を有した Python プログラミング実習用システムの開発

毛利　元昭（愛知大学経営学部）

要旨

プログラミング実習系授業の実施において，担当教員にかかる負担の軽減や学習者へのフィードバックの早期化は重要である．既存のプログラミング実習システムの多くは CLI アプリケーション用の自動採点機能を有しているが，GUI アプリケーション用は実装例が少なく，利用可能な状況も限られている．そこで筆者は，GUI 自動操作とスクリーンショットに基づく GUI 採点支援機能を有した Python プログラミング実習用システムを開発した．最終的な採点は人が行う必要があるものの，手元の環境での実行を減らせるため，採点時間が短縮され，負担軽減やフィードバックの早期化を図ることができた．

キーワード： プログラミング学習，採点支援，Python，GUI，PHP，シェルスクリプト

1. はじめに

プログラミングの実習系授業の実施において，担当教員にかかる負担は大きい．資料・教材作成と講義だけでなく，課題の細やかな出題，学習者の課題進捗状況の把握，採点とフィードバック，質問やトラブルへの対応などが求められる．また，Try and Error を繰り返すプログラミング学習では，合否のフィードバックはできるだけ早期であることが望まれる．情報工学や情報科学を専攻していない初心者 50〜60 人規模の複数クラスを単独で担当するような場合，授業を正常に実施するには何らかの効率化・自動化が必要と言える．

このような背景から，これまでに多くのプログラミング実習用システムが開発され，広く利用されている．日本人の利用が多いものとしては，Aizu Online Judge[注 1)]，Progate[注 2)]，paiza[注 3)] などがあり，世界的には AtCoder[注 4)] が有名である．しかしこれらのほとんどは，ソースコードあるいはコマンドラインインターフェース（CLI）上でのテキスト入出力に基づいた自動採点機能を有するにとどまっている．その理由は下記のように考えられる．

1. 任意のプログラミング言語で記述された GUI を，ウェブブラウザ上に構成できない

2. 初心者である学習者に，出題者の意図通りのテスト（GUI の操作）を，正確に行わせることが困難
3. 結果の判定・評価が困難

箇条書きの 1 について詳細を述べる．まず，プログラミング学習をウェブサービスとして提供する場合，学習者側の環境はウェブブラウザが前提となることが多い．学習者がウェブブラウザを介してコードを提出すると，その実行結果や評価結果がウェブブラウザ上に表示される形である．しかし，ウェブブラウザ上で GUI を構成可能なプログラミング言語は，現在では実質的に JavaScript に限定される．その JavaScript も，GUI に対しては実際には DOM[注 5)] を介して HTML 文書内の要素にアクセスする形である．つまり，任意のプログラミング言語で記述された GUI をブラウザ上に構成するには，GUI の各パーツを HTML 要素に変換することが必要となり，それらを扱うにはプログラム全体を JavaScript に変換することが必要となる．これは，技術的には可能かもしれないが，そのようなツールの開発コストやテストにかかるリソースについて勘案すると，授業・サービスを提供していく上で現実的でない．実際に，筆者が担当する授業で教えている Python に限定しても，標準の GUI ライブラリである Tkinter を HTML および JavaScript に変換するツールは，2025 年現在で見つけられなかった．

ここで，ウェブブラウザ上に GUI そのものを構成せず，画像として表示，あるいはレイアウト情報として表示することは，さほど難しくない．実際に，筆者が以前に開発したプログラミング実習用システム[1)]でも，GUI の起動時のウィンドウをスクリーンショット（SS）し，実行結果として表示させていた．また，立石らが開発したシステム[2)]では，Java ソースファイルから GUI のレイアウト情報を XML 形式で出力させていた．しかし，GUI が構成されているわけではないため，これらの方法ではウェブブラウザ上での GUI のテストを行うことができない．

箇条書きの 2 について詳細を述べる．GUI を備えたアプリケーション（以下，アプリ）の実行・テストには，GUI の操作や GUI 上への表示が伴う．GUI が想定するマウスやキーボードを介した操作を，ウェブサーバ側からウェブブラウザ上で自動で行わせることは，セキュリティ上の理由からできない．学習者による実際の操作，あるいはローカル環境での何らかのユーザ操作自動化ツールが必要となる．

ここで，学習者自身の手でテストを行わせることは，プログラミングへの理解を深めることには繋がるものの，課題ごとに異なるテストを正確に行わせる必要があり，初心者を対象とする授業・サー

ビスを提供していく上で現実的でない．大学での成績評価には公平性と公正性が求められることからも，採点結果を自己申告させる形は避けるべきであろう．

ユーザ操作自動化ツールを用いる場合も，その環境の構築・設定を学習者に求めることになり，あるいは学習者の多様な環境に応じて対処する必要があり，やはり初心者に対しては現実的でない．例えば，Selenium[注 6)]と呼ばれる操作自動化ツールは，その操作内容を Python で記述することができるが，操作対象はウェブブラウザに限定されている．Ranorex[注 7)]と呼ばれるテスト自動化ツールは，操作対象を問わないが，利用環境が Windows に限定されている上に，有償である．xdotool[注 8)]と呼ばれる操作自動化ツールは，操作対象を問わないが，利用環境が Linux または macOS に限定されている上に，CLI 上で実行する必要がある．PyAutoGUI[注 9)]と呼ばれるライブラリは，Python プログラム内に GUI 操作を記述できるが，「指定タイトルのウィンドウを見つける」などのウィンドウ操作が環境依存であり，環境に応じて他のライブラリと組み合わせる必要がある．

箇条書きの 3 について詳細を述べる．何らかの方法で，ユーザ側あるいはサーバ側で GUI の操作を自動的に行えるようになったとする．しかし，その結果となる GUI 上への表示をどのように判定・評価すべきかについては，未だ決定的な方法が確立されていない．

前述の立石らの GUI アプリ自動採点システム[2)]は，解析した GUI のレイアウト情報を利用してサーバ側で GUI の自動操作を行っており，複数のテスト結果に基づいて評価を行っていた．また，テストケースを GUI の操作ベースで生成するツールも付属していた．しかし，対象となる言語は Java に，GUI ライブラリは JavaFX に限定されており，Python を教えている筆者の授業では利用できない．水谷は，大規模言語モデルの一種であり，画像の言語化を得意とする Vision Language Model (VLM) を用いた GUI アプリ自動採点システム[3)]を提案した．しかし，計算リソースへの要求が高い VLM をオンプレミスで構築する必要があり，計算量・計算時間が膨大になる見込みで，かつ，システムとしても検証段階で成熟していない．

これらの理由から，GUI アプリの自動採点は現状では困難であり，GUI アプリの最終的な採点は未だ教員が行う必要があると判断した．同時に，筆者はプログラミング実習用システム[注 10)]に GUI アプリの採点支援としての機能実装を行うこととした．本稿では，開発したシステムの設計・構成を説明するとともに，採点時間を比較することで本システムの有用性を示す．

2. 採点者にかかる負担とその軽減

プログラミング実習用システムに自動採点が期待される理由は，採点する教員にかかる負担の軽減にある．それでは，採点者はどのような状況に負担を感じるのか．筆者は次のように考えた．

1. 採点を必要とする対象が多い
2. 採点に必要なステップ数が多い
3. 正解の状態との比較がしにくい
4. コメントでのフィードバック

それぞれへの対策について，方針を述べる．

2.1. 採点を必要とする対象が多い

そもそもの学習者の数や課題の数を減らすことは考えるべきではないため，手動採点の必要性を減らすことを考える．学習者は，課題が合格の要件を満たしているという自信が無い場合でも提出することがある．特に初心者はテストを自主的に十分行おうとせず，実質的にテストを採点者に委ねるために提出することが多い．逆に言えば，提出したプログラムが合格の要件を満たしていないことが分かれば，学習者は修正作業に取り掛かることができる．ここで，明らかに要件を満たしていない場合にも採点の実施を多数要求されることは，採点者にとって大きな負担となる．よって少なくとも，自動判定可能な要件については自動判定を実施し，満たしていない場合は「不合格」とする形（足切り）が良いと言える．

プログラムが合格の要件を満たしているかどうかを判定するには，プログラムを解析する必要がある．本システムの前身[1)]では Java をビルドした JAR 形式で提出させており，その JAR は中間言語へのコンパイルと最適化が行われたものであった．そのため，JAR を javap [注 11)] コマンドで解析した結果は，穴埋め形式のコーディングであれば学習者ごとの揺らぎが少なく，自動採点に活用できた．一方，本システムで前提とする Python は，表面上，プログラムをソースコードの形で保持しつつ実行する．そのため，ソースコードそのものを提出させる形が最も自然であるが，そのままでは学習者ごとの揺らぎが大きい．ソースコードを自動解析するにあたって，前処理として表記のある程度の画一化処理が必要である．

ここで，Python には isort [注 12)]，autoflake [注 13)]，black [注 14)] などの自動整形ツールが開発されている．自動整形ツールは，チーム開発などの際にソースコードの表記の画一化を目的として導入されるものであるため，今回の目的にも合致する．よって，自動整形ツールを適用した後にソースコードを解析し，合格の要件を満たしていない場合は即座に「不合格」とすればよい．

2.2. 採点に必要なステップ数が多い

前述の足切りによって手動採点の対象が減った場合でも，採点 1 件当たりに必要なステップ数が多いと，採点者にとって大きな負担となる．ウェブブラウザ上で実行できない GUI アプリの手動採点にかかる作業は，一般的に次の通りであろう．

1. 採点を必要とする対象を探す
2. 答案をダウンロードする
3. プログラムを実行・操作する
4. 正解の状態と比較する
5. 評価やコメントを登録する

箇条書きの 4 と 5 については，完全な自動採点を目指さない以上，手動での作業を無くすことはできない．これらの負担の軽減については，第 2.3節および第 2.4節で述べる．ここでは，箇条書きの 1〜3 について，以降に詳細を述べる．

2.2.1. 採点を必要とする対象を探す

手動採点の画面の作り方にもよるが，採点を必要とする対象を探すのに手間取ることは，採点者にとって負担である．筆者の場合，GUI アプリ作成の科目には毎年 30 人程度の履修者がおり，授業中のガイドに従ってほぼ一斉に課題が提出される場合もあれば，任意のタイミングの場合もある．いずれにせよ，そのままでは採点を必要とするものと不要とするものが入り乱れている状況となる．ここで，「探す」をステップ数に含めなくてよい状態とは，例えば任意の課題の答案一覧をページ表示した際，採点を必要とする答案が分かりやすく表示されている状態である．つまり，何らかの強調機能やフィルタ機能が備わっていることが望ましいと言える．

2.2.2. 答案をダウンロードする

ウェブブラウザ上で実行できないため，手動での動作確認をするには採点者のローカル環境で実行する必要がある．最近のウェブブラウザは，セキュリティの関係から，プログラムのダウンロードおよび実行の際には拒否を前提とする確認ダイアログが表示される．セキュリティの設定を変更することで確認不要で許可とすることもできるが，学内備品の PC 上での設定変更は問題となりやすい．加えて，アプリの実行に必要なファイルがプログラム本体の他にも存在する場合，それらも正しく配置しておくことが必要であり，実質的なステップ数が増える．これらをステップ数に含めないようにするには，プログラムをダウンロードをすることなく，ウェブブラウザ上で目視で採点できることが望ましいと言える．

2.2.3. プログラムを実行・操作する

GUI アプリの動作確認をするには，採点者が確認事項とそのための操作について正しく理解し実施する必要がある．現状で，筆者の科目では筆者のみが採点者で，同じ課題の答案を連続して採点していくとしても，数をこなすことはやはり負担である．これをステップ数に含めないようにするには，確認事項に対応する操作が自動でなされた様子を確認できることが望ましいと言える．つまり，GUI の操作自動化ツールを導入する必要がある．

2.3. 正解の状態との比較がしにくい

GUI アプリの動作確認をするには，確認事項に対応する操作が自動でなされた様子を確認するとしても，正解の状態と比較することが必要である．筆者は出題者でもあるため，課題ごとに自身で設けた採点基準に基づいて採点を実施しているが，その負担は大きい．採点者にかかる負担を減らすには，正解例を操作した際の様子が，比較しやすい形で提供されていることが望ましいと言える．

ここで，GUI の自動操作ツールを利用したとして，それによって正しく操作できるかは事前の確認が必須である．その様子を動画あるいはそれに準ずるものとして記録しておき，学習者の提出したプログラムが同様に操作された様子と並べて表示させればよい．ただし，動画だとして，その再生タイミングにズレが生じていると，比較が困難となる．再生タイミングを同期させる仕組みを組み込む必要がある．

2.4. コメントでのフィードバック

採点においては，特に「不合格」とした際に，学習者にその理由等の説明や修正のための助言を与えたい場合がある．筆者がこれまで受け持った学習者の多くは，手元で実行してもエラーメッセージを細かく読むことを敬遠していた．直接的な修正方法を述べたものではないことに加え，専門用語を含んだ英文のため，「読んでも分からない」との印象があったためである．その結果，正解例が示されるまで取り組まない者や，質問対応で正解例ばかりを求める者も多々現れた．その状況は授業進捗の遅延と教員の負担を生じさせるため，削減が望まれる．

ここで，前述のように自動判定可能な要件については自動判定を実施する形であれば，その判定理由は明らかであり，修正のための助言も多くは定型的である．つまり，ある程度であればコメントでのフィードバックは自動化可能であり，採点者にかかる負担を軽減できると言える．

3. システム設計

3.1. システム要求

GUI 採点支援機能を有したプログラミング実習用システムの開発にあたり考慮したシステム要求は，次の通りである．

1. 学習者がウェブブラウザを介してアクセスでき，課題の表示や提出，評価の確認も可能
2. 60 人程度がほぼ同時にアクセスしてもシステムダウンが発生しない
3. 利用にはログインを必要とし，テスト用ユーザを除いて学内の認証サービスにて認証
4. プログラム（および実行に必要なファイル群）が提出された場合，プログラムの解析および実行を自動で行い，その結果をファイルとして保持
5. 課題の提出から採点結果等の表示までが 20 秒以内
6. GUI を含め，課題の実行結果を課題用ページにて並べて表示
7. 管理者の画面から，提出されたファイルや答案の確認，採点の修正や手動採点，コメントが可能
8. 管理者の画面から，クラスごと・授業の回ごとに課題の公開・提出期間等の設定が可能

これらの要求が満たせるように，オンプレミスにて環境構築およびシステム設計を行った．現行の環境は表 1の通りである．

表 1: システムの環境

本体	PC/AT 互換機
OS	Lubuntu 24.04 64bit
ウェブサーバ	Apache 2.4
サーバ側処理系	PHP 8.3, Bash 5.2, Python 3.12
GUI 自動操作	xdotool 3
画面・画像処理	scrot 1.10, ImageMagick 6.9, APNG Assembler 2.91
ブラウザ側処理	JavaScript, jQuery 3.7
データベース	MariaDB 15.1

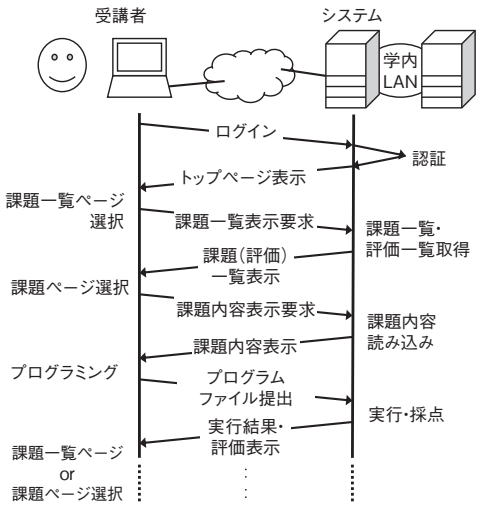


図 1: 学習者によるシステム利用の流れ

学習者によるシステム利用の流れは図 1の通りであり，本システムの前身と同様であるため，詳細な説明を割愛する．

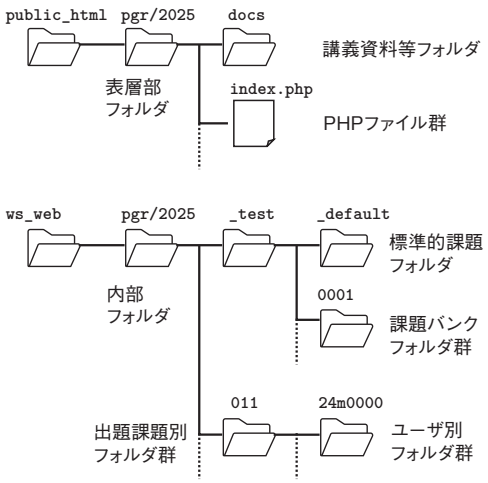


図 2: システムのフォルダ構成

3.2. 課題バンクフォルダ群

フォルダ構成は図 2の通りであり，基本的に本システムの前身と同様であるため，ファイル群に変更点のある課題バンクフォルダ群についてのみ説明する．

「_test」フォルダ内の課題バンクフォルダ群においては，各課題の識別子（数字 4 桁）をフォルダ名として課題ごとにフォルダを用意している．課題ごとのフォルダには，主に下記のファイルが指定の名前で収められている．

- setting.txt
  DB への登録用設定ファイル
- direction.txt
  課題の説明（専用）
- figure.png
  説明の補足用の画像
- code.txt
  ソースコード（提示用）
- answer.txt
  正答例
- test_cmd.sh
  自動採点スクリプト（専用）
- input*.dat（*は任意の数値）
  自動実行時の入力
- output*.dat（*は input に対応）
  自動実行時の想定出力（比較用）
- xdotool.txt
  GUI 自動操作内容（xdotool 準拠）
- ss_win.png
  GUI 実行時の想定ウィンドウ SS（アニメーション）
- output_pc2.txt
  クラスおよび関数・メソッドの情報（比較用）
- その他，配布ファイルなど

本システムの前身からの変更点は，「xdotool.txt」および「ss_win.png」が追加されていることである．これらの利用については，第 5章で詳細を述べる．

また，課題ごとのフォルダとは別に，標準的な課題用の「_default」フォルダを用意してある．そこには課題の説明（direction*.txt）と採点スクリプト（test_cmd*.sh）を課題のタイプ別（*はタイプの番号に対応）で収めてあるほか，共用のプログラムなどを収めてある．課題ごとのフォルダ内に各ファイルが存在しなければ，これらが利用される．課題

のタイプは次の 8 通りとする.

1. プログラムファイルを受け取るのみ
2. CLI アプリのファイルを受け取って自動採点
3. GUI アプリのファイルを受け取って自動実行・自動操作
4. テキストデータを受け取るのみ
5. テキストデータを行ごとに自動採点
6. PNG ファイルを受け取るのみ
7. ドキュメントを受け取るのみ
8. ZIP ファイルを受け取るのみ

共用のプログラムは次の 2 つである.

