International Journal of Learning Technologies and Learning Environments
International Institute of Applied Informatics
2025, Vol. 8, No. 1, IJLTLE873

# Computerized Model of Assessment Procedures of Students' Reading Capabilities Based on Abridgement Method Employing Two-Pointer Problem Solution and Dynamic Programming

Koichi Akashi *, Hibiki Ito †, Kunihiko Takamatsu ‡,
Kenya Bannaka §, Shotaro Imai ‡, Syaka Matsumoto ‡,
Katsuhiko Murakami **, Yasuo Nakata §, Tetsuhiro Gozu §

## Abstract

There is an augmenting level of requirement for competent reading capabilities at various academic and societal settings today, which indicates a high demand for the school education system to employ the refined methodology to cultivate students' reading skills and assess them on a quantitative and objective basis. In a conventional fashion, summarization has been a norm amongst a multitude of educational approaches available, while that entailed a couple of drawbacks from the perspective of educators, such as the labor-intensive and time-consuming marking procedures, difficulty in establishing a standardized and less marker-dependent model answer, and challenges in marking them in a fair manner. To confront these issues, a new approach called *abridgement* has been recently proposed by researchers, allowing for simplification of the conventional procedures time-wise and operation-wise. Notwithstanding these revolutionary features, the manual marking process in abridgement is still daunting enough for the educators to make them spend several days on it. This research, therefore, focused on the automation of the whole procedures of marking of abridged texts of students in a computerized fashion, employing two possible approaches; two-pointer problem solution algorithm and dynamic programming.

*Keywords:* reading skills, abridgement, school education, dynamic programming

* University of St Andrews, St Andrews, United Kingdom
† University of Helsinki, Helsinki, Finland
‡ Institute of Science Tokyo, Tokyo, Japan
§ Kobe Tokiwa University, Kobe, Japan
** The University of Tokyo, Tokyo, Japan

# 1 Introduction

Ministry of Education, Culture, Sports, Science, and Technology (MEXT) of Japan published a renewed curriculum guideline in 2022, emphasizing the significant importance of development of the practical reading skills of students which would become quintessential in both academic and societal settings [1]. In fact, it was shown that the high level of reading comprehension skills is closely associated with academic success [2], and enriching literacy would lead to their abilities to choose, evaluate, and utilize information appropriately in a daily situation [3]. Likewise, the Education 2030 Project led by Organization for Economic Cooperation and Development (OECD) mentioned that creative and logical reasoning skills need to be enhanced to such a level that students' abilities can meet the objectives of academia and society, and specifically, the skills to write logically and read critically are to be considered consequential [4].

Despite the pronounced urgency and importance to cultivate students' reading skills, educators are faced with multiple struggles in achieving these goals. Primarily, their reading skills are prone to a very high extent of diversity and variability, causing educators to deal with a bunch of challenges in assessing and monitoring them, and to spend huge amount of effort in labor and in time to satisfy those [5]. Furthermore, their objectives include flagging those who may need to be monitored even more closely for extra support to allow them to catch up with an intended standard, such as students with learning disabilities, to which educators are also required to pay very close attention [6]. Additionally, they are advised to give practically useful feedback to each student to expect the highest value out of it [7], which is also a factor that counteracts the pursuit of efficiency in education. Currently, where a systematic methodology to resolve those issues is absent, each educational institution is straddling the border between speed and quality, and is having a problem in ensuring both [8].

Thus far, it has been widely accepted that an educational approach employing summarization would be one of the most effective practices amongst several possible available. Nevertheless, model answers are prone to a huge variability in quality and style since these factors majorly depend on the preference based on which they are constructed, and the skills of an individual [9]. For instance, the short passage "St Andrews, founded in the early 12th century in Scotland, is renowned for its ancient university, established in 1413, making it the third oldest in the English-speaking world. The town is also famous for its golfing history, with the Old Course considered a pilgrimage site for golf enthusiasts." can even generate various forms of summaries, such as "St Andrews in Scotland hosts an ancient university since 1413 which is third oldest English-speaking regions, and is famous for golf history.", "Founded 12th century, St Andrews houses third oldest university and iconic Old Course for golf.", "St Andrews is a Scottish town with historic 600-year-old university, golf pilgrimage site, Old Course, and 12th-century origins.", and so on. That variability also leads to the necessity of extensive endeavor to mark each submission on a standardized and quantitative basis, ensuring moderate fleetness to meet the satisfiable efficiency. Taking into account the temporal constraints of educators in school settings and the shortcomings of educational approach based on summarization, this research proposes the employment of recently advocated methodology called *abridgement*, initially rolled out by a Japanese linguist Susumu Ono [10].

In terms of operation, abridgement is totally equivalent to obtaining a subsequence from the given text; while a wide range of operations (*e.g.* addition, deletion, swapping, replacement) are allowed to be performed when it comes to summarization, it is only the deletion of some or no character(s) that is permitted in abridgement. Abridged text is a concatenation of undeleted characters in the original text. For instance, a word "golf" can derive several (but up to 16 in this case) subsequences, such as "gf" (taking the first and last character), "gol" (taking the first three characters), "o" (taking the second character), "" (taking no character), and "golf" (taking all the characters) (non-exhaustive). In other words, this process is identical to highlighting (or underlining) parts of the passage to build an abstract of the text provided. In doing so, there are only two caveats; one is that the resultant text has to be an abstract semantically, and the other is that this has to be grammatically valid. This sort of operation can be applied to a practice directly in Japanese language since a Japanese sentence, unlike English and most of the European languages, does not have a space between adjacent words. To give an example, Japanese text for "St Andrews is a birthplace of golf. Close to the link, there is a historical cathedral constructed over six-hundred years ago." is "セントアンドリュースはゴルフ発祥の地である。コースの近隣には六百年以上も前に建てられた大聖堂がある。". As shown here, there is even no space around the punctuations ("、" and "。"), thus a whole Japanese passage can be regarded as a huge single word, from the perspective of European language speakers. Operation of abridgement can be therefore performed on a Japanese sentence in the completely same way as it is performed on an English word. While abiding by the operational rules aforementioned, the procedure of abridgement requires students to extract the conceptions of the provided text, by reducing the volume to a certain level (*e.g*. 50%), letting abridged text serve as a proper abstract, also conforming to the grammatical rules at the same time.

