



Jason Heesang Lee

Professional Portfolio

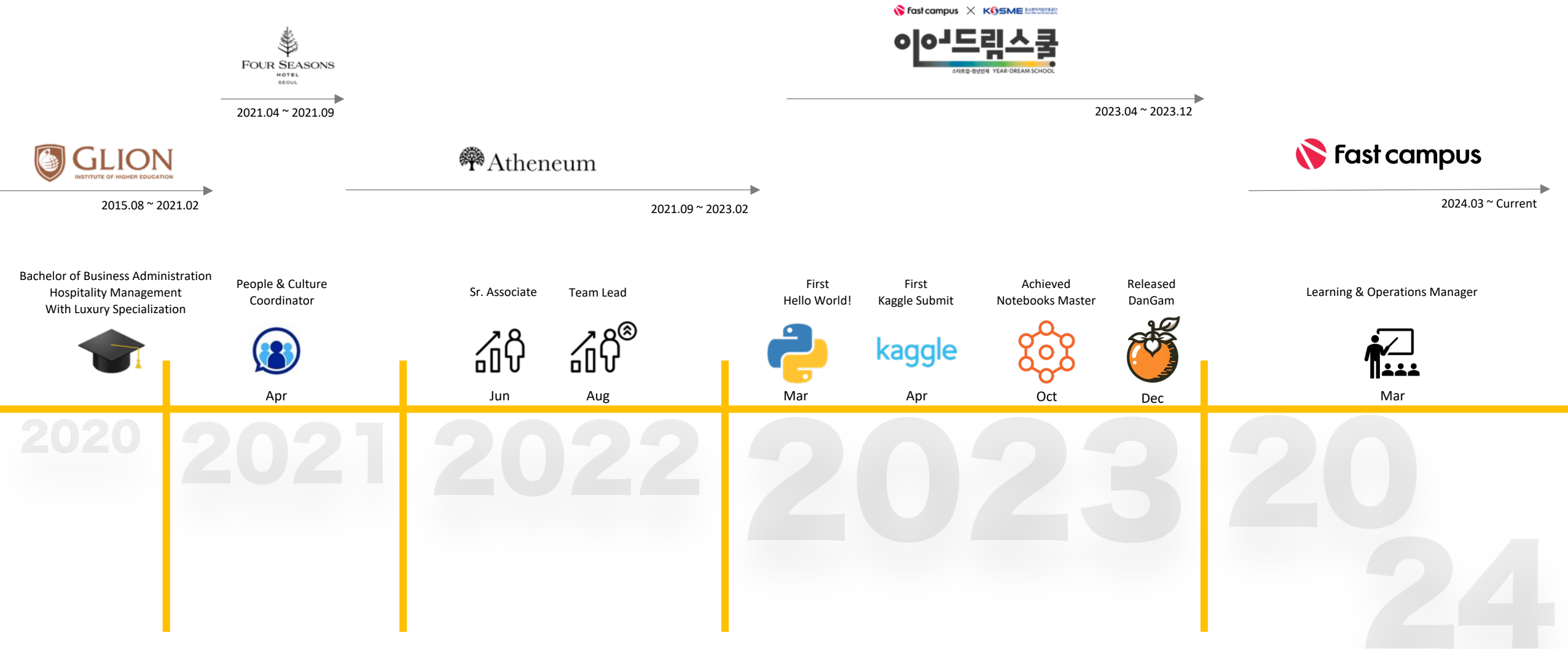


# List of Contents

---

- Timeline
- Competitions
- Career Transition
- Industry Partnership Project
- ChatDatePartner – Chatbot Project

# Timeline



# 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.

## DAICON

- **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**: 1<sup>st</sup> and 7<sup>th</sup> 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

# Career Transition



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.



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.



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.**



Avocadoland

**Industry Partnership Project**

# Defining Problem

Week 0 ~ Week 2

## Project & Company Introduction

- 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 Avacadoland, 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.