- make_pc2.py
  Python コードからクラスおよび関数・メソッドの情報を抜粋
- test_xdo.sh
  xdotool.txt に従って GUI を自動操作しながらウィンドウ SS のアニメーションを生成

これらを含む，採点スクリプトおよび GUI の自動操作については第5章で詳細を述べる.

3.3. データベース構成

データベース構成も，基本的に本システムの前身と同様であるため，説明を割愛する.

4. PHP ファイル群

PHP ファイル群は，大きく分けて次の 4 つに分けられる.

1. ユーザ情報および認証関連
2. 課題および評価の提示関連
3. 課題の提出および採点関連
4. 管理関連

1 は論旨に関わらないため，説明を割愛する．2〜4 は，主に本システムの前身からの変更が大きな点について述べる.

4.1. 課題および評価の提示関連

該当するファイルは次の通りである.

- show_eu.php
  課題（評価）一覧ページ
- show_p.php
  課題内容ページ
- show_p9.php
  授業への意見等を入力するページ
- disp_png.php
  システムの内側に保存された PNG 画像の表示
- dl_file_d.php
  課題の配布ファイルのダウンロード
- dl_file_u.php
  提出ファイルのダウンロード

この中で論旨と関係の深い show_p.php について，詳細を説明する．

show_p.php は，課題の詳細な内容が表示されるとともに，答案の提出を受け付けるページである．その一例を図 3に示す．課題番号はクラスと授業回（2 桁）と課題の順番からなるが，学習者に作成させる課題用のフォルダ名も兼ねている．そのため，課題の説明文中のプロジェクト名が課題番号になるよう，表示時に置換している．メインの表には，タイトルに続き，説明，コード，図，配布物，提出，判定，コメント，正解例の欄が並ぶが，コード以降の欄はそれぞれ不要な場合には表示されない．判定欄には，下記のマークのいずれかが表示される．

- ”×”：　未提出
- ”▲”：　不合格
- ”確 (遅)”：　確認中（遅刻提出）
- ”●”：　合格（遅刻提出）
- ”確認中”：　確認中・未評価
- ”◯”：　合格
- ”◎”：　合格（加点対象）

メインの表の下には，前回提出したプログラムの実行結果と正解プログラムの実行結果を横に並べて表示する．テキストが出力されるアプリの場合は，相違点の背景をオレンジ色にすることで，プログラムの誤りを検討しやすくしてある．提出物が GUI アプリの場合は，自動実行

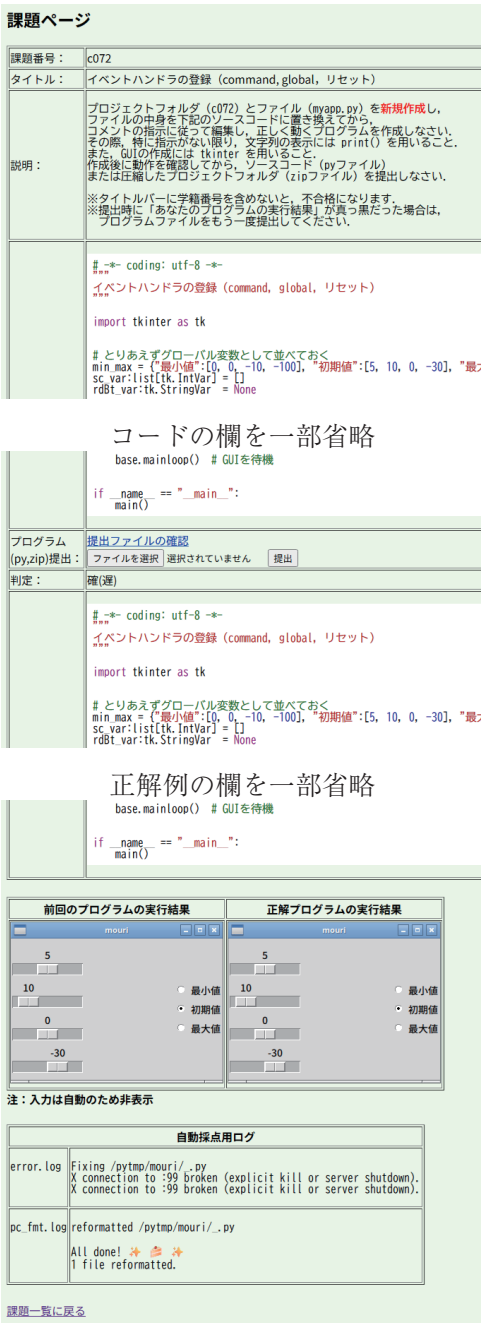


図 3: GUI を含む課題の内容ページ（show_p.php）の例

時のアニメーション（提出時の自動操作を指定していない場合は起動時のSS）を表示する．ここで，HTMLドキュメント中のimgタグのloading=”lazy”オプションによって，実質的にアニメーションの開始タイミングである表示タイミングを同時にすることで，動作状況の正解例との比較がしやすくしてある．課題の提出時だけでなく，事後にも「前回の結果」を確認・比較できる状況を用意することで，学習者個人へのフィードバックがなされるとともに，自主的な修正が促される．

これらの他，採点スクリプト実行時のログファイル群（*.log）の中身を列記する．本来の用途はデバッグだが，第2.4節で述べた自動的なコメントとしての，評価結果に対する学習者への説明にも活用している．よくある誤りへの指摘などを，採点スクリプトを介してログファイルに出力させることで，実質的に学習者へのコメントを自動化する形である．これにより，学習者個人へのフィードバックを早期化できる．

なお，評価結果が合格（加点対象）の際は，花火のアニメーションを表示する．些細な報酬ではあるが，学習者のモチベーション向上に役立っている．

4.2. 課題の提出および採点関連

該当するファイルは次の通りである．

- ul_code.php
  プログラムファイルの提出，採点スクリプトの実行，結果表示
- ul_text.php
  テキストエリアのデータの投稿，採点スクリプトの実行，結果表示
- ul_file.php
  その他の指定ファイルの提出
- submit_p9.php
  授業への意見等を提出

この中で論旨と関係が深いul_code.phpについて，詳細を説明する．

ul_code.phpは，課題タイプが1〜3の場合に，show_p.phpからプログラムファイルをPOSTすると呼び出される．出題課題別・ユーザ別フォルダへのファイルの格納，採点スクリプトの実行とその結果の表示を行う．

具体的には，まず，提出されたファイルの拡張子が[py, pyw, txt]の場合は「myapp.py」に，[pyz[注15)], zip]の場合は「a.zip」にリネームし，出題課題別・ユーザ別フォルダへ格納する．圧縮ファイルである[pyz, zip]を許容する理由は，アプリの実行に必要なファイル群をまとめて提出させるためである．ここで，圧縮ファイルの場合は展開も行うが，展開後にmyapp.pyが存在しない場合はエラー表示を行う．なお，この一連の工程は，採点スクリプトを実行するにあたり，プログラムファイル名を固定するために行

うものである．

続いて，実行フォルダを出題課題別・ユーザ別フォルダに移し，timeout コマンド注 16)とともに採点スクリプトを実行する．これにより，無限ループなど実行時のトラブルが発生してもシステムが機能不全に陥らない．GUI が含まれる課題タイプ 3 の場合は描画領域が必要なため，別途 xvfb-run コマンド注 17)とともに自動操作スクリプトを実行する．このとき，ブラウザの応答がタイムアウトすることを防ぐため，プロセスを分離させる．

採点スクリプトは，リダイレクトを用いて途中の標準出力や標準エラー出力を全てそれぞれ指定のファイルに吐き出し，最終的な評価値を表す 1 文字（不合格：”F”，確認中：”K”，合格：”G”，合格（加点対象）：”S”）のみを標準出力する形をとる．この評価値を PHP 側で受け取り，それに基づいてデータベースへの評価の登録と，ユーザへ示す評価結果ページを作成・表示する．

評価結果ページには，課題内容ページ「show_p.php」と同様に，GUI の画像を含め，提出されたプログラムの実行結果と正解プログラムの実行結果を並べて表示する．採点スクリプト実行時のログファイル群や加点対象への花火アニメーションも同様である．ただし，ページロード時には GUI の自動操作が未完了の場合があり，その場合は起動時の SS（静止画）が表示される．学習者が課題内容ページに戻る頃にはアニメーションが完成していることが多いため，この対処で問題は生じにくい．

4.3. 管理関連

該当するファイルは次の通りである．

- admin.php
  管理用の機能へのアクセスページ
- show_ls.php
  授業情報の一覧ページ
- edit_l.php, update_l.php
  授業ごとの設定ページと更新
- edit_el.php, update_es.php
  授業ごとの評価の管理ページと更新
- show_ap.php
  課題ごとの答案の一覧ページ
- read_p9.php
  授業ごとの意見の確認ページ
- update_ps.php
  課題バンク情報の DB への登録
- show_es.php
  数値化した評価の一覧ページ
- show_swd.php
  内部フォルダの表示
- achive_swd.php, dl_file_s.php
  内部フォルダの圧縮とダウンロード
- edit_us.php, update_us.php
  ユーザ情報の管理ページと更新
- remove.php

ユーザ情報等の削除

この中で論旨と関係の深い show_ap.php について，詳細を説明する．

show_ap.php は，同一課題における同一クラスの全学習者の答案・評価を表示することに加え，各評価の登録・編集が可能なページである．その一例を図 4に示す．横方向には更新用チェックボックス，学籍番号（学習者の ID），評価，ファイル，コメントの欄が並んでおり，縦方向に各学習者の答案情報が並んでいる．表の 2 行目にあるドロップダウンリストは，第 2.2.1項で述べた，答案を評価でフィルタリングする機能である．これにより，「確認中」のみを表示させるなどして，効率的に採点できる．

ファイルの欄は，標準では提出されたファイルへのリンクやテキストの答案を表示し，課題別・ユーザ別フォルダ内の画像ファイル「a.png」「ss_win.png」が存在する場合は評価が「確認中」の場合にそれを表示する．これも，手動採点が必要な対象を優先的に確認するための仕様である．ただし，図 4中であれば表の 2 行 4 列目のオプションをクリックすることで，全員分の画像や，課題別・ユーザ別フォルダの任意のテキストファイルの中身を表示できる．具体的には，全員の画像を閲覧したい場合は「全」を，ソースコードを閲覧したい場合は「py」を，出力を閲覧したい場合は「dat」項の数字を，採点スクリプトによって生成される「output_pc*.txt」を閲覧したい場合は「pc」項の数字を，エラーログを閲覧したい場合は「err」をクリックすればよい．これにより，手動のダウンロードと実行を経ずとも自動実行された結果から教員が判定を下せるケースが生じるため，第 2.2.2項で述べたように採点者の負担軽減とフィードバックの早期化に繋がる．

課題バンクフォルダ内に，GUI 実行時の想定ウィンドウ SS（アニメーション）である ss_win.png が存在する場合，学習者のプログラム由来の ss_win.png の右側に並べて表示させている．課題ページと同様に，HTML ドキュメント中の img タグの loading=”lazy”オプションによって，実質的にアニメーションの開始タイミングである表示タイミングを同時にすることで，第 2.3節で述べたように動作状況の比較がしやすくしてある．ここで，想定画像は同じ課題で本来すべて同一となり，キャッシュによってそのアニメーションのタイミングは同一画像で一致してしまう．隣接する学習者由来の ss_win.png とタイミングが異なると，その両者の比較，ひいてはウェブブラウザ上での教員の目視による採点が困難になる．そこで，img タグの src の指定時に学籍番号をパラメータに加えることで，キャッシュの使いまわしを回避させ，アニメーションの開始タイミングが隣接する学習者由来の ss_win.png と同期させて

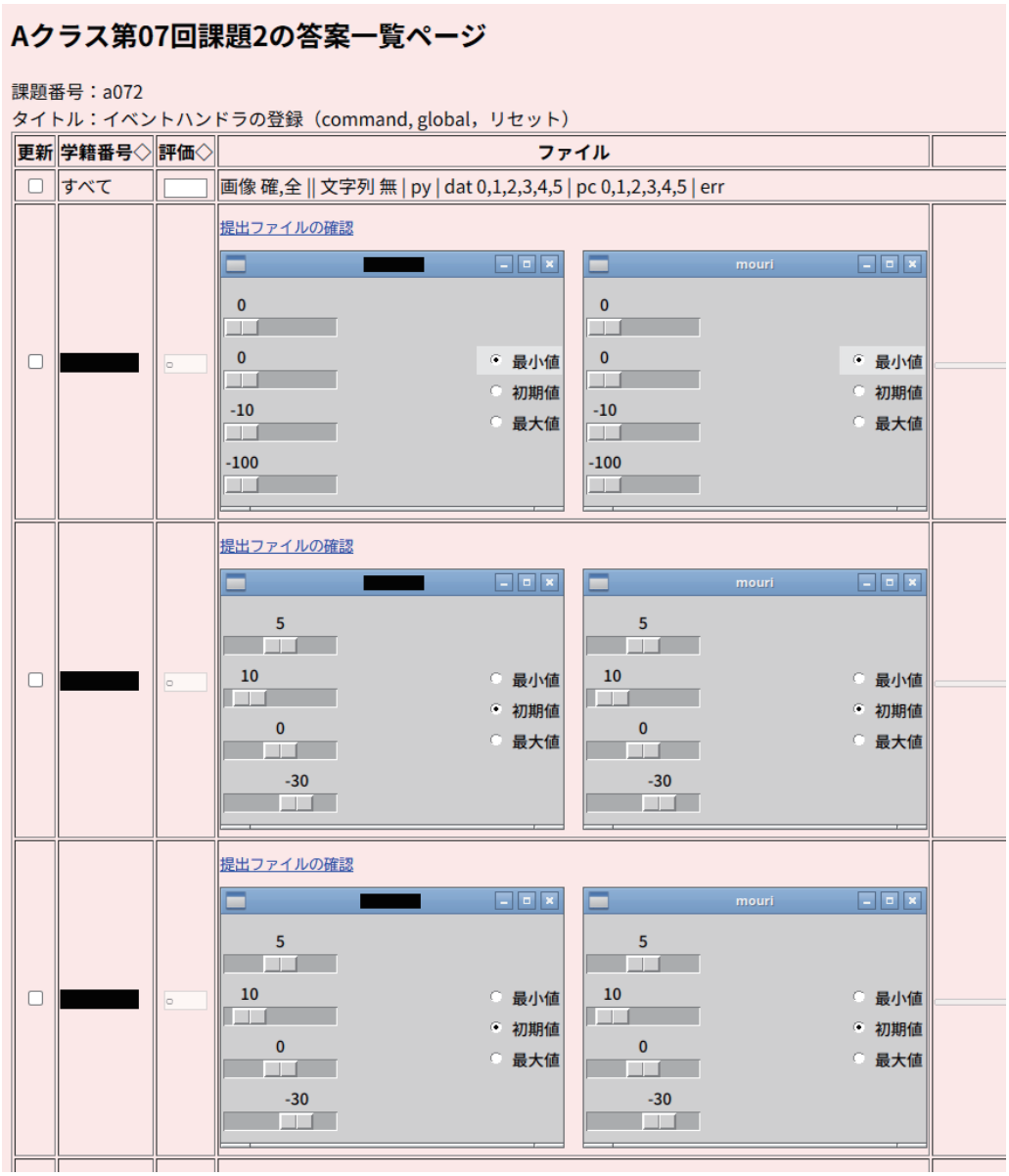


図 4: 答案一覧ページ（show_ap.php）の例

ある．実際に，図 4でも，横並びは画像が一致しつつも縦並びではアニメーションのタイミングが異なっていることが分かる．

コメント欄には，学習者個人へのコメントを書きこむことができる．筆者は主に，手動採点で評価を「不合格」にした場合に，その理由および助言を伝えるために利用している．

ページの最下段には，評価やコメント

の登録のためのボタンが配置してある．一覧の形で操作する都合上，評価やコメントは課題ごとにまとめての入力となる．

5. 自動採点・GUI 採点支援機能

5.1. 要求

自動採点機能および GUI 採点支援機能を実装するにあたって要求としたことは次の通りである．

1. Linux で利用できる無償のソフトウェア群・コマンド群で構成・構築
2. PHP から呼び出し，その採点結果を PHP から参照可能
3. Python プログラムを整形・解析し，その結果を採点に利用可能
4. Python プログラムを実行し，主に標準入力からデータを与え，標準出力は指定のファイルに保存
5. GUI アプリを自動操作し，その様子を APNG[注 18)]で記録
6. テキストの自動採点は別解に対応

それぞれの理由を簡潔に説明する．まず 1 については，筆者がシステム用の特別な予算を有していないためである．また，サーバの OS が Linux であることは前提のため，その上で利用可能な形で全てを構成・構築する必要がある．

2 については，表層部を PHP で構成することが前提のためである．PHP からの呼び出し方法や参照方法の自由度は高いため，容易な要求と言える．

3 については，プログラムの構造も採点対象に含めるためである．第 2.1節で述べたように，要件を満たしていない場合は「不合格」を学習者に伝えられた方がよい．また，解析に先駆けてソースコードを整形し，提出物とは別に保持しておく必要がある．

4 については，入出力に着目した採点のためである．処理用のデータは実行時引数かファイルか標準入力で与える必要があるが，実行時引数は初心者にとってハードルが高い．ファイルで与える場合は，ファイル入出力の単元以降とする必要がある．標準入出力は比較的早いタイミングで単元を済ませることができるため，これをベースとするのがよい．また，実行時の標準出力を指定の名前のファイルへ保存することで，実行と採点を切り離して考えることができる．

5 については，第 2.2.3項および第 2.3節で述べたように，GUI アプリの採点を省力化するためである．MP4 などの純粋な動画形式を利用しない理由は，現状で音声が不要であること，実時間ベースでなくてもよいこと，実装に難があることが挙げられる．要点について自動・短時間で切り替わる形なら，比較・採点が短時間で済ませられる．

6 については，答案に自由度を持たせ

ておきたいためである．詳細については前身のシステム[1)]を参照されたい．

5.2. タイプ 2: CLI アプリ提出時の動作

タイプ 2 の課題，つまり，アプリのファイルを受け取って CLI 上で自動実行および採点する場合の動作は，次のような流れである．

1. ファイルを受け取り，ZIP ファイルの場合は展開
2. ファイルを解析して自動採点に必要な情報を抽出
3. 提出された CLI アプリを入力用データファイルに基づいて自動実行
4. 出力および諸情報に基づいて自動判定しつつ，必要に応じてコメント