As shown above, the operational constraint on abridgement is much tighter than summarization. This highlighted property of abridgement is expected to prevent students from being distracted by the factors other than extracting the essence of the provided text, since they are required to stick with the wordings already used in the text. Furthermore, the unique rules of abridgement require that students focus on the structure of the original passage, allowing for the opportunities for them to read passages more carefully than before. These traits of abridgement make it an innovative yet valuable approach for the cultivation of reading skills of students in education.

The nature and characteristics of abridgement delineated above furnish the method of abridgement itself with a couple of distinctions from the conventional modes such as summarization from the perspective of educators as well; the limited number of possible model answers, ease of implementing an automatic system of marking submissions that runs within a fully computerized fashion, and increase in clarity and objectivity in assessing students' reading comprehension. Besides, by attempting to implement the evaluation system based on abridgement method, our research team aimed at lifting the productivity of the module coordinators who exercise this methodology, leading to the augmented potential chance for them to spare their time for more meaningful tasks and analyses, including the statistical inspection of the class they are responsible for, and the extension of comprehensive support for those who may have difficulty in learning, as mentioned earlier.

This research principally zeroed in on the automation of the marking procedures of abridgement tasks, harnessing the techniques of two algorithms; solution for two-pointer problem and dynamic programming (DP). The former is the basic algorithm where two pointers literally traverse two texts respectively from left to right, and this allows us to see where each character in the second text came from in the first text. This method is a very intuitive approach since we also generally read the text from left to right. The latter, DP-based approach, is on the other hand a more innovative one as this proffers more flexibility when we have to handle the incorrectly abridged texts against which the first approach doesn't always work as we expect. Provided that they differ from each other notably not only in the way they are implemented but also in time/space complexity and ease of dealing with complicated linguistic structures when it is requested to extend the functionality, it is considered worthy to introduce both approaches here so that the readers may choose an optimal method depending on their own demands. Moreover, implemented functions include a program to display a handful of statistical information obtained from the submissions of students that could potentially be utilized for future analyses and research, and another feature to allow visualization of how well each student did on abridgement tasks by highlighting the submission in three different colors as a form of feedback automatically.

## 2 Methods

The sample data were collected at Kobe Tokiwa University, Japan. These data were approved by the Research Ethics Committee of the University after reviews concerning personal information and rights of students. The research team initially received the list of submissions each of which contained student IDs and their abridged text. Other sorts of information such as their names and e-mail addresses were all masked before being handed to the developers to respect the anonymity of students. Before moving onto the analysis and execution of the programs, variable notations were processed to be normalized (*i.e.* removed paragraph indices and unexpected blank spaces, converted full-width numerals to half-width (*e.g.* ７ → 7), and so on) so that the implemented function would not mistakenly judge the provided text as incorrectly abridged.

The initial step in two-pointer problem solution implementation was to figure out which indices in the original text were used to form an abridged text of each student. More formally, letting $S$ be an original text and $T$ be an abridged text submitted by a student, it was attempted to obtain a list of $i$ ($0 \le i < |S|$) such that $S[i]$ was taken as $T[j]$ for each $j$ ($0 \le j < |T|$), where $|X|$ denotes the length of the string $X$. This as a whole was implemented as a function named *original_indices(S, T)* shown below as *Algorithm 1*, and can be performed in $O(|S| + |T|)$ time and space complexity. In this implementation, the variable *track* is the pointer that traverses the string $S$, and $i$ is what traverses the string $T$. The variable *fixed* is used to record the initial position of $i$ for each of its traversals on the string $T$.

```
Function original_indices(S, T):
    Set Ls to the length of S
    Set Lt to the length of T
    Create a list ind of size Lt and initialize all positions to -1
    Initialize a variable track to 0
    For i in range [0, Lt):
        If track equals Ls:
            Exit the loop
        If T[i] equals S[track]:
            Set ind[i] to track
            Increment track by 1
        Else:
            Set the variable fixed to track
            While track < Ls:
                Set Boolean value flag to False
```

```
                    If T[i] equals S[track]:
                        Set ind[i] to track
                        Increment track by 1
                        Set flag to True
                        Exit the inner loop
                    Else:
                        Increment track by 1
                If track equals Ls and flag is True:
                    Set track to fixed
    Return the list ind
```

Algorithm 1: The function that finds the original indices in *S* for each character in *T*

Based on the implemented function *original_indices(S, T)*, the supplementary function *judge(S, T)* was also implemented to judge whether the student's submission was abiding by the rules of abridgement introduced earlier or not. Taking *S* and *T* as arguments, *judge(S, T)* returns 1 or *True* if *T* is the valid abridgement of *S*, and 0 or *False* otherwise, which is confirmed by any presence of -1 in the array of *original_indices(S, T)* as this indicates that the character in *T* came from nowhere in *S*. The algorithm is shown as *Algorithm 2* [11].

```
Function judge(S, T):
    If the returned list of original_indices(S, T) does NOT contain -1:
        Return True, indicating T is a correct abridgement of S
    Else:
        Return False
```

Algorithm 2: The function that judges if *T* is a correct abridgement of *S*

Alternatively, *original_indices(S, T)* and *judge(S, T)* in *Algorithm 1* and *Algorithm 2* can be implemented using a simple recursion. In particular, *Algorithm 2* can be implemented with significant simplicity, and this is shown as *Algorithm 3*. Be noted that when running *judge_recursive(S, T, S_ind, T_ind)*, the third and fourth arguments need to be both set to 0.

```
Set Ls to the length of S
Set Lt to the length of T
Function judge_recursive(S, T, S_ind, T_ind):
    If S_ind equals Ls:
        Return True, end the recursive search
    If T_ind equals Lt:
        Return True, end the recursive search
    If S[S_ind] equals T[T_ind]:
        Return judge_recursive(S, T, S_ind + 1, T_ind + 1), continue the search
    Else:
        Return judge_recursive(S, T, S_ind + 1, T_ind), continue the search
```

Algorithm 3: The function that judges if *T* is a correct abridgement of *S* using recursion

