Final Project Report

Title: Analysis of NCT members from the comment sections of their *Welcome to Sun&Moon* YouTube series

Author: Anna Batra

Summary of Research Questions and Results:

1. **What are the top keywords associated with each NCT member over all the videos? What are the top keywords per member present, per video, how do they change?**
   Finding accurate top keywords per member present, per video, proves to be challenging as the model we use may not have enough varied information to distinguish the members from one another. However, we can still get some good information from this, as explained in the results section. The top keywords over all the videos seems to do a much better job at finding keywords, and I will focus more on analyzing how they are more accurate in the results section.

2. **What members make good chemistry together? Which member do fans associate most with another member? (over all the videos)**
   This proved to be pretty accurate from my opinion as a long-time fan (5 years) of NCT, however there are some surprising members I didn't realize would be paired together. This does make sense as my opinions on chemistry between members may not be in the majority of other fans. However, this only proves that these results can be of possibly more use to the company, as they, like me, may not currently have a good idea of what members inflict more emotions in the majority of fans. Specific results shall be explained in the results section.

3. **What is the popularity of each member throughout time? (over all the videos)**
   The popularity of members is well showcased, and results can be very useful. In fact, there is a surprising result that member Winwin was mentioned in the comments section over 3000 times in the span of one day, well above the rest of the other members. More on this to come in the results section.

Motivation and Background:

NCT is a highly popular K-pop boy group (23 members) that has for this past year making regular YouTube variety content. It is not only created to appease current fans, but also used as marketing strategy to keep fans invested in the idols and bring in more fans as well. These questions are to be answered with data over the entire *Welcome to Sun&Moon* series.

Figuring out what the top keywords associated with each member from the YouTube comments can be helpful in a variety of ways. From the side of the company the group is under, it can be used to figure out what kind of personality/ face the fans make of each member, as well as make more spinoff variety content based on the response of the fans. Kind of how other companies may use techniques to find out the response of products they put out, as well as how to improve

and connect with their customers. On the side of the fans, top keywords can be used to update other fans on what the current big inside jokes are, as well anything major they may have missed in the video and wish to rewatch.

Figuring out which member is most associated with another member could also be a useful strategy in marketing new content videos by putting those members together in activities that inflict more emotions in the fans.

Last of all, it is also useful to figure out who's more mentioned widely in the comments. This can give the group's company an idea of who was more popular for this series over time, and who they might want to focus more on in future videos.

Dataset: There are ten videos in this series. The data is currently in the form of YouTube comments. I plan on using the YouTube Data API to gather the data and doing the pre-processing myself. Here are the links:

EP.1: https://www.youtube.com/watch?v=omFh70hiAG4

EP.2: https://www.youtube.com/watch?v=mSxR1l-vyuI

EP.3: https://www.youtube.com/watch?v=ecQWoZzLXSM

EP.4: https://www.youtube.com/watch?v=5FoeLnS2opU

EP.5: https://www.youtube.com/watch?v=1ElE5HV5d4o

EP.6: https://www.youtube.com/watch?v=w_52Ca6gkPE

EP.7: https://www.youtube.com/watch?v=kbl_N17Zru0

EP.8: https://www.youtube.com/watch?v=6vP9KGw9Lv8

EP.9: https://www.youtube.com/watch?v=dv9ydZstIYg

EP.10: https://www.youtube.com/watch?v=jluLTOq_f28

Challenge Goals:

1. **Messy Data:** My data is currently in the form of YouTube comments. Not in any form to be able to analyze anything yet. I will use the *YouTube Data API* to scrape it off the web and do the pre-processing myself. I will also use the library *langdetect* to limit myself to using comments written in the English language and the library *demoji* to remove the emojis.
2. **New Library:** My project involves using NLP techniques to look at word embeddings (Word2Vec) that can capture the semantic relationships between words. For research questions 1 and 2, I will use the library *gensim* to implement this Word2Vec Deep

Learning model and produce the results of the most similar words from the trained model. I will use the *nltk* library to help getting rid of stop-words that can affect the results. I will the library *wordcloud* to create visualizations of the top keywords. I will use the library *plotly* to create a table visualization for the associated members.

Methodology/ Work Plan:

Step 1: Get the data (Estimated time: already completed in 6 hours)

To get the data, a developer key for the YouTube Data API is needed, so that must be made first. Then using the small bit of python code provided by the API, we must alter it to fit our needs. We need to make sure we get all the top-level comments. We also should save every item we may possibly need returned from the API. I saved the video id, original text, author name, like count, date published, and date updated into a file for use later, per video. It is likely we may not use each of the data in our analysis, but in case our question changes we won't need to get the data again. The YouTube Data API has a limited quota on how much data we can get, so we must be careful on how many times we ask it for data.

Step 2: Clean the data (Estimated time: already completed in 4 hrs)

Next, we must clean the data. At this point, we can reduce to columns we have decided on, else it may make the process slower. I chose to save the video id, original text, and the date published. I made the decision of using the date published for the analysis, rather than the date updated. Next, we must remove the emojis from the text, I used the library *demoji* for this. Afterwards, we must keep only the English language comments, we can filter the comments we need using the library *langdetect*. We should also make the text lowercase and remove the punctuation and stopwords. To remove stopwords, we can use a list provided by the nltk.corpus library. Next, we should clean up the date published column. I kept only the date aspect of it, as the time is not needed in our analysis. Once again, we can save this cleaned data into a file for the analysis part.

Step 3: Analyze the data (Estimated time: 10 hrs)

For questions that are answered over all the video, we must feed the comment data over all the videos into the *gensim* Word2Vec model. For questions that are answered per video, we must make Word2Vec models for each video.

To answer the question of finding the top keywords per member, per video, we must ask the gensim model for the top 30 similar words to the member's name, and take out all keywords are another member's name. Those lists for those members will be the ending result for this question. Afterwards, using the library *wordcloud*, a wordcloud should be made for each member mc-ing/guesting, and be saved into a png file. There should be a file of wordclouds per episode (10 episodes, so 10 files). For finding keywords for members over all the episodes, we must instead use the model over all the video and make one wordcloud per member. This time, we shall save each member's wordcloud into their own png file (23 members, so 23 files).

For finding the most associated member, we must compare the similarity ratings of each member with all the members, using the gensim model over all the videos. The member with the highest rating for the given member will be the most associated member. Using the *plotly* library, we should create a table figure and save it to png file (1 file). As an example, for clarity, I pick one of the members, Haechan, then I compare the similarity ratings of him with each of the members and pick the top one. The result is that Haechan is most associated with that one member.

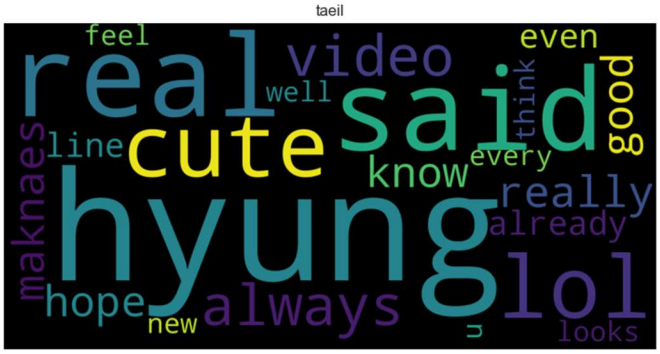
To find the popularity of the members over time, we must group the data by the date. Then we must sum the number of times each member's name appears in the comments, per date. We can then use a relplot from *seaborn* to make a data visualization that showcases all 23 members popularity over time and save it to a png file (1 file). Due to the popularity of the videos mainly dying off at a few days after the end of the series, we should plot the dates from 9/24/2020-10/24/2020.

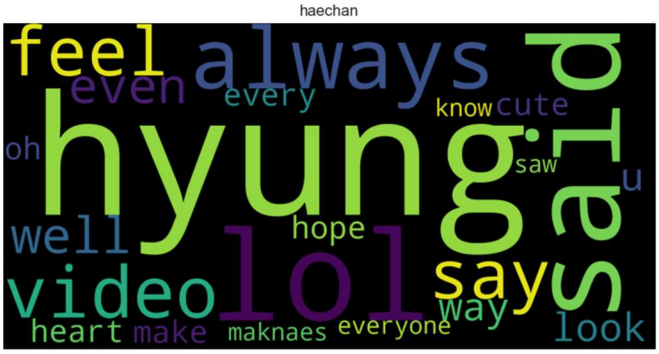
The outcome of this should be 35 png files saved to the results directory.

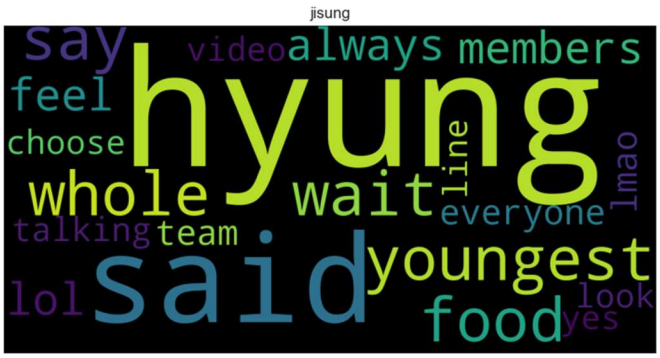
Results:

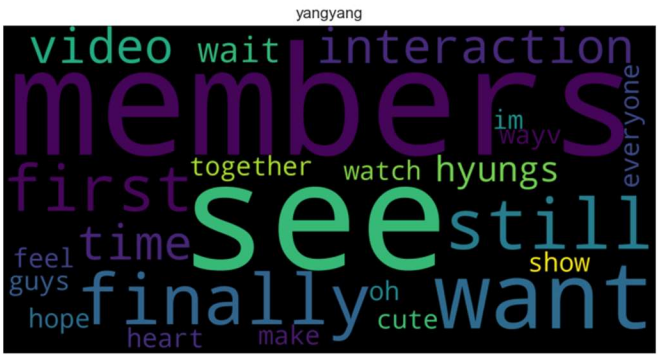
Here are the visualizations for the top keywords, per member, per video, in order from episode 1 through 10.

Episode 1:









Looking at episode 1's keywords, you can see many of the keywords are the same for each member. This may be due to not having enough varied information. However, there is a little bit of information you can still obtain. Such as with Yangyang's keywords "first," "finally," and "interaction." Having been a long time fan, I can infer that many fans in the comments talked about Yangyang's first interactions with Taeil, Haechan, and Jisung. Taeil, Haechan, and Jisung have had much more interactions as they are with the more Korean based units, while Yangyang is always with the Chinese based unit. This caused excitement in the fans for seeing what kind of interactions Yangyang would have the other members as they have never been placed together in a variety video before.