タイプ 2 標準スクリプトの前半をコード 1に示す．1〜5 行目では，変数を初期化し実行フォルダ（課題別・ユーザ別フォルダ）へ移動する．12〜17 行目では，解析に先立って，isort, autoflake, black を用いてソースコード群を自動整形し，コメント行を削除しつつファイルを output_pc.txt として結合する．このとき，ファイル名が「myapp.py」のものを予め「_.py」に変更しておくことで，モジュールファイルが同梱されていても課題のメインプログラムを先頭にして結合できる．

19〜24 行目では，まず，output_pc.txt を解析してクラスや関数・メソッドの情報を抽出し，output_pc2.txt として保存する．その具体例をコード 2に示す．

ここで，「class」の後はクラス名を，「def」の後はメソッド名を表し，「class ~; def」の後は関数名を表している．続いて，課題バンクフォルダ内に output_pc2.txt が存在している場合，diff コマンド[注 19)]を用いて比較した結果を diff_pc2.log として保存する．これにより，クラスや関数・メソッドについて指示に従っているかを確認できる．

タイプ 2 標準スクリプトの後半をコード 3に示す．26〜34 行目では，課題バンクフォルダ内に標準入力用データファイル「input*.dat」（*はワイルドカード）が存在する場合に，その個数分の自動実行を繰り返す．その際の標準出力は，装飾用の制御コードを取り除いた上で「output*.dat」（*は input に対応）保存される．対応する標準出力比較用のファイルが存在する場合は，diff コマンドを用いて比較した結果を diff_res.log として保存する．これにより，実行結果が想定のものであるかを確認できる．なお，実行には各 5 秒のタイムアウトを設定し，出力ファイルサイズにも上限を設けている．アプリの無限ループやページに実行結果を表示する際のトラブルを回避するためである．

36〜47 行目では，これまでに得られた

コード 1: 標準の自動採点スクリプト前半（タイプ 2，タイプ 3 共通）

```
# 変数を初期化し実行フォルダへ移動
d_sh=$(pwd)
p_id=$(basename ${d_sh})
flag_k=0
cd ../../${1}/${2}

# ソースコード群を空の一時フォルダへコピー
rm -fr /pytmp/${2} 2>> error.log
mkdir -p /pytmp/${2}
cp -fr *.py /pytmp/${2} 2>> error.log

# ソースコード群を自動整形した後に結合
mv /pytmp/${2}/myapp.py /pytmp/${2}/_.py
isort /pytmp/${2}/*.py 2>> pc_fmt.log 1>> error.log
autoflake -ri --remove-all-unused-imports --ignore-init-module-imports --remove-unused-variables /pytmp/${2}/*.py 2>> pc_fmt.log 1>> error.log
black --line-length 159 /pytmp/${2}/*.py 2>> pc_fmt.log 1>> error.log
cat /pytmp/${2}/*.py | sed -e '/^[␣␣]*#/d' -e 's/[␣␣]*$//' 1> output_pc.txt 2>> error.log

# クラスや関数・メソッドの情報を抽出
python3 ${d_sh}/../_default/make_pc2.py < output_pc.txt 1> output_pc2.txt 2>> error.log
if [ -s ${d_sh}/output_pc2.txt ]; then
  flag_k=1
  diff -diN --strip-trailing-cr output_pc2.txt ${d_sh}/output_pc2.txt >> diff_pc2.log
fi
```

結果から評価値を決定している．何らかの比較が行われた場合，それらの差分が存在すれば（差分ファイルが空でないならば）不合格（F），存在しなければ合格（G）とし，比較が行われていない場合は確認中（K）としている．第 2.4節で述べたように，評価値が F の場合，その理由や助言をログファイルに追記している．学習者が課題ページまたは提出後のページでこれを読むことで，修正と再提出を早期に行うことができる．

この標準の自動採点スクリプトによって，入出力の関係やプログラムの構造の大枠のみ確認したい課題のほぼ全てが，比較用のファイルを用意するだけで自動採点できる．

コード 2: output_pc2.txt の例

```
1 class Animal:
2 class Animal; def aruku()
3 class Animal; def naku()
4 class Dog(Animal):
5 class Dog; def ouchi()
6 class Dog; def sanpo(,)
7 class ~; def main()
```

5.3. タイプ 3: GUI アプリ提出時の動作

タイプ 3 の課題，つまり，アプリのファイルを受け取ってディスプレイ上で自動実行・自動操作する場合の動作は，次のような流れである．

1. ファイルを受け取り，ZIP ファイルの場合は展開
2. ファイルを解析して自動採点に必要な情報を抽出
3. 諸情報に基づいて足切りの自動判をしつつ，必要に応じてコメント
4. 提出された GUI アプリを実行し，操作指示に基づいて自動操作
5. 要所での SS 群からアニメーションを生成

タイプ 3 の標準の自動採点スクリプトの後半をコード 3に示す．26〜30 行目では，GUI の自動操作に必要な要素

1. 実行できる
2. Tkinter がインポートされている
3. ウィンドウタイトルに学籍番号が含まれている
4. GUI を起動・待機させている

が揃っているかを確認している．ウィンドウタイトルに学籍番号（学習者の ID）を含ませる理由は，SS を撮るウィンドウを検索するためである．グループワークの課題では全員の学籍番号を併記させることで，いずれの学習者が提出しても SS を撮影できる．

32 行目以降では，第 2.1節で述べたように，これまでに得られた結果から足切りを行うかを決定している．タイプ 2 の場合と同様に，評価値が F の場合，その理由や助言をログファイルに追記している．

タイプ 3 では自動採点スクリプトを呼び出した後に，GUI 自動操作のスクリプトを呼び出す．その前半をコード 5に示す．6〜8 行目では，軽量な openbox[注 20)] 上で GUI アプリを起動させている．28 秒のタイムアウトを設定しているが，立ち上がりで 2 秒のスリープを挟むこと，この後にも幾度かスリープさせることから，自動操作は実質的に 20 秒程度で終える必要がある．

10〜17 行目では，提出者の学籍番号（学習者の ID）を含むウィンドウを探して，scrot コマンド[注 21)] で SS を撮り，

コード 3: 標準の自動採点スクリプト後半（タイプ 2）

```
26 # input*.dat があれば自動実行
27 for f_i in ${d_sh}/input*.dat; do
28   f_o=$(basename ${f_i/input/output})
29   cat ${f_i} | timeout 5 python3 myapp.py | sed -r 's/\x1b\[[0-9;]*[a-zA-
       Z]//g' | head -n 5000 | head -c 256K 1> ${f_o} 2>> error.log
30   if [ -s ${d_sh}/${f_o} ]; then
31     flag_k=1
32     diff -dBwN --strip-trailing-cr ${f_o} ${d_sh}/${f_o} >> diff_res.log
33   fi
34 done
35
36 # 判定
37 if [ ${flag_k} -eq 0 ]; then
38   echo -n "K"
39 elif [ -s diff_pc2.log ]; then
40   echo -e "\nクラスや関数・メソッドを見直してください" >> diff_pc2.log
41   echo -n "F"
42 elif [ -s diff_res.log ]; then
43   echo -e "\n実行結果が異なります" >> diff_res.log
44   echo -n "F"
45 else
46   echo -n "G"
47 fi
```

ss_win.png として保存している．GUI の自動操作が指示されていない場合は，これが起動時の SS となる．ウィンドウが見つからない場合は起動に失敗していると言えるため，その場合にはその旨をログファイルに追記している．

コード 6に，GUI 自動操作スクリプトの後半を示す．19〜38 行目では，課題バンクフォルダに操作指示用のファイル xdotool.txt が置いてある場合に，その指示に従いながら GUI を操作し，その過程をアニメーションとして保存している．xdotool.txt の具体例をコード 7に示す．

空行以外は xdotool 上の指示であり，これによって GUI を操作する．空行の際は，連番を付けつつ scrot コマンドで SS を撮る．こうすることで，課題制作者が意図したタイミングの GUI の様子が画像的に記録できる．

コード 7に対応する SS を図 5に示す．また，対応付けは次のようになる．

0. 操作前（起動時）
1. メニューをクリックしていくこと

コード 4: 標準の自動採点スクリプト後半（タイプ 3）

```
# GUIの自動操作に必要な要素があるか確認
gui_check0=$(python3 -m py_compile myapp.py 2>&1) >> error.log
gui_check1=$(grep -c "^␣*import␣tkinter" output.pc.txt) 2>> error.log
gui_check2=$(grep -c ".title(.*${2}.*)" output_pc.txt) 2>> error.log
gui_check3=$(grep -c '.mainloop()' output_pc.txt) 2>> error.log

# 判定
if [ -s diff_pc2.log ]; then
  echo -e "\nクラスや関数・メソッドを見直してください" >> diff_pc2.log
  echo -n "F"
elif [ -n "${gui_check0}" ]; then
  echo -e "${gui_check0}\nプログラムが起動できないようです" >> error.log
  echo -n "F"
elif [ ${gui_check1} -lt 1 ]; then
  echo -e "\nTkinterを利用してください" >> error.log
  echo -n "F"
# 以降省略
```

コード 5: GUI の自動操作用スクリプトの前半

```
# 変数を初期化し実行フォルダへ移動
d_sh=$(pwd)
p_id=$(basename ${d_sh})
cd ../../${1}/${2}

# 軽量なopenbox上でGUIアプリケーションを起動
timeout 29 openbox 2>> error.log 1>>e1.log &
timeout 28 python3 myapp.py 2>> error.log 1>>e1.log & sleep 3 2>> error.
 log 1>>e1.log

# タイトルに基づいてウィンドウを探し，スクリーンショット
w_id=$(xdotool search --onlyvisible --name ${2}) 2>> error.log 1>> e1.log
if [ -z "${w_id}" ]; then
  echo -e "\nプログラムまたはGUIが正しく起動していません" >> error.log
fi
xdotool windowactivate ${w_id} 2>> error.log 1>> e1.log
xdotool windowmove ${w_id} 0 0 2>> error.log 1>> e1.log
scrot -ubz ss_win.png 2>> error.log 1>> e1.log
```

コード 6: GUI の自動操作用スクリプトの後半

```
19 # GUIの自動操作が指示されている（xdotool.txtがある）場合
20 if [ -s ${d_sh}/xdotool.txt ]; then
21
22   # 行ごとにxdotool向けの命令を実施するが，空行の場合はスクリーンショット
23   count=1
24   while read line1; do
25     if [ "${line1}" ]; then
26       xdotool ${line1} 2>> error.log 1>> e1.log
27     else
28       scrot -ubz /pytmp/${2}/ss_${count}.png 2>> error.log 1>> e1.log
29       count=$((count+1))
30     fi
31   done < ${d_sh}/xdotool.txt
32
33   # 画像をトリミングしてアニメーションを生成
34   wh_tmp=(‘identify -format ’%w␣%h’ ss_win.png‘)
35   mogrify -extent ${wh_tmp[0]}x${wh_tmp[1]} /pytmp/${2}/ss_*.png
36   cp ss_win.png /pytmp/${2}/ss_0.png 2>> error.log 1>> e1.log
37   apngasm ss_win.png /pytmp/${2}/ss_*.png 1 2 2>> error.log 1>> /dev/null
38 fi
39
40 # ウィンドウを閉じて終了
41 xdotool sleep 1 windowkill ${w_id} 2>> error.log 1>> e1.log
42 sleep 1
43 exit
```

でファイル読み込みダイアログを開き，ファイル名を入力

2. Enter キーで処理を進める
3. テキストエリアをクリックして文字列を入力
4. メニューをクリックしていくことでファイル書き込みダイアログを開き，ファイル名を入力
5. Enter キーで処理を進める

1 コマ目と 4 コマ目ではダイアログが立ち上がってアクティブになっているため，それらが撮られている．また，トリミングにより右側が切れている．

コード 6の 33〜35 行目では，ImageMagick で SS 群をトリミングし，そのサイズを ss_win.png に揃えている．37 行目では，APNG Assembler[注 22)] で SS 群からアニメーションを生成し，ss_win.png を上書きしている．アニメーションは，0.5 秒ごとに画像を送りつつループ再生する形である．こうすることで，GUI の

コード 7: xdotool.txt の例

```
mousemove 20 40
click 1
mousemove 20 70
click 1
type output_pc2.txt

key Return

click 1
type hogehoge
key Return

mousemove 20 40
click 1
mousemove 20 100
click 1
type output.dat

key Return

```

自動操作の様子をブラウザ上で素早く確認できるようになる.

6. システムの有用性の検証

6.1. 検証方法

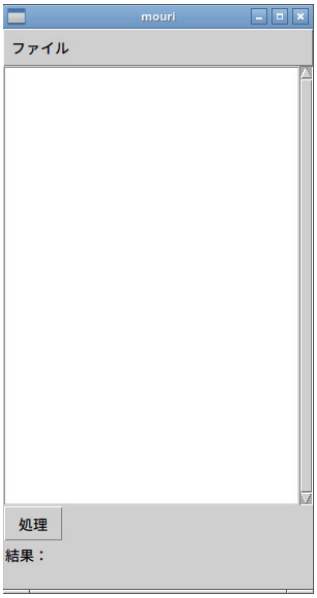
本システムの開発目的は，特に GUI アプリが提出された際の，採点者の負担軽減と学習者へのフィードバックの早期化である．両方に共通する指標として，採点に要した時間から本システムの有用性を測ることとした．具体的には，GUI アプリを提出する課題について，次のような調査を行い，結果を比較した.

1. 支援あり：答案一覧ページの表示から全件の評価の登録までにかかった時間を計測
2. 支援なし：答案一覧ページの表示から，答案のダウンロード・実行・操作を行い，全件の評価の登録までにかかった時間を計測

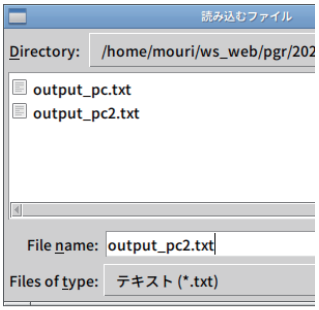
ここで，ブラウザ上のアニメーションにて不合格が疑われる答案に対しては，コメントのため支援ありの場合もソースコードの表示または提出物のダウンロード・実行・操作を行った．なお，実際に授業で提出された答案群を用いた関係上，個人情報保護の観点から筆者が採点者と計測者を兼務した．第 4.3節で述べたように，採点結果を課題ごとにまとめて入力する形をとっている関係上，答案 1 件ごとの時間計測は行っていない.

6.2. 調査結果

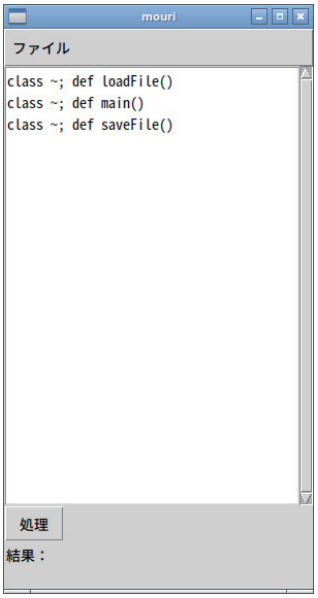
前述の調査を複数の課題について実施した．各課題の採点時間の総和の比較を表 2に示す．支援機能を用いることで，いずれの課題も 30%未満の時間で一通りの採点を終えられている．ここで，被験者・採点者である筆者は支援なしの採点を 8 年間程度経験しており，支援なしの採点を開発期間を含めて実験時点で 1 年



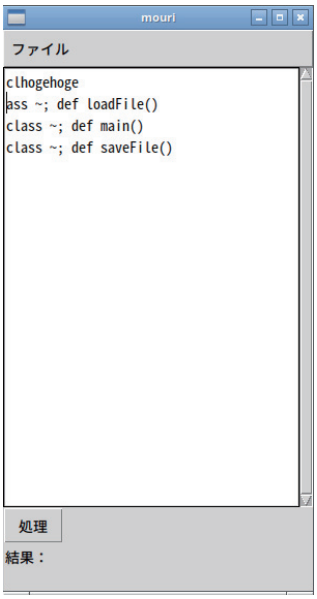
（起動時）



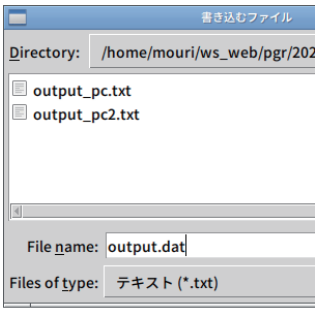
（1 コマ目）



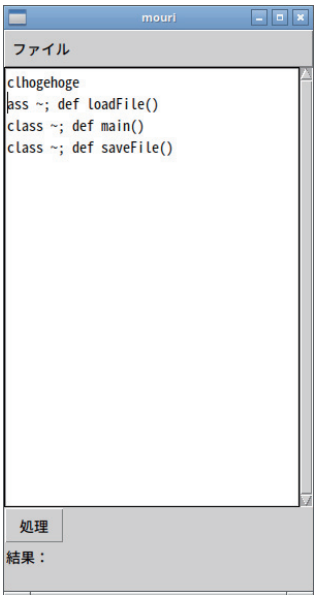
（2 コマ目）



（3 コマ目）



（4 コマ目）



（5 コマ目）

図 5: スクリーンショットの例（コード 7に対応）

表 2: 各課題の採点時間の総和 [秒]

課題	件数	支援あり	支援なし	比
a064	18	85	386	0.220
a071	19	117	436	0.269
a072	18	94	345	0.272
a074	16	103	390	0.264
a075	13	81	477	0.170
a076	10	59	262	0.225

間程度経験している．本実験では，被験者が支援の有無に関わらず慣れているために，操作ミスなどが無く安定した結果が得られたと言うことができる．逆に，採点や本システムの支援に不慣れな利用者の場合には，採点時間の揺らぎが大きくなりうる．特に，不合格が疑われる答案を採点する際には，支援ありの場合もソースコードの表示または提出物のダウンロード・実行・操作を行うため，採点のフローが変化する．その判断と対処に慣れていない場合は，アニメーションの再確認や操作ミスなどにより，支援なしの場合より時間を要する可能性もある．逆に，正解の動作が限定的で，かつ，不合格が疑われる答案が少ない場合，採点時間の比はより 0 に近づくと考えられる．

比が最も小さい課題 a075 は図 5にも示した課題であるが，ダイアログを利用する点とファイル入出力を伴う点で，支援なしの場合には大きな時間を使うことになった．支援ありの場合は，第 4.3節で述べた，出力されたテキストファイルの中身をブラウザ上に表示させる機能を利用して確認できるため，短時間で採点できたと言える．