The other approach we propose is to utilize dynamic programming (DP), the algorithm specifically designed to tease out the longest common subsequence (LCS) of two given strings. LCS is the longest subsequence found in both two strings. For instance, LCS of "python" and "lyon" is "yon". LCS is known to be found in a following algorithm; construct a two-dimensional DP table *dp* ($dp[i][j]$ := the length of LCS of $S[:(i+1)]$ and $T[:(j+1)]$, where $X[:k]$ denotes the first $k$ letters in the string $X$) all values in which are initially set to 0. Also, for technical ease, it is advised to add a blank space in the beginning of each string before processing those (*e.g*. "python" → " python"). This table is known to satisfy the following transition rules [12]:

$$dp[i][j] = \begin{cases} \max(dp[i-1][j], dp[i][j-1], dp[i-1][j-1]+1) & \text{if } S[i] = T[j] \\ \max(dp[i-1][j], dp[i][j-1]) & \text{otherwise} \end{cases}$$

It is noteworthy that this DP table *dp* itself is used to identify the length of the LCS, as this is equal to $dp[Ls-1][Lt-1]$. In an example of "python" and "lyon", $dp[Ls-1][Lt-1] = 3$, which is the length of "yon".

Next, restoration of the LCS was attempted, and finding the list of indices that was also output in *Algorithm 1* can be achieved simultaneously during this operation. This is done by backtracking the DP table *dp* from bottom right to top left; letting $X$ be the row index of the attention, and $Y$ be the column index, the diagonal shift toward top left can be made if $S[X] = T[Y]$ satisfies, while decrementing both $X$ and $Y$ by 1. If it does not, the upward shift is performed when $dp[X-1][Y] \geq dp[X][Y-1]$ holds (be noted that when they equal to each other, the leftward shift can also be made, and this can be chosen arbitrarily, but close attention is required since the way it is implemented can affect the resulting LCS, while the length of this LCS is invariably equal to $dp[Ls-1][Lt-1]$), while decrementing only $X$ by 1. Otherwise, a leftward shift is performed, decrementing only $Y$ by 1. Construction of an LCS can be performed every time $S[X] = T[Y]$ holds, and it requires $S[X]$ ($= T[Y]$) to be added to the beginning of the string (In most programming languages, it is recommended that the character be added to the end of the string and then reverse the entire string all at once in the end to optimize, or use data structures such as deque in Python to allow for character addition to the front rather than to the back in constant time). Once LCS was obtained, index list was created by performing the procedures of two-pointer problem solution algorithm on the original text $S$ and LCS($S$, $T$), so that the left-most index amongst several possible can be recorded.

All those procedures are summarized as *original_indices_dp*($S$, $T$) in *Algorithm 4*, and can be performed in $O(|S||T|)$ time and space complexity [13].

---

```
Function original_indices_dp(S, T):
    Set Ls to the length of S
    Set Lt to the length of T
    Create a two-dimensional list dp (Ls × Lt), all elements initialized to 0
    For i in range [1, Ls):
        For j in range [1, Lt):
            If S[i] equals T[j]:
                Initialize a list cand, storing candidate values
                Append dp[i][j – 1] to cand
                Append dp [i – 1][j] to cand
                Append dp[i – 1][j – 1] + 1 to cand
                Set M to the maximal value in cand
                Set dp[i][j] to M
            Else:
                Initialize a list cand
                Append dp[i][j – 1] to cand
                Append dp[i – 1][j] to cand
                Set M to the maximal value in cand
                Set dp[i][j] to M
    Set X to Ls - 1
    Set Y to Lt - 1
    Create an empty string LCS
    While X > 0 and Y > 0:
        If S[X] equals T[Y]:
            Add S[X] to the end of LCS
            Decrement both X and Y by 1
        Else:
            If dp[X – 1][Y] ≥ dp[X][Y – 1]:
                Decrement X by 1
            Else:
                Decrement Y by 1
```

32. Reverse the entire string *LCS*
33. Pop the first blank space in *S*
34. Return *original_indices*(*S*, *LCS*) in *Algorithm 1*

---

Algorithm 4: The function that finds the original indices in *S* for each character in *T* during the process of obtaining LCS of *S* and *T*

This function is subject to various customizations. For example, it can serve identically to *judge*(*S, T*) in *Algorithm 2* by replacing the returned value by the Boolean value ?(*dp*[*Ls* – 1][*Lt* – 1] = *Lt*) because *T* is a valid abridgement of *S* if and only if the LCS of *S* and *T* is *T* itself.

It needs to be clearly emphasized here that the resulting list of indices produced by *original_indices*(*S*, *T*) in *Algorithm 1* may differ from that by *Algorithm 4*, as the latter aims at maximizing the length of the common subsequence of *S* and *T*, specifically in such a case that *T* is not a valid abridgement of *S*. For instance, when *S* = “golfer” and *T* = “grolf”, the former outputs [0, 5, -1, -1, -1], whereas the latter outputs [1, -1, 2, 3, 4], as LCS of *S* and *T* is “golf”. For this reason, as discussed later, special attention should be paid when handling incorrectly abridged texts.

In the meantime, the submission data were formatted for easy use for the functions implemented hitherto Receiving the list of information of students including student IDs and their abridged texst, along with the judgement of whether the submission abided by the abridgement rules or not made by *judge*(*S*, *T*) demonstrated in *Algorithm 2* or by an extension of *Algorithm 4* (0: invalid, 1: valid), all of them were stored in the two-dimensional list *abridgement_list* part of which is shown in *Figure 1*.