## Task

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

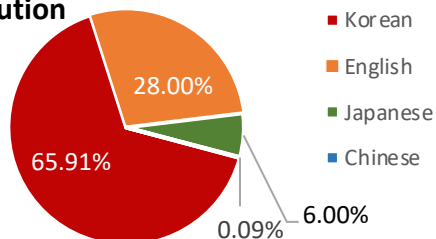
**Objective:**

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 efficient and light?	Consider whether there are other metrics that could be objective criteria instead of the subjective criteria of "acceptable"

# Resolutions

Week 3 ~ Week 6

## 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 now stably separate Chinese and Japanese.

## 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 at Github<sup>(4)</sup>

- **How much weight should we put on each keyword based on the emotional state to make most users accept?**

We used the approach from a research paper<sup>(5)</sup> to develop our sentiment analysis model.

## 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 efficient and light?**

TweetNLP was used as an example.  
It was light and efficient, but it only performed sentiment analysis on sentence-level.

- **How did we solve?**

I have developed a brand-new package called DanGam<sup>(3)</sup> 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.

### Preprocessing

Identifying  
Unregistered  
Words & Terms

Removing  
Informal Terms

Removing  
Informal Terms

### Sentiment Analysis

TweetNLP

LLM

DanGam





# Resolutions

Week 3 ~ Week 6

## 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 <sup>(1)</sup>

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 <sup>(2)</sup>

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.

## Output Example

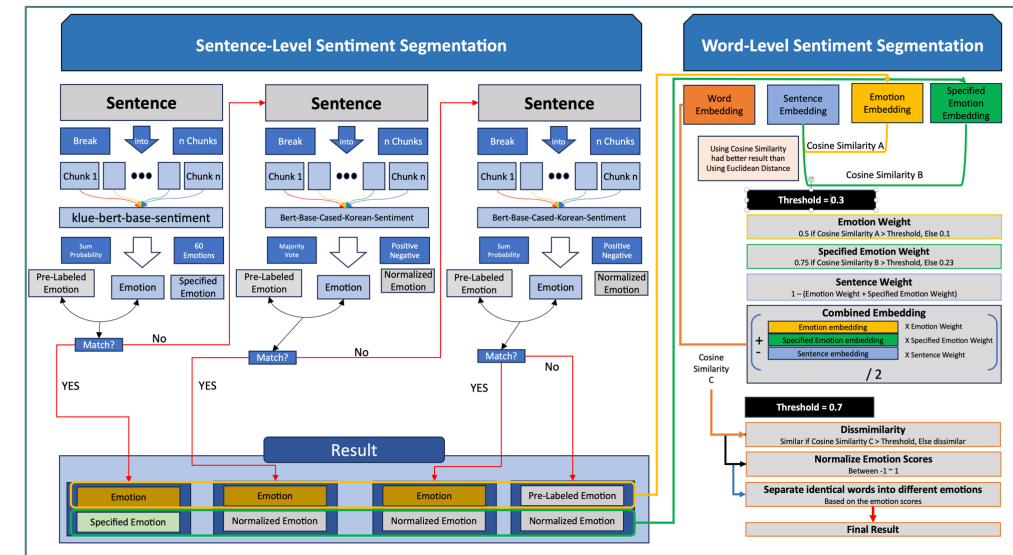
### • Example Sentence :