6.3. 考察

本システムを利用することで，採点時間はもちろん，採点時のストレスが大幅に減ったことを実感している．授業中に採点する場合，その間は説明や学生のケアができないため，時間がかかると「早く終わらせたいという」プレッシャーが生じる．その中で，ファイル入出力を扱う課題 a075 のようにステップ数が多い場合は，心理的負荷が特に大きいものであった．それらが軽減されることの意義は大きいと言える．

学習者側の反応としては，以前に授業の SA[注 23)]から，次のような意見を得ている．

> 先生が質問対応に早く入れるようになって良かった．自分の時には動画ではなく，提出したプログラムがちゃんとしているか不安だった．提出した側でも確認できる点が良い．正解の動画と並んで表示されていると，もっと分かりやすくて良い．

図 3および図 4のように，GUI の場合でも提出者にも採点者にも正解プログラムの

実行結果と並べて表示する仕様にしたのは，この意見を受けてのことである．SA以外からは，システムに対する意見が特に出ていないが，学習者にも実益があったのであれば，本システムの導入に意義があったと言える．

6.4. 課題点

システムそのものではないが，GUIの自動操作について適切な指示を作成する労力が大きな点が，本システムの課題と言える．まず，普段使っているウィンドウマネージャがopenboxではなく，作成に際して切り替える必要がある．また，クリックなどの際の画面上の位置をはじめ，操作について丹念に検討・調査することが必要である．加えて，xdotoolについての知識も必要であり，指示ファイルの作成には試行錯誤も必要である．つまり，他者はもちろん，筆者自身が課題を追加・編集するにも労力が大きくなってしまう．コンテンツを充実させるためには，自動操作の指示の作成支援機能が必要であろう．

前向きな課題としては，自動採点の対象を増やせる可能性が挙げられる．自動化アニメーションやスクリーンショット群について，正解プログラムのものと上手く比較できれば，それに基づいて自動採点ができる可能性がある．現行より負担軽減やフィードバックの早期化を目指す場合，そのような機能が必要であろう．

7. まとめ

本稿では，GUI自動操作とスクリーンショットに基づくGUI採点支援機能を有したPythonプログラミング実習用システムについて，設計・構成を説明するとともに，採点時間を比較することで本システムの有用性を示した．まず，採点者にかかる負担について分類し，ぞれぞれの軽減について述べた．続いて，外部・内部のシステム設計と自動採点・GUI採点支援機能について，抜粋しながら詳細に示した．そして，採点時間が短縮されたことから本システムの有用性を示した．

今後の課題としては，コンテンツを充実させるためにはGUI自動操作の指示の作成支援機能の開発が挙げられる．また，さらなる負担軽減・フィードバック早期化を目指すためには，画像ベースでの自動採点についての検討が挙げられる．

謝辞

本稿の執筆にあたり，本学国際コミュニケーション学部の梅垣敦紀教授に作成して頂いたupLaTeXのクラスファイル[4)]を利用し，またその際に様々なご助力も頂いた．同氏に厚く感謝を申し上げる．

注

注 1）Aizu Online Judge,
https://judge.u-aizu.ac.jp
注 2）Progate, https://prog-8.com
注 3）paiza, https://paiza.jp
注 4）AtCoder, https://atcoder.jp
注 5）Document Object Model のことであり，HTML などのドキュメントの要素を木構造で表現し操作可能にする仕組み.
注 6）Seleinum,
https://www.selenium.dev
注 7）Ranorex,
https://ranorex.techmatrix.jp
注 8）xdotool,
https://github.com/jordansissel/xdotool
注 9）PyAutoGUI,
https://pyautogui.readthedocs.io
注 10）ソースコードや課題バンク等は下記の URL にて公開している.
https://halo.aichi-u.ac.jp/~mouri/works/programming_training_system_python.zip
注 11）javap, Java の SDK に付属する，バイトコードで書かれたクラスファイルを解析・逆アセンブルして人間が理解できる文字列に変換するツール.
注 12）isort,
https://pycqa.github.io/isort/
注 13）autoflake,
https://github.com/PyCQA/autoflake
注 14）black,
https://github.com/psf/black
注 15）Python として実行可能な圧縮ファイル用の拡張子．実際には特殊なファイルの追加後に ZIP 圧縮されていることを表す.
注 16）timeout, プログラムの実行に時間制限を設ける Linux コマンド.
注 17）xvfb-run, 仮想のディスプレイフレームバッファを生成してプログラムを実行するためのコマンド.
注 18）Animated Portable Network Graphics のことであり，画像形式である PNG を動画に拡張したもの.
注 19）diff, 2 つのテキストファイルを比較し，差分を出力する Linux コマンド．空白や改行を無視した比較も可能.
注 20）openbox, Linux 用のウィンドウマネージャ．シンプルかつ軽量で立ち上がりが早い．https://openbox.org
注 21）scrot, スクリーンショットを撮るための Linux コマンド．アクティブウィンドウに対しタイトルを含めた撮影も可能
注 22）APNG Assenbler, PNG ファイル群を 1 つの APNG ファイルに変換する.
https://apngasm.sourceforge.net
注 23）SA, 有償で授業の手伝いをする学生.

参考文献

1） 毛利 元昭: “JAR ファイルを対象とする自動採点機能を有した Java プログラミング実習用システムの構築と運用”, 愛知大学情報メディアセンター紀要「COM」, Vol. 31, No. 1, pp.11-35, 2022.

2） 立石 良生, 井上 潮: “模範解答を利用した GUI プログラミング課題の自動採点システム”, DEIM2020 最終論文集, I1-1, 2020.

3） 水谷 晃三: “Vision Language Model を用いたプログラミング課題の自動採点システムの設計と予備的評価”, 情報処理学会 情報教育シンポジウム SSS2025, pp.72-79, 2025.

4） 梅垣 敦紀: “愛知大学情報メディアセンター紀要「COM」への upLaTeX での投稿”, 愛知大学情報メディアセンター紀要「COM」, Vol. 34, No. 1, pp.1-10, 2025.

Title: **Development of Python Programming Training System with GUI Scoring Support Function Based on GUI Automatic Operation and Screenshots**

Author: **Motoaki MOURI**

生成 AI 協働型の「データ分析論」における教育実践
― 地域政策学部の学生を対象に ―

蒋　湧（愛知大学地域政策学部）

要旨

地域政策学部の「データ分析論」講義は、根拠に基づく政策立案（Evidence Based Policy Making、EBPM）に必要なデータ分析手法を学ぶ演習科目である。2 年次以上の学部生を対象に、これまで表計算ソフトを用いた統計分析とグラフ作成など、いわゆる情報リテラシーをベースにした演習科目であった。2025 年秋学期の「データ分析論」は、実験的に AI コーディングと Python プログラミングを導入し、生成 AI 協働型のデータ分析講義の教育実践を試みた。人数を制限した受講者と複数の教員が参加し、共に生成 AI 協働型演習科目の在り方を探ってみた。本稿は、半年の教育実践活動をまとめたものであり、今後生成 AI 時代の文系大学におけるデータ分析科目の在り方への議論と実践に寄与することを目指す。

キーワード： 生成 AI との協働，Anaconda と JupyterLab を用いた Python プログラミング環境，AI コーディング、Pandas によるデータ分析、Matplotlib によるデータの可視化

1. はじめに

1.1. 教育実験の背景

地域政策学部の「データ分析論」講義は、根拠に基づく政策立案（Evidence Based Policy Making、EBPM）に必要なデータ分析手法を学ぶ演習科目であり、これまで、表計算ソフトを用いた統計分析とグラフ作成など、いわゆる情報リテラシーをベースにした演習を行ってきた。2025 年秋学期の「データ分析論」は、実験的に AI コーディングと Python プログラミングを用いたデータ分析を導入した。人数を制限した受講者、複数の教員が演習科目に参加し、学生と教員が生成 AI 協働型の演習科目の在り方を探ってみた。

講義は、生成 AI との協働で Python プログラミングを用いた地域データ解析と可視化を目指す。これまで、Python プログラミングを用いたデータ解析は、高度なデータ分析の知識とスキルを要することで、主に理工系学生向けの講義内容とされてきた。本講義は、文系の学生を対象に、とりわけ Python プログラミング経験のない学生に向けて、こうした講義内容を体験させる背景として、以下の 2

点が挙げられる。

近年、「根拠に基づく政策立案（EBPM）の重要性が注目されている。愛知大学地域政策学において、データサイエンスに基づいた地域データ分析講義は、学部の特色ある教育として重視されていることが、一つ目の背景である。もう一つの背景は、生成 AI の飛躍的な発展と普及により、これまでハードルの高かったデータ分析に関する知識とスキルが、生成 AI の助けを借りて、文系の学生も扱いやすくなった現実がある。

生成 AI との協働」の実践は本講義の目的である。地域政策の研究を志している教員と学生にとって、まず、自分と生成 AI の「関係意識」からはじめる。つまり、生成 AI の時代に向けて、教員はどのように教えるべきか、学生はどのように学ぶべきか、自らの課題と目標を持って、生成 AI との協働を探ってみる。

教育実践に参加したのは、18 名の 2 年次学生と数名の教員であった。生成 AI を自分の「伴走者」、あるいは「代走者」であるように、AI 協働型の Python プログラミングによるデータ解析の学び方、教え方を実践してみる。

1.2. 教育実践の主な内容

図 1 には 15 回講義の主な内容を示す。教育実践は以下の 4 つのテーマに沿って行われた。一つ目は生成 AI の協働である。生成 AI の概念と特徴を学び、生成 AI 協働型の教え方・学び方を実践の中に確認する。二つ目は文系の学生に対する Python プログラミングに取り巻く情報処理の「生態系」を紹介する。三つ目は Python プログラミング環境の構築を演習形式で学ぶ。四つ目は生成 AI 協働型の Python によるデータ解析とデータ可視化を 4 つの演習を通して実践する。以下の章は教育実践の詳細を報告する。



図 1: 講義の内容

2. 生成 AI との協働

進化し続ける今日の生成 AI をどのように認識するのか、それは直面する最初の課題であった。権威的な論説を探すよりは、これまで皆さんが実際に生成 AI に触れてみた体験から得た感想と認識を

表 1: 人間と生成 AI の「知恵」の特徴 （2025 年春学期地域政策学部「研究法 07」）

人間の知恵 (Human Intelligence)	生成 AI の知恵 (Generative AI Intelligence)
経験と感情に基づく 喜怒哀楽などの感情や身体的体験から学びを深める	**大量のデータに基づく** 学習データの統計的パターンに基づいて推論・生成を行う
価値判断が可能 倫理、道徳、美意識など主観的・社会的判断ができる	**感情や意識はない** 感情や主観的判断は模倣できるが、実体としては持たない
目的意識がある 自己実現や他者貢献など、内発的な動機に基づく行動ができる	**目的は外部に設定される** 与えられた指示や設計に従って作動する
創造性と直感を持つ 既存知識の枠を超えた発想や「ひらめき」が生まれる	**創造は組合せ的** 過去の情報を組み合わせて新しい表現を生み出すが、本質的な意味での「創造」ではない
文脈を超えて学ぶ 断片的な情報から意味を構築し、未知の状況に柔軟に対応する	**文脈理解は表面的** 文法や語彙のパターンから文脈を推定するが、深層的理解はない
身体と結びつく 五感や身体感覚を通じて環境と相互作用するは	**身体性を持たない** 環境との直接的な相互作用せず、センサーなどを介して間接的に対応するのみ

ベースに、議論を深めることは本教育実践の第 1 歩として位置付けた。

表 1 は、2025 年春学期の「研究法」において、「地域研究のための生成 AI の活用」をテーマに、17 名の 2 年次学生がまとめた「人間と生成 AI の「知恵」の特徴」である。春学期の研究法では、学生の地元の自治体を対象に、自ら 1 つの「地域課題」を選定し、生成 AI との協働で、①課題の現状、②課題存在の社会的背景、③課題が地域に与える影響、④課題を解決するための方策、4 点を踏まえた教育実践を行った。

表 1 の人間と生成 AI に関する「知恵」の理解を踏まえ、本講義では、人間の知恵の補完という視点により、以下のように生成 AI の役割を整理した。

- 情報処理の高速化・効率化
  煩雑な作業の代行（大量のデータから瞬時に要約・検索・分析、文書作成、資料整理など）
- スキルと発想支援・創造のきっかけ提供
  専門分野スキルの提供と多様な視点からの提案（思考の幅を広げ、人間創造性の刺激）
- 知識の可視化と整理
  専門知識や複雑な概念を簡潔に構造化し可視化
- 時間・場所に依存しない助言者
  人間の思考や判断の伴走者として、24 時間アクセス可能な「知識ベース」

その結果から、図 2 の「自分と生成 AI 協

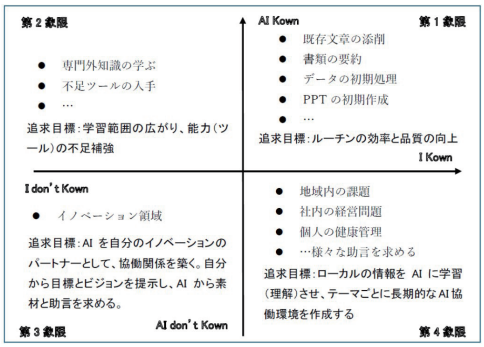


図 2: 自分と生成 AI 協働の 4 象限

働の 4 象限」を学生に提示し、今回の教育実践における自分と生成 AI との協働関係を確認させた。横軸と縦軸、それぞれは「I Know」と「AI Know」とする。それより、第 1 象限は、自分と AI がともに知って、できる分野になる。第 2 象限は、AI が知っているが、自分は知らない、できない分野であり、AI の助けを求める領域になる。例えば、データ解析の第 1 段階には、データ収集やデータ加工などの事前作業が発生する。それらは、「データ処理のルーチン作業」として、第 1 象限に示した作業効率の向上のために、生成 AI の活用は可能になる。また、Python によるデータ解析について、第 2 象限に示されたように、生成 AI が自分の能力補強として活用されることになる。学習の目標に合わせて、生成 AI の結果だけを受け入れることも可能になる。

3. Python を用いたデータ解析に関するソフトウェアの「生態系」

PC に動作するソフトウェアの「生態系」について、文系学生にとっては経験と知識の浅い領域である。情報リテラシー教育の一貫として、それらの内容を取り入れることは欠かせない。図 3 に示した概念と内容は、一回の座学講義で紹介した。そのとき、以下の項目に説明を加えた。

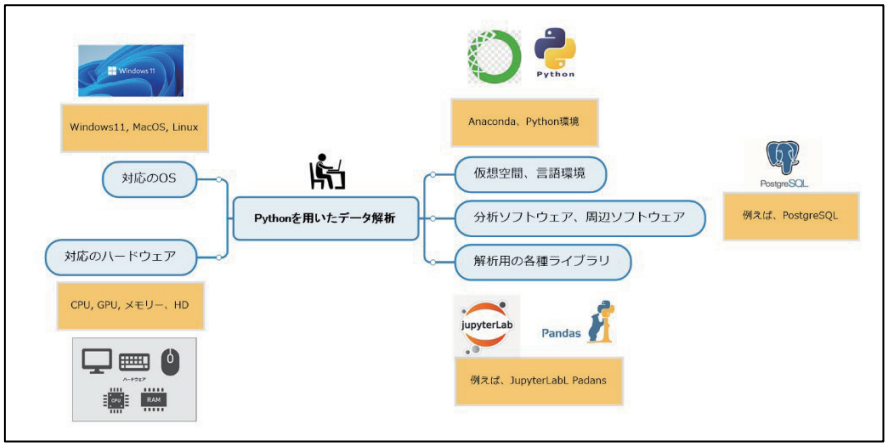


図 3: Python を用いたデータ解析に関するソフトウェアの「生態系」

- PC のハードウェア
  プログラミングにおいて、特に重視されるメモリ、CPU と GPU の基礎知識を解説した。
- OS の選択
  特に Windows と Mac を中心に、特徴を説明する。
- オープンソースの知識
  オープンソースの歴史、オープンソースの社会的な意味、オープンソースのバージョン管理とその課題、Anaconda の役割を説明した。
- Python プログラミングの仮想環境
  タスクごとのプログラミング作業環境、ソフトウェアの互換性を考慮した作業環境、Anaconda による仮想環境を作成する（図 4）。

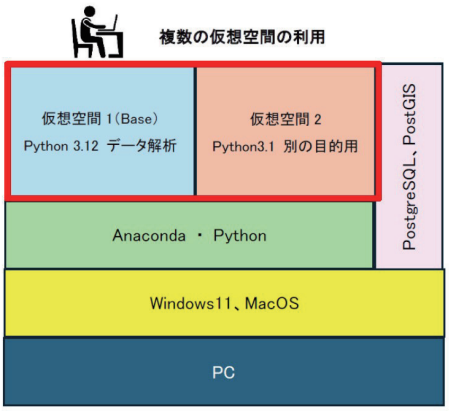


図 4: Python を用いたデータ解析に関するソフトウェアの「生態系」

4. Python プログラミング演習環境の構築

次は演習として、図 5 に示した Python プログラミング作業環境を構築した。以下は、主な作業の手順を示す。

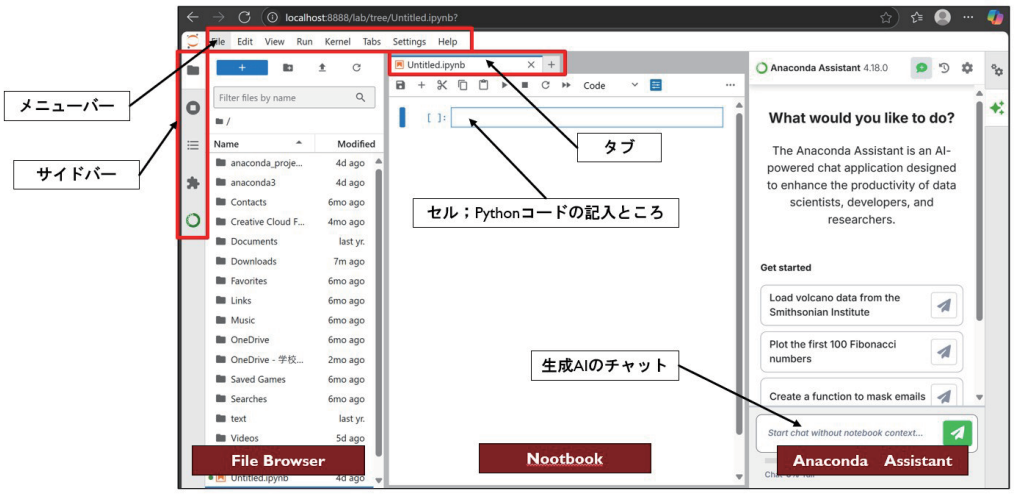


図 5: Jupyter Lab による Python を用いたデータ解析の画面

(ア) Python3.12 ベースの Anaconda3 システムをインストールする
(イ) 仮想環境 base とテスト用の test_env 仮想環境を作成する
(ウ) JupyterLab をインストールする
(エ) Anaconda にユーザーアカウントを作成する
(オ) Anaconda Assistant の開設、ChatGPT との連携可能
(カ) OpenAI から ChatGPT、あるいは Google から Gemini の使用を準備しておく。