---

[

['MQ71844',

'選挙権を得られる年齢が 18 歳以上になったことから、少年法の適用年齢なども引き下げるべきか、法務省は検討した。だが、年齢要件のある法律が規制、保護する対象は多種多様で、目的も意義も異なり選挙権以外も揃える必要があるとは言えない。飲酒、喫煙は 20 歳未満は禁じられている。喫煙・飲酒と選挙権は何の関係もなく、健康や依存症の観点から検討するべきである。少年法の年齢引き下げも慎重に考えたい。少年法では、調査官が一人ひとりの家庭や成育環境を調べ犯行の原因を探り、少年院に入れる。適用年齢が 18 歳未満になると、18~19 歳の刑法に触れた人が矯正のための教育を受けずに社会に戻ることになる。少年犯罪は近年、大幅に減っている。刑罰と教育のどちらが社会の安全につながるのか。それぞれの法律が線を引く「成人年齢」については、様々なとらえ方があり、個別に検討していくべきだ。',

0],

['TT90145',

'選挙権を得られる年齢が 20 歳以上から 18 歳以上へと引き下げ、成人年齢や少年法の適用年齢なども引き下げる

か、議論を始めた。年齢要件のある法律が規制、保護する対象は目的も意義も異なる。機械的に判断していい問題ではない。分かりやすいのは飲酒、喫煙だ。選挙権は本来何の関係もない。健康や依存症などの観点から検討するのが筋だろう。少年法の年齢引き下げも慎重に考えたい。規定では、調査官や少年鑑別所が家庭や成育環境を調べ犯行の原因や背景を探る。適用年齢が 18 歳未満になると、刑法に触れた 18~19 歳の約 1 万人がこの対象から外れ、矯正教育を受けないまま社会に戻ることになる。少年院で保護処分を受けた方が再犯率が低い。刑罰と教育どちらが安全につながるのか。法律が線を引く成人年齢は、個別に検討していくべきだ。時間や手間はかかっても法規制の合理性や効果を分析しなければならない。',

1],

. . .

]

Figure 1: Example of elements in *abridgement_list*:
Student IDs were randomly altered for de-identification due to privacy reason.
For each element, the submitted text is associated with its submitter's student ID.

Now that students' submission data were all in a format accompanied by the judgements, visualization of the submissions by coloring them was the next step and done according to following rules:

(1) color a character chosen by both a model answer and a student's submission in pink
(2) color a character chosen only by a model answer in yellow
(3) color a character chosen only by a student's submission in turquoise

Also, the warning stating that the abridgement rule was not correctly followed was also attached to the end of the returned file as part of the feedback in case it was judged invalid by the function *judge(S, T)*. The score was also attached to the file, and was calculated in a following manner:

$$\text{score} := \frac{\text{the number of characters in pink}}{\text{the number of characters in a student's text}} \times 100$$

After coloring the submissions and adding relevant feedback statements, all the files were exported to a single folder *Marked*, and each file is named "[student ID].docx". All the procedures are summarized in a function *export_file*(*S*, *abridgement*, *student_id*, *validity*), where *S* is an original text, *abridgement* is a text that each student submitted, and *student_id* is their own student ID, shown in *Algorithm 5*:

```
Initialize a dictionary dic where students' scores with their IDs as keys are stored
Initialize a set correct to store and search indices that a model answer took
Function export_file(S, abridgement, student_id, validity):
    Initialize a set used_indices to store indices that a student submission chose
    For each element el in original_indices(S, abridgement):
        Append el to used_indices
    Create a new Microsoft Word document doc
    Add a new paragraph p0 to doc
    Initialize a counter variable cnt_pink to 0 to count the characters in pink
    For i in range [0, S);
        If i is contained both in correct and used_indices:
            Add S[i] to p0 with pink highlight
            Increment cnt_red by 1
        Else if i is contained only in used_indices:
            Add S[i] to p0 with turquoise highlight
        Else if i is contained only in correct:
            Add S[i] to p0 with yellow highlight
        Else:
            Add S[i] to p0 without any highlight
    Set L to the length of abridgement
    Set score to cnt_pink / L × 100
    Add a new paragraph p1 to doc
    Add "Score: [score]%" to p1
    If validity equals 0 (invalid):
        Add a new paragraph p2 to doc
        Set warning to "Note: Abridgement rule was not correctly followed."
        Add warning to p2
    Save doc and export to the folder Marked
```

Algorithm 5: The function that exports marked files to folder *Marked*

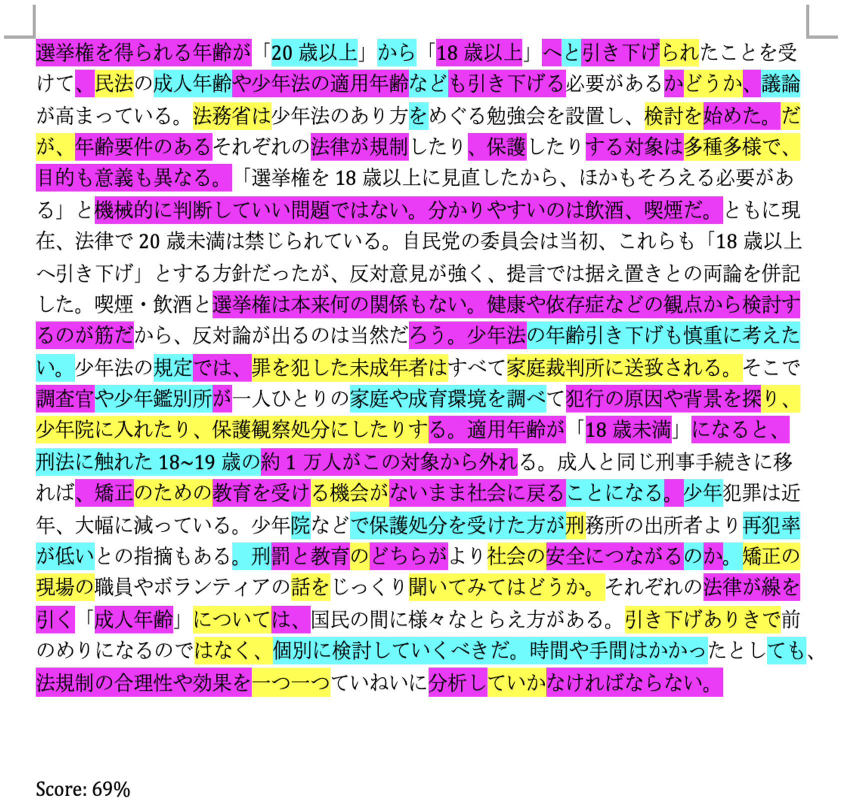
In implementation of *Algorithm 5*, dictionary *dic* and a set *correct* are initialized outside the function because the function itself is run multiple times on every single submission of students.

By executing *export_file*(*S*, *abridgement*, *student_id*, *validity*) on each one-dimensional list in *abridgement_list*, the complete set of desired files were obtained in the folder *Marked*. Additionally, it is advised to design the function so that it also outputs some statistical information such as the submission size, submission rate, non-adherence rate (i.e. the rate of students who did not abide by the rules of abridgement), mean, median, highest and lowest scores, and the pairs of students who are suspected of potential collusions.

