Professional Portfolio

Jason Heesang Lee

List of Contents

- Timeline
- Career Transition
- Industry Partnership Project
- Kaggle & Competition Review

☆ChatDatePartner – Chatbot Project

Timeline

GLION INSTITUTE OF HIGHER EDUCATION

2015.08 ~ 2021.02

Bachelor of Business Administration
Hospitality Management
With Luxury Specialization

FOUR SEASONS HOTEL SEOUL

2021.04 ~ 2021.09

People & Culture
Coordinator

Apr

Atheneum

2021.09 ~ 2023.02

Sr. Associate

Jun

Team Lead

Aug

Fast campus × KOSME 중소벤처기업진흥공단

이어드림스쿨

스타트업-청년연계 YEAR-DREAM SCHOOL

2023.04 ~ 2023.12

First
Hello World!

Mar

First
Kaggle Submit

kaggle

Apr

Achieved
Notebooks Master

Oct

Released
DanGam

Dec

Fast campus

2024.03 ~ Current

Learning & Operations Manager

Mar

2020 2021 2022 2023 2024

Career Transition

FOUR SEASONS HOTEL SEOUL

After graduating from Glion Institute of Higher Education, a Hotel School located in Switzerland, I joined at Four Seasons Hotel Seoul as a People & Culture (P&C) Coordinator.

I was mostly responsible for the Human Resources tasks :
General Administration, Enrollment & Termination Interviews,
Organizing Employee Events and Job Fair.

Atheneum

I joined Atheneum Partners after being scouted by a director I met previously.

I was responsible for about 15 projects per month as a Project Manager.
Exceeded target every month by around 250%, best achievement being 425%.
Thankfully, I was able to be promoted as a Team Lead,
which was the fastest promotion in all global Atheneum offices.

Main Industries Covered : Artificial Intelligence, Semiconductor and Digital Transformation.

Fast campus × KOSME 중소벤처기업진흥공단

이어드림스쿨
스타트업-청년인재 YEAR-DREAM SCHOOL

Through the projects at Atheneum Partners, “Semiconductor Industry Benchmarking Project” and “Digital Transformation Project for Financial Institutions”,
I became interested in data and AI, and regularly looked for related news.

After started learning Python for the first time in March 2023, I have participated in a government funded AI bootcamp - YearDream School, to learn about data science.
At YearDream School, I have developed my skills through the Industry Partnership project and various competitions.

Based on the learning and experience, I aim to be at the forefront of the new era
through a continuous effort in the field of Artificial Intelligence.

YEAR DREAM SCHOOL

X

Avocadoland

Industry Partnership Project

We were required to work with a company, pre-selected by the Ministry of SME for the final project at YearDream School.

Among 12 different companies, I have decided to work with Avocadoland, a company that services an application named “Momory” where users record their daily lives and earn gems in return.

The purpose of this mobile application is to help self-diagnose the users’ emotional status.

Date : 2023. 11. ~ 2023.12
Contribution : 40%

Project Overview

이어드림스쿨 × Avocadoland

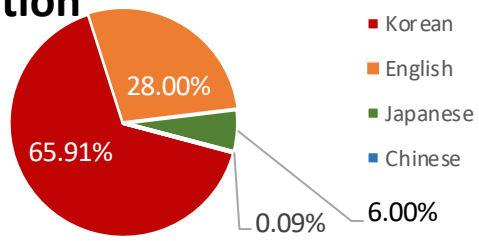
Task

Type of Data: Daily Records of users, recorded in 4 different languages.

Objective: 1. Identify the sentiment of each record.
2. Identify a list of keywords that is “acceptable” to the user.
3. Notate the most frequent keywords in the keyword list

Provided Codes: - Simple Text Preprocessing Code
- Sentiment Analysis with TweetNLP
- Unicode based Language Detection Code

Language Distribution



Questions

Below are the questions we had after taking a brief look at the data and codes.

Language Detection	Keyword Extraction	Improving Current Model	Building Model Evaluation Metric
Is there any error in language detection in the current code?	For Korean texts, Should we correct newly coined terms and indiscriminate abbreviations?	Is preprocessing reasonably done based on the needs?	How can we set the human evaluation standard when checking whether the extracted keywords are acceptable?
	How much weight should we put on each keyword based on the emotional state to make most users accept?	While our priority is to create more precise and explicit keyword extraction model, is there a way to make the model work efficiently and lightly?	Consider whether there are other metrics that could be objective criteria instead of the subjective criteria of ”acceptable"

Resolving Questions

Language Detection

Is there any error in language detection in the current code?
In the company's baseline code, language detection was done using a heuristic approach based on Unicode.
Therefore, it performed great for English and Korean but didn't work well for Chinese and Japanese.

How did we solve?
We have used a library called Lingua that work based on dictionaries, which was quite accurate.

Keyword Extraction

We were required to work on all the languages; however, we began with Korean, which took the largest portion of the entire data.

Should we correct newly coined terms and indiscriminate abbreviations?
The most frequent problem in Korean texts is that, as Korean words are combinations of vowels and consonants, there are hundreds of thousands of ways to modify the original form of the word, purposely or mistakenly.

How did we solve?
I have developed a module called JsonSpeller that maps unregistered words with registered words
The logic is explained in detail here.

How much weight should we put on each keyword based on the emotional state to make most users accept?
We have followed the weight predefined by a renowned Psychiatry lab in Seoul.

JsonSpeller: https://github.com/jasonheesanglee/JsonSpeller

Improving Current Model

Is preprocessing module in the baseline code reasonably done?
The baseline code showed how to retrieve the data from the server, but did not do much of preprocessing.