学生は持参の PC において、図 4 と図 5 に示した Python プログラミング作業環境を構築した。

5. 生成 AI 協働型の Python によるデータ解析とデータ可視化の教育実践

ここまで、生成 AI 協働型のデータ解析の実践に関する基本知識の学習と作業環境整備の演習を行った。

次は、表 2 に示された 4 つのテーマを通して、生成 AI 協働型のデータ解析を実践する。以下は、教育実践の方針を踏まえて、4 回の演習の目標、詳細内容を紹介する。

表 2: 生成 AI 協働型の Python データ分析における演習内容

演習	テーマ	主な内容と目標
1	Pandas データフレームを学ぶ：1975 年〜2024 年日本年齢 3 区分人口データの分析と可視化	a) e-Stat から統計データを CSV ファイルでまとめる（手動）、b) CSV データを DataFrame への読み込み（AI 協働）、c) Pandas によるデータ分析（AI 協働）、d) Matplotlib による折れ線の作成（AI 協働）、e) JupyterLab による演習レポートの作成（手動）
2	年齢構成の地理的偏在:2025 年度、名古屋市、豊橋市、新城市、設楽町における年齢 5 歳階級・男女別の人口データの分析と可視化	a) e-Stat から統計データを CSV ファイルでまとめる（AI 協働）、b) CSV データを DataFrame への読み込み（AI 協働）、c)Pandas による「移動目的別の特化係数」の計算（AI 協働）、d)Matplotlib による移動目的別特化係数のヒートマップ作成（AI 協働）、e) Matplotlib による名古屋圏への転入ギャラリー作成（AI 協働）、f)JupyterLab による演習レポートの作成（手動）
3	名古屋圏における人々の地域移動：2025 年度 47 都道府県から名古屋圏へ転入の年齢 5 歳階級人口の分析と可視化	a) e-Stat から統計データを CSV ファイルでまとめる（手動）、b) 分析シナリオの設計（AI 協働）、c) Pandas によるデータ分析（AI 協働）、d) Matplotlib による縦棒グラフと人口ピラミッドグラフの作成（AI 協働）、e) JupyterLab による演習レポートの作成（手動）
4	地理空間情報の収集と可視化：タウンページから豊橋のコンビニエンスストアデータを収集し、店舗住所から緯度経度を取得し、店舗立地マップを作成する	a) タウンページから店舗データを収集する（AI 協働）、b) 住所から緯度経度を入手する（AI 協働）、c) QGIS でポイントレイヤを作成（手動）、d) GeoPandas を用いて空間データを GeoJSON に出力（AI 協働）、e) GeoJSON データと QGIS でマップ作成（手動）、f) JupyterLab による演習レポートの作成（手動）

5.1. 教育実践の方針

- 「学び方」を自ら学ぶ
  従来の教員指示によりデータ処理やプログラミングの手法を代わりに、生成 AI 協働の「学び方」を学生自ら学び、目的を達成するための道筋を探ってみる。
- 多様な学習目標への試み
  講義には、統一的なスキル学習の目標は設けない。生成 AI との向き合い方は、学生の目的に合わせて自ら決める。例えば、地域政策学を志している学生であれば、生成 AI から得た Python コードを深く理解する必要はない。生成 AI は、能力補強の「助手」として活用し、その結果を利用する。一方、データアナリストを志している者対しては、その状況は変わる。生成 AI は、Python プログラミングの「教師」になり、コード技術を深く学び、自分の能力を伸ばすことは可能になる。
- 成果物よりは生成 AI 協働のプロセスを重視
  教育実践の目標は、生成 AI 協働のプロセスにある。演習ごとに「演習レポート」の提出が課せられる。ここでは、成果物よりは、生成 AI との協働のプロセスを整理し、その中から新しい「学び方」の発見が期待される。
- 教員の役割
  生成 AI 協働の実践において、教員と学生は同じスタートラインに立っている。教員の役割は、従来の「教える」から共に学びの「同伴者」に変わる。場合によって、教員は「経験者」として学生にアドバイスを与えるが、「指示」はしない。

教員が事前に作成した 4 回の演習レポートを学生に提示した。それは教員自らの「協働記録」であり、学生に 1 つの事例として参考に使われる。以下は、教員の演習レポートを中心に、教育実践に詳細を報告する。

5.2. 演習 1：1975 年〜2024 年日本年齢 3 区分人口データの分析

最初の演習として、Pandas の DataFrame の基本概念と Matplotlib 使い方を中心に、生成 AI 協働型の学び方の確認に目標を設ける。初回の演習に、手動のデータ収集とデータ整理を行った。

【課題 1】 Python による csv ファイルの読み込み
ChatGPT に整理した csv データを渡し、Python による csv データを DataFrame への読み込みを依頼する（図 6）。

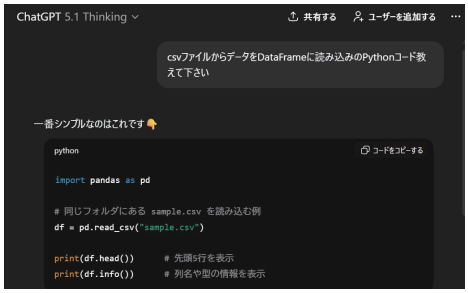


図 6: Python による csv ファイル読み込みの依頼



図 7: Pandas の DataFrame によるデータ一覧

生成 AI から得られた Python コードに、演習環境の Workspace パスを入れて、JupyterLab で実行すると図 7 の結果が現れる。

【課題 2】DataFrame の基本概念と使い方

DataFrame の基本概念と操作方法を、生成 AI の指示に従って、確認していく。図 8 には、DataFrame に index の設置方法を確認する様子である。

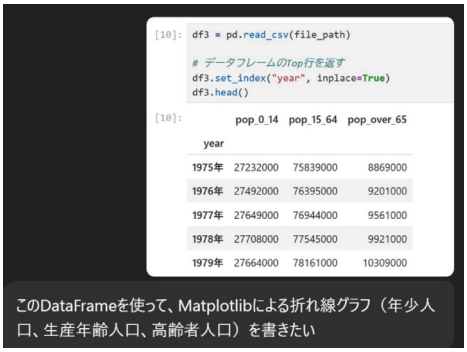


図 8: index の設定と折れ線の作成（AI 協働）

【課題 3】 Matplotlib によるグラフ作成

次は、Matplotlib によるグラフ作成を体験する。

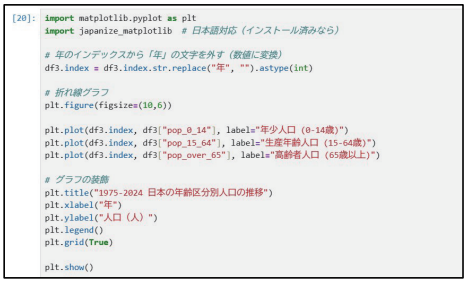


図 9: AI コーディングの結果（折れ線作成）

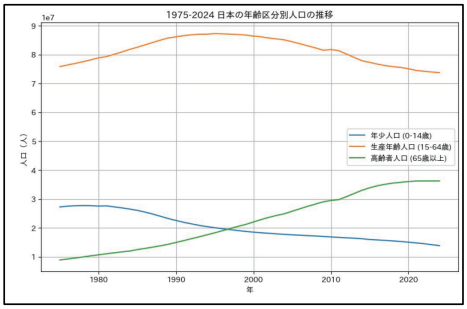


図 10: 実行で得られたグラフの結果

5.3. 演習 2：年齢構成の地域偏在

演習 2 からは、演習のシナリオを重視した生成 AI との協働を実践する。初回目は、教員から問題意識やシナリオの構想案を提示する。

【課題 1】演習のシナリオ

地域の「少子高齢化」は本来、時系列でその推移を追うべき現象である。本演習では視点を変え、複数自治体における横断的な比較により、年齢構成の地理空間的偏在（＝どの地域に若年層が希薄化し、高齢層が集積しているか）を確認する。政府統計総合窓口 e-Stat の統計を用い、愛知県の名古屋市・豊橋市・新城市・北設楽郡を対象に、2025 年の年齢 5 歳階級・男女別人口を分析し、可視化する。

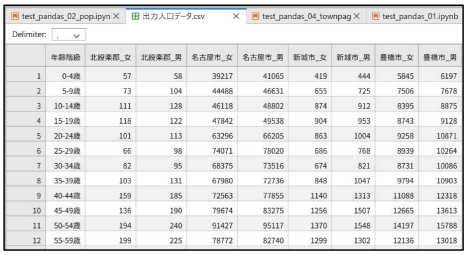
【課題 2】生成 AI を活用した統計データの事前加工

データの事前加工などルーチン的な作業を生成 AI にまかせてみる。その際、生成 AI への指示は、簡潔かつ明確であることが重要になる（図 11）。



図 11: 生成AIにデータ抽出の指示を出す

図 12 は生成 AI が整理、加工された csv データを示す。



test_pandas_02_pop.ipyn × 出力人口データ.csv × test_pandas_04_townpag × test_pandas_01.ipynb

Delimiter: ,

	年齢階級	北設楽郡_女	北設楽郡_男	名古屋市_女	名古屋市_男	新城市_女	新城市_男	豊橋市_女	豊橋市_男
1	0-4歳	57	58	39217	41065	419	444	5845	6197
2	5-9歳	73	104	44488	46631	655	725	7506	7678
3	10-14歳	111	128	46118	48802	874	912	8395	8875
4	15-19歳	118	122	47842	49538	904	953	8743	9128
5	20-24歳	101	113	63296	66205	863	1004	9258	10871
6	25-29歳	66	98	74071	78020	686	768	8939	10264
7	30-34歳	82	95	68375	73516	674	821	8731	10086
8	35-39歳	103	131	67980	72736	848	1047	9794	10903
9	40-44歳	159	185	72563	77855	1140	1313	11088	12318
10	45-49歳	136	190	79674	83275	1256	1507	12665	13613
11	50-54歳	194	240	91427	95117	1370	1548	14197	15788
12	55-59歳	199	225	78772	82740	1299	1302	12136	13018

図 12: 生成 AI で整理された csv データ

【課題 3】 Pandas によるデータ解析

演習シナリオに従って、生成 AI に対し、Pandas を用いたデータ解析の指示を出す。

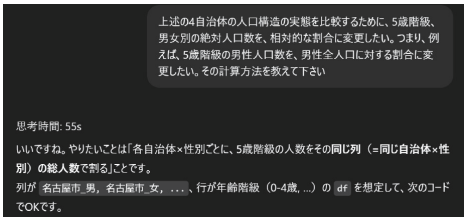


図 13: 生成 AI で統計分析を行う

図 14 は得られた解析結果を示す。

年齢階級	北設楽郡_女	北設楽郡_男	名古屋市_女	名古屋市_男	新城市_女	新城市_男	豊橋市_女	豊橋市_男
0-4歳	0.014790	0.015629	0.033655	0.036133	0.019739	0.021044	0.032047	0.033773
5-9歳	0.018941	0.028025	0.038179	0.041031	0.030857	0.034362	0.041154	0.041844
10-14歳	0.028801	0.034492	0.039577	0.042941	0.041174	0.043225	0.046028	0.048367
15-19歳	0.030618	0.032875	0.041057	0.043589	0.042587	0.045168	0.047937	0.049746
20-24歳	0.026207	0.030450	0.054319	0.058254	0.040656	0.047585	0.050760	0.059245

図 14: 生成 AI で得られた年齢階級・男女別の人口割合

【課題 4】 Matplotlib によるグラフギャラリーの作成

人口ピラミッドで人口の年齢構造を可

視化する。「年齢構成の地域偏在」というシナリオの趣旨にそって、4 地域の人口ピラミッドによるグラフギャラリーを使って表現したい。グラフギャラリーの作成は、表計算ソフトによる情報リテラシーの範疇を超え、やや高度な Python プログラミングの技術が必要とされる。そのタスクを生成 AI に任せてみる。

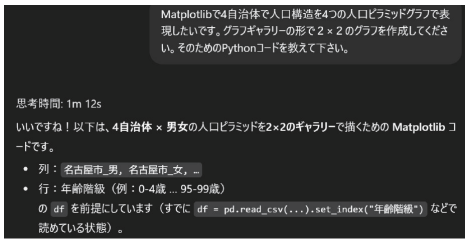
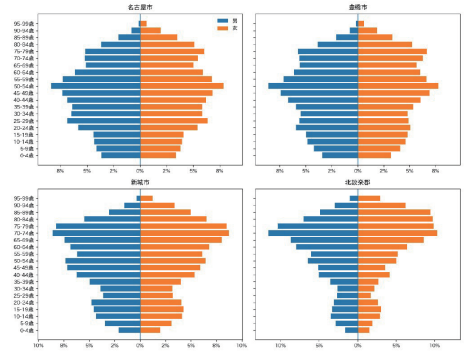


図 15: 生成 AI に人口ピラミッドギャラリーの作成依頼


図 16: 完成された人口ピラミッドギャラリー

期待された通りに、4 自治体における人口ピラミッドのグラフギャラリーが生成された（図 16）。

5.4. 演習 3：名古屋圏における人々の地域移動

演習 2 では、教員が提示した演習シナリオに沿って、データの解析と可視化を行った。演習 3 は、学生が自ら、生成 AI との協働で演習のシナリオを作成することを目指す。

【課題 1】演習の目的

名古屋圏を中心に、①どのような人々が、②なんの目的で、③どこから or どこへ移動するかを明らかにする。入手したデータからは「移動目的」は明確になっていないので、移動目的グループの概念を入れてほしい。

【演習 2】生成 AI 協働で演習シナリオの作成

これまでの演習には、データ収集、分析方法、グラフ作成方法などで生成 AI との協働を体験してきた。今回の演習では、「研究シナリオ」の検討に生成 AI との協働を試みる。

人間と比べ、生成 AI は受動的に動く特徴があり、我々自ら研究の目的、ほしい結果などをわかりやすく生成 AI に伝える必要がある。また、そのプロセスは一回きりではなく、状況により生成 AI との協議が、満足するまで複数回繰り返す。ここでは、教員の協働記録を学生に共有する。

【質問】薦める分析方法を教えて下さい
【回答】おすすめ分析メニュー
（地域 × 年齢の両面で“特徴”を抽出）

- 年齢プロファイルの偏り（LQ：特化係数）
- 「指標化」でストーリーを作る（目的別サマリー）
- クラスタリング（似た年齢構成の転入元どうしをグループ化）
- 年齢×地域のヒートマップで“濃淡”を俯瞰

など、7 つの分析メニューが提示された。

それに対し、著者は①特化係数による年齢プロファイルの偏り、②目的別の「標準化」、③年齢×地域のヒートマップで可視化、3 つの要素を融合した分析を提案した（図 17）。

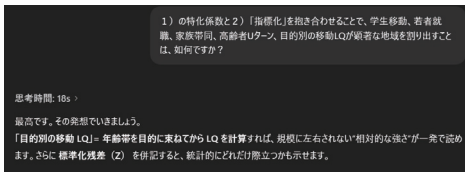


図 17: 生成 AI との協働で演習シナリオを作成する

生成 AI が以下の目的別の年齢グループとそれに用いた LQ 計算の案を提示してくれた（図 18）。

【演習 3】 移動目的別のヒートマップの作成

次には、生成 AI に 47 都道府県×移動目的グループに関するヒートマップ作成を依頼し、図 19 の結果が得られた。

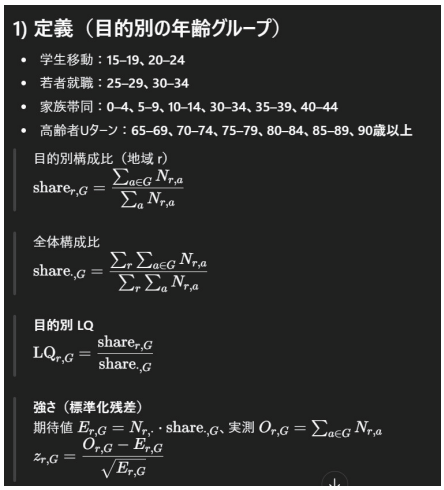


図 18: 生成 AI から提案された移動目的別の LQ 計算

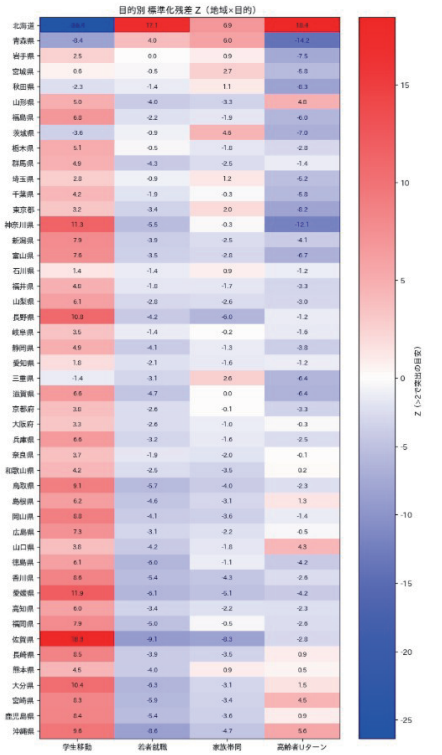


図 19: 47 都道府県×移動目的グループのヒットマップ

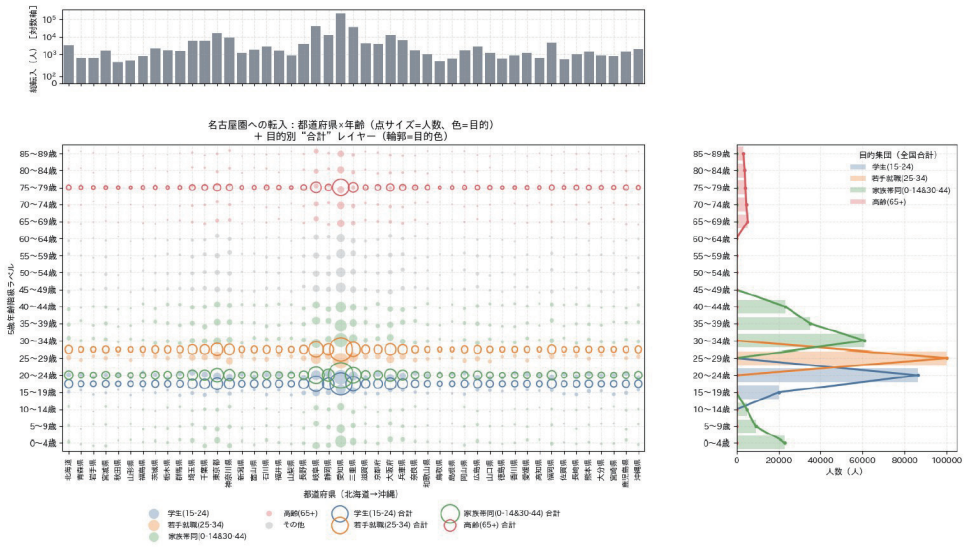


図 20: 都道府県×移動目的グループにおける散布図とヒストグラムのギャラリー

【演習 4】 移動目的別人口数のギャラリー作成

最後に、都道府県別、移動目的グループごとの移動人口絶対数に対し、散布図とヒストグラム図に合わせたグラフギャラリーの作成を生成 AI に依頼した。かなり高度なグラフの作成にもかかわらず、目的に合った結果が入手できた（図 20）。