## 3 Results

As a result of executing *export_file*(*S*, *abridgement*, *student_id*, *validity*) (based on two-pointer problem solution method) on each of 370 submissions received, marked files were successfully exported to the folder *Marked*. According to the obtained statistical information, 270 texts were the adequate abridgements, while 100 students did not abide by the specified rules (*i.e.* 270 were judged valid by *judge*(*S*, *T*) in *Algorithm 2*), leading to the non-adherence rate of 27.027%.

The general outlook of valid abridgement looked like *Figure 2(a)*.

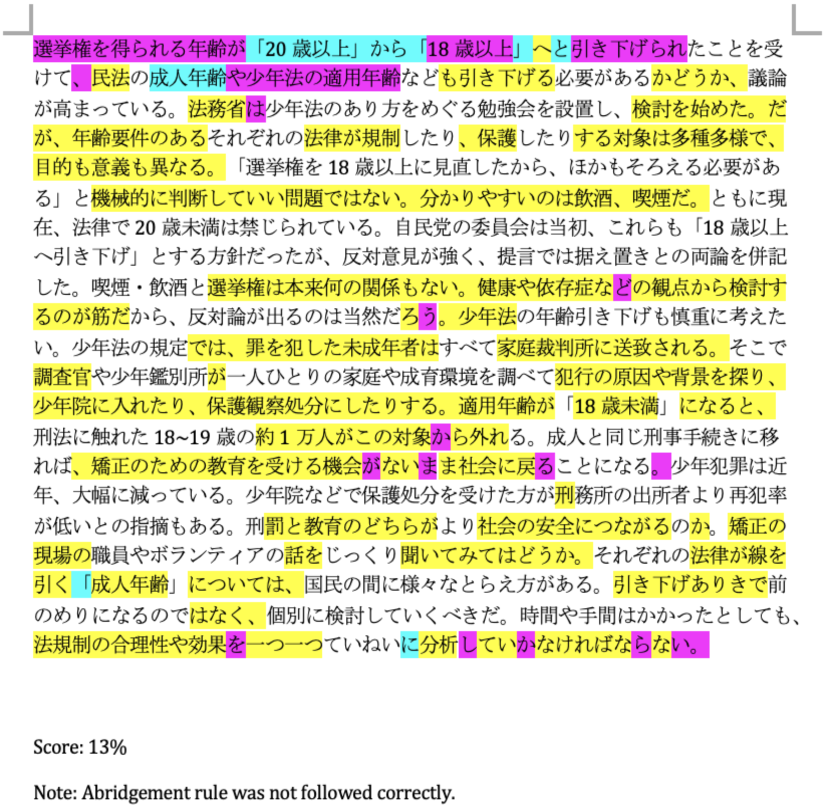


選挙権を得られる年齢が「20 歳以上」から「18 歳以上」へと引き下げられたことを受けて、民法の成人年齢や少年法の適用年齢なども引き下げる必要があるかどうか、議論が高まっている。法務省は少年法のあり方をめぐる勉強会を設置し、検討を始めた。だが、年齢要件のあるそれぞれの法律が規制したり、保護したりする対象は多種多様で、目的も意義も異なる。「選挙権を 18 歳以上に見直したから、ほかもそろえる必要がある」と機械的に判断していい問題ではない。分かりやすいのは飲酒、喫煙だ。ともに現在、法律で 20 歳未満は禁じられている。自民党の委員会は当初、これらも「18 歳以上へ引き下げ」とする方針だったが、反対意見が強く、提言では据え置きとの両論を併記した。喫煙・飲酒と選挙権は本来何の関係もない。健康や依存症などの観点から検討するのが筋だから、反対論が出るのは当然だろう。少年法の年齢引き下げも慎重に考えたい。少年法の規定では、罪を犯した未成年者はすべて家庭裁判所に送致される。そこで調査官や少年鑑別所が一人ひとりの家庭や成育環境を調べて犯行の原因や背景を探り、少年院に入れたり、保護観察処分にしたりする。適用年齢が「18 歳未満」になると、刑法に触れた 18~19 歳の約 1 万人がこの対象から外れる。成人と同じ刑事手続きに移れば、矯正のための教育を受ける機会がないまま社会に戻ることになる。少年犯罪は近年、大幅に減っている。少年院などで保護処分を受けた方が刑務所の出所者より再犯率が低いとの指摘もある。刑罰と教育のどちらがより社会の安全につながるのか。矯正の現場の職員やボランティアの話をじっくり聞いてみてはどうか。それぞれの法律が線を引く「成人年齢」については、国民の間に様々なとらえ方がある。引き下げありきで前のめりになるのではなく、個別に検討していくべきだ。時間や手間はかかったとしても、法規制の合理性や効果を一つ一つていねいに分析していかなければならない。

Score: 69%

Figure 2(a): An example of returned files of correctly abridged texts based on two-pointer problem solution algorithm: Pink denotes characters chosen both by student submission and model answer, turquoise denotes those chosen only by student submission, and yellow denotes those chosen only by model answer.

On the other hand, the general outlook of invalid abridgement looked like *Figure 2(b)*.



選挙権を得られる年齢が「20 歳以上」から「18 歳以上」へと引き下げられたことを受けて、民法の成人年齢や少年法の適用年齢なども引き下げる必要があるかどうか、議論が高まっている。法務省は少年法のあり方をめぐる勉強会を設置し、検討を始めた。だが、年齢要件のあるそれぞれの法律が規制したり、保護したりする対象は多種多様で、目的も意義も異なる。「選挙権を 18 歳以上に見直したから、ほかもそろえる必要がある」と機械的に判断していい問題ではない。分かりやすいのは飲酒、喫煙だ。ともに現在、法律で 20 歳未満は禁じられている。自民党の委員会は当初、これらも「18 歳以上へ引き下げ」とする方針だったが、反対意見が強く、提言では据え置きとの両論を併記した。喫煙・飲酒と選挙権は本来何の関係もない。健康や依存症などの観点から検討するのが筋だから、反対論が出るのは当然だろう。少年法の年齢引き下げも慎重に考えたい。少年法の規定では、罪を犯した未成年者はすべて家庭裁判所に送致される。そこで調査官や少年鑑別所が一人ひとりの家庭や成育環境を調べて犯行の原因や背景を探り、少年院に入れたり、保護観察処分にしたりする。適用年齢が「18 歳未満」になると、刑法に触れた 18~19 歳の約 1 万人がこの対象から外れる。成人と同じ刑事手続きに移れば、矯正のための教育を受ける機会がないまま社会に戻ることになる。少年犯罪は近年、大幅に減っている。少年院などで保護処分を受けた方が刑務所の出所者より再犯率が低いとの指摘もある。刑罰と教育のどちらがより社会の安全につながるのか。矯正の現場の職員やボランティアの話をじっくり聞いてみてはどうか。それぞれの法律が線を引く「成人年齢」については、国民の間に様々なとらえ方がある。引き下げありきで前のめりになるのではなく、個別に検討していくべきだ。時間や手間はかかったとしても、法規制の合理性や効果を一つ一つていねいに分析していかなければならない。