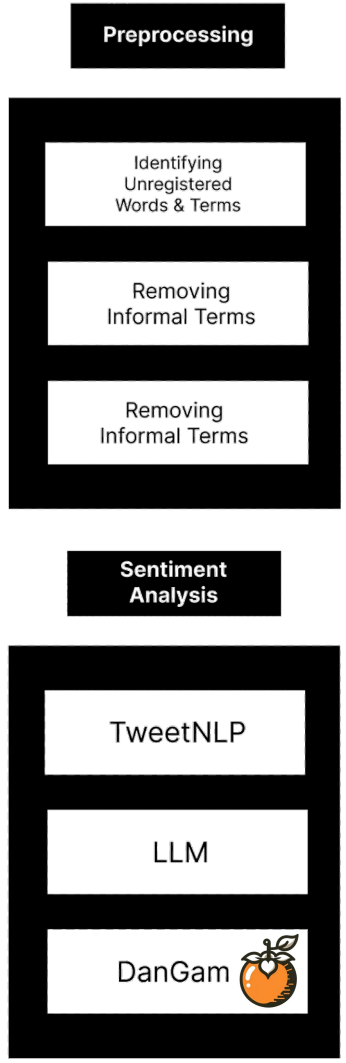
While our priority is to create more precise and explicit keyword extraction model, is there a way to make the model work efficiently and lightly?
TweetNLP was used as an example.
It was light and efficient, but it only performed sentiment analysis on sentence-level.
I have developed a brand-new package called **DanGam** for word-level sentiment analysis.
Its logic is briefly explained in the next slide.

Building Model Evaluation Metric

How can we set the human evaluation standard when checking whether the extracted keywords are acceptable?

Whether there are other metrics that could be objective criteria instead of the subjective criteria of "acceptable"

Unfortunately, these questions remain unsolved.
As each person has a different standard for the level of emotion, it is nearly impossible to create objective standards.



DanGam

이어드림스쿨 × Avocadoland

Compared with other existing research

- **TweetNLP**
  - TweetNLP supports multiple languages including Korean and works well at the sentence level. However, it does not support word-level sentiment analysis.
- **HuggingFace Text Classification Models**
  - Similar to TweetNLP, they support sentence-level sentiment analysis, but not word-level.
- **Word-Level Sentiment Analysis with Reinforcement Learning**
  - This research is similar to DanGam, but DanGam offers sentiment analysis for all the words in each sentence.
- **Word-Level Contextual Sentiment Analysis with Interpretability**
  - The result of this research research is similar to those of DanGam.
    However, DanGam is an inference tool, in contrast to a Deep-Learning Model that requires training..

How does it work

- DanGam takes a sentence as an input and identifies the general emotion as well as specific emotions within that sentence.
- DanGam calculates the cosine similarity between the sentence and the emotion, and between the sentence and the specific emotion.
- It combines sentence embedding, emotion embedding and specific emotion embedding with weights based on the calculated similarities.
- Then it calculates the cosine similarity between word embedding and the combined embedding.
- If the similarity is high, it suggests that the word has an emotion similar to that of the combined embedding.

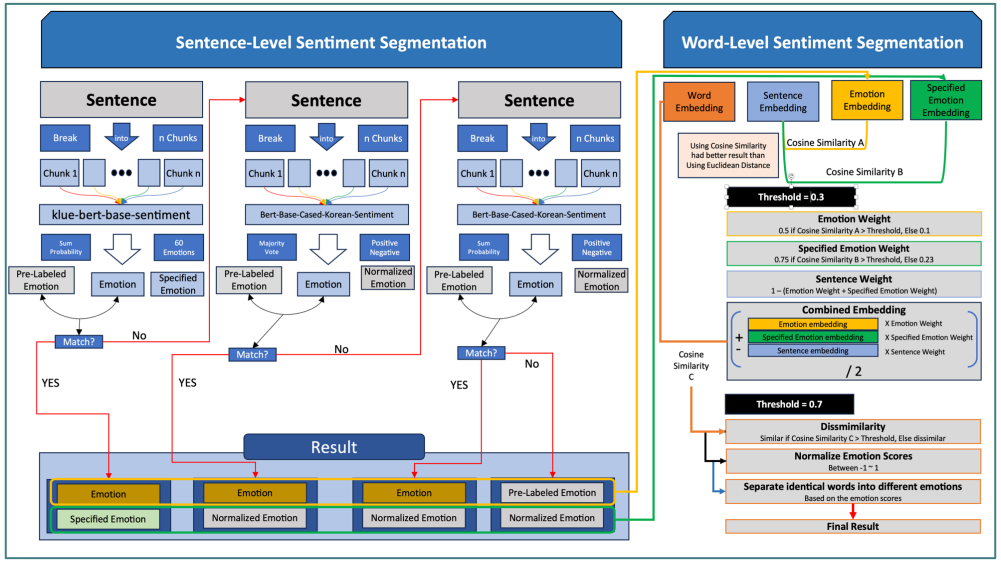
DanGam: https://github.com/jasonheesanglee/dangam

Output Example

- Example Sentence :
  "나는 방금 먹은 마라탕이 너무 좋다.
  적당한 양념에 알싸한 마라향이 미쳤다.
  그런데 고수는 진짜 싫다"
- The resulted output is in a range as below.
  Positive Emotion (1) ~ Negative Emotion (-1)

```
# {'나는': 1.0,
# '방금': 0.8419228076866834,
# '먹은': 1.0,
# '마라탕이': 0.8522973110543406,
# '너무': 1.0,
# '좋다': 1.0,
# '적당한': 0.965806179144829,
# '양념에': 0.7151325862316465,
# '알싸한': 0.4678710873322536,
# '마라향이': 0.328179239525493,
# '미쳤다': 0.34263925379014165,
# '그런데': -0.07491504014905744,
# '고수는': -0.7992964009024587,
# '진짜': -0.9295882226863167,
# '싫다': -0.9120299268217638}
```

Visualized Diagram



Final Output and Recap

이어드림스쿨 × Avocadoland

Final Output

"태형이와 시오니와 찬혁기, 배나영, 이희상과 함께 롯데월드에 갔다.
롯데월드 좋아졌더라. 마라탕을 먹고 오꼬노미야끼도 먹고, 탕후루도 먹었다.
나영이와 희상이가 싸우게 되어 분위기가 좀 그랬다."