5.5. 演習 4：空間情報の収集と可視化

演習 4 は、学部の GIS（地理情報システム）カリキュラムに合わせて、生成 AI 協働における地理空間データの扱いを実践した。学部の GIS 演習 I でも扱った豊橋コンビニエンスストアをテーマに、生成 AI 活用した方法を提示する。

【課題 1】生成 AI による店舗データの入手

タウンページから豊橋市のコンビニエンスストアのデータを抽出するが、これまでの GIS 演習 1 には、およそ演習講義一回分の時間を使って、データ収集、加工とアドレスマッチングの作業を行ってきました。本教育実践は、生成 AI を活用したデータ収集と加工を体験する。

図 21 のようにタウンページの URL、それに合わせた詳細の要望を生成 AI へ渡す。結果として、図 22 のような csv フ

タウンページ　https://itp.ne.jp/zenkoku/aichi/から以下の情報を抽出したい

（1）愛知県豊橋市
（2）コンビニエンスストア
（3）抽出したいデータ項目：系列名、店舗名、住所
（4）出力：CSV形式

図 21: 生成 AI に店舗データダウンロードの依頼

ァイルが入手できた。一部無料版の生成 AI は、タウンページからコピー/貼り付けのテキスト情報を渡す必要がある。


townpage.ipynb | toyohashi_conbini.csv | test_pandas_01.ipynb

Delimiter: ,

	系列名	店舗名	住所
1	セブン - イレブン	豊橋東脇店	愛知県豊橋市東脇１丁目２１－９豊橋駅
2	酒店たばこ店ローソン	豊橋二川店	愛知県豊橋市大岩町字小山塚６２－８二川駅
3	セブン - イレブン	豊橋高師本郷町店	愛知県豊橋市高師本郷町字竹ノ内７１芦原駅
4	酒店ファミリーマート	豊橋江島店	愛知県豊橋市江島町１４１－１
5	ファミリーマート	豊橋柱八番町店	愛知県豊橋市柱八番町４７愛知大学前駅 /
6	セブン - イレブン	豊橋草間町店	愛知県豊橋市草間町字平南７－１南栄駅
7	ローソン	豊橋老津店	愛知県豊橋市老津町字沖田７３－３老津駅
8	セブン - イレブン	豊橋石巻本町店	愛知県豊橋市石巻本町字北市場４５－６
9	セブン - イレブン	豊橋忠興店	愛知県豊橋市忠興１丁目５－１６赤岩口駅 /

図 22: 収集された店舗データ

【課題 2】アドレスマッチング

GIS 演習 I において、アドレスマッチングは東京大学のアドレスマッチングサービスサイトを利用してきた。本実践では、Google の Gemini を利用し、緯度経度の情報を入手した（図 23）。ChatGPT は、Open Street Map を利用したアドレスマッチング Python コードを入手したが、変換成功までは至らなかった。



	A	B	C	D	E
1	系列名	店舗名	住所	緯度	経度
2	セブン - イレブン	豊橋東脇店	愛知県豊橋市東脇1丁目21-9豊橋駅	34.75311	137.36491
3	酒店たばこ店ローソン	豊橋二川店	愛知県豊橋市大岩町字小山塚62-8二川駅	34.71898	137.44606
4	セブン - イレブン	豊橋高師本郷町店	愛知県豊橋市高師本郷町字竹ノ内71芦原駅	34.71629	137.39622
5	酒店ファミリーマート	豊橋江島店	愛知県豊橋市江島町141-1	34.73412	137.40749
6	ファミリーマート	豊橋柱八番町店	愛知県豊橋市柱八番町47愛知大学前駅 /	34.74447	137.3786
7	セブン - イレブン	豊橋草間町店	愛知県豊橋市草間町字平南7-1南栄駅	34.73641	137.37775
8	ローソン	豊橋老津店	愛知県豊橋市老津町字沖田73-3老津駅	34.69244	137.33023
9	セブン - イレブン	豊橋石巻本町店	愛知県豊橋市石巻本町字北市場45-6	34.80471	137.4413
10	セブン - イレブン	豊橋忠興店	愛知県豊橋市忠興1丁目5-16赤岩口駅 /	34.77465	137.43065
11	ファミリーマート	新豊橋駅店	愛知県豊橋市駅前大通1丁目138新豊橋駅	34.76205	137.38316
12	ローソン	豊橋天伯店	愛知県豊橋市天伯町字高田山57-1	34.71156	137.41431
13	ファミリーマート	豊橋大清水店	愛知県豊橋市大清水町字大清水238-3大清水駅	34.70345	137.35141
14	セブン - イレブン	豊橋北山町店	愛知県豊橋市北山町字黒浦7-42南栄駅 /	34.73682	137.39027
15	セブン - イレブン	豊橋神野新田町店	愛知県豊橋市神野新田町字ヌノ割38-1豊橋駅 /	34.73608	137.33665
16	セブン - イレブン	豊橋西岩田２丁目店	愛知県豊橋市西岩田2丁目1-1運動公園前駅	34.75443	137.4156

図 23: 変換した店舗の住所と緯度経度の情報

【課題 3】GeoPandas の使用

これまで Pandas ライブラリを利用し、DataFrame によるデータ解析を行った。地理空間情報を扱うには、対応する GeoPandas ライブラリを使うことになる。

```python
import pandas as pd
import geopandas as gpd
from shapely.geometry import Point

# CSV読み込み (encodingを明示)
file_path_in = r'.\data\toyohashi_conbini_with_geo_pending.csv'
df = pd.read_csv(file_path_in, encoding="cp932")  # または encoding="shift_jis"

# 緯度・経度の列名
lat_col = "緯度"
lon_col = "経度"

# GeoDataFrameに変換 (EPSG:4326 = WGS84)
gdf = gpd.GeoDataFrame(
    df,
    geometry=[Point(xy) for xy in zip(df[lon_col], df[lat_col])],
    crs="EPSG:4326"
)

# GeoJSONとして保存
file_path_out = r'.\data\toyohashi_conbini.geojson'
gdf.to_file(file_path_out, driver="GeoJSON")
```

図 24: csv から GeoPandas への読み込み

図 24 では、緯度経度付きの CSV ファイルから GeoPandas へ読み込みのコード示す。空間データ解析に欠かせない緯度経度と参照系の情報が取り入れていることが確認できる。

図 25 は、GeoPandas から書き出した DataFrame の構造を示し、その中、緯度経度と Geometry の地理空間情報が含まれている。

	系列名	店舗名	住所	緯度	経度	geometry
0	セブン - イレブン	豊橋東脇店	愛知県豊橋市東脇1丁目21-9豊橋駅	34.75311	137.36491	POINT (137.36491 34.75311)
1	酒店たばこ店ローソン	豊橋二川店	愛知県豊橋市大岩町字小山塚62-8二川駅	34.71898	137.44606	POINT (137.44606 34.71898)
2	セブン - イレブン	豊橋高師本郷町店	愛知県豊橋市高師本郷町字竹ノ内71芦原駅	34.71629	137.39622	POINT (137.39622 34.71629)
3	酒店ファミリーマート	豊橋江島店	愛知県豊橋市江島町141-1	34.73412	137.40749	POINT (137.40749 34.73412)
4	ファミリーマート	豊橋柱八番町店	愛知県豊橋市柱八番町47愛知大学前駅 /	34.74447	137.37860	POINT (137.3786 34.74447)

図 25: GeoPandas の DataFrame の模様

【課題 4】 GeoPandas の DataFrame から GeoJSON への変換

GIS 演習 I では、Shape File 形式の空間データを使ってマッピングをしたが、本実践では、図 26 の GeoJSON 形式の空間データを使用する。従来の Shape File 形式に比べ、GeoJSON 形式はコードの可読性とデータの軽量性の面で優れている。なお、GeoPandas から GeoJSON 形式へのデータ書き出しは、図 24 の下部に参考できる



図 26: GeoJson データの模様

QGIS ソフトウェアも GeoJSON 形式の空間データが対応しており、図 27 は完成した豊橋市コンビニエンスストアの分布図を示す。

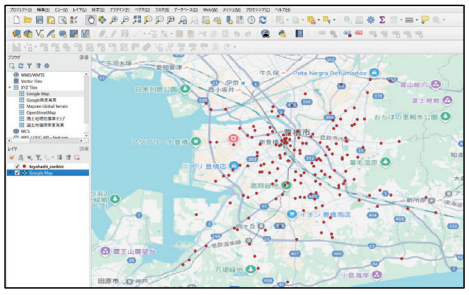


図 27: QGIS でマップ表現

6. まとめ

生成 AI 協働型の「データ分析」講義を試みた。教員と学生がともに生成 AI とのかかわり方を模索しながら、生成 AI 時代にふさわしい教え方と学び方を考え、実践を行った。

学生に対し、「学び方」を自ら学ぶことを目標として、強調した。その際、まず、自らやりたいことを明確にする。次は、自分やりたいことに対し、このどの部分、どのように生成 AI の助けを求めるかを考え、的確な指示を出す。生成 AI の役割は、自分目標により変わる。ときにはデータ分析やグラフをやってくれる「助手」になり、ときにはプログラミングコードを教える「教員」になる。生成 AI に対し、分析結果やグラフなどの「成果物」を求めるのだけではなく、研究シナリオを一緒に考えるなど、研究「パートナー」としての役割を求める。

教員は、学生と同じスタートラインに立っている。「教育者」よりは「同伴者」のように学生と接触すべき。学生たちは、多様な目標を持って生成 AI と向き合うので、教育の目標は多様化する。

これまで「情報系科目」のほとんどは、特定のスキルを取得するための科目であった。これまでの教室には、教員と学生しかいない。今後の生成 AI 時代において、状況の変化は予想できる。

一つ目は、講義に関わる「メンバー」の

変化である。生成 AI は講義の一員に加えることで、教員、学生、生成 AI 間の協働は、今後の講義進行の基調になる。もう一つはスキル学習の目標に関する考え方の変化である。学習目標の多様化に伴い、それにふさわしい教員の役割が求められる。

本寄稿執筆の際に、教育実践はまだ終わっていない。学生たちの感想を本稿に含まれないことは残念に思う。別の機会で皆さんの感想をまとめたい。

謝辞

教育実践に参加した 18 名の学生、並びに、多忙なスケジュールから抜けて教育実践に参加した龍昌治先生、駒木伸比古先生、佐藤弘隆先生にお礼申し上げる。複数の教員は、時間割制限のため、実践に参加できないものの、Moodle に公開された講義資料に目を通して、貴重なコメントを頂くことに謝意を表す。また、外部講師として、総合株式会社アクセンチュアの劉光博先生から、「AI 時代の働き方～今からできる即戦力になる準備」のご講演をいただくことにお礼申し上げる。

参考文献

1） Andy Kirk, “Data Visualization: A Handbook for Data Driven Design”, SAGE Publication, 2019.

2） イワタ ヨウスケ、「ChatGPT がよくわかる本」、秀和システム、2023.

3） 竹村貴也、「ChatGPT による Python プログラミング」、オーム社、2024.

4） 増田智明、「GitHub Copilot × Python 入門」、日経 BP、2025.

Moodle 4.5 および 5.x バージョンにおける主要機能の進化に関する報告

森野　誠之（Moodle サポート担当）

要旨

愛知大学（以下，本学）で導入している Moodle において，バージョン 4.5 から 5.1 に至るまでの主要な機能的変化を整理し，その特徴を報告する．長期サポート（Long-Term Support, LTS）版である Moodle 4.5 において導入された AI サブシステム，5.0 および 5.1 にかけて進められた AI 関連機能の拡充，教育コンテンツ再利用性向上のための機能，UI ／ UX 面の改良についてまとめる．Moodle は AI 技術を学習体験向上のための手段として統合する「人間中心」の方針を明確にしつつ，問題バンクのコース間共有や成績ペナルティ機能など，教育現場の具体的な要望に対応した改良を並行して進めていることが確認された．

キーワード： Moodle，学習管理システム（LMS），AI（人工知能），e ラーニング

1. はじめに

Moodle は，世界各国の教育機関および企業で利用されているオープンソースの学習管理システム（LMS）であり，活発な開発コミュニティによって継続的に改良が行われている．おおむね 6 か月ごとに新機能を含むメジャーバージョンアップが実施され，その間にバグ修正やセキュリティ対応を目的としたマイナーバージョンアップが提供されるという開発サイクルを採用している[1)]．

このように変化の速い開発体制のもとでは，システムの安定運用を重視する教育機関において，どのバージョンを採用するかが重要な検討事項となるが，本学では標準のサポート期間（約 1.5 年）よりも長いサポートが提供される長期サポート（Long-Term Support, LTS）版を採用している．

一方，近年は教育分野においても人工知能（AI：Artificial Intelligence）技術の活用が急速に進み，LMS に求められる役割も変化している．Moodle はこうした状況を踏まえ，2023 年にリリースされたバージョン 4.5（LTS）において，AI 機能を統合するための基盤となる「AI サブシステム」を導入した．以降，5.0，5.1 と連続して，AI 関連機能の強化とあわせて，教育現場の運用効率を高めるための実務的な機能改善が行われている．

本稿では，Moodle 4.5 から 5.1 に至る

までの主要な機能的進化について，① AI 機能の統合，② 教育コンテンツ再利用性の向上，③ UI ／ UX の改善，という三つの観点に着目して整理する．

2. Moodle 4.5（LTS）の位置づけと主要な変化 — AI 統合の基盤整備

Moodle 4.5 は，次期 LTS である 5.3 がリリースされる 2026 年 10 月まで，LTS として 2027 年 10 月までの長期サポートが提供される重要なバージョンである[1]．このバージョンでは，従来の機能改善に加えて，将来の AI 活用を見据えた基盤整備が大きな特徴となっている．

2.1. AI サブシステムの導入

Moodle 4.5 では，AI サブシステムが新設された[1]．これは，サードパーティプラグイン開発者が AI 関連機能を実装しやすくするための共通基盤であり，今後の AI 機能拡張を支える中核的な仕組みである．

主な特徴は以下のとおりである．

- **対応プロバイダ**
  リリース時点で，OpenAI と Microsoft Azure AI の 2 種類の AI プロバイダに対応している．サイト管理者は API キーを設定することで，これらの外部サービスと Moodle を連携させることができる[2]．
- **AI プレースメント**
  AI サブシステムを利用した標準機能として，次の三つの「プレースメント」が提供されている[2]．
  1. テキスト生成：プロンプト（指示文）に基づき，コース概要などの文章を生成する機能
  2. 画像生成：教材やコース内で利用する画像を自動生成する機能
  3. 要約：ページやブックリソースなどの長文コンテンツを自動的に要約する機能
- **エディタとの統合**
  これらの機能は，Moodle 標準の TinyMCE テキストエディタに新設されたアイコンから呼び出すことができる．そのため，教員や学習者は特別な技術的スキルを必要とせず，通常の編集操作の延長として AI 機能を利用できる設計となっている[1]．ただし，この機能は Moodle の管理画面で AI プロバイダの設定をしたときのみで，設定していない場合は表示されない．

さらに，AI 機能利用時には，ユーザが初回利用時に AI 利用ポリシーへ同意することが必須となっており，ガバナンス面への配慮も行われている[2]．

2.2. コース構造の柔軟性向上

コース内コンテンツをより階層的に整理したいという要望に対応して，「サブセクション」機能が新たに導入された[2)]．これにより，従来のセクション（章）の下位に，さらに細かい節を設けることが可能となる．既存コースへの影響を考慮し，本機能はデフォルトでは無効化されており，管理者が明示的に有効化することで利用できるようになっている．筆者がテスト環境で利用した限りではスマートフォンでは見づらく，サブセクションごとにユニークな URL を持つためにコース全体が見づらくなった印象である．

2.3. UI/UX および管理機能の改善

Moodle 4.5 では，教育現場での扱いやすさを高めるために，以下のような改善も行われている[2)]．

- 課題提出状況ページのレイアウト刷新（検索・フィルタ機能の強化，ヘッダ／フッタの固定表示 等）
- 課題の締切に関する自動通知機能の強化
- 標準録音・録画ツール RecordRTC における画面録画機能の追加

これらの改良は，Moodle 創設者である Martin Dougiamas が CEO を退任し，AI 研究開発に専念するという組織変更とも無関係ではない[1)]．同氏は，AI を単なる自動化の道具ではなく，学習体験を豊かにするための補助的ツールとして位置づける「人間中心アプローチ」を強調しており，4.5 での AI サブシステム導入は，その方針を具現化する第一歩と位置づけられる．課題機能の改善は，今までの問題点が解消されており，ユーザーメリットが大きいものの，RecordRTC に関してはスマートフォン中心の学生にとっては使うことはなさそうである．

3. Moodle 5.0 における機能強化とその方向性

Moodle 5.0 は LTS ではないものの，4.5 で整備された基盤の上に，教育現場の運用を変えうる複数の機能が追加されたバージョンである．松崎[3)]が指摘するように，本バージョンの特徴は「派手さ」よりも，現場の要望に応える堅実な機能強化と，本格的な AI 利用に向けた土台固めにある．

3.1. 教育コンテンツ再利用性の向上

大規模な運用環境では，コース数や問題数の増加に伴い，コンテンツの管理や再利用が課題となる．Moodle 5.0 では，この課題に対応するため，問題バンクに関する機能が拡張された．