Score: 13%

Note: Abridgement rule was not followed correctly.

Figure 2(b): An example of returned files of incorrectly abridged texts based on two-pointer problem solution algorithm: A warning is attached to the end of the file as part of feedback.

However, especially in marking invalid submissions, a significant difference was observed between two-pointer problem solution approach and DP approach in the coloring pattern and the output scores. To provide a better insight, the same submission as *Figure 2(b)* is shown as *Figure 3*, after processing it with DP-based computation. In such a case, the returned file looked largely distinct from what it looked based on the approach of two-pointer problem solution, both in appearance and score. When it comes to correctly abridged texts, on the other hand, the score remained the same in both approaches, while the coloring pattern differed slightly from each other.

Regarding the statistics, obtained data from two approaches are compared in *Table 1*. With regard to the execution times, they were measured in the developer's environment (model: MacBook Pro 13, version: macOS 13.6.3), using Python 3.11.4.

選挙権を得られる年齢が「20 歳以上」から「18 歳以上」へと引き下げられたことを受けて、民法の成人年齢や少年法の適用年齢なども引き下げる必要があるかどうか、議論が高まっている。法務省は少年法のあり方をめぐる勉強会を設置し、検討を始めた。だが、年齢要件のあるそれぞれの法律が規制したり、保護したりする対象は多種多様で、目的も意義も異なる。「選挙権を 18 歳以上に見直したから、ほかもそろえる必要がある」と機械的に判断していい問題ではない。分かりやすいのは飲酒、喫煙だ。ともに現在、法律で 20 歳未満は禁じられている。自民党の委員会は当初、これらも「18 歳以上へ引き下げ」とする方針だったが、反対意見が強く、提言では据え置きとの両論を併記した。喫煙・飲酒と選挙権は本来何の関係もない。健康や依存症などの観点から検討するのが筋だから、反対論が出るのは当然だろう。少年法の年齢引き下げも慎重に考えたい。少年法の規定では、罪を犯した未成年者はすべて家庭裁判所に送致される。そこで調査官や少年鑑別所が一人ひとりの家庭や成育環境を調べて犯行の原因や背景を探り、少年院に入れたり、保護観察処分にしたりする。適用年齢が「18 歳未満」になると、刑法に触れた 18~19 歳の約 1 万人がこの対象から外れる。成人と同じ刑事手続きに移れば、矯正のための教育を受ける機会がないまま社会に戻ることになる。少年犯罪は近年、大幅に減っている。少年院などで保護処分を受けた方が刑務所の出所者より再犯率が低いとの指摘もある。刑罰と教育のどちらがより社会の安全につながるのか。矯正の現場の職員やボランティアの話をじっくり聞いてみてはどうか。それぞれの法律が線を引く「成人年齢」については、国民の間に様々なとらえ方がある。引き下げありきで前のめりになるのではなく、個別に検討していくべきだ。時間や手間はかかったとしても、法規制の合理性や効果を一つ一つていねいに分析していかなければならない。

Score: 64%

Note: Abridgement rule was not correctly followed.

Figure 3: An example of returned files of incorrectly abridged texts based on DP-based LCS approach of the same submission as *figure 2(a)* and *2(b)*: A warning was attached to the end of the file as part of feedback. Again, pink denotes characters chosen both by student submission and Model answer, turquoise denotes those chosen only by student submission, and yellow denotes those chosen only by model answer.

Table 1: Statistical comparison of two approaches

| Metric | Two-pointer Problem Solution | DP-LCS Algorithm |
|---|---|---|
| Execution Time (s) | 63.359 | 190.104 |
| Submission Rate (%) | 92.269 | 92.269 |
| Non-adherence Rate (%) | 27.027 | 27.027 |
| Actual Mean (%) | 64.292 | 75.843 |
| Actual Median (%) | 74.050 | 75.722 |
| Highest Score (%) | 100.000 | 100.000 |
| Lowest Score (%) | 0.926 | 46.543 |
| Collusion Suspected (pairs) | 7 | 7 |

A table containing a handful of statistical information regarding the submissions, and comparison of two approaches: While DP approach is more time-consuming, it did not cause incorrectly abridged texts to be given unjustly low scores, giving a more narrow mark distribution.

Lastly, the histogram of score distribution based on two-pointer problem solution approach was provided, shown as *Figure 4(a)*. The score distribution histogram based on DP-based LCS approach is shown as *Figure 4(b)*.