However, it is difficult to come up with good keywords for the rest of the members because of lot of them are very similar and general. The same could be said over the rest of the results for episodes 1-10. The results are like this only because the model I chose to use isn't exactly for finding the top keywords, best instead for findings word in a similar distribution to each other. Therefore, it is best that keywords only be analyzed for members over all the episodes, as the information for each member will be much more varied and the model should have a much better time coming up with more accurate keywords. I will still provide the visualizations for episodes 2-10, if you wish to take a look.

Episode 2:

taeil

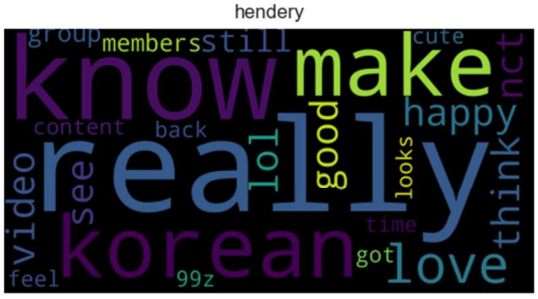
haechan

mark

lucas

xiaojun

hendery

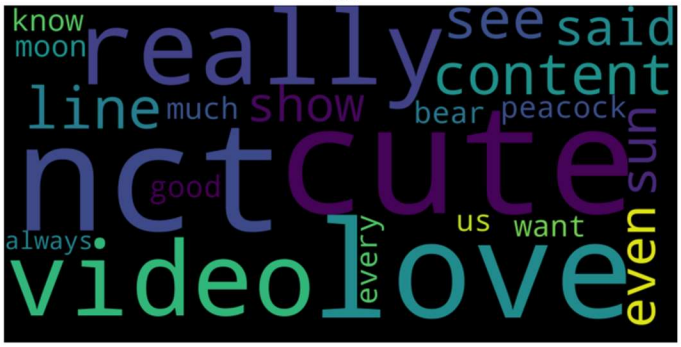


Looking at these keywords, per episode, I realize it may actually be better fit to analyze in general what are the keywords per episode, instead of per episode per member. Due to so many of the members having really similar keywords. For example, “markhyuck” was a prominent keyword for 4 of the members. Markhyuck is a friendship pairing between Mark and Haechan, whose real name is Donghyuck. They are known to be paired as best friends, and judging that it was a popular keyword, there was probably some fun interaction between them in this episode.

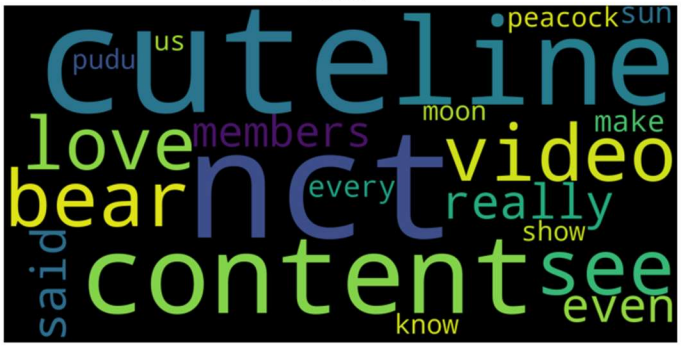
To better improve this project in the future, I would instead try to find popular keywords per episode and create a wordcloud based off of that. However, for now, I will provide the wordclouds per episode, per member, and move on to analyzing keywords for members over all the videos.

Episode 3:

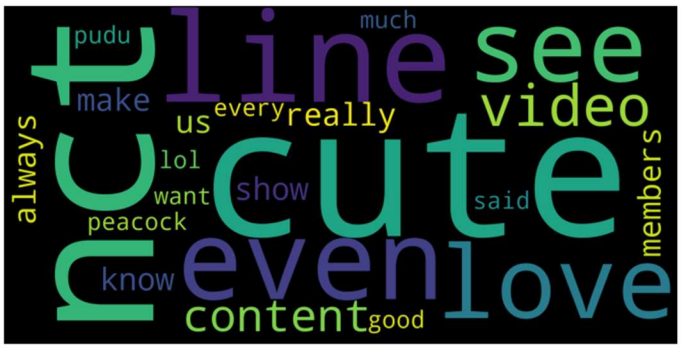
taeil



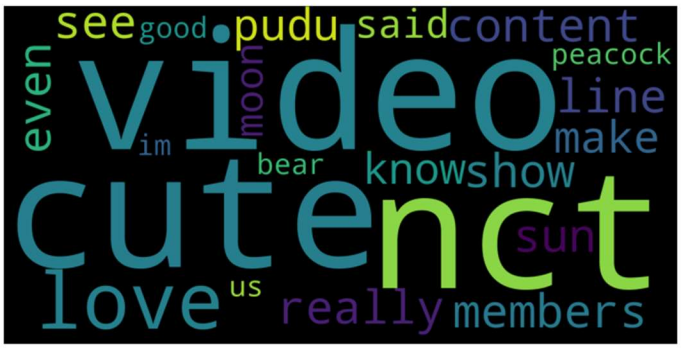
haechan



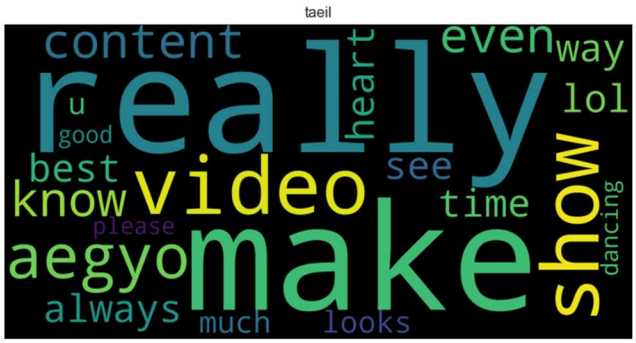
chenle

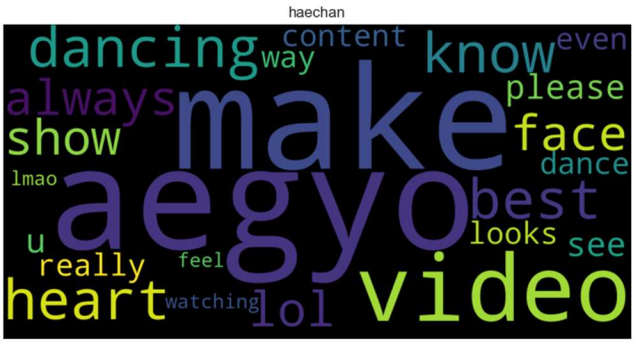


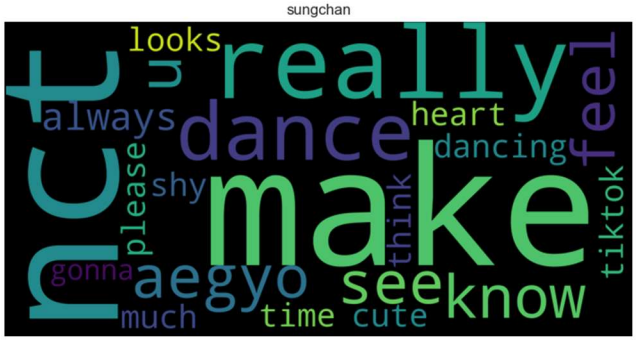
kun

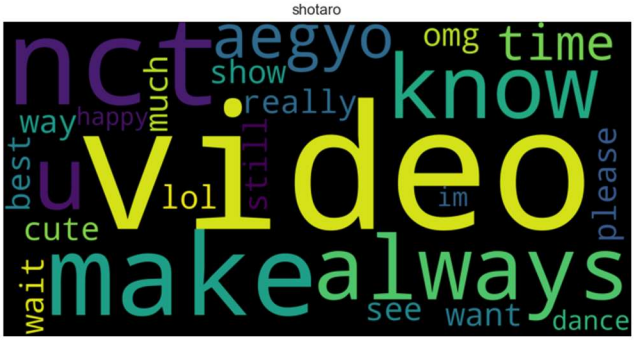


Episode 4:


taeil
content
even
way
really
heart
lol
u
good
best
know
video
see
time
show
dancing
please
aegyo
make
always
much
looks

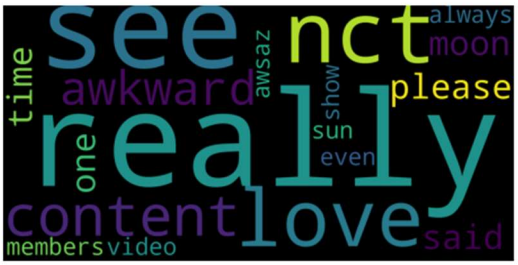

haechan
dancing
content
way
know
even
please
always
make
face
show
dance
lmao
aegyo
best
looks
see
u
really
feel
heart
watching
lol
video


sungchan
looks
really
nct
u
always
dance
heart
feel
dancing
please
shy
make
think
tiktok
gonna
aegyo
see
know
much
time
cute


shotaro
nct
aegyo
omg
time
show
know
really
much
way
happy
video
best
u
lol
still
cute
im
please
make
always
wait
see
want
dance

