Chung Innovation Challenge #1

## SUMMARY

| **Title** | Korean OCR |
| --- | --- |
| **Goal** | To develop an effective OCR technology for Korean language |
| **Permitted Technologies** | Participants may use:   * Their own custom-designed OCR models and software. * Open source libraries and frameworks released under permissive licenses, including MIT and Apache 2.0, or other OSI-approved licenses permitting commercial use. * All third-party source code must be properly credited, and any code copied or adapted from public repositories must comply with license requirements and attribution standards. * Submissions based on code or libraries with restrictive licenses (such as GPL) may not be eligible, depending on competition rules.   All final solutions must be submitted with documentation and appropriate licensing information, ensuring full compliance with the competition terms |
| **Participants** | Open to all HCMUT students  Students may form teams. A student may not join multiple teams. |
| **Scoring** | * 1st phase: accuracy compared to the original text * 2nd phase: accuracy and OCR speed (cost) * final phase: in-person presentation |
| **Timeline** | * Note: we will give them the dataset on 10 * september. * 1st phase: 21st September, from 8am to 12pm * 2nd phase: yyyy ~ yyyy * final phase: zzzz ~ zzzz |
| **Prizes** | The top 5 teams based on the 2nd phase scores will present their work to judges. And the prize winner will be decided by the judges.   * 1st prize winner: $500 * 2nd prize winner: $300 * 3rd prize winner: $100 * 2 consolation prize winner   Winners will receive certificates. |
| **Computing Resources** | * 1st phase: Participants will use free Google Colab provided by Kaggle. * 2nd phase: The top 10 teams based on 1st phase accuracy scores will receive $150 worth of dedicated GPU resources. ***The teams not in the top 10 may continue the competition to the final.*** |

## CONTENTS

[**SUMMARY 1**](#_mvw3igk4yox5)

[**CONTENTS 2**](#_luevf1q28vxm)

[**DATA 3**](#_xqzmiz33ln96)

[**EVALUATION 4**](#_9j7r2cygjmj4)

[Interpreting the Output 4](#_km1hunnekvnp)

[Interpreting Verbose Match 5](#_22h1l0sbpbao)

[Interpreting Verbose Un-match 6](#_5nyw71y53o6l)

## DATA

Image and text data for OCR technology development and evaluation

**Location**: <https://drive.google.com/drive/folders/1GC6YbbrWKfvMs6ILh0So-Q1wMJkcfIun?usp=drive_link>

**Files**:

1. ORIGINAL\_IMAGE.zip : images from several e-commerce sites
2. ORIGINAL\_TEXT.zip : text extracted from the above images
3. ocr\_eval\_20250903.py : python script that compares the 2 input files(original text file and ocr output text file) and calculate the accuracy of the ocr output

**How the data were prepared**:

The images are collected from several Korean e-commerce web sites by crawling the detail item information section of the product item web page, where each product was chosen randomly. Corresponding text is generated by gemini 2.5 Pro, Google’s multi-modal LLM. It may contain some errors.

**Korean e-commerce web sites**

1. OliveYoung: [www.oliveyoung.co.kr](http://www.oliveyoung.co.kr)
2. LFmall: [www.lfmall.co.kr](http://www.lfmall.co.kr)
3. Hyecho Travel: [www.hyecho.com](http://www.hyecho.com)

**Prompt used in Google gemini 2.5 pro to extract text within the image**:

| *첨부 이미지에 있는 모든 문자를 수정없이 적어주세요. 단, 작아서 읽기 어려운 문자, 제품 위에 직접 인쇄된 문자, 어워드 로고에 있는 문자와 웹사이트의 메뉴, 카테고리 탭과 같은 인터페이스(UI) 관련 문자는 제외해주세요. 이미지에 없는 문자도 제외해주세요* |
| --- |
| ***English Translation****: Write down all the text in the attached image without any modifications. However, exclude the following: text that is too small to read, text printed directly on the product, text within award logos, and user interface (UI) related text such as website menus and category tabs. Please also exclude any text that is not present in the image.* |

## EVALUATION

OCR output text is compared to the text in original To evaluate the OCR technology

Output of python script ( ocr\_eval\_20250903.py )

Explanation with example

car cr 1 edit distance

### **Interpreting the Output**

| ===== OCR Evaluation Results (v1.5.1) =====  FILES: Original File: DATA/texts\_hyecho/TJP201654\_01.txt.text, OCR File: DATA\_ABCStudio/hyecho/qwen2.5-vl-7b-instruct/TJP201654\_01.json.txt  TOTAL ERROR SCORE: 2047  ERROR SCORE BREAK DOWN : 29, 2003, 15  NO. OF CHARS : 1915, 2003, 15  NO. OF LINES : 72, 105, 2  NO. OF LINES IN FILE: 177, 102  NO. OF CHARS IN FILE: 3918, 1892 |
| --- |

Let's break down your example line by line:

* FILES: Original File: ..., OCR File: ... This simply confirms the two files that were compared.
* TOTAL ERROR SCORE: 2047 This is the final, most important metric. It's the sum of all detected errors and penalties. In this case, the total error is **2047**.
* ERROR SCORE BREAK DOWN : 29, 2003, 15 This line shows the three components that add up to the total score.
  + **29 (Matched Errors)**: This is the Levenshtein distance, or the number of single-character edits (insertions, deletions, substitutions), for all the lines that the script successfully paired between the two files. A low number here means the matched text was very accurate.
  + **2003 (Deletion Penalty)**: This is a penalty for all the content that was in the original file but completely missed by the OCR. The penalty is equal to the total number of characters in those missing lines.
  + **15 (Insertion Penalty)**: This is a penalty for content that the OCR produced but wasn't in the original text (e.g., hallucinations). The penalty equals the total number of characters in these extra lines.
  + **Calculation**: 29 + 2003 + 15 = 2047
* NO. OF CHARS : 1915, 2003, 15 This line shows the character counts corresponding to the error breakdown.
  + **1915**: The total number of characters in the original lines that were successfully **matched**.
  + **2003**: The total number of characters in the original lines that were **unmatched** (deletions).
  + **15**: The total number of characters in the OCR lines that were **unmatched** (insertions).
* NO. OF LINES : 72, 105, 2 This gives a similar breakdown, but for lines instead of characters.
  + **72**: The number of lines from the original file that were **matched**.
  + **105**: The number of lines from the original file that were **unmatched**.
  + **2**: The number of lines from the OCR output that were **unmatched**.
* NO. OF LINES IN FILE: 177, 102 These are the total line counts for each file after text normalization (like removing blank lines).
  + **177**: The original file has 177 lines. (Check: 72 matched + 105 unmatched = 177)
  + **102**: The OCR file has 102 lines.
* NO. OF CHARS IN FILE: 3918, 1892 These are the total character counts for each normalized file.
  + **3918**: The original file has 3918 characters. (Check: 1915 matched + 2003 unmatched = 3918)
  + **1892**: The OCR file has 1892 characters.

### **Interpreting Verbose Match**

With -v option, the python script outputs details of how original and ocr outputs are matched

| Match 13: Type=N-to-1, Error=0  Original (Indices: [34, 35])  L034: 라이쵸 산장을 지나 오야마 벳산을 거쳐  L035: 다시 라이쵸산장으로 돌아오는 서클 종주입니다  => Combined: "라이쵸 산장을 지나 오야마 벳산을 거쳐 다시 라이쵸산장으로 돌아오는 서클 종주입니다"  OCR (Indices: [18])  L018: 라이쵸 산장을 지나 오야마 벳산을 거쳐 다시 라이쵸산장으로 돌아오는 서클 종주입니다 |
| --- |

This block shows one specific pairing the algorithm decided was the best possible match for the lines involved.

* Match 13: Type=N-to-1, Error=0 This is the header for the match.
  + **Match 13**: A simple counter for the matches in the report.
  + **Type=N-to-1**: This is the most important piece of information here. It describes the structure of the match. N-to-1 means that **multiple (N) lines from the Original file were combined to match one (1) line from the OCR file**. This typically happens when an OCR engine correctly merges a sentence that was split across multiple lines in the source text.
  + **Error=0**: This is the Levenshtein distance between the combined original text and the OCR text. A score of **0 means it's a perfect match**, with no character differences.
* Original (Indices: [34, 35]) This section details the lines from the ground truth file.
  + **Indices: [34, 35]**: These are the specific line numbers from the original file that were used in this match.
  + **L034 & L035**: The script displays the exact text of lines 34 and 35.
  + **=> Combined**: This shows how the script concatenated the original lines to perform the comparison. This combined string is what was actually compared against the single OCR line.
* OCR (Indices: [18]) This section shows the corresponding line from the OCR output.
  + **Indices: [18]**: This is the line number from the OCR file.
  + **L018**: The script displays the exact text of the OCR line.

### **Interpreting Verbose Un-match**

| --- Unmatched Original Lines (Deletions) ---  L023: 다녀온 분들의 만족하는 트레킹 여정 (Penalty: 19)  L024: 모든 게 만족스러운 일정이었습니다 트레킹 코스 숙소 산장 가이드와 인솔자까지 다테야마 종주 상품은 북알프스 입문으로는 최고의 상품이라고 생각합니다 (Penalty: 81)  L127: a 다테야마 종주 트레킹은 산장 2연박으로 산장에 짐을 놓고 당일 트레킹에 필요한 짐만 배낭에 넣은 후 트레킹합니다 모든 짐을 들고 산장까지 걸어야 하며 마지막날은 모든 짐을 들고 트레킹 후 하산 합니다 2박에 필요한 짐과 보온의류 및 도시락 물 간식 등을 넣을 수 있는 공간이 필요합니다 30리터40리터 배낭을 준비해주시는 것이 좋습니다 산장에서 사용하지 않는 물건은 캐리어에 미리 넣어 오시면 산행 종료 후 찾을 수 있습니다 (Penalty: 239)  --- Unmatched OCR Lines (Insertions) ---  L017: 다테야마 트레킹 (Penalty: 8)  L023: 조도산 트레킹 (Penalty: 7) |
| --- |