- **問題バンクのコース間共有**
  従来は，コースをまたいだ問題共有には管理者権限による設定が必要であった．5.0 では，教員権限でコース単位の「問題バンク」を作成し，他コースから参照できるようになった[3) 4)]．たとえば，教員が自分専用の「問題バンクコース」を用意し，そこに問題を蓄積したうえで，担当する複数科目で共通に利用するといった運用が可能になる．
- **問題バンクの詳細なフィルタリング**
  問題数が増加した場合でも目的の問題を探しやすくするため，フィルタリング機能が強化された．タグだけでなく，ステータス，作成者，問題名，問題タイプ，本文などの条件で検索できるようになっている[3) 4)]．

3.2. パフォーマンスと運用効率の改善

- **小テストの事前作成機能**
  多数の受講者が同時に受験する小テストでは，開始時にサーバ負荷が集中することが課題となる．5.0 では，テスト開始前に小テストデータをあらかじめ生成しておく「事前作成機能」が追加され，受験開始時の負荷軽減が図られている[3) 4)]．
- **活動タイプ別表示機能**
  コースページに「活動」タブが追加され，小テスト，課題，フォーラムなどの活動タイプごとにコンテンツを一覧できるようになった[3)]．これにより，学生が特定の活動のみを続けて実施したり，教員が全課題をまとめて採点するといった作業が容易になる．
- **成績ペナルティ機能**
  提出遅延に対して自動的に減点を行う成績ペナルティ機能が導入された．管理者がルール（例：1 日遅延ごとに 30 % 減点）を定義し，教員は課題ごとに当該機能の有効／無効を選択できる[4)]．

3.3. AI 機能の拡張

- **ローカル LLM「Ollama」への対応**
  AI プロバイダとして，新たに Ollama がサポートされた．Ollama はローカル環境で大規模言語モデル（LLM）を動作させることが可能であり，成績情報など機微なデータを外部に送信せずに AI 機能を利用できる利点がある[4)]．これは，学習履歴に基づく個別フィードバックなど，プライバシーに配慮した高度な AI 活用の可能性を広げる．
- **「説明」機能の追加**
  既存の「要約」機能に加え，コンテンツ内容をより平易な言葉で説明する「説明」機能が追加された[3) 4)]．これは，学習者の理解を補助するための

新たな支援機能と位置づけられる．

- **管理者向け AI 利用レポート**
  AI 利用状況を可視化するレポート機能が導入され，誰がいつどの機能をどの程度利用したか（トークン量を含む）を確認できるようになった．また，AI ポリシー同意状況を一覧表示するレポートも用意されている[3)4)]．

3.4. 機能整理とシステム近代化

Moodle 5.0 では，利用頻度の低い活動モジュール（チャットや投票など）が標準インストールから外され，必要に応じてプラグインとして追加する形に整理された[3)]．また，従来のテキストエディタ Atto が廃止され，TinyMCE に一本化されるなど，システム全体の近代化も進められている[3)]．

4. Moodle 5.1 の最新動向

Moodle 5.1 は，5.0 までの変更点をさらに洗練させ，とくに教員および管理者の利用体験を高めるための細かな改善が加えられたバージョンである．

4.1. AI 機能の高度化

- **AI プロバイダの追加**
  5.1 では，新たに DeepSeek が AI プロバイダとして追加され，利用可能な選択肢がさらに拡大した[5)]．
- **コース・活動レベルでの AI ツール制御**
  教員はコース設定や各活動の設定画面から，当該コンテキストで許可する AI 機能（要約のみ許可しテキスト生成は無効にする等）を細かく制御できるようになった[5)]．これにより，教育目的に応じて AI の介入度合いを調整する運用が可能となる．

4.2. 教育者向け UI/UX の洗練

- **目的別に整理された活動チューザ**
  活動やリソースを追加する際の選択画面が，「評価」「協働」「コミュニケーション」などの目的別に分類される構成へと変更された[5)]．これにより，特に経験の浅い教員でも目的に合った機能を選択しやすくなっている．
- **活動概要ページの情報拡充**
  5.0 で導入された活動概要ページに表示される情報が拡張され，フォーラムではディスカッション数や未読投稿数，小テストでは受験者数や平均受験回数などが一覧されるようになった[5)]．コース全体の進捗把握が容易になると考えられる．
- **課題グレーディング画面の改善**
  課題採点画面において，採点パネル

（ルーブリック等）と提出物の表示領域の境界をドラッグしてリサイズできるようになり，詳細な評価基準を用いつつ効率的な採点作業が可能となった[5]．

4.3. 管理者・教員向け機能の改善

- **手動登録時の機能強化**
  学生を手動でコースに登録する際に，登録と同時に所属グループや登録開始日時を指定できるようになった．
- **ソフトウェアディレクトリ構造の変更**
  セキュリティと柔軟性向上を目的として，Moodle のディレクトリ構造が再編成された．一部の環境では，これに伴い Web サーバ設定の調整が必要となる場合がある．

5. 考察

Moodle 4.5 から 5.1 にかけての一連の変化を俯瞰すると，その方向性は大きく二点に整理できる．

第一に，AI 技術の教育への戦略的統合である．Moodle と AI の原則[6]に基づき，AI サブシステム導入を起点に，プロバイダの多様化（とくにローカル環境で動作する Ollama への対応）や，コース・活動レベルでの利用制御機能の充実が進められている．これは，Moodle が AI を単なる業務効率化ツールとしてではなく，学習体験の質的向上や個別化支援を実現するための中核的要素とみなしていることを示している．創設者 Martin Dougiamas 自らが AI 開発に注力している事実は，この方向性の重要性を裏づけるものである[1]．

第二に，教育現場のニーズに即した着実な機能改善である．問題バンクの共有機能，小テストの事前作成による負荷軽減，成績ペナルティ機能，活動チューザや活動概要ページの改善などは，いずれも教員から長年寄せられてきた要望に応える機能であると理解できる．Moodle は最新技術の導入と並行して，日常的な教育活動における具体的な不便を解消するための改良を継続しており，オープンソースプロジェクトとして健全な開発姿勢を維持していると考えられる．

今後の展望として，5.2 では LTI（Learning Tools Interoperability）の改善が予定されており[5]，次期 LTS である 5.3（2026 年 10 月リリース）に向けて，機能の安定化と拡充が進むと見込まれる[5]．とくに AI 関連機能は急速な変化が予想され，2025 年から 2026 年は「人間がどのような役割を担うべきか，どのような教育が必要か」[1]という根源的な問いに向き合うための一つの転換点となる可能性がある．

6. おわりに

本稿では，Moodle LMS のバージョン 4.5 から 5.1 にかけての主要な機能的進化について，AI 統合と現場ニーズへの対応という二つの観点から整理・考察した．LTS である 4.5 において AI 統合の基盤を整備し，5.0 で問題バンク共有機能などの実務的な機能強化を図り，5.1 で UI／UX をさらに洗練させるという流れは，Moodle が将来の教育のあり方を見据えながら，現在の教育現場における負担軽減にも配慮して開発を進めていることを示すものといえる．

Moodle の進化は，一つの LMS の機能強化にとどまらず，オープンソースコミュニティが技術革新と教育実践のあいだのギャップをいかに埋めていくか，という観点からも示唆に富む事例である．今後も継続的なバージョンアップを通じて，AI 技術との関係性や教育現場での活用方法がどのように変化していくかを注視していく必要がある．

参考文献

1）株式会社イーラーニング公式 YouTube チャンネル（2025）「AI-powered 'Moodle 4.5' – The Future and Possibilities of Education.【Moodle Evangelist's Explanation】」YouTube. https://www.youtube.com/watch?v=ryIlOqhd9dw

2）Moodle（2024）「What's new in Moodle 4.5 | Moodle Academy」YouTube. https://www.youtube.com/watch?v=n14R5PHsUcg

3）株式会社イーラーニング公式 YouTube チャンネル（2025）「Moodle LMS 5.0 AI 進化の堅実なプラットフォーム【Moodle エバンジェリストが解説】」YouTube. https://www.youtube.com/watch?v=ZXYIQaaMcRo

4）Moodle（2025）「What's new in Moodle 5.0 | Moodle Academy」YouTube. https://www.youtube.com/watch?v=LX5-6xyrkxQ

5）Moodle（2025）「What's New in Moodle 5.1 | Moodle Academy」YouTube. https://www.youtube.com/watch?v=ui8iQmKjeOI

6）Moodle's AI principles – Moodle https://moodle.com/moodle-and-our-ai-principles/

1. ICT委員会 会議報告

愛知大学情報メディアセンターの事業および運営は、ICT企画会議のもと、三校舎合同のICT委員会を設置し、豊橋および名古屋(車道メディアゾーン含む)情報メディアセンターの事業を推進する。（2024年10月から2025年9月まで）

2024年度
◇第6回10月3日
議題:
1. 補正予算申請について
2. 2025年度メディアセンター開室時間について
3. 豊橋校舎新棟移転後のPC教室について
4. 情報メディアゾーン利用内規の改正について

協議･報告:
1. 学生の推奨ノートパソコンの環境について
2. 2025年度実習室アンケートについて
3. 2025年度予算申請について

◇第7回11月28日
議題:
1. 2025年度情報メディアセンター開館カレンダーについて
2. 2025年度予算申請について
3. 豊橋校舎パソコン実習室(514、523教室)のPC整備について
4. 車道校舎パソコン実習室(K802、K804教室)のPC整備について
5. 大学ICT推進協議会（AXIES）の加入について

協議･報告:
1. 教育用ソフトウェア利用申請書について

2025年度
◇第1回6月26日
議題:
1. COM編集委員選出について
2. 情報セキュリティ部会委員選出について
3. 教職員メールシステム更新について

協議･報告:
1. 2024年度事業報告書について
2. 2025年度実習室アンケートについて
3. 2025年度予算申請について

◇第2回9月12日 (メール会議)
議題:
1. 名古屋校舎サブコア･エッジスイッチ更改について

２．情報メディアセンター主催行事

（2024年10月～2025年9月）

◆名古屋校舎

開　講　日	講　習　会　名	教室	参加人数
2024年10月16日(水)	Word講習会・卒論編	L712	３人
2024年10月28日(月)	Word講習会・卒論編	L710	２人
2024年11月12日(火)	Excel講習会・グラフ編	L710	４人
2024年11月19日(火)	Word講習会・卒論編	L712	１人
2024年12月18日(水)	Excel講習会・関数編	W401	１人
2025年１月21日(火)	Excel講習会・グラフ編	L713	１人
2025年１月30日(木)	Excel講習会・関数編	L713	１人
2025年４月４日(金)	新入生向けパソコン利用ガイダンス	W402	32人
2025年５月15日(木)	Word講習会・レポート編	L710	11人
2025年５月16日(金)	Word講習会・レポート編	L710	11人
2025年５月19日(月)	PowerPoint講習会	L712	７人
2025年５月28日(水)	Word講習会・レポート編	L711	２人
2025年６月17日(火)	PowerPoint講習会	L713	３人
2025年６月30日(月)	Word講習会・レポート編	L711	２人
2025年７月７日(月)	Word講習会・卒論編	L710	１人

◆豊橋校舎：主催行事なし

◆車道校舎：主催行事なし

3．2024年度Moodle（LMS）運営業務報告

1．Moodle講習会

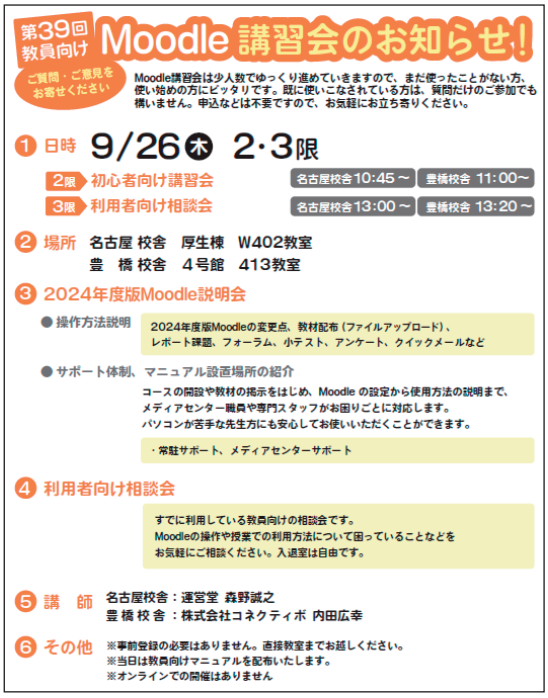
Moodleの利用促進および遠隔授業サポートのため，Moodle講習会を以下の通り実施した。

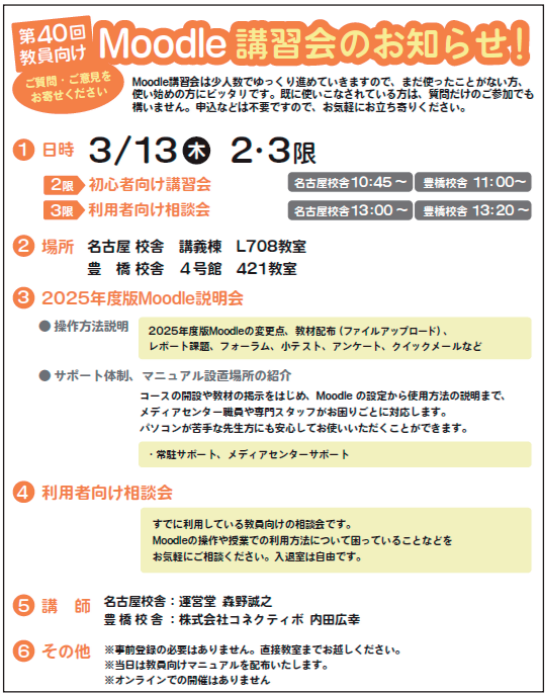
第39回Moodle講習会

校舎	開催日時	場所
豊橋	2024年9月26日 2・3限	4号館 413教室
名古屋		厚生棟W402教室

第40回Moodle講習会

校舎	開催日時	場所
豊橋	2025年3月13日 2・3限	4号館 421教室
名古屋		講義棟L708教室





2. Moodle 利用状況

（A）コース利用状況

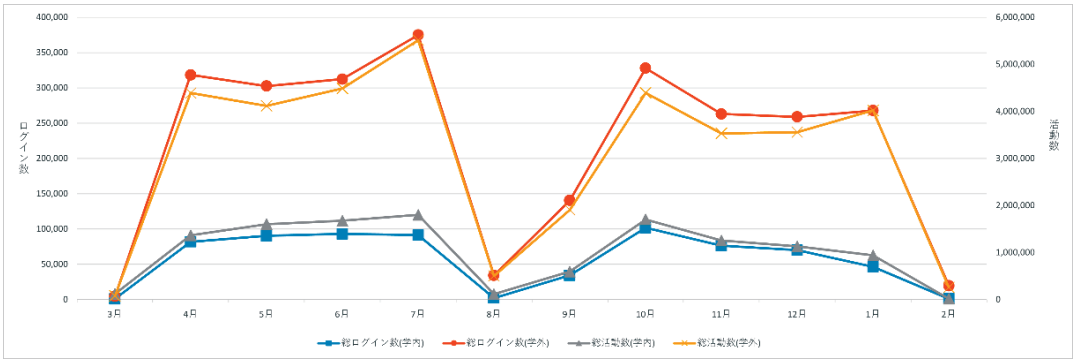
運用開始 16 年目の 2024 年度も，2023 年度に続き全ての授業についてコース作成をすることとなった。春学期・秋学期・通年の内訳は異なるが，コース合計は昨年と同程度の数となった。

カテゴリ	2024年度 利用コース数			
	春学期	秋学期	通年	合計
共通教育科目（名古屋）〈法・経済・経営・現中・国際〉	441	433	6	880
共通教育科目（豊橋）〈文・地域・短大〉	245	263	0	508
法学部	82	84	46	212
経済学部	69	73	59	201
経営学部	145	164	33	342
現代中国学部	183	140	7	330
国際コミュニケーション学部	203	171	11	385
文学部	231	220	22	473
地域政策学部	136	130	8	274
短期大学部	73	70	2	145
大学院	130	133	80	343
法科大学院	63	54	2	119
資格課程	87	81	21	189
協定留学生日本語コース	3	2	0	5
自習用教材	0	0	33	33
ヘルプ	0	0	3	3
その他	0	0	12	12
合計	2,091	2,018	345	4,454

（B）サイトアクセス状況

2022 年度から 2024 年度の総ログイン数の推移を確認すると，学内からのアクセス数が増加しており，総活動数も増加していることから，学内での授業利用及び自己学習での利用頻度が増えていることが分かる。

2024年度 学内・学外からのログイン数・活動数推移（月別）



		3月	4月	5月	6月	7月	8月	9月	10月	11月	12月	1月	2月	合計	平均
2022年度	総ログイン数(学内)	9,573	54,594	82,049	79,164	66,275	3,271	51,287	74,780	55,785	52,108	24,276	1,015	554,177	46,181
	総ログイン数(学外)	4,687	383,652	416,864	387,634	437,170	56,448	285,497	340,012	268,321	262,827	240,576	13,481	3,097,169	258,097
	総活動数(学内)	67,048	1,053,213	1,683,918	1,491,317	1,347,595	156,507	923,600	1,395,511	941,911	924,951	553,307	28,540	10,567,418	880,618
	総活動数(学外)	247,705	5,764,765	6,291,599	5,919,690	7,178,573	908,085	3,893,479	4,658,463	3,579,325	3,764,434	3,654,233	207,114	46,067,465	3,838,955
	ログインあたり活動数(学内)	7.00	19.29	20.52	18.84	20.33	47.85	18.01	18.66	16.88	17.75	22.79	28.12	19.07	21.34
	ログインあたり活動数(学外)	52.85	15.03	15.09	15.27	16.42	16.09	13.64	13.70	13.34	14.32	15.19	15.36	14.87	18.03
2023年度	総ログイン数(学内)	782	66,646	98,014	89,623	81,983	2,497	46,362	89,379	65,874	55,040	34,465	1,071	631,736	52,645
	総ログイン数(学外)	3,847	313,049	333,134	321,060	373,625	38,184	190,812	302,799	243,753	246,014	237,781	18,047	2,622,105	218,509
	総活動数(学内)	113,717	1,119,470	1,607,484	1,532,618	1,581,938	128,481	756,574	1,499,616	1,066,212	912,327	717,805	22,585	11,058,827	921,569
	総活動数(学外)	141,053	4,254,154	4,399,265	4,551,903	5,546,865	629,520	2,671,257	4,254,039	3,333,208	3,526,161	3,711,019	318,347	37,336,791	3,111,399
	ログインあたり活動数(学内)	145.42	16.80	16.40	17.10	19.30	51.45	16.32	16.78	16.19	16.58	20.83	21.09	17.51	31.19
	ログインあたり活動数(学外)	36.67	13.59	13.21	14.18	14.85	16.49	14.00	14.05	13.67	14.33	15.61	17.64	14.24	16.52
2024年度	総ログイン数(学内)	899	81,709	90,368	92,935	91,256	2,089	34,264	101,701	76,194	69,903	46,244	1,038	688,600	57,383
	総ログイン数(学外)	4,451	318,230	302,392	312,225	374,823	34,163	140,505	327,754	263,000	258,740	267,716	19,693	2,623,692	218,641
	総活動数(学内)	120,369	1,364,035	1,600,593	1,674,266	1,798,986	113,696	594,969	1,701,532	1,253,989	1,129,646	937,999	17,061	12,307,141	1,025,595
	総活動数(学外)	76,510	4,390,650	4,117,886	4,487,311	5,512,690	500,786	1,907,154	4,392,445	3,531,708	3,556,934	4,020,278	258,722	36,753,074	3,062,756
	ログインあたり活動数(学内)	133.89	16.69	17.71	18.02	19.71	54.43	17.36	16.73	16.46	16.16	20.28	16.44	17.87	30.32
	ログインあたり活動数(学外)	17.19	13.80	13.62	14.37	14.71	14.66	13.57	13.40	13.43	13.75	15.02	13.14	14.01	14.22
前年同月比	総ログイン数(学内)	115.0%	122.6%	92.2%	103.7%	111.3%	83.7%	73.9%	113.8%	115.7%	127.0%	134.2%	96.9%	109.0%	109.0%
	総ログイン数(学外)	115.7%	101.7%	90.8%	97.2%	100.3%	89.5%	73.6%	108.2%	107.9%	105.2%	112.6%	109.1%	100.1%	100.1%
	総活動数(学内)	105.8%	121.8%	99.6%	109.2%	113.7%	88.5%	78.6%	113.5%	117.6%	123.8%	130.7%	75.5%	111.3%	111.3%
	総活動数(学外)	54.2%	103.2%	93.6%	98.6%	99.4%	79.6%	71.4%	103.3%	106.0%	100.9%	108.3%	81.3%	98.4%	98.4%