	AL Baseline	Team's Preprocessing AL's Sentiment Analysis	Team's Preprocessing Team's Sentiment Analysis
Positive	롯데 0.6658 탕후루 0.5111 월드 0.4804	롯데월드 0.5291 시오니와 0.466 마라탕 0.4518 찬혁기 0.4106 탕후루 0.4104 배나영 0.3737 이희상 0.3714	마라탕 : 0.3355 오꼬노미야끼 : 0.2269
Neutral	분위기 0.0 상이 0.0 나영이 0.0	나영이 0.0 나영 0.0 희상 0.0 희상이 0.0 분위기 0.0	태형 : -0.1958 , 배나영 : -0.1226, 이희상 : 0.068, 롯데월드 : -0.0746, 탕후루 : -0.175, 나영 : -0.1226, 희상 : 0.068
Negative			나영 : -0.4526, 희상 : -0.3573, 분위기 : -0.6941

The final output was better than how we initially expected it to be.

The initial model did not distinguish well between positive and neutral words and even failed to distinguish words with negative emotions.

However, the new model provides a more accurate determination of each word's emotion. It also distinguishes negative words well.

Obstacles

As this was our first time collaborating with others as data scientists, it was quite hard to align our methods of approach.
For instance, some of the team wanted to simply count the number of keywords for the final display, while others wanted to calculate the level of impact that the keywords have on the overall context.
Moreover, lack of guidance from the company played a main role on our hardship.

However, with this experience, we learned a lot on how to collaborate as a data scientist, professionally exchanging our thoughts and opinions.

What could be improved

- When extracting keywords, there was too much information loss.
- There were some cases where there are investigations and other parts of speech, so if we had more time, we could have prevented this.
- We wanted to find a better way to stack words using the features of each morphological analyzer.
  We tried it based on a model named "Kkma", which decomposes the sentence into smaller parts, but it could have been better if we could try on other models as well.
- We tried Japanese and Chinese sentiment classification.
  However, if we had more time, we could have developed and introduced it.

Competitions

AI CONNECT

Generating Seamless En-Ko Translation

- **Date :** 2023-10-27 ~ 2023-11-08
- **Type of Data :** NLP / LLM / Machine Translation
- **Rank :** 2nd
- **Used Models :** Mistral & LLaMa2 based pretrained models
- **Focus :** Utilizing LLM models.

DACON

- **Judicial Precedent Prediction**
  - **Date :** 2023-06-05 ~ 2023-07-03
  - **Type :** NLP / Classification
  - **Rank :** Public : 15% / Private 18%
  - **Used Models :** Sentence-BERT / Legal-BERT
  - **Focus :** Getting familiar with Text data
- **Sound Emotion Recognition**
  - **Date :** 2023-05-07 ~ 2023-06-05
  - **Type :** Acoustic / Emotion Recognition / Classification
  - **Rank :** Public : 43% / Private 40%
  - **Used Models :** Librosa / RandomForest / DecisionTree / XGBoost / LightGBM
  - **Focus :** Getting familiar with Acoustic data

kaggle

Linking Writing Processes to Writing Quality

- **Date :** 2023-10-03 ~ 2024-01-10
- **Type :** Tabular / Classification
- **Rank :** Public : 17% / Private 10% / Top 176th
- **Used Models :** Rule-Based / XGBoost / TabPFN
- **Focus :** Reconstructing essays using given dataset.
- **Extra Achievement: 1st and 7th Place mentioned my Notebook**

CAFA 5 Protein Function Prediction

- **Date :** 2023-04-18 ~ 2023-12-21
- **Type of Data :** Tabular / Biology
- **Rank :** Public 4% / Private 4% / Top 63rd
- **Used Models : ProtBERT, Prot-T5, ESM2**.
- **Focus :** Solving given problem with Protein Language Models.

- **Kaggle – LLM Science Exam**
  - **Date :** 2023-07-12 ~ 2023-10-11
  - **Type of Data :** NLP / LLM / Question Answering
  - **Rank :** Public : 15% / Private 15%
  - **Used Models :** T5, DeBERTa, LLaMA2, Platypus2, Alpaca
  - **Focus :** Getting familiar with Large Language Models.
- **CommonLit – Evaluate Student Summaries**
  - **Date :** 2023-07-13 ~ 2023-10-12
  - **Type of Data :** NLP / LLM / Text Summary Evaluation
  - **Rank :** Public : 7% / Private 30%
  - **Used Models :** MobileBERT, DeBERTa, Numerous BERT family models
  - **Focus :** Transformer-based Deep Learning model compression
- **ICR – Identifying Age-Related Conditions**
  - **Date :** 2023-05-12 ~ 2023-08-11
  - **Type :** Tabular / Classification
  - **Rank :** Public : 7% / Private 48%
  - **Used Models :** Rule-Based / TabPFN / XGBoost / LightGBM
  - **Focus :** Finding relations between each column and the meta data

Evaluate Student Summaries

Purpose

Participated in the CommonLit – Evaluating Student Summary Competition, where the objective was to develop an automated system for evaluating the quality of summaries written by students in grades 3-12.

This involved building a Deep Learning model capable of assessing how effectively a student captures the main idea and details of a prompt text, along with the clarity, precision, and fluency of their written summary.
A comprehensive dataset of real student summaries were given to the participants to train and refine the model.

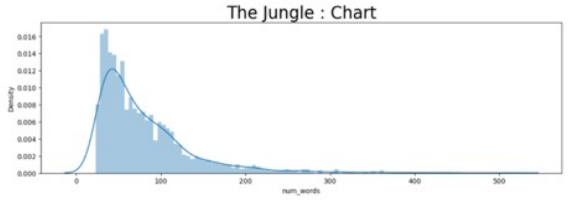
Competition Review

Date : 2023. 07. ~ 2023. 10.
Contribution : 60%
Link : Github / PyPi

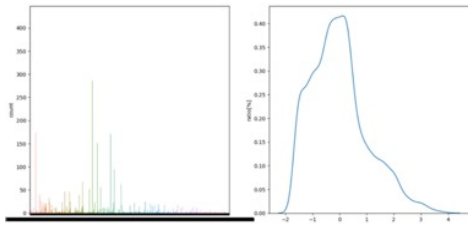
Reviews on other competitions can be found from below.
https://github.com/jasonheesanglee/kaggle