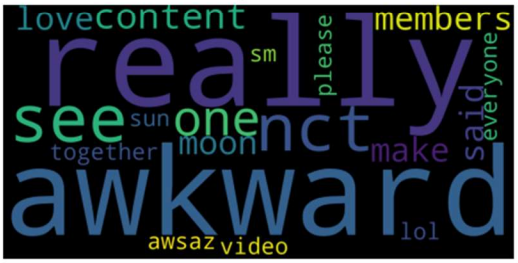
Episode 5:

taeil



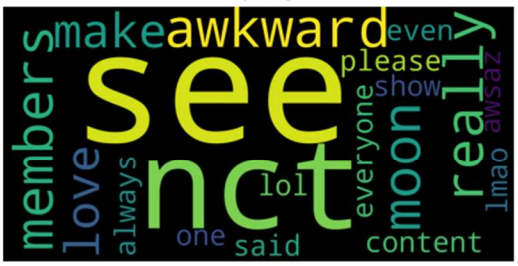
haechan

jeno



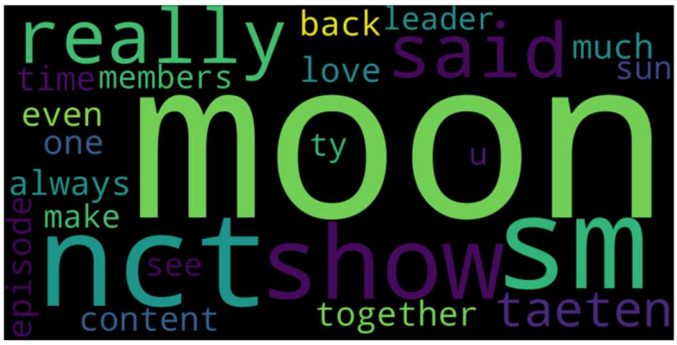
renjun

doyoung

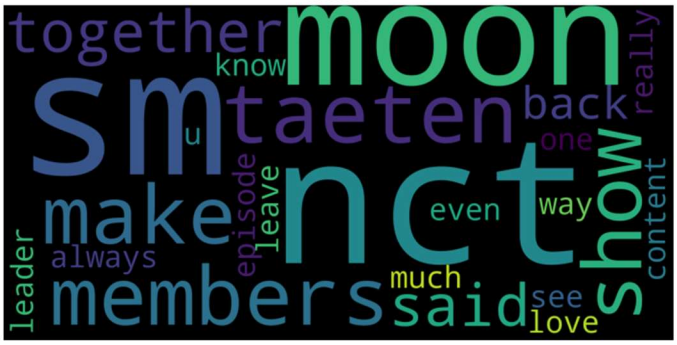


Episode 6:

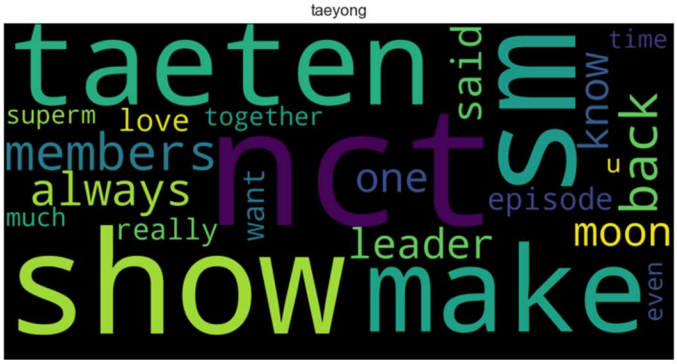
taeil



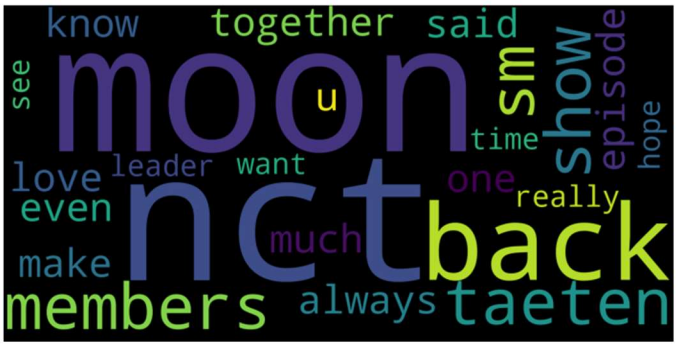
haechan



taeyong



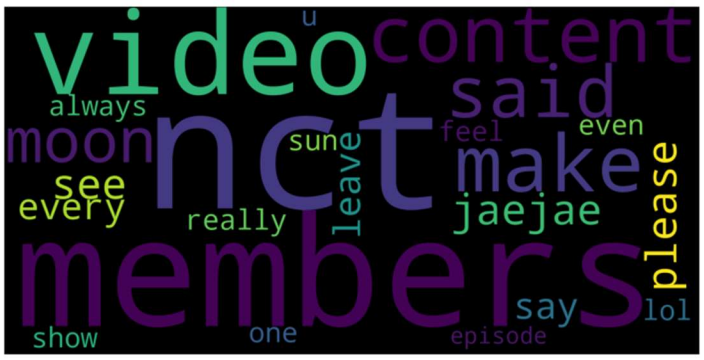
ten



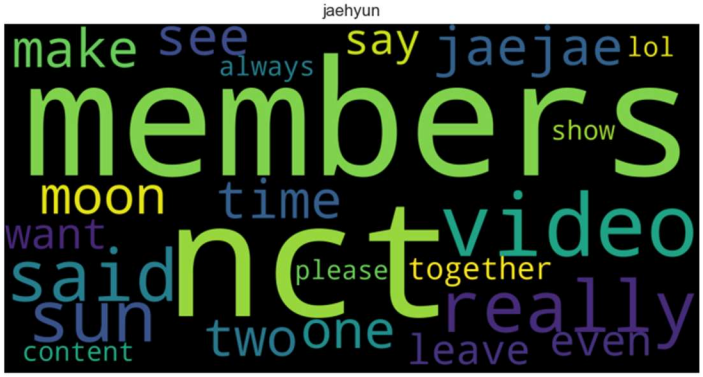
Episode 7:

taeil

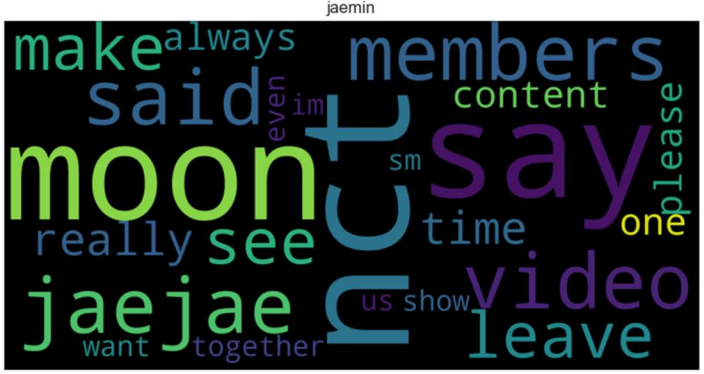
haechan



jaehyun



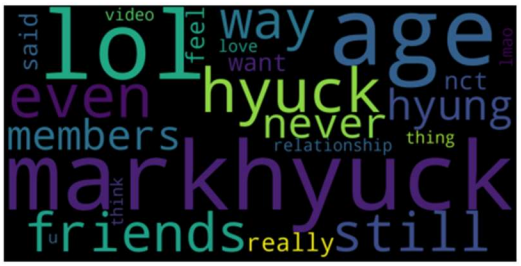
jaemin



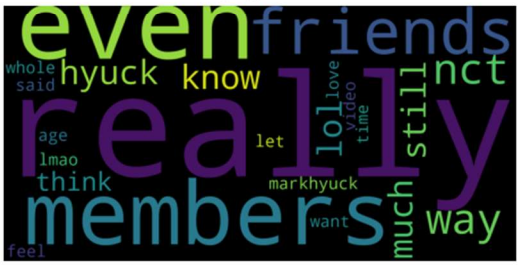
Episode 8:

taeil

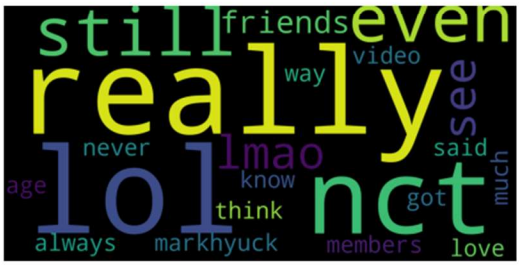
haechan



jungwoo



lucas



mark

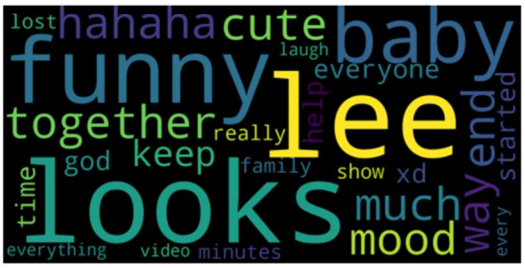
Episode 9:

taeil

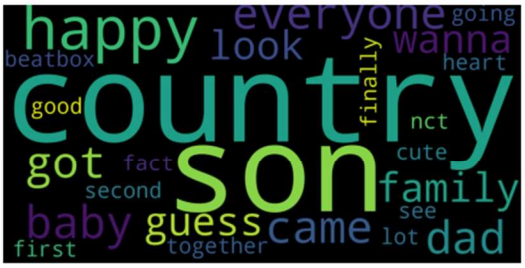
haechan

johnny

mark



ten



yangyang

Episode 10:

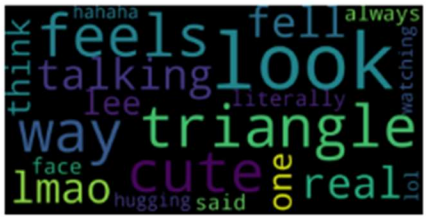
taeil

haechan

jungwoo

lucas

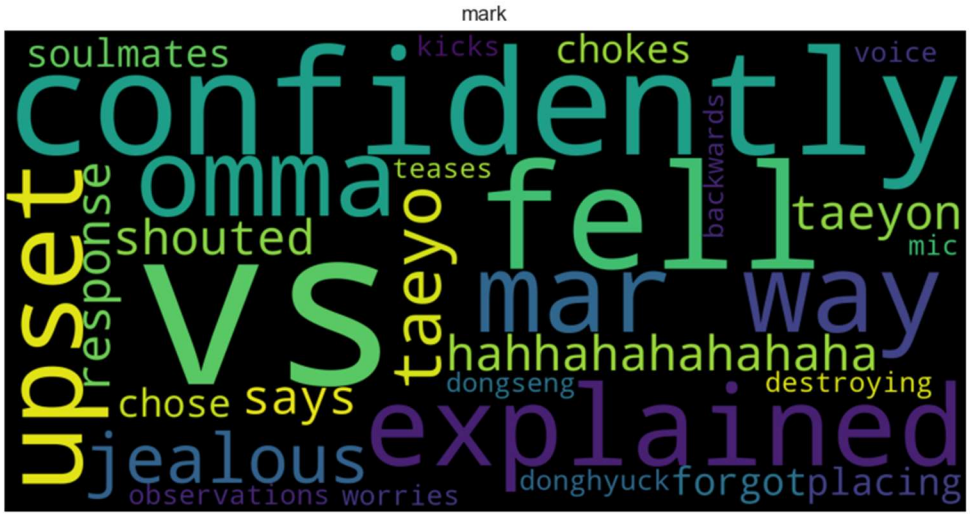
mark



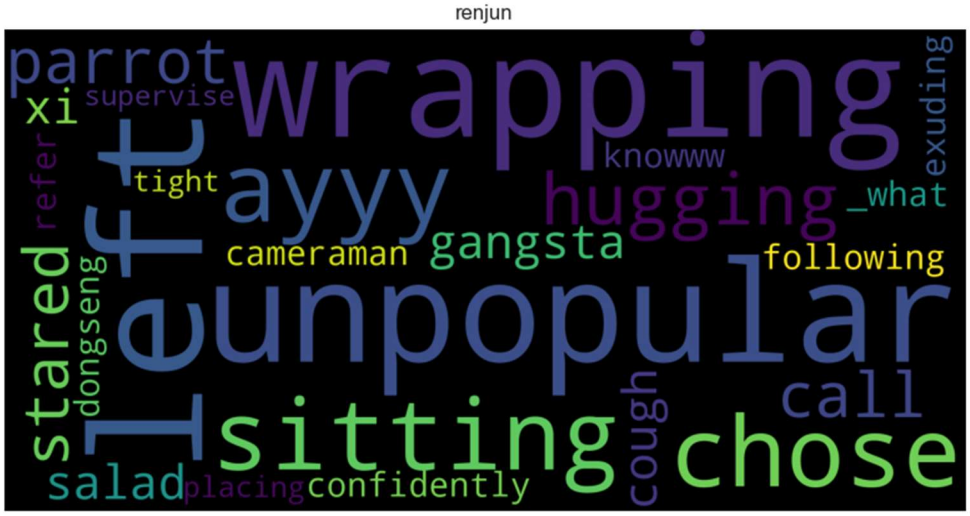
yuta

winwin

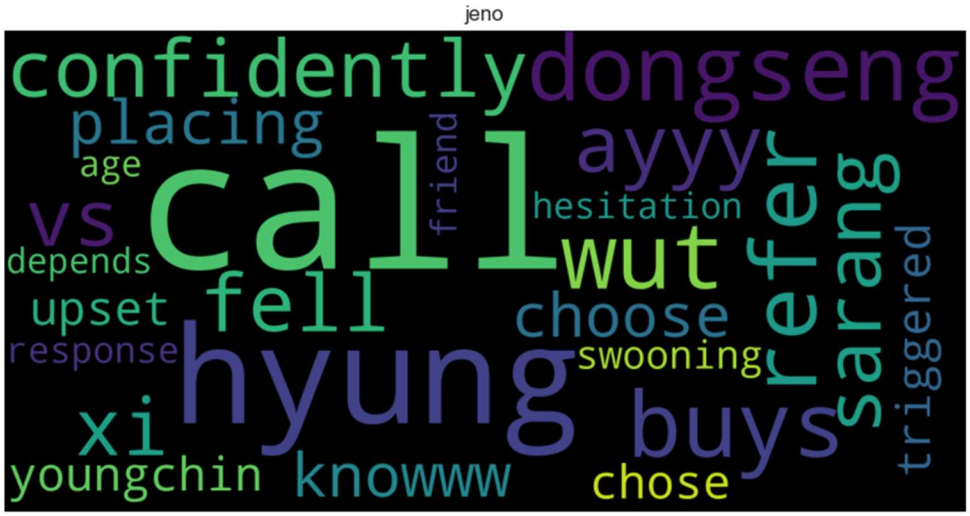
Below, I will now analyze top keywords for each member over all the videos, by listing some of the more unique ones per member



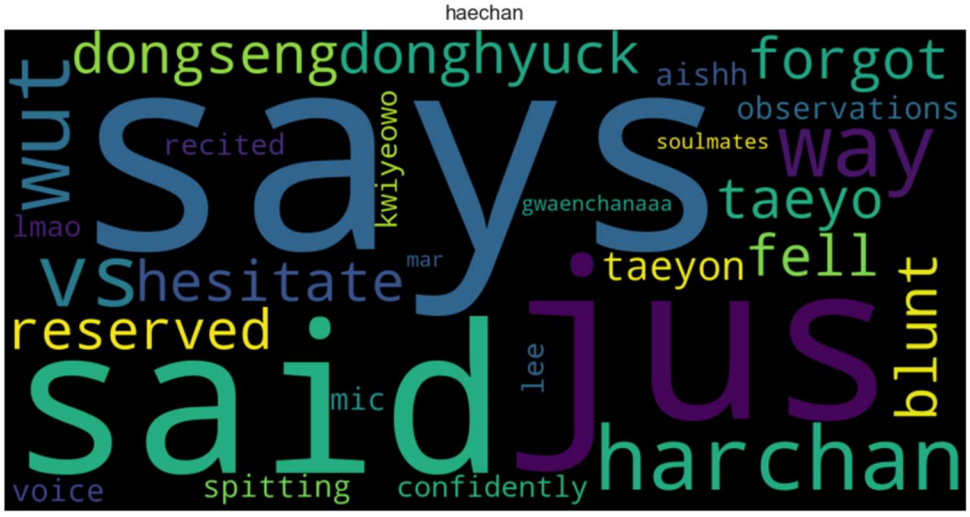
Top keywords for Mark: upset, soulmates, confidently, vs, chokes, forgot, omma (mom in Korean)



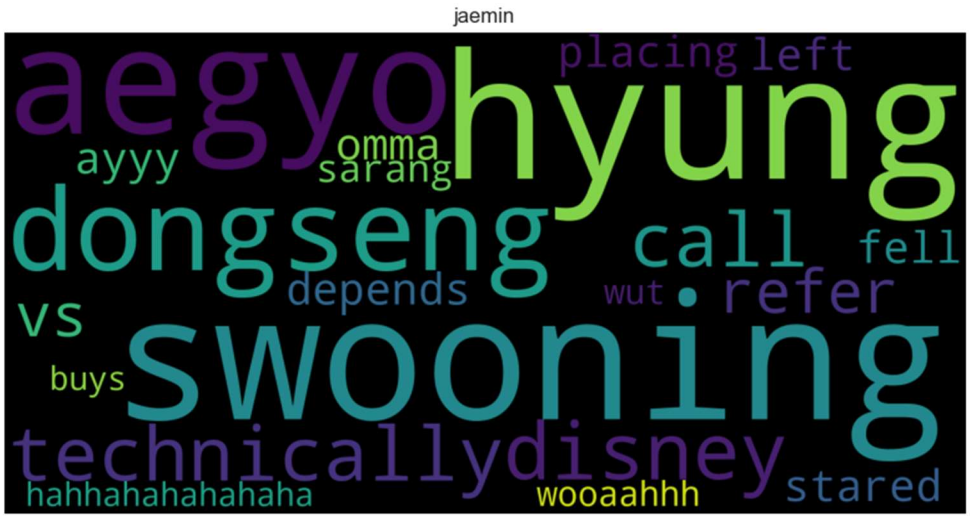
Some top keywords for Renjun: parrot, gangsta, cough, supervise, wrapping, cameraman, unpopular, salad



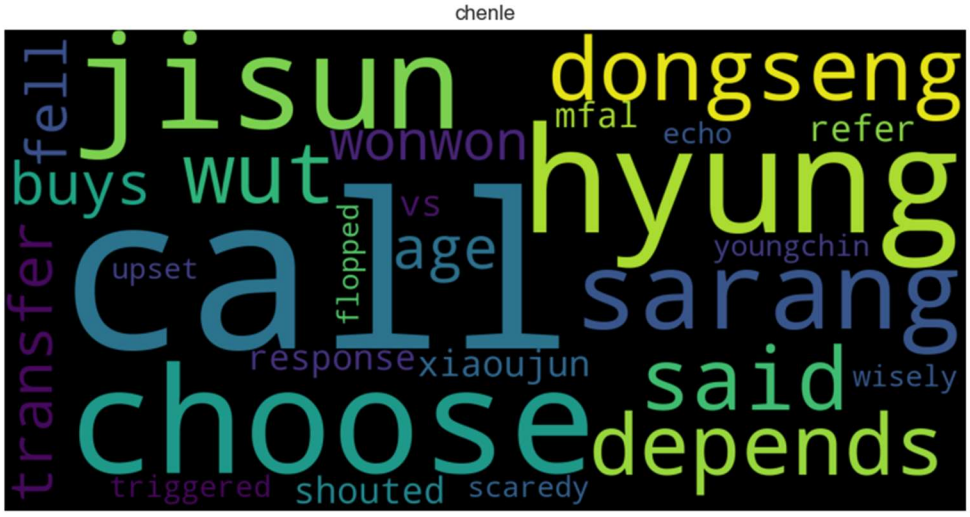
Top keywords for Jeno: call, buys, triggered, hesitation, upset, response, fell



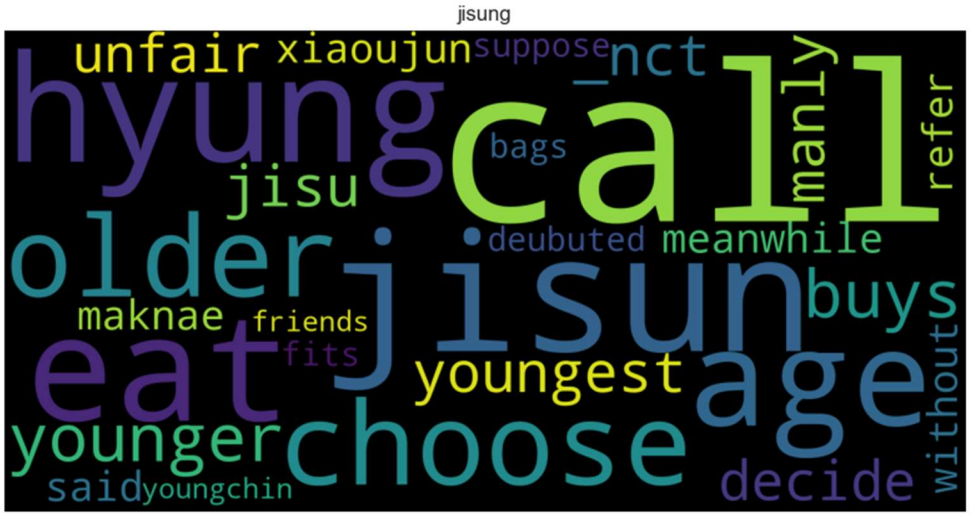
Top keywords for Haechan: dongsaeng (younger person in Korean), blunt, soulmates, reserved, recited, spitting, gwaenchana (it's okay in Korean), says