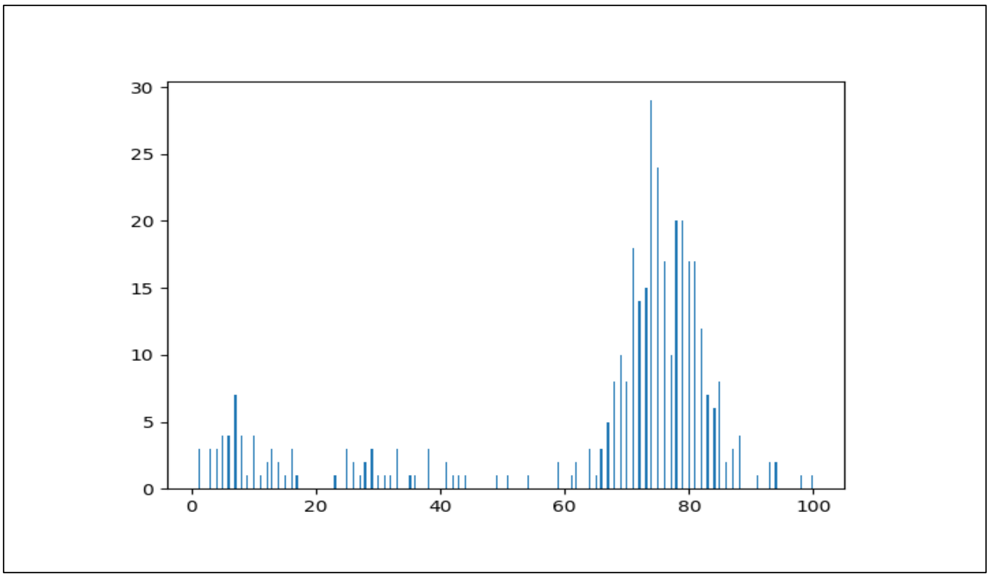



Figure 4(a): Score distribution histogram based on two-pointer problem solution algorithm: Observable clusters of students can be found on the left hand side, indicating the presence of submissions whose scores were made unjustly low because of inadequate abridgement.

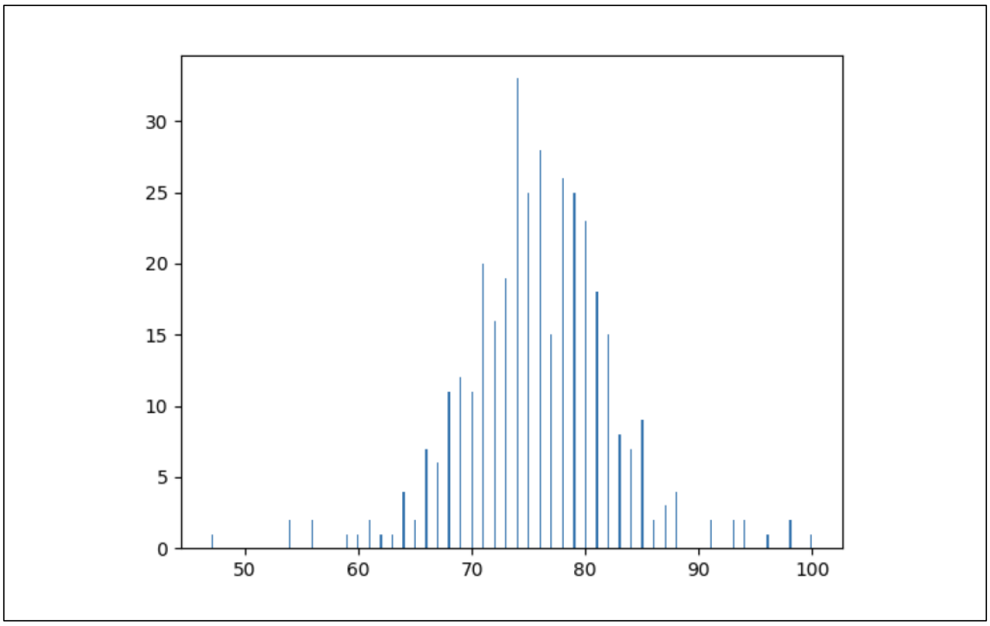



Figure 4(b): Score distribution histogram based on DP-LCS algorithm: Separate clusters observed in *Figure 4(a)* were all merged into one (hence the lower region is not shown).

## 4 Discussion

Above all, it was noteworthy that the marking procedures were fully automated from the point of receiving submissions from students to outputting the marked files with visually helpful feedback with a warning where necessary. In fact, there was a report to the research team that it used to take a module coordinator of Kobe Tokiwa University, where the experiment was conducted, approximately 15 hours to mark all the submission of around 400 students taking the module in a single year when this research was initially brought to the attention of the team, which was successfully reduced to a few minutes even in the worst case thanks to the endeavors above. Furthermore, parts of the functions implemented above provide educators with a handful of statistical information that could be utilized for future analyses, and potentially can help them identify those who may need to be monitored for extra support. It can be undoubtedly declared that this research laid a foundation to an enhanced efficiency and efficacy for speedy and standardized reading skill assessment with the deployment and actual operation of abridgement in various educational fields, all in a relatively simplified fashion.

This research also shed light upon the newly introduced methodology of dynamic programming (DP) that was ushered in the field of eduinformatics, notably LCS problem solution using DP that has been primarily focused on in the field of bioinformatics so far [14].

Regarding the differences observed between two approaches introduced here, however, the close attention turned out to be required. Despite the two methodologies giving highly similar results when it comes to marking adequately abridged texts, they gave the immensely different results on invalid submissions, as discussed and demonstrated in the previous chapter. The underlying reason for the different behavior can be even found in a simple example introduced earlier, in which two approaches return distinct index list where ($S$, $T$) = (“golfer”, "grolf"). The discussion is still at quite an early stage, and it leaves a room for further discussion whether to allow relatively high marks to be given to incorrectly abridged texts by taking DP-LCS method or to handle those submissions with severity by taking two-pointer problem solution algorithm. Opinions on the former side proposed within the research team included that students at an early stage in the module are prone to the inadequate techniques and errors in abridgment, and they may be deprived of motivation when they receive extremely low scores, whereas DP approach could be even useful as a "reference score" although not officially recorded. The nature of two-pointer problem solution algorithm, in fact, unjustly lowered the scores just because of a single addition or omission of the characters resulting in the indices of the large part of the text being not properly stored for score computation. Conversely, opinions on the latter side included that students may think that they do not necessarily have to conform to the abridgement rules perfectly because they can still receive acceptably high marks notwithstanding the inadequate techniques being exercised. To mitigate these opinions, one suggestion that could be made is to introduce small penalty for invalid abridgements (*e.g.* - 5marks) to DP-based approach so that this would not drastically punish the rule-breaking students but to discourage them to proceed without further attention to the specified rules of abridgement. At this stage, at least, it is proposed that the hybrid method of those approaches to maximize the educational effect be the most assistive.