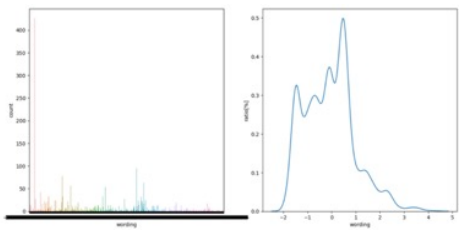
Data Overview

Exploratory Data Analysis



	num_words	num_unique_words	num_chars	num_stopwords	num_punctuations	num_words_upper	num_words_title	mean_word_len
count	1996.0000	1996.0000	1996.0000	1996.0000	1996.0000	1996.0000	1996.0000	1996.0000
mean	79.4509	54.5040	420.7560	42.1242	10.3472	0.0521	3.6844	4.2630
std	55.6905	30.2666	302.9265	29.4349	9.2133	0.3155	3.3914	0.3110
min	23.0000	18.0000	115.0000	8.0000	0.0000	0.0000	0.0000	3.2581
25%	41.0000	32.0000	214.0000	22.0000	4.0000	0.0000	2.0000	4.0522
50%	63.0000	47.0000	330.0000	33.0000	8.0000	0.0000	3.0000	4.2500
75%	100.0000	68.0000	527.2500	53.2500	14.0000	0.0000	5.0000	4.4458
max	509.0000	252.0000	2775.0000	257.0000	88.0000	8.0000	30.0000	5.8947






```
Train content description:
count    7165.000000
mean       -0.014853
std         1.043569
min        -1.729859
25%        -0.799545
50%        -0.093814
75%         0.499660
max         3.900326
Name: content, dtype: float64
```

```
Train wording description:
count    7165.000000
mean       -0.063072
std         1.036048
min        -1.962614
25%        -0.872720
50%        -0.081769
75%         0.503833
max         4.310693
Name: wording, dtype: float64
```

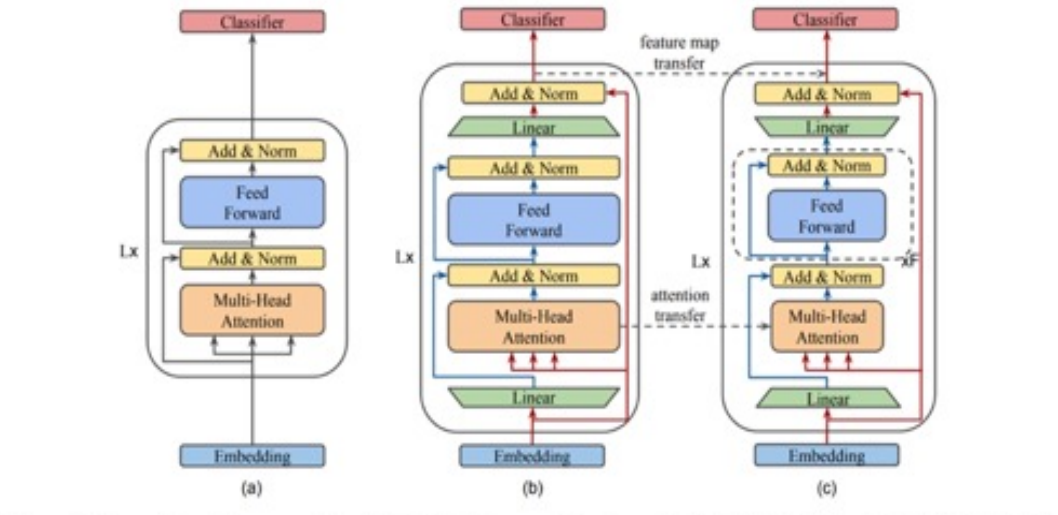
Preprocessing & Spellcheck

- Cleansing the text is a must to run the NLP model effectively
- Many misspelled words were in text:
  ex) “Pharaoh” -> “Pharoh” or “Pharoah”.
- Accuracy : SymSpellPy > PySpellChecker > TextBlob > AutoCorrect
- Speed : AutoCorrect > PySpellChecker > SymSpellPy > TextBlob

Tool	Detected	Correctly Replaced	Misjudged	Time
Grammarly	16 words	16 words	Less 1 word	-
TextBlob	20 words	14 words	2 words 2 names	4 sec
PySpellChecker	18 words	16 words	2 words	1 sec
SymSpellPy	18 words	17 words	1 name	2 sec
AutoCorrect	15 words	12 words	2 words 1 name	0 sec

Model Application

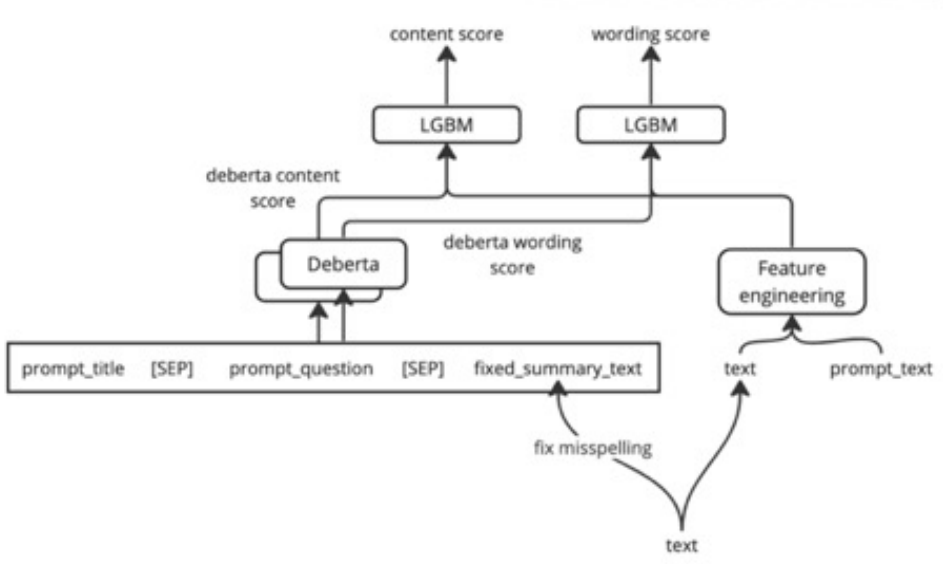
Solving a Real-World Problem with MobileBERT



MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices (Sun et al., 2020)

- As the purpose of this competition is to enhance the teaching and scoring experience of the teachers, I believed making this model lightweight is the key.
- By implementing MobileBERT, the model would run on individual mobile devices, making it accessible to teachers in environments where computer access is difficult.