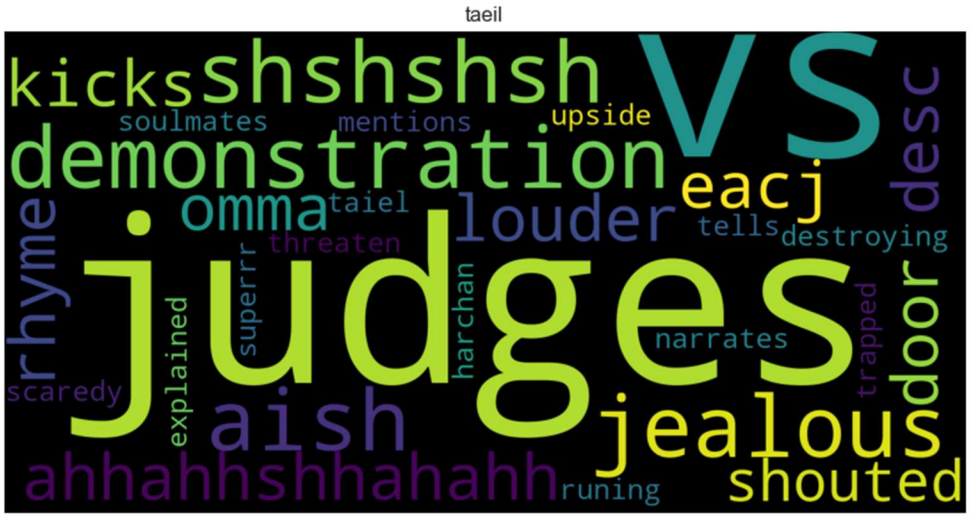
Top keywords for Jaemin: aeygo (acting cute in Korean), swooning, disney, dongsaeng (younger person)



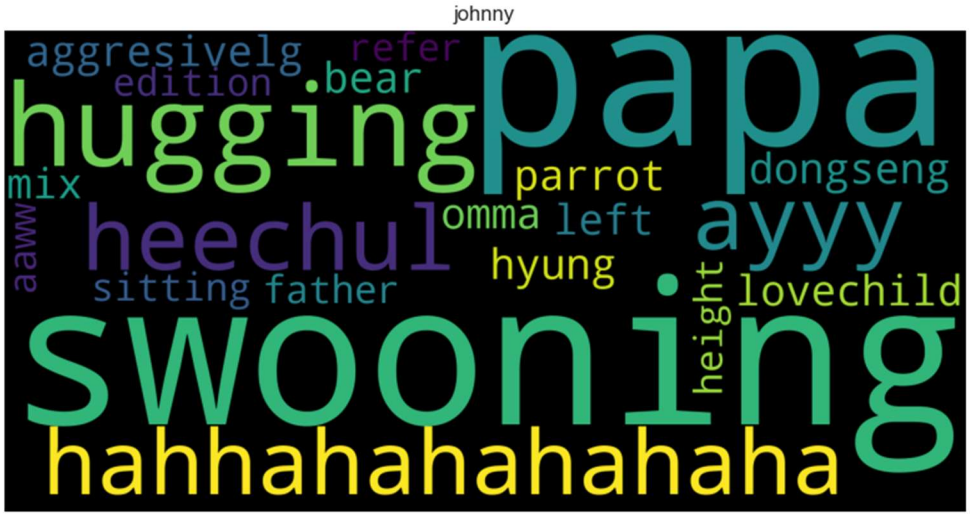
Top keywords for Chenle: transfer, shouted, echo, wisely, scaredy, age, call



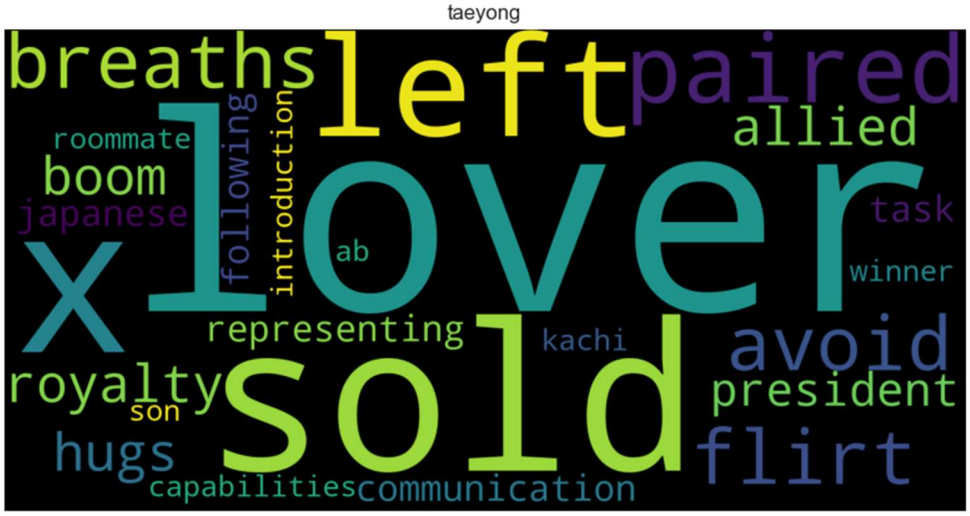
Top keywords for Jisung: unfair, maknae (youngest in Korean), younger, youngest, age, eat, buy, manly



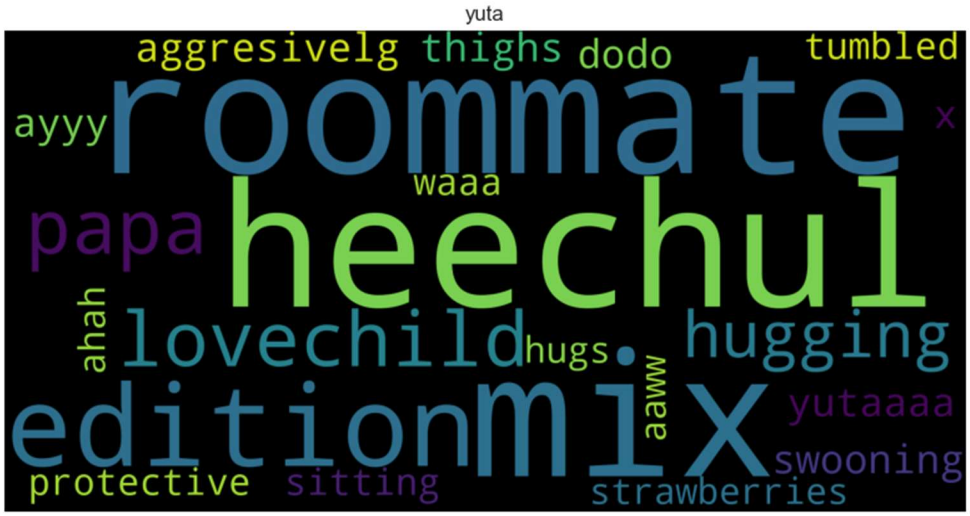
Top keywords for Taeil: aish (sound of irritation in Korean), judges, jealous, shouted, demonstration, vs, narrates, omma (mom in Korean)



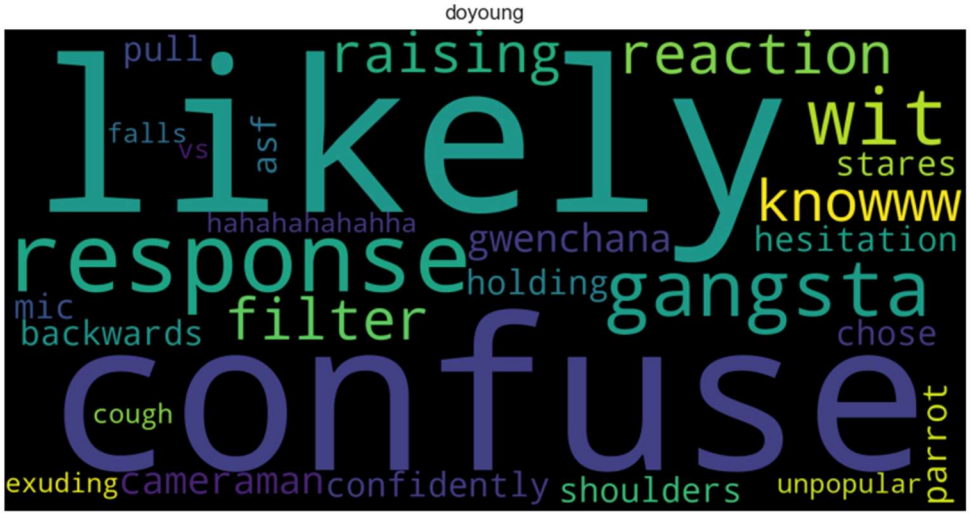
Top keywords for Johnny: aggressive, heechul (famous Korean celebrity), father, papa, swooning, hugging



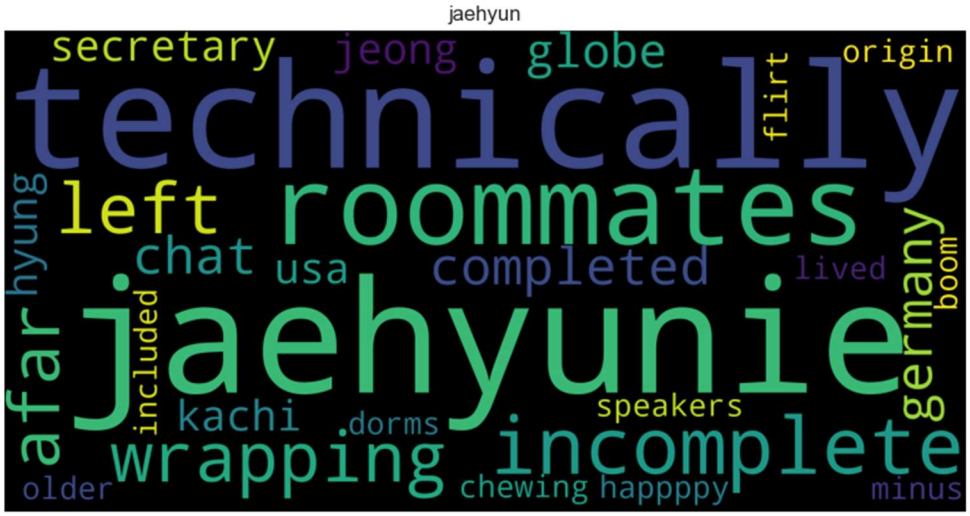
Top keywords for Taeyong: breaths, paired, lover, sold, hugs, flirt, capabilities, communication, president, winner, allied