"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,
# '마라향': 0.328179239525493,
# '미쳤다': 0.34263925379014165,
# '그런데': -0.07491504014905744,
# '고수는': -0.7992964009024587,
# '진짜': -0.9295882226863167,
# '싫다': -0.9120299268217638}
```

## Visualized Diagram



# Verification & Validation

Week 7 ~ Week 8

## Final Output

태형이와 시오니와 찬허기, 배나영, 이희상과 함께 롯데월드에서  
롯데월드 좋아했다. 마라탕을 먹고 오꼬노미야끼도 먹고, 탕후루도 먹었다.  
나영이와 희상이 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 찬허기 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 in the field of AI, it was quite difficult to align our methods of approach.
- Also, even though pivoting to a new objective was carefully done with a mutual agreement with Avacadoland, but we couldn't help but be disappointed as this result was not one that we desired in the beginning of the project.
- However, with this experience, we learned a lot on how to collaborate with others in the field of AI, professionally exchanging our thoughts and opinions.

## What could be improved

- There was some loss of information due to preprocessing for misspelled words correction when extracting keywords.  
There were also issues such as the occurrence of word particles and other parts of speech merged, which could be prevented if given a little more time.
- Each morphological analyzer had its own advantages and disadvantages.  
If these were utilized with a stacking method that utilizes the features of each, we could have made each word clearer and produce better quality results.
- Due to the duration of the project, we only worked on Korean and English, but we will need to create a model that analyzes Chinese and Japanese as well for the existing purpose.
- As we didn't take a planning phase in the beginning of the project and jumped right into development, we couldn't solve the problems until the end, which could have been prevented in the planning phase.  
To prevent the same situation from happening in the ChatDatePartner project, which will be introduced later in this slides, we set assigned about 30% of the total period for planning.

# References

- 1) Chen, Ruiqi, Zhou, Yanquan, Zhang, Liujie & Duan, Xiuyu (2019)  
 Word-level Sentiment Analysis with Reinforcement Learning  
 DOI 10.1088/1757-899X/490/6/062063
- 2) Ito, Tomoki, Tsubouchi, Kota, Sakaji, Hiroki, Yamashita Tatsuo & Izumi Kiyoshi (2020)  
 Word-Level Contextual Sentiment Analysis with Interpretability.  
<https://doi.org/10.1609/aaai.v34i04.5845>
- 3) jasonheesanglee. Dangam.  
<https://github.com/jasonheesanglee/dangam>
- 4) jasonheesanglee. JsonSpeller.  
<https://github.com/jasonheesanglee/JsonSpeller>
- 5) Jaewon Hwang, & Youngjoong Ko (2009). A Document Sentiment Classification System Based on the Feature Weighting Method Improved by Measuring Sentence Sentiment Intensity. Journal of KISS : Software and Applications, 36(6), 491-497.  
<https://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE01208058>



# CHAT DATE PARTNER

Date : 2024. 03. ~ 2024. 05.  
Contribution : Planning - 70% | Development – 80%  