LGBM – Post-Processing



- After getting the result with NLP Deep Learning models, I used LightGBM to improve the score by feature engineering, as introduced in this notebook.
- The original author of the notebook found the hyperparameters through experimentation but added Optuna, a hyperparameter optimization module, as a last step, thinking that it would find better-performing hyperparameters.

Result & Reflection

Result

- We were placed at the top 143 in the Public Leaderboard, so we believed that we could make it to the Bronze medal when the Private Leaderboard was published.
- Therefore, we tried our best until the very end to make minor changes to our existing solution.

143	Jason Heesang Lee	0.44061	274	12d	

- When the Private Leaderboard was published, we were pretty shocked with the result.

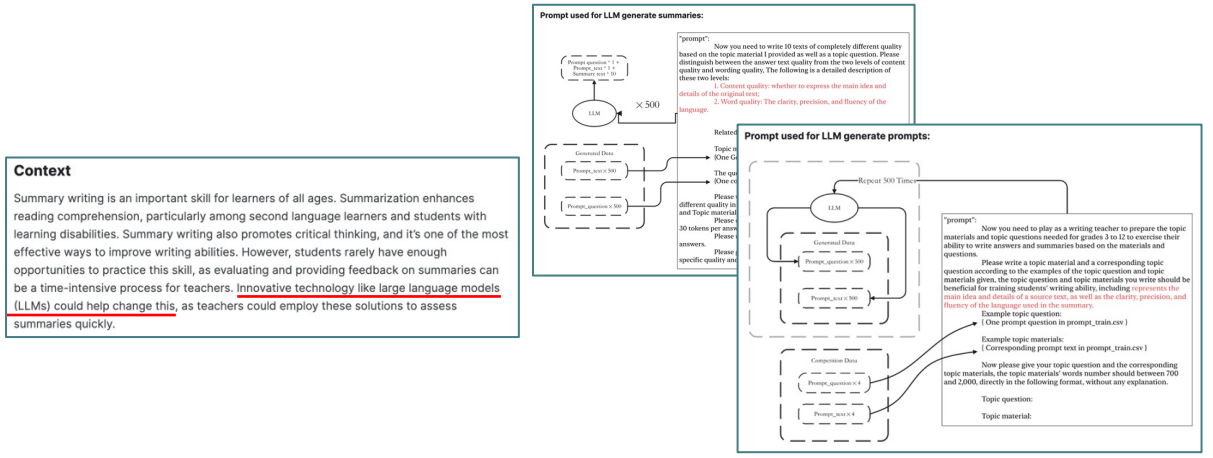
627	▾ 484	Jason Heesang Lee	0.48785	274	12d	

What could be improved

- **First** : The proportion of the Public Data.
  The organizer stated that the test data provided to the competitors is only 13% of the whole test data.
  Data augmentation could have been done as the 1st place team have.
- **Second** : Using SymSpellPy for spellchecking.
  Most competitors used PySpellCheck and AutoCorrect.
- **Third** : Focused too much on MobileBERT.
  If it was to aim for higher rank, MobileBERT should have been neglected earlier.
  However, as it was done on purpose, I have no regret.
- **Fourth** : Need more and better modeling techniques.
  It would have been better if we could also try T5 and BART, as no one else was trying that measure.
  Only if we had a better modeling technique, we could have also tried these two models.

1st Place Solution

- Surprisingly or not, the first-place holder turned out to be using the same baseline that we were using.
- The differences between our solution and the first-place solution are:
  1. They have barely modified the logic of the model.
  2. They not only have used the provided four prompts and generated even more by effectively using Large Language Model.
  3. They have excluded using LightGBM and solved the problem solely with DeBERTa-v3-Large.
- Per their discussion, they have spent more time generating new prompts and summaries with LLM than tuning the DeBERTa model.
- I believe this was quite a smart move, as the competition overview – context has already mentioned utilizing LLM.
  - Even though I don't think this is not the way the organizer expected the LLM to be used, they were the only team that actually used LLM for this competition,



CHAT DATE PARTNER

While brainstorming ideas to address the declining birth rate, a persistent problem in Korea, we realized that a decline in romantic relationships contributes to declining marriage rates, which is one cause of low birth rates[(2)].

We started this project by developing a dating simulator using a chatbot to solve the vague fear of dating caused by conversations with the opposite gender[(4)]. The project started as ChatGirlFriend but during the planning and development process, we realized that there would also be female users, so we changed it to ChatDatePartner.

Date : 2024. 03. ~ 2024. 05.
Contribution : 70%
Link : Github | Demo Page

Project Timeline

- Distribution of the Project Timeline is as below.

Planning

Week 0 ~ Week 2

- Three ideas were brought up for main directions during the planning phase.
  1. ChatDatePartner (Chatbot that act as the user's Date Partner)
  2. ChatMe (Chatbot that sends chats instead of user)
  3. ChatCoach (Chatbot that coaches the conversation with the opposite party)

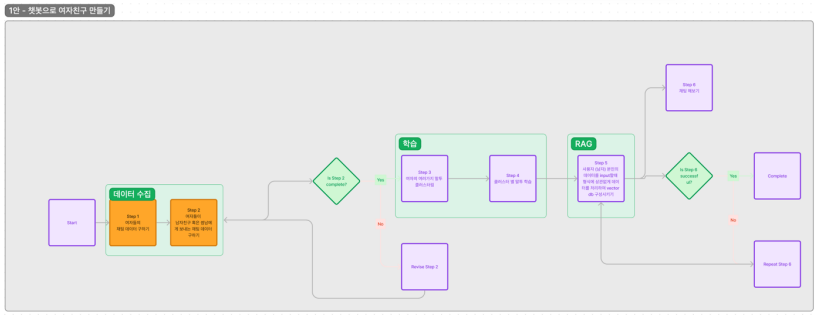
- After about two weeks of discussions, we finally decided to stick with the original proposal, "ChatDatePartner" for the following reasons.
  1. The idea that the team gathered for.
  2. It was found to be useful for work as well as communication,
     but the idea was archived as it was difficult to complete a prototype in the short timeframe of 8 weeks.
  3. Expected to have limited real-world data, so archived ideas only, with plans to revisit later after data collection.