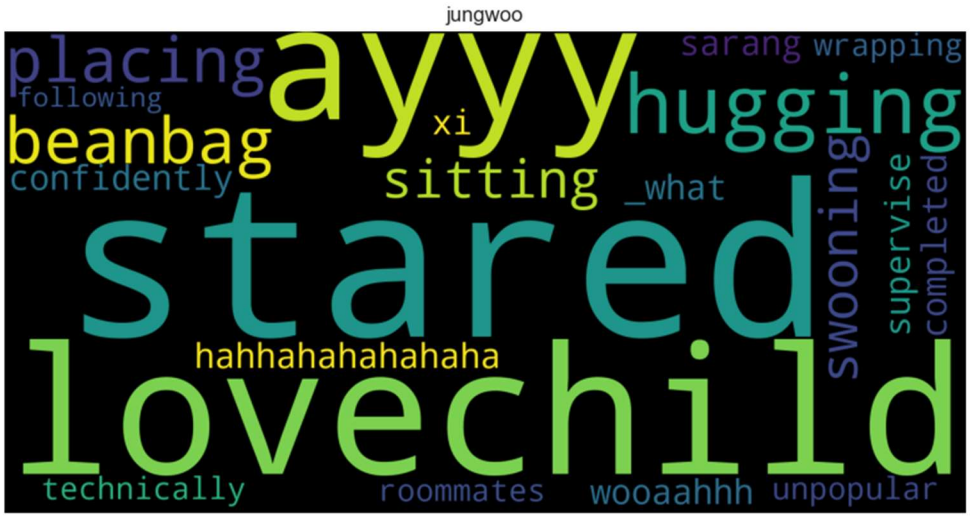
Top keywords for Yuta: dodo, tumbled, thighs, papa, heechul (famous Korean celebrity), protective, strawberries, hugging



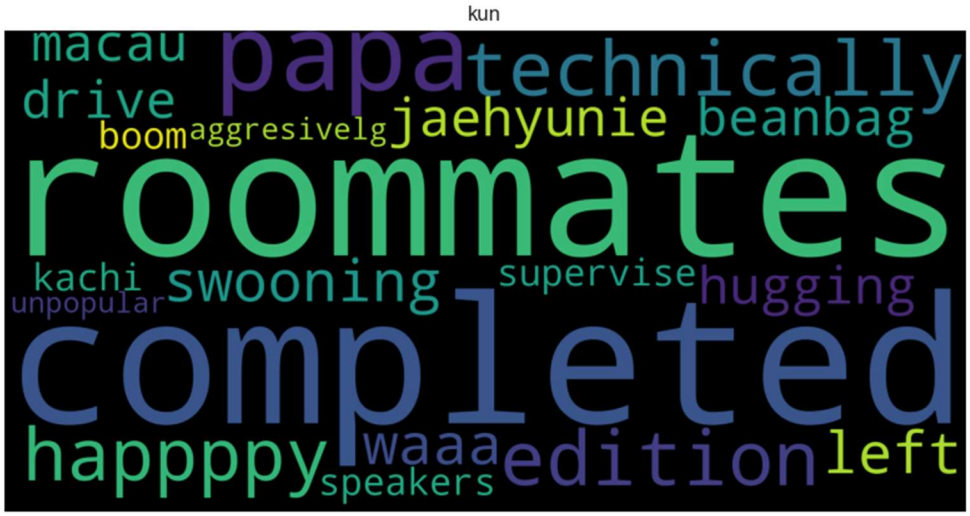
Top keywords for Doyoung: raising, reaction, mic, backwards, confuse, cough, shoulders, unpopular, hesitation, filter



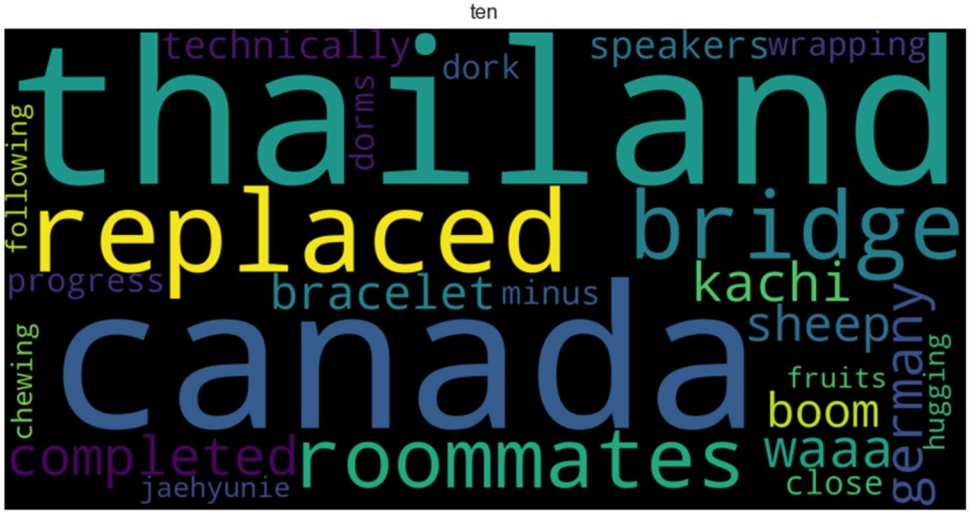
Top keywords for Jaehyun: secretary, globa, usa, included, lived, flirt, origin, older, chat, incomplete



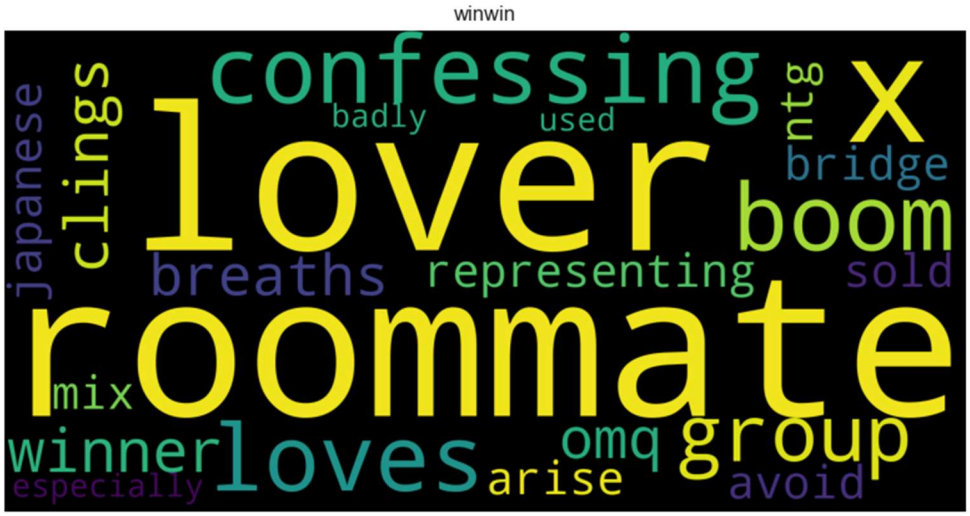
Top keywords for Jungwoo: sarang (love in Korean), hugging, stared, lovechild, confidently



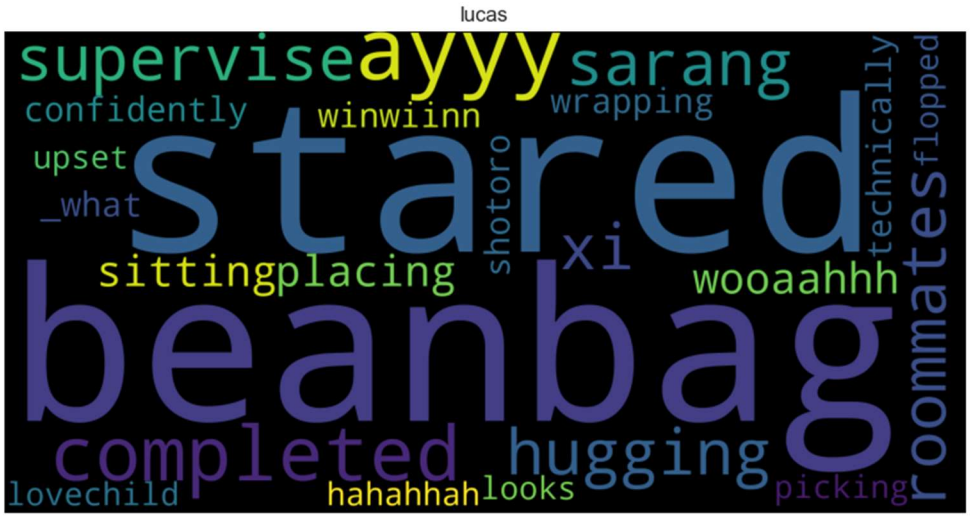
Top keywords for Kun: macau, papa, driver, edition, supervise



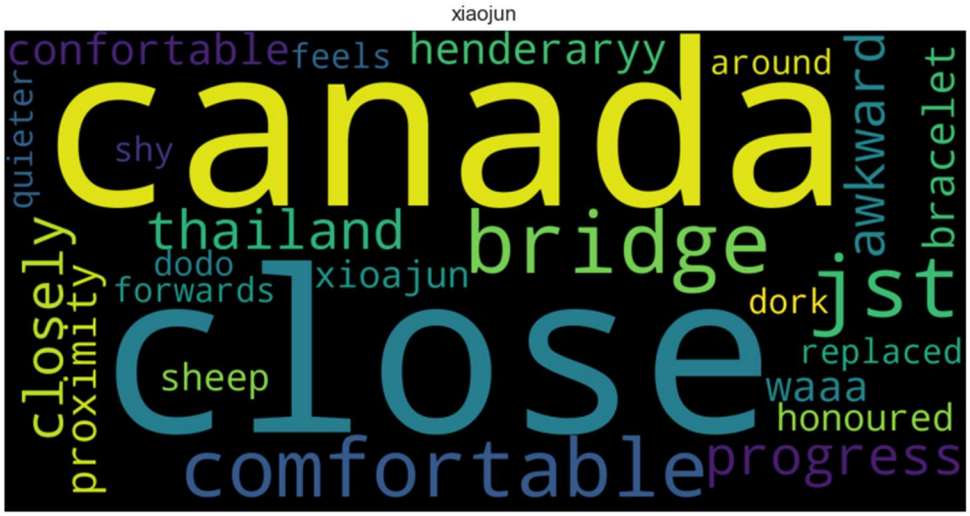
Top keywords for Ten: Thailand, chewing, bracelet, fruits, germany, speakers, canada, dorms, replaced