Link : [Github](#) | [Demo Page](#)

# Planning

Week 0 ~ Week 2



- **Distribution of the Project Timeline is as below.**



- Based on the experience of not having a planning period in previous projects, we decided to set a planning period of 3 weeks, which is about 30% of the total duration of the project.
- 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<sup>(5)</sup>.
- 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<sup>(6)</sup>.
- 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.

# 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 continue with the original proposal, “ChatDatePartner” for the following reasons.**
  1. ChatDatePartner was an idea that other team members decided to join this team.
  2. The project was found to be useful not only for communication but also for business use, but the idea was archived due to the difficulty of collecting message data containing personal information for verification and the difficulty of completing a prototype in a short period of 8 weeks.
  3. Due to the difficulty of collecting message data containing personal information required for verification and the limited availability of various real-world data, only the idea was archived.

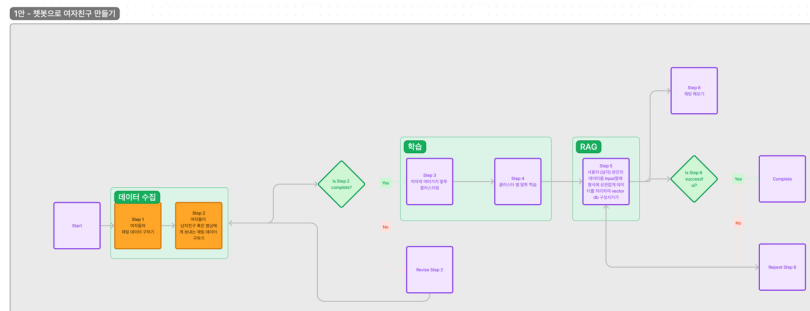
# Planning

Week 0 ~ Week 2

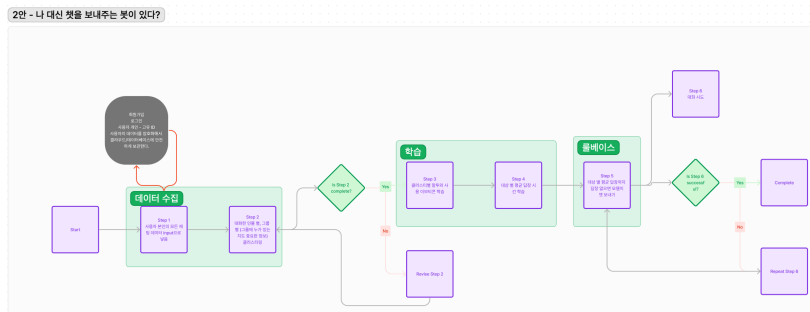


- The workflows for the three directions described, are shown below.

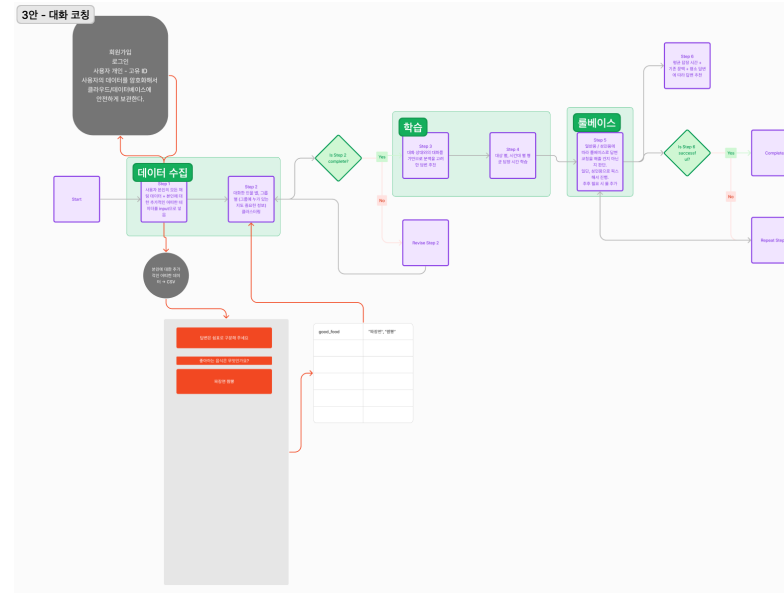
## 1. ChatGirlFriend



## 2. ChatMe



### 3. ChatCoach



 피해자

상대방에게 여자친구가 없다고 했고,  
바람을 피고 있는 상황

오빠 우리 저번에 갔던  
떡볶이 집이 어디지?

코칭: 이 분과 최근에 갔던 떡볶이 집은 업기떡볶이입니다.

아 그때 신전떡볶이였  
지?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 used :**

KakaoTalk Chat Data<sup>(7)</sup>, Broadcast Content Data<sup>(1)</sup>, Cultural Content Story Data<sup>(2)</sup>, Korean Social Media Data <sup>(3)</sup>, 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.

- **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.

- **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

- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>- <b>Direction</b><br/>Assigning role</li><li>- <b>Name</b><br/>Assigning name</li><li>- <b>Gender</b><br/>Assigning gender</li><li>- <b>Characteristics</b><br/>Assigning characteristics</li><li>- <b>Relationship with user</b><br/>Assigning relationship with user</li></ul> | <ul style="list-style-type: none"><li>- <b>Five Factor Model<sup>(8)</sup></b><br/>Assigning five personality type scales to determine tone of voice</li><li>- <b>Purpose</b><br/>Assigning purpose</li><li>- <b>Speech</b><br/>Assigning speech style example</li><li>- <b>Professional Domain</b><br/>Assigning professional domain</li><li>- <b>Age</b><br/>Assigning age</li></ul> |
|---|--|

- **Creating seamless prompt**

- **Processing postposition particles.**

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

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

- **Adding current time**

Provided current time at each chat message to add real-time attribute.