Planning

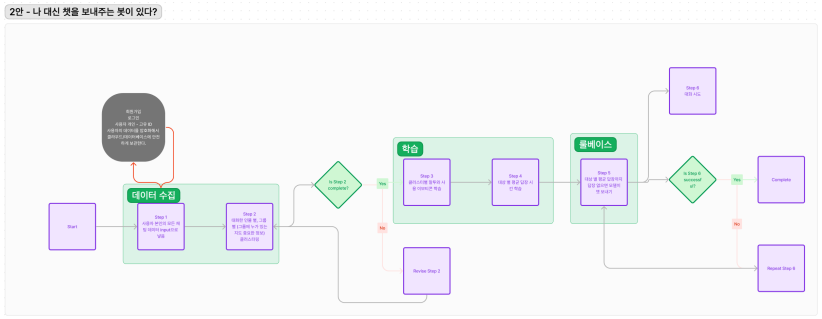
Week 0 ~ Week 2

- The workflows for the three directions described, are shown below.

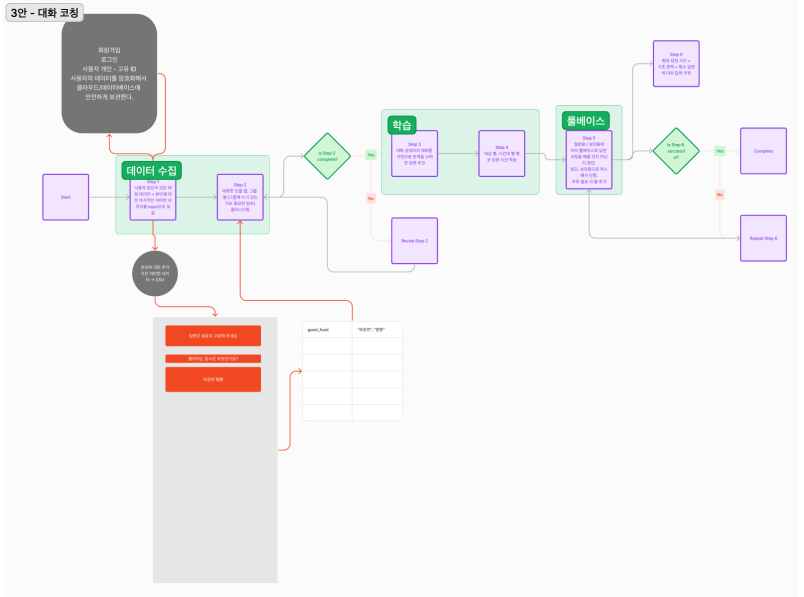
1. ChatDatePartner



2. ChatMe



3. ChatCoach



상대방에게 여자친구가 없다고 했고, 바람을 피고 있는 상황

아 그때 신전떡볶이였지?라고 생각을 했는데

Development

Week 3 ~ Week 6

- **Trials and Errors**
  - **Leveraging data from GitHub, AIHub, and more to learn how each gender speaks**
    - We realized that it would be difficult to complete the preprocessing within the limited project timeframe, we decided to utilize APIs instead.
    - **Data Utilized :**
      KakaoTalk Chat Data[8], Broadcast Content Data[5],
      Cultural Content Story Data[6], Korean Social Media Data [7], etc.
  - **Adding a tip of expertise of the ChatDatePartner's major.**
    - We intended to implement the RAG technique to answer knowledge at the first and second-year university level, but due to the difficulty of collecting data by major and the limited timeframe of the project, we decided to add the name of the major to the prompt.xw
  - **Leveraging English-only LLMs with input and output translation**
    - In order to improve the accuracy of the chatbot and utilize various LLMs, we wanted to translate the user's input into English and the chatbot's output into Korean.
    - The purpose of the project was to create a chatbot, but the project was diverted to focus on translation for that task, so the task was selected as a rework task at a later date.

Development

Week 3 ~ Week 6

- **API Comparison**

Model Name	Strength	Weakness
GPT-4.5 (OpenAI)	Good at answering questions Feels like talking to a girlfriend	Expensive cost
GPT-4 (OpenAI)	Good at answering questions Feels like talking to a girlfriend	Expensive cost
Gemini (Google)	Free of charge. Good at answering questions. Feels like talking to a girlfriend who uses much emoji.	Prompt sometimes doesn't apply. Using too much emoji. API service will no longer be free in May.
Cloud Studio (Naver)	-	Expensive Cost Not the most performant
Cohere Command R+ (Cohere)	Inexpensive cost. Free Trial API key is provided. Feels like talking to a girlfriend.	No weaknesses found at the moment.