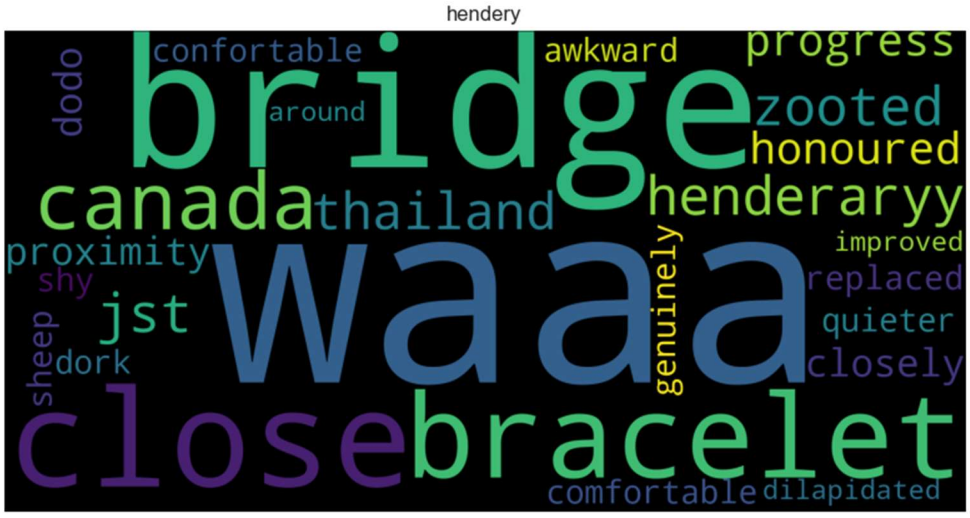
Top keywords for Winwin: clings, roommate, confessing, badly, winner, loves, avoid, lover, group, representing



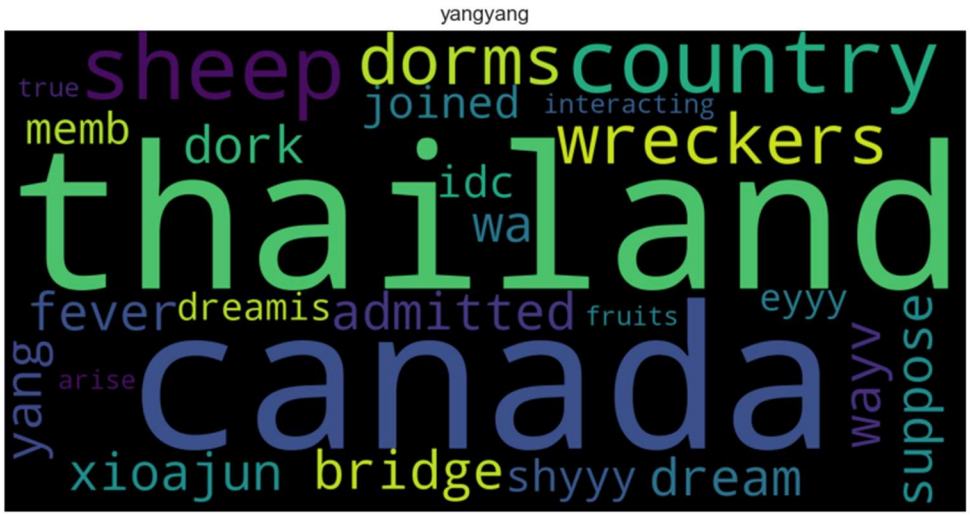
Top keywords for Lucas: stared, beanbag, sitting, flopped, wrapping, sarang (love in Korean), looks, hugging, stared



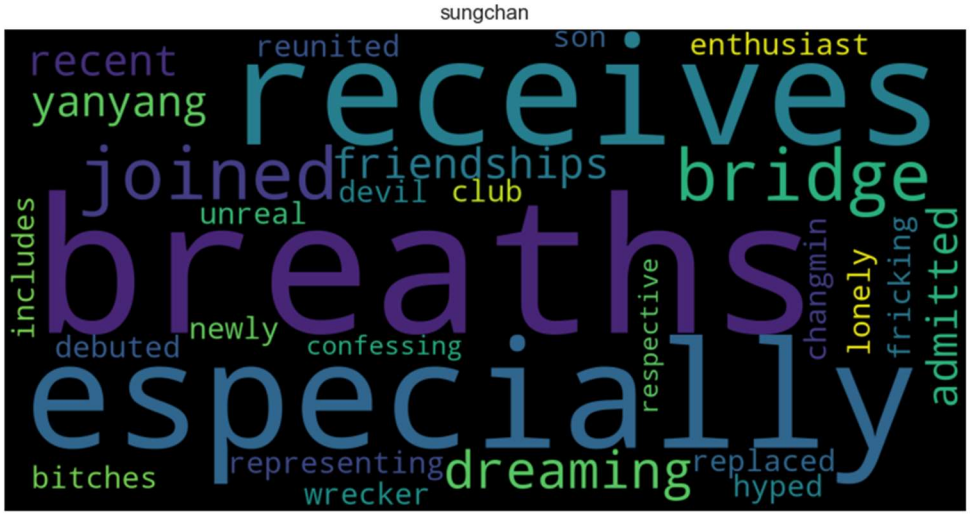
Top keywords for Xiaojun: comfortable, bridge, close, bracelet, progress,



Top keywords for Hendery: shy, proximity, close, dork, bracelet, thailand, canada, awkward, bridge, honored, improved, quieter. progress



Top keywords for Yanyang: country, Thailand, Canada, dream, fever, interacting, fruits, admitted, joined



Top keywords for Sungchan: recent, newly, friendships, bridge, debuted, wrecker, hyped, breaths

Top keywords for Shotaro: friendships, confessing, adore, sheep, shy, improved, family, breathes, endearing, antics

With each of these top keywords per member, the company can easily pick out what kind of image each member has.

Now below, I will include the visualization indicating close associations between members:

member	associated member
mark	jeno
renjun	jeno
jeno	renjun
haechan	mark
jaemin	jeno
chenle	jisung
jisung	chenle
taeil	mark
johnny	yuta
taeyong	johnny
yuta	johnny
doyoung	renjun
jaehyun	ten
jungwoo	lucas
kun	johnny
ten	lucas
winwin	taeyong
lucas	jungwoo
xiaojun	hendery
hendery	xiaojun
yangyang	ten
sungchan	shotaro
shotaro	sungchan

A lot of associations are expected but there are quite a few surprising as well. Due to there being many members, I will only state the ones that are considered to be surprising/ usually less seen together, with my knowledge of which members have what kinds of activities, interactions, and what subgroups each member is in. For easy reference, I have listed all the subgroups.

NCT Dream: Mark, Haechan, Renjun, Jeno, Jaemin, Chenle, Jisung

NCT 127: Taeil, Johnny, Taeyong, Yuta, Doyoung, Jaehyun, Jungwoo, Mark, Haechan

WayV: Kun, Ten, Winwin, Lucas, Xiaojun, Hendery, Yangyang

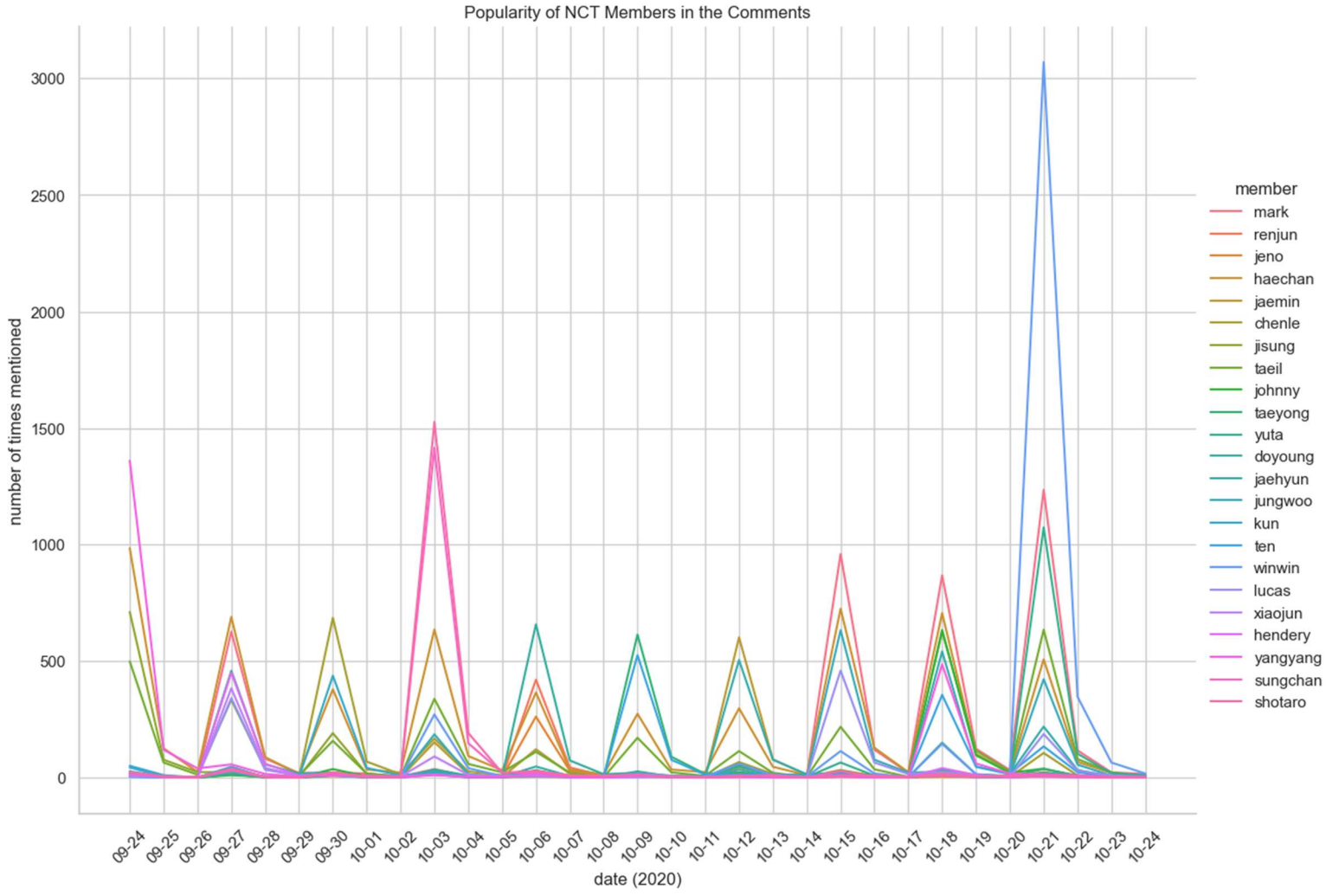
SuperM: Mark, Taeyong, Ten, Lucas

No group: Sungchan, Shotaro

- **Mark and Jeno:** Despite being in the same group, these two don't interact much outside of their group, fans may want to see more interaction
- **Taeil and Mark:** same reason as Mark and Jeno
- **Doyoung and Renjun:** This is new pairing. Recently there was a new content video focusing on members that are awkward together, and these two hit it off well together. This association may indicate that fans may appreciate watching more content with them together, despite them coming from two different subgroups.
- **Jaehyun and Ten:** Jaehyun and Ten are in different subgroups, but have had a few interactions nevertheless, especially with their commonality of being fluent in English. Fans may want to see more interaction with them speaking English together, as well as with other English-speaking members like Mark, Johnny, and Yangyang.
- **Kun and Johnny:** I have honestly no idea what to make of this, but there must be some reason for them to be paired together. Could be just because fans like seeing more interactions between members who don't meet often.