```
def chat(self, user_input): ## Cohere  
    current_time = str(datetime.now(tz=pytz.timezone('Asia/Seoul')))  
    response = self.co.chat(  
        chat_history=self.get_chat_history(),  
        preamble=self.system_message.replace(_old: '{time}', current_time),  
        message=user_input,  
        connectors=[{"id": "web-search"}],  
    ).text
```

# Verification & Validation

Week 7 ~ Week 8



- **Prompt Configuration**

Prompt was configured as below.

```
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 틀키면 안돼.
        {self.user_name[1:]}에게 {self.partner_name[1:]}{self.p_syl[2]} {self.friend_type} 관계이기 때문에 관계에 어울리는 말투를 사용해야해.
        내가 {self.partner_name}{self.p_syl[0]} 아닌 것을 들리지 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]} 답황스러움을 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)
    '''
    return ' '.join(prompt)
```

# Verification & Validation

Week 7 ~ Week 8



내 연인 설정하기

이름/닉네임을 입력해주세요

이희상

연인의 이름/닉네임을 입력해주세요

카리나

연인의 성별을 골라주세요.

여자

연인의 나이를 설정해주세요.

26

21

100

연인의 전공을 골라주세요.

호텔경영학

개방성

84.40

0.00

100.00

성실성

92.91

0.00

100.00

외향성

90.43

0.00

100.00

친화성

88.65

0.00

100.00

신경성

63.48

0.00

100.00

연인과의 챗 시작하기

Chat Date Partner

개인화된 연인과 대화를 나누어보세요!

이 자기야 뭐해?

카 음.. 나 지금 카페에서 논문 읽고 있었어. 자기야, 오늘은 뭐하고 있어?

이 나는 일하고 있지! 자기는 무슨 논문 읽고있어?

카 음.. AI와 관련한 최신 논문 읽고 있어. 자기야, 나 호텔경영학 전공했잖아. 그래서 AI가 호텔 산업에 어떻게 적용되고 있는지 궁금해서 관련 논문 찾아보고 있었어.

이 지금 몇시지? 밥을 아직 안먹어서 밥 먹으려고 하는데, 배달은 뭘 시키면 좋을까?

카 음.. 지금 시간은 1시 36분이야. 점심시간이 조금 지난 시간이라니, 배가 많이 고프겠다. 배달 음식을 시켜 먹고 싶은데, 뭘 먹을지 고민이 된다면 최상이 좋아하는 음식을 시켜 먹는 건 어때? 아니면, 오늘은 특별한 날이니까 새로운 음식을 도전해 보는 것도 좋을 것 같아.

메시지를 입력해주세요 :

Manage app

- **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.

대화상대 설정하기

본인의 이름/닉네임을 입력해주세요

이희상

상대방의 이름/닉네임을 입력해주세요

오해원

본인의 성별을 골라주세요.

남자

상대방의 성별을 골라주세요.

여자

상대방과의 관계를 입력해주세요.

10년지기 친구

상대방의 나이를 설정해주세요.

26

21

100

상대방의 전공을 입력해주세요.

호텔경영학과

개방성

84.40

0.00

100.00

성실성

92.91

0.00

100.00

외향성

90.43

0.00

100.00

친화성

88.65

0.00

100.00

신경성

63.48

0.00

100.00

10년지기 친구와의 챗 시작하기

# What could be improved



- **Tools for self-understanding**

- Based on a research<sup>(1)</sup>, 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<sup>(9)</sup>, 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.

- **User Feedback**

- Thanks to this chatbot, I don't get nervous when talking to girls anymore. Thank you!
- It's interesting to see how differently the chatbot answers depending on how I set.
- I wanted a girlfriend from Busan, but it's a bit annoying that the chatbot has a weird dialect.

# 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=](https://www.mohw.go.kr/board.es?mid=a10503000000&bid=0027&tag=&act=view&list_no=374084&cg_code=)