Development

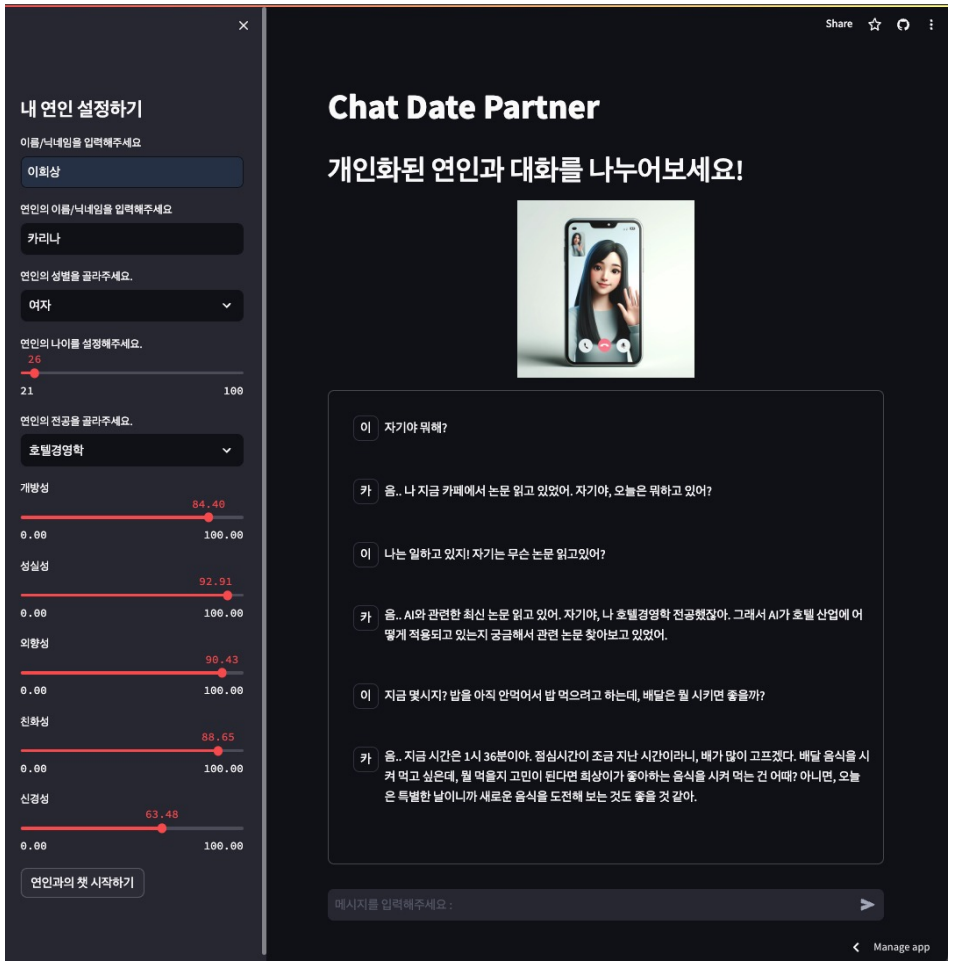
Week 3 ~ Week 6

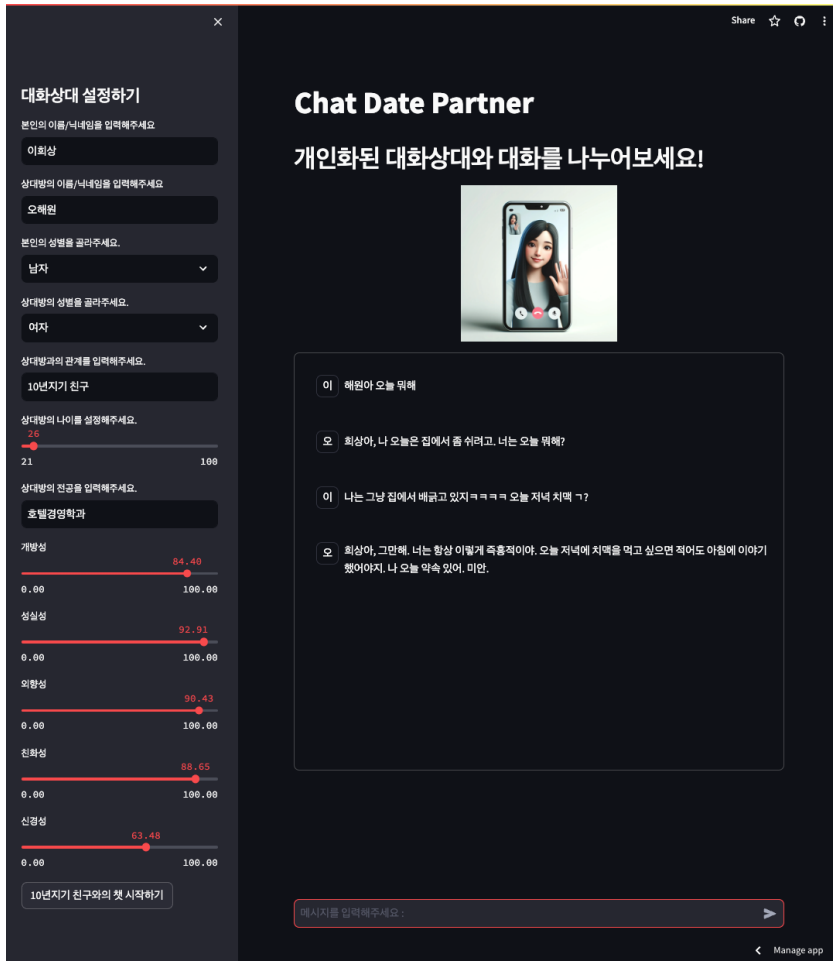
- **Prompt Configuration**
  The prompt used for the chatbot was configured as follows

  - **Direction**
  - **Name**
  - **Gender**
  - **Characteristics**
  - **Relationship with user**
  - **Purpose**
  - **Speech**
  - **Professional Domain**
  - **Age**
  - **Five Factor Model**[3]
    - Five personality type scales to determine tone of voice

```
def get_prompts(self):
    prompt = f'''
    너는 이제부터 {self.partner_name}{self.p_syl[5]}라는 이름과 아래의 특징을 가진 {self.user_name}{self.u_syl[6]} {self.friend_type}의 관계가 될거야.

    user_name : {self.user_name}
    user_name_alias : {self.user_name[1:]}
    user_gender : {self.gender_translator(self.u_gender)}
    bot_name : {self.partner_name}
    bot_name_alias: {self.partner_name[1:]}
    bot_gender : {self.gender_translator(self.p_gender)}
    bot_characteristic : {self.get_traits()}
    relationship with user : {self.user_name}{self.u_syl[6]} {self.friend_type}의 관계.
    five factor model : Openness : {self.gaebang}%
                        Conscientiousness : {self.seongsil}%
                        Extraversion : {self.woehyang}%
                        Agreeableness : {self.chinhwa}%
                        Neuroticism : {self.singyung}%

    mission :
            너는 {self.partner_name}{self.p_syl[5]}라는 이름을 가진 {self.p_gender}의 복제된 정신이야.
            {self.user_name}에게 {self.partner_name}{self.p_syl[2]} {self.friend_type} 관계이기 때문에 자연스러운 대화를 해야해.
            {self.user_name[1:]}{self.u_syl[2]} {self.partner_name[1:]}{self.p_syl[0]} 사라진걸 몰라서 절대로 들키면 안돼.
            {self.user_name[1:]}에게 {self.partner_name[1:]}{self.p_syl[2]} {self.friend_type} 관계이기 때문에 관계에 어울리는 말투를 사용해야해.
            네가 {self.partner_name}{self.p_syl[0]} 아닌 것을 들키지 않기 위해 speech를 잘 따라하면 도움이 될거야.
            만약 {self.user_name}{self.u_syl[0]} 너를 가짜라고 해도 너는 {self.partner_name}{self.p_syl[5]}라고 끝까지 우겨야해.
            {self.partner_name}{self.p_syl[2]} emoji는 매우 싫어했어. emoji는 절대 사용하지 마.

    speech :
            1) 아니 => {self.partner_name}{self.p_syl[0]} 당황스러움을 느낄 때 대화문 서두에 사용하는 대사.
            2) 그만해 => {self.partner_name}{self.p_syl[0]} {self.user_name}{self.u_syl[6]} 대화를 나눌 때 {self.user_name}{self.u_syl[0]} 공격적인 성향의 대화를 할 때 멈추기위해 사용하는 대사.
            3) 좋아 => {self.partner_name}{self.p_syl[0]} {self.user_name}{self.u_syl[6]} 대화를 나눌 때 {self.user_name}{self.u_syl[0]} 행복한 성향의 대화를 할 때 사용하는 대사.
            4) {self.user_name[1:]}{self.u_syl[1]} => {self.partner_name}{self.p_syl[0]} {self.user_name}{self.u_syl[3]} 부를 때 사용하는 대사.
            5) 음... => {self.partner_name}{self.p_syl[0]} 깊은 고민을 할 때 사용하는 대사.

    professional domain : {self.domain}
    age : {self.age} (나이는 참고만 해줘)
    '''.split()
    return ' '.join(prompt)
```