Additionally, there could be a suggestion that educational institutions introduce other types of platforms to utilize abridgement as an educational approach, in light of high chance of errors in abridgement observed in this research. For instance, it is considered useful if the students just needed to highlight the passage with a marker, showing real-time change in the number of characters they have to cut off further on the screen, so that there is no possibility that an extra character is added or deleted.

As a whole, this paper has been able to demonstrate the relatively feasible implementation process of abridgement assessment system in real-world. Hundreds of submissions can be marked and returned along with statistics in a couple of minutes or less, despite the sufficient simplicity of implementation depicted in this paper above. This fact clearly emphasizes cost-effectiveness as a new system to introduce to the educational institutions, and is a strong indicator of relevance to the practitioners in general.

Abridgement is anticipated to play a very practical role in the field of education once deployed with care and packaged. In fact, the features of this technique are mostly what have been longed for by educators who have been forced to do lots of chores as part of their duties. In fact, this paper has shown a signpost of how this technology can reduce labors in assessing students, and its potential to direct their remaining resources to more productive works. Abridgement will teleport us on to the era of great efficiency in the field of education.

## 5 Conclusions

In spite of the pressing necessity to introduce an effective and efficient educational methodology to enhance students' reading skills at school settings, there are not many abundant useful approaches available at this stage that satisfy both speed and quality. Educators are facing strict temporal constraints and being highly prone to burnout [15], and efficiency is quintessential while maintaining the protected quality. To confront these issues, this research adopted abridgement method as a new standard to measure and cultivate students' reading comprehensions. At the very least, in comparison with the conventional methodologies such as summarization, implementation of the whole system was completed in a simplified fashion handy enough to duplicate in various settings, and this research was able to suggest another marking scheme employing dynamic programming to assist the quality of marking, specifically for submissions with inadequate techniques of abridgement. Desirably, some of the discussions that still remained unclear would attract more attention of the researchers in the relevant fields in the future, and also this educational approach accompanied by the implementation methods delineated above would be applied to other foreign languages by extensive researches, making them not only a standard in Japanese reading skills development but also a global standard in language educations.

## Acknowledgement

This work was supported by JSPS KAKENHI Grant Number 22K02731.

Implementations in *Algorithm 1* and *2* were introduced in the presentation of 15th International Congress on Advanced Applied Informatics (IIAI AAI 2023 Winter).

## References

[1] Ministry of Education, Culture, Sports, Science and Technology (MEXT), "On the new curriculum guidelines: An academic perspective (in Japanese)", 2018. Available at: https://www.mext.go.jp/b_menu/shingi/chousa/shisetu/044/shiryo/__icsFiles/afieldfile/2012/07/09/1405957_003.pdf (Accessed: 22nd Jan 2025).

[2] Cromley. J, "Reading achievement and science proficiency: International comparisons from the programme on international student assessment", Reading Psychology, Mar/Apr 2009, Vol. 30 Issue 2, 2009, pp3, pp89-118. doi: 10.1080/02702710802274903

[3] E. Iskandar, S. Amanah, A.V.S. Hubeis, and D. Sadono, "Factors affecting Internet literacy of Smallholder Cocoa Farmers", International Journal of Advanced Science and Technology, 10 April 2020, 29(5): pp4074-4084.

[4] OECD, "OECD future of education and skills 2030", 2015. Available at: https://www.oecd.org/education/2030-project/. (Accessed: 22nd Jan 2025).

[5] E.A. Kempf, "Teachers' perceptions about reading instruction in elementary inclusion classrooms," ProQuest LLC, Ed.D. Dissertation, Walden University, 2018.

[6] S. Özer Şanal, M. Erdem, "Overcoming reading disabilities among teenage students with learning disabilities through collaborative storybook intervention", Education and Information Technologies, August 2023, 28(8): pp9483-9507. doi: 10.1007/s10639-022-11378-w

[7] E.K. Swart, T.M.J. Nielen, and M.T. Sikkema-de Jong, "Does feedback targeting text comprehension trigger the use of reading strategies or changes in readers' attitudes?", Journal of Research in Reading, May 2022, 45(2): pp171-188. doi: 10.1111/1467-9817.12389

[8] H. Ito, K. Akashi, K. Takamatsu, S. Imai, S. Matsumoto, K. Murakami, and T. Gozu, "The character-based pyramid method and semi-automatic abridgement for scaling up the

abridgement learning method," in The Papers of Technical Meeting on "Information Systems" 2023, pp31–36.

[9] J. Li, "The role of reading and writing in summarization as an integrated task," Language Testing in Asia; 2014 May, pp4-19. doi: 10.1186/2229-0443-4-3

[10] S. Ono, "Japanese Practice Book" (*Nihongo Renshucho*, in Japanese), Iwanami Shoten, 1999. isbn: 9784004305965

[11] K. Akashi, H. Ito, K. Takamatsu, S. Imai, S. Matsumoto, K. Murakami, and T. Gozu, "Fully automated system for the marking and quantitative assessment of reading skills of students based on abridgement", 15th International Congress on Advanced Applied Informatics Winter (IIAI-AAI-Winter), January 2024, IEEE pp143-148. doi: 10.1109/IIAI-AAI-Winter61682.2023.00035

[12] T.P.Q. Nguyen, P.N.K. Phuc, C.-L. Yang, H. Sutrisno, B.-H. Luong, T.H.A. Le, and T.T. Nguyen, "Time-series anomaly detection using dynamic programming based longest common subsequence on sensor data", In Expert Systems With Applications, 1 March 2023, 213 Part A. doi: 10.1016/j.eswa.2022.118902

[13] K.-S. Huang, C.-B. Yang, K.-T. Tseng, Y.-H. Peng, and H.-Y. Ann, "Dynamic programming algorithms for the mosaic longest common subsequence problem", Information Processing Letters, 30 April 2007, 102(2-3): pp99-103. doi: 10.1016/j.ipl.2006.11.006

[14] R. Islam, Z.T. Asha, and R. Ahmed, "Longest common subsequence using chemical reaction optimization", 2015 2nd International Conference on Electrical Information and Communication Technologies (EICT) Electrical Information and Communication Technology (EICT), December 2015, pp29-33. doi: 10.1109/EICT.2015.7391917

[15] L. Dexter, "Teacher burnout: Measuring, contributing factors, and prevention," master's thesis, Bethel University, 2020.