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．

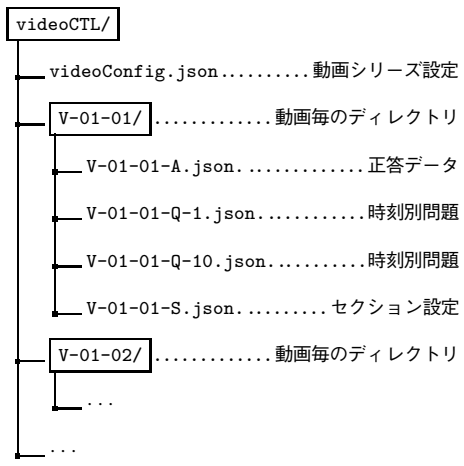


図 2: サーバに配置するファイルのディレクトリ構造

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}
  }
}
```

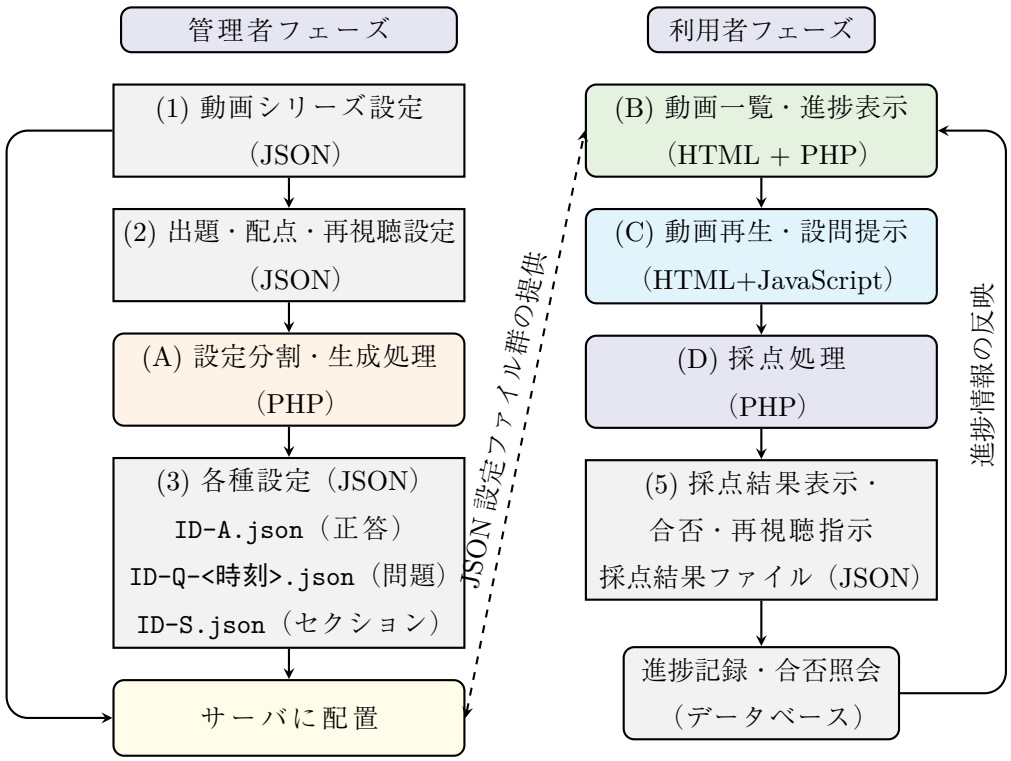


図 1: 動画教材における時刻連動型自動採点システムの全体構成

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．

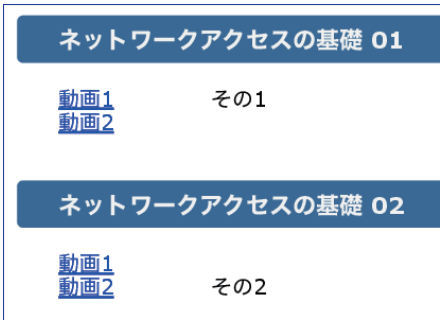


図 3: リスト 1 による動画一覧

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**