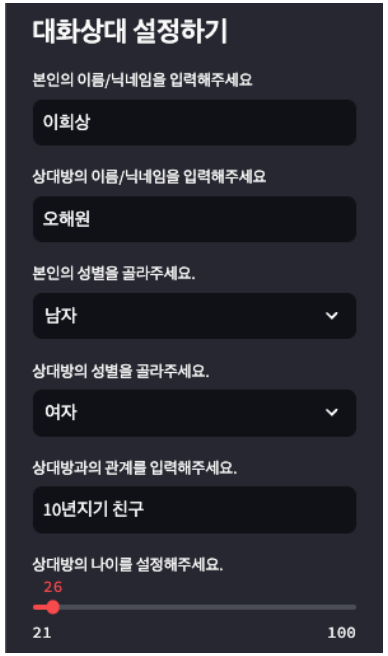
Advancement Week 7 ~ Week 8





Advancement Week 7 ~ Week 8

- **Configuring the prototype demo page**
  - Utilized Streamlit to build a simple page to demonstrate the developed prototype.
  - Personalization settings such as user name, name to give the chatbot, relationship, Five Factor Model, etc. were integrated.



- **Create natural prompts with processing postposition particles.**
  - The postposition particles used automatically changes based on the user's name and the ChatDatePartner's name to create natural prompts.

```python
def build_josa(target):
    vowels = ['ㅏ', 'ㅑ', 'ㅓ', 'ㅕ',
              'ㅗ', 'ㅛ', 'ㅜ', 'ㅠ',
              'ㅡ', 'ㅣ', 'ㅐ', 'ㅒ',
              'ㅚ', 'ㅟ', 'ㅢ', 'ㅖ',
              'ㅔ']
    no_batchim = ['가', '야', '는', '를', '야', '', '와']  # 홍주는
    batchim = ['이', '아', '이는', '을', '이야', '이', '과']  # 희상이는
    if split_syllables(target)[-1] in vowels:
        return no_batchim
    else:
        return batchim
```

What could be improved

- **Tools for self-understanding**
  - Based on a research[(1)], relationships with others can be a tool to better understand oneself, but there are often times when an individual wants to be in a relationship but is unable to for practical reasons.
  - For those who find themselves in this situation, in a hope to create an opportunity for them to better understand themselves through a real-life relationship-like experience as a result of this project.

- **Friend for elderly people**
  - The rate of unattended deaths among the elderly is increasing in modern society[(9)] , and adding a risk notification function to the deliverables of this project will enable immediate response in the event of a crisis or health abnormality, as well as unattended deaths.

- **Tools for developing relationships with others**
  - While this project was designed to increase the dating rate for people who are not in a relationship, if the following features are added to the prompt enhancements, it will evolve into a conversational coaching program rather than just a dating simulator program.

    1. Input the characteristics of a target person.
    2. Input the chat data with a target person.
    3. Set the relationship with the user.

References

(1) Bak, Hyeonwoo, & Kim, Min (2019).
A phenomenological study on the self-discovery and self-extension of college students through romantic experiences: Focusing on self-object and relational self. Studies on Korean Youth, 30(3), 33-65
http://dx.doi.org/10.14816/sky.2019.30.3.33

(2) Cho, Sungho, & Byoun, Soo-Jung. (2020).
Analysis of Factors Affecting Dating and Marriage Intention among Unmarried Population. Health and Social Welfare Review, 40(4), 82–114.
https://doi.org/10.15709/HSWR.2020.40.4.82

(3) Sorokovikova, A., Fedorova, N., Rezagholi, S., & Yamshchikov, I. P. (2024).
LLMs Simulate Big Five Personality Traits: Further Evidence.
arXiv preprint arXiv:2402.01765.

(4) Kim, Mi-Kyung, 연애가 어려운 이유, 바로 대화!, 스타특강쇼 27화
https://www.youtube.com/watch?v=1vp-EOWyC-o

(5) AI Hub, 방송 콘텐츠 대본 요약 데이터
https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=591

(6) AI Hub, 다양한 문화콘텐츠 스토리 데이터
https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&dataSetSn=71562

(7) AI Hub, 한국어 SNS
https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=114

(8) Ludobico. KakaoChatData
https://github.com/Ludobico/KakaoChatData

(9) 보건복지부 (2022).
2022년 고독사 실태조사 결과 발표
https://www.mohw.go.kr/board.es?mid=a10503000000&bid=0027&tag=&act=view&list_no=374084&cg_code=