4．ICT委員会構成員

◆ICT 委員 （2025 年 10 月 1 日現在）

役職名	所　　属	氏　　名
情報メディアセンター所長	経　営　学　部	岩田　員典
委　　員	文　　学　　部	横山　麻衣
	地 域 政 策 学 部	湯川　治敏
	短 期 大 学 部	迫田　耕作
	法　　学　　部	松井　吉光
	経　営　学　部	毛利　元昭
	現 代 中 国 学 部	吉川　　剛
	経　済　学　部	池森　　均
	国際コミュニケーション学部	梅垣　敦紀
	法 科 大 学 院	石口　　修
	情報システム課	石原有希子

◆情報メディアセンター事務室

情報 システム課	課　　長	石原有希子
	係　　長	伊神　真悟
	課　　員	岩田　大輝
		石川　彰吾
情報システム課 豊橋分室	係　　長	宮部　浩之

5．愛知大学　情報メディアセンター沿革・歴代所長

年度	組織		所長(任期)		システム沿革
			豊橋	名古屋	
1978					IBM製ホストコンピュータ4331　導入
1979					
1980	電子計算機センター	電子計算機センター委員会	津村　善郎 (1980.4.1～1982.4.30)		
1981					
1982					
1983			福田　治郎 (1982.5.1～1985.3.31)		
1984					
1985			高橋　正 (1985.4.1～1989.3.31)		
1986					
1987					
1988					第1期教育研究情報システム稼動　1988.4-1991.3
1989	情報処理センター	情報処理センター委員会 豊橋情報処理センター委員会 名古屋情報処理センター委員会	藤田　佳久 (1989.4.1～1994.9.30)	坂東　昌子 (1989.4.1～1990.9.30)	日立製ホストコンピュータ(HITAC M-640/20)導入
1990					
1991				浅野　俊夫 (1990.10.1～1992.9.30)	第2期教育研究情報システム稼動　1991.4-1994.3
1992					
1993				有澤　健治 (1992.10.1～1994.9.30)	
1994					第3期教育研究情報システム稼動　1994.10-1997.3 (全校舎学内LAN敷設)
1995			樋口　義治 (1994.10.1～1998.9.30)	長谷部　勝也 (1994.10.1～1998.9.30)	
1996					
1997					第4期教育研究情報システム稼動　1997.4-2000.9 (延長6ヶ月)
1998			宮沢　哲男 (1998.10.1～2000.3.31)	有澤　健治 (1998.10.1～2000.9.30)	
1999					
2000			小津　秀晴 (2000.4.1～2002.9.30)	田川　光照 (2000.10.1～2002.9.30)	10月　第5期教育研究情報システム稼動
2001					
2002			龍　昌治 (2002.10.1～2008.9.30)	坂東　昌子 (2002.10.1～2006.9.30)	
2003					
2004	情報処理センター	情報メディアセンター委員会 豊橋情報メディアセンター委員会 名古屋情報メディアセンター委員会			4月　第6期教育研究情報システム稼動
2005					
2006		情報メディアセンター運営会議 豊橋情報メディアセンター運営会議 名古屋情報メディアセンター運営会議		中尾　浩 (2006.10.1～2008.9.30)	
2007					
2008		ICT企画会議 豊橋ICT委員会 名古屋ICT委員会			4月　第7期教育研究情報システム稼動
2009			蒋　湧 (2008.10.1～2010.9.30)	伊藤　博文 (2008.10.1～2012.9.30)	
2010					
2011			沓掛　俊夫 (2010.10.1～2012.9.30)		
2012		ICT委員会			4月　新名古屋校舎システム稼働
2013			中尾　浩 (2012.10.1～2014.9.30)		
2014					
2015			松井　吉光 (2014.10.1～2018.9.30)		
2016					
2017					
2018					
2019			岩田　員典 (2018.10.1～)		
2020					
2021					
2022					
2023					
2024					
2025					9月　豊橋情報メディアセンター移転 (4号館 → Center.Bldg.2階)

編集後記

今回のCOMは記念すべき第50号です。私自身がCOMに携わるのはここ最近のことですが，長い歴史の中でこの特別なタイミングに編集に関わることができた幸運に感謝いたします。とともに，ここまで原稿をお寄せ下さった皆様と刊行を支えて下さった皆様すべてにお礼申し上げます。本当にどうも有難うございました。

今号のCOMには3本の論文と情報教育実践報告および研究ノートの各1本が掲載されています。論文では「自動採点システム」・「レポート分析」・「プログラミング実習用システム」など，教育上必要かつ不可欠で，実際の講義にも応用できる非常に有用な内容が含まれています。

また，情報教育実践報告では昨今の喫緊の課題である「生成AI」について，実際の講義を通じて教育上の具体的な向き合い方・付き合い方が非常に丁寧に纏められています。さらに，研究ノートでは本学のLMSであるMoodleの最近の発展について詳細に論じられています。教員として利用するだけではわからない開発側の意図も記載されています。

どの稿も，最近の情報科目教育および教育の中で必要な情報技術の知識だけでなく，具体的な設計・実装や実践を通じてわかる問題点と実際の教育現場で「学生にきちんとわかりやすく伝わるように」という意志やきめ細やかな配慮が根底にありました。今後の本学の教育に大きく寄与するものと思いますので，多くの人の目に留まることを期待します。

さて，このCOMは今号もLaTeXを用いて多くのページが編集されています。電子出版を意識してのことですが，出版校を実際に手元で見ながら原稿を作成できるというメリットは著者としてかなり大きいものと思います。そもそもTeXは開発者であるDonald E. Knuthが技術や芸術を意味するギリシア語「$\tau\acute{\epsilon}\chi\nu\eta$（テクネ）」から名付けたという話もあります。さらに，これに「〜学」を意味する「$\lambda o\gamma\acute{\iota}\alpha$」を組み合わせた言葉がテクノロジー(Technology)だそうです。編集を通してこのテクノロジーを如何に産み出し使いこなすかはあくまで人であることを改めて認識する機会なりました。

今後もCOMの発展のため皆様のご寄稿をお待ちしております。

(U.A.)

愛知大学情報メディアセンター紀要＜ＣＯＭ＞　原稿募集要項

情報メディアセンター紀要＜ＣＯＭ＞は、下記の要領で原稿を募集しています。詳細につきましては、情報メディアセンターまでお問い合わせください。

１．著者の資格

(1) 本学教職員および本学教職員との共著者
(2) 本学非常勤教員
(3) 本学学生（教員と共著とする。）
(4) 編集委員会が認めたもの

２．投稿原稿の内容

投稿原稿は未発表のもので、下記に関係する内容とする。
(1) 情報教育に関する理論と実践
(2) 情報科学や情報工学に関する理論とその応用
(3) 情報システムに関する調査、分析、理論
(4) コンピュータを活用した研究、教育、および業務等の実践報告
(5) 本学のコンピュータ利用に関して必要と思われる情報メディアセンターの報告
(6) その他（編集委員会が認めたもの）

３．投稿原稿の区分

投稿された原稿は編集委員会の審査に従って、下記のように区分して掲載する。ただし、法令等に抵触する、内容に著しい不備がある、執筆要項に従わないなどの問題があるものは、原稿の修正を依頼することや、掲載を見あわせることがある。
(1) 論文
(2) 研究ノート
(3) 情報教育実践報告
(4) 書評（新刊・古典・ソフトウェア）
(5) 学会動向
※ 原稿の体裁と見本については別紙を参照のこと。

４．原稿の提出要領

(1) 原稿は、電子ファイルで提出すること。
(2) 完成された投稿原稿のみを受理する。
(3) 提出する電子ファイル名は、投稿原稿のタイトルとすること。
(4) 図版等がある場合は、その電子ファイルもあわせて提出すること。
図版等のファイル形式は jpeg、png、pdf とする。
(5) 提出ファイルは、原則 Microsoft Word、テキスト形式、TeX 形式のいずれかとする。ただし、TeX を利用する場合は所定のクラスファイルを利用し pLaTeX あるいは upLaTeX によって作成した PDF ファイルと共にソースファイル一式を提出すること。また、その他の形式であっても編集委員会が認めた場合は受理する。
(6) 裏表紙（目次用）として、タイトル、著者名の欧文を添えること。
(7) 著者は連絡先（ゲラ等の送付先）の住所、電話番号を申し込み先の担当者まで連絡すること。

５．投稿原稿の体裁

投稿原稿は横書きとし、図・表などは適切な場所に分かりやすく挿入すること。なお、投稿原稿は COM 編集委員会にて共通したフォーマットに統一する。

6．校正

(1) 校正は著者校正を 2 回とする。

(2) 校正段階での内容の変更は、編集作業に支障をきたさない範囲で行なうこと。

7．著作権

(1) 提出された論文の著作権は、原則として愛知大学情報メディアセンターに属し、無断で複製あるいは転載することを禁じる。

(2) 論文作成に際して用いたコンピュータソフトや映像ソフト等の著作権に関する問題は、著者の責任において処理済みであること。他人の著作権の侵害、名誉毀損、その他の問題が生じないよう十分に配慮すること。

(3) 万一、執筆内容が第三者の著作権を侵害するなどの指摘がなされ、第三者に損害を与えた場合、著者がその責を負う。

(4) 著作人格権は著者に属する。

(5) 本誌に掲載された原稿は、学内においては、愛知大学情報メディアセンターホームページおよび愛知大学リポジトリにてデジタル公開するものとする。

(6) 本誌に掲載された原稿は、学外においては国立情報学研究所等へ登録される。

(7) 本誌に掲載された原稿の著者抄録（要旨・Abstract）は、国立研究開発法人科学技術振興機構における文献データベースに収録される。その際、第三者および人工知能（AI）に関連する利用について許諾する。

8．その他

(1) 別刷りは著者に対して希望を調査し、30 部を上限として無料進呈する。

(2) 著者には紀要を 2 部進呈する。ただし希望があれば 10 部を限度として進呈する。

以上

申し込み・問い合わせ：愛知大学情報メディアセンター
担当 ：情報システム課豊橋分室　宮部浩之
E-mail：johosystem@ml.aichi-u.ac.jp
TEL ：0532-47-4124(内線 1532)
FAX ：0532-47-4125(内線 1549)

2014 年 8 月 8 日改訂

（別紙）

愛知大学情報メディアセンター紀要＜COM＞ 執筆要項

1．執筆言語

和文もしくは英文とする。

2．原稿

(1) 論文・・・和文の場合は 30,000 文字程度、英文の場合は 15,000 words 程度を上限とする。ただし、図版等の数量に応じて調節すること。

(2) 研究ノート・・・和文の場合は 20,000 文字程度、英文の場合は 10,000 words 程度を上限とする。　ただし、図版等の数量に応じて調節すること。

(3) 情報教育実践報告・・・和文の場合は 20,000 文字程度、英文の場合は 10,000 words 程度を上限とする。　ただし、図版等の数量に応じて調節すること。

(4)書評（新刊・古典・ソフトウェア）・・・和文の場合は 5,000 文字程度、英文の場合は 3,000 words 程度を上限とする。書評（新刊・古典）には図版等を挿入することはできないが、ソフトウェアレビューについては若干の図版を添えることが出来る。

(5) 学会動向・・・COM のフォーマットに従う。

長文の場合は分裁や再提出等の措置を求めることがある。

3．著者と所属

著者名と所属を記載し、著者名のあとにカッコ（）に入れて所属を記載する。

4．セクションタイトルとセクション記号

本文中の章、節、項、目などの立て方は、原則として以下のとおりとする。

（例）

1.　章タイトル

1.1　節タイトル

1.1.1　項タイトル

(1)　目タイトル

5．図・表・写真

図・表・写真は、本文中の適当な箇所に挿入すること。または、挿入箇所を明確にすること。ただし、COM 編集委員会にて挿入位置、サイズを変更する場合があるが、変更不可の場合は明記のこと。

(1)表について

表の上部に「表○　表名」（○は表の一連番号）を記載すること。

(2)図・写真について

図・写真の下部に「図○　図名」（○は図の一連番号）または「写真○　写真名」（○は写真の一連番号）を記載すること。

６．要旨とキーワード

論文と研究ノートには要旨とキーワードをつける。要旨は 400 字以内（200words 以内）で執筆し、本文と同じ言語でもよいし、異なった言語でもよい。キーワードは国立情報学研究所の CiNii 等への正確な登録のために、５～７語程度のキーワードをつける。

７．謝辞

謝辞を記載する場合は、本文の最後に謝辞と小見出しを使い記載する。

８．注

注を記載する場合は、以下の方法による。

本文の末尾に後注として一括して記載する。本文の後に１行空けてから「注」という見出しを立て、その次の行から、注を一括して記載する。上記の場合、本文中の該当箇所には、番号と右丸括弧を使い注１）のように上付きで記すこと。

９．参考文献

参考文献の記載は、本文の後（注がある場合は注の後）に１行空けてから「参考文献」という見出しを立て、その次の行から、参考文献を一括して記載すること。本文中の該当箇所には、番号と右丸括弧を使い１）のように上付きで記すこと。

参考文献は原則として、雑誌の場合には、著者、標題、雑誌名、巻、号、ページ、発行年を、単行本の場合には、著者、書名、ページ数、発行所、発行年を、この順に記す。引用番号の記し方は本文上に出現した順番とし、次の例を参照にされたい。

（例）

1) 山田太郎：偏微分方程式の数値解法，情報処理，Vol.1，No.1，pp.6-10（1960）．
2) Feldman，J.and Gries，D.：Translater Writing System，Comm. ACM，Vol.11，No.2，pp.77-113（1968）．
3) 大山一夫：電子計算機，p.300，情報出版，東京（1991）．
4) Wilkes，M. V：Time Sharing Computer Systems，p.200，McDonald，New York（1990）.

以上

愛知大学情報メディアセンター紀要 COM〔コム〕
Vol.35 No.1 第50号

2026年７月３日 印刷
2026年７月10日 発行

編集 愛知大学情報メディアセンター
「COM」編集委員会

発行 愛知大学情報メディアセンター
（名古屋）名古屋市中村区平池町四丁目60-6
〒453-8777 TEL（052）564-6117（直通）
FAX（052）564-6217
（豊 橋）豊橋市町畑町1-1
〒441-8522 TEL（0532）47-4124（直通）
FAX（0532）47-4125
（車 道）名古屋市東区筒井二丁目10-31
〒461-8461 TEL（052）937-8120（直通）
FAX（052）937-8121

制作 山悦印刷株式会社

情報メディアセンター教育用パソコン　機種および設置台数

（2025年10月1日時点）

○豊橋校舎

設　置　場　所		機　　種	台数
Center.Bldg. 2F	12A情報ゼミ室	LIFEBOOK A5513/RX	24
	12B情報ゼミ室	LIFEBOOK A5513/RX	24
	121教室	ESPRIMO D7014/RX	60
	122教室	ESPRIMO D7014/RX	50
	123教室	ESPRIMO D7014/RX	50
	124教室	ESPRIMO D7014/RX	50
	メディアゾーン	ESPRIMO D7014/RX	62
		DAIV FX-I7G7A	2
5号館	523教室	Dynabook B65/HS	30
豊橋　計			344

○名古屋校舎

設　置　場　所		機　　種	台数
厚生棟4F	W401教室	ESPRIMO D7014/RX	60
	W402教室	ESPRIMO D7014/RX	60
	W403教室	ESPRIMO D7014/RX	60
	W404教室	ESPRIMO D7014/RX	60
	メディアゾーン	ESPRIMO D7014/RX	112
		Endervor MR8400	2
講義棟7F	L707教室	dynabook B65/HU	80
	L708教室	dynabook B65/HU	80
	L709教室	dynabook B65/HU	80
	L710教室	dynabook B65/HU	24
	L711教室	dynabook B65/HU	24
	L712教室	dynabook B65/HU	24
	L713教室	dynabook B65/HU	24
名古屋　計			690

○車道校舎

設　置　場　所	機　　種	台数
メディアゾーン	ESPRIMO D6011/H	6
車道　計		6

Journal of Aichi University Media Center
vol.35 No.1

CONTENTS

Preface ····· Director : Kazunori Iwata

Articles

Design and Implementation of a Time-Synchronized Automatic Grading System for Video-Based nstructional Materials ····· Kazunori Iwata 1

A Comparative Study of Text Similarity Metrics and Visualization Techniques Using Artificially Generated Student Reports ····· Kazunori Iwata 15

Development of Python Programming Training System with GUI Scoring Support Function Based on GUI Automatic Operation and Screenshots ····· Motoaki Mouri 33

Educational Practice in 'Data Analysis Theory' through Generative AI Collaboration: Targeting Students in the Faculty of Regional Policy ····· Yong Jiang 61

Notes ····· Seiji Morino 77

Miscellaneous ····· 85

Editorial ····· 92

Aichi University Media Center