The rest of the associations are expected to my knowledge.

Below is the visualization of the popularity of members over the releasing dates of the videos.



What is the most surprising is the Winwin had a very big popularity boost in the last video, with over 3000 mentions in the comments in one day. Members who had 1000-1500 mentions in one day are: Yangyang, Sungchan, Shotaro, Mark, and Yuta. Members who had between 500-1000 comments in one day are: Jisung, Haechan, Mark, Chenle, Doyoung, Taeyong, Ten, Jaehyun, Jungwoo, Johnny, and Taeil.

With this information, the entertainment company can deem which members they may want to promote more to increase their popularity, or members they may continue promote because of their popularity.

Work Plan Evaluation:

My work plan I created in part 1 was split into three parts, just like in the methodology section: getting the data, cleaning the data, and analyzing the data. I had already completed getting the data and cleaning the data before turning in part1, which took about 6 hours and 4 hours respectively. I had also completed a good chunk of analyzing the data and estimated the total amount of time to complete it all would be ten hours. I didn't exactly time myself on completing the project, but I believe it was around maybe 12 hours. I added the extra 2 hours (I originally predicted 10) because I didn't realize that nearing the end of the project, when I started running my program over all the data files instead of a few, it would take a much longer time to process. To work efficiently, I ran it and always worked on homework for other classes while waiting. I would check back about 1 hour later to see if it finished running. However, I didn't really time myself, so these are all just guesses. I started my project quite early so I never felt pressured for time.

Testing:

To test my code, I created a small clean test file of data called clean_test.csv, with a smaller set of members to test with. I used these on all my methods to test if they produced what I wanted correctly. I tested this at the bottom of the main in analyze_data.py, and it is now commented out. Visualizations created from this test will not be provided in my results directory, as many names of the files produced are the same as produced by the results with the real data files. However, I have provided the visualizations created by the test file below. To recreate these visualizations, comment everything in main, and uncomment the testing section at the bottom of the main.

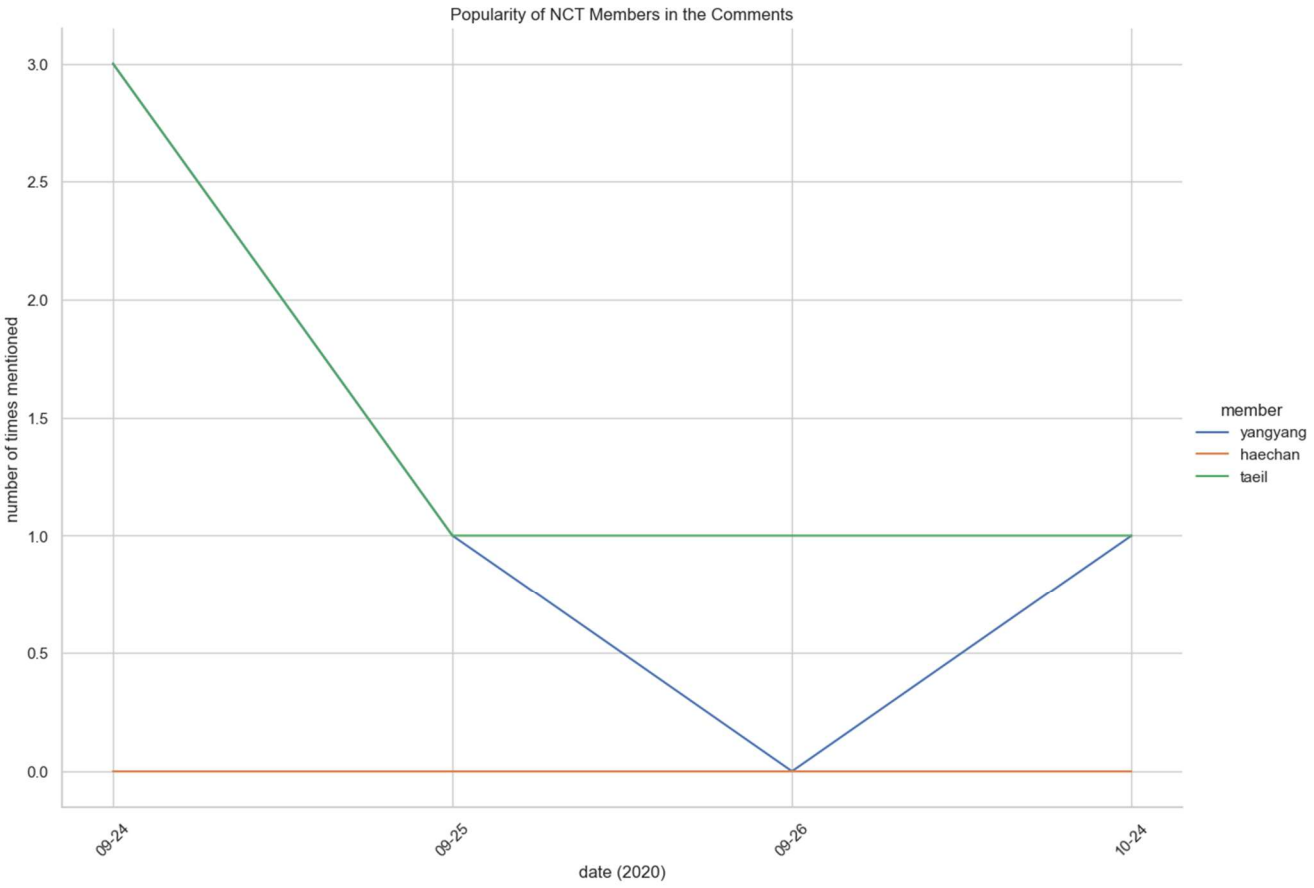
Using the test code, I could see that my visualizations were produced beautifully in the way I wanted. I also could accurately count each member's name per range of dates expected to graph and check my popularity visualization, which produced to be accurate. To check the accuracy of the association table, I place two of the member's names, taeil and yangyang, almost always next to each other in my test data file, and placed one member's name, haechan, by itself once. This way I could easily verify association as taeil and yangyang should definitely be associated with each other both ways, and haechan wouldn't have enough data to find an association. As for checking keywords, since I used the data as both as one episode's worth of data and "all episode's" worth of data, I expected the keywords for each member for the episode and for each member over "all" the episodes to be the same, which proved to be correct.

However, there is no way to decisively verify that the keywords themselves are accurate, but by skimming through them they do seem to use the words and fit the data I created. Pictures below show the test data and results that they produce.

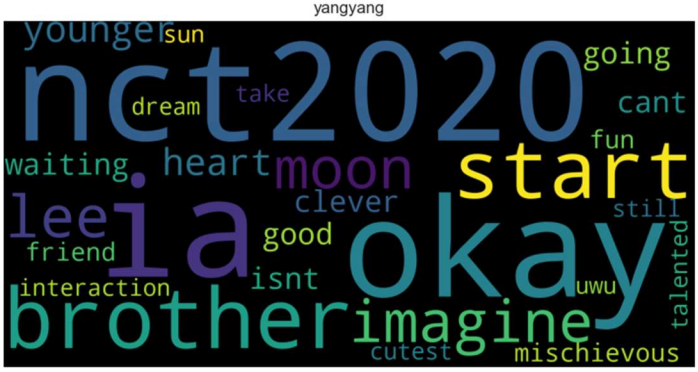
data > clean_test.csv

```
,text,date_published
7828,always cutest yangyangie,2020-09-24
7830,imagine going get start heart cant take,2020-09-24
7833,one week friend talking ia gonna seeing wayv members dream 127 believe,2020-09-24
7835,isnt real,2020-09-24
7838,sun moon cute help,2020-09-24
7840,omg soooo possible see wayv awkward okay waaaah gonna fun uwu taeil yangyang,2020-09-24
7842,nct wayv interaction taeil yangyang yes want,2020-09-24
7846,wayv members seriously take good care taeil yangyang even really crack among,2020-09-24
7847,wished kind content still cant believe taeil yangyang,2020-09-25
7848,even though kinda short thank content taeil 3,2020-09-26
7853,uwu cute adorable taeil yangyang,2020-10-24
7854,continue mischievous guys stay cute clever witty stay talented younger brother lee haechan ace nct2020,2020-10-25
7855,1 48 talk cute taeil yangyang waiting called,2021-09-24
7856,one think taeil yangyang adorable,2021-10-24
```

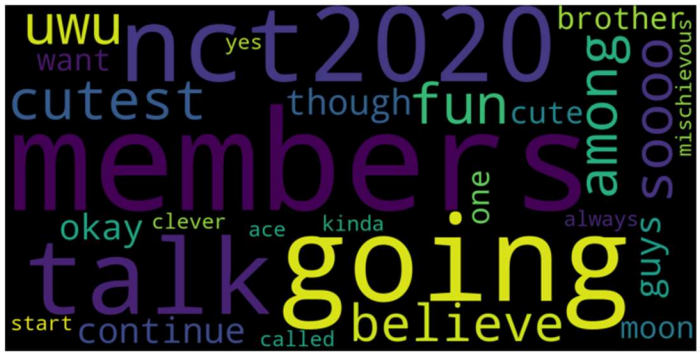
member	associated member
yangyang	taeil
haechan	
taeil	yangyang



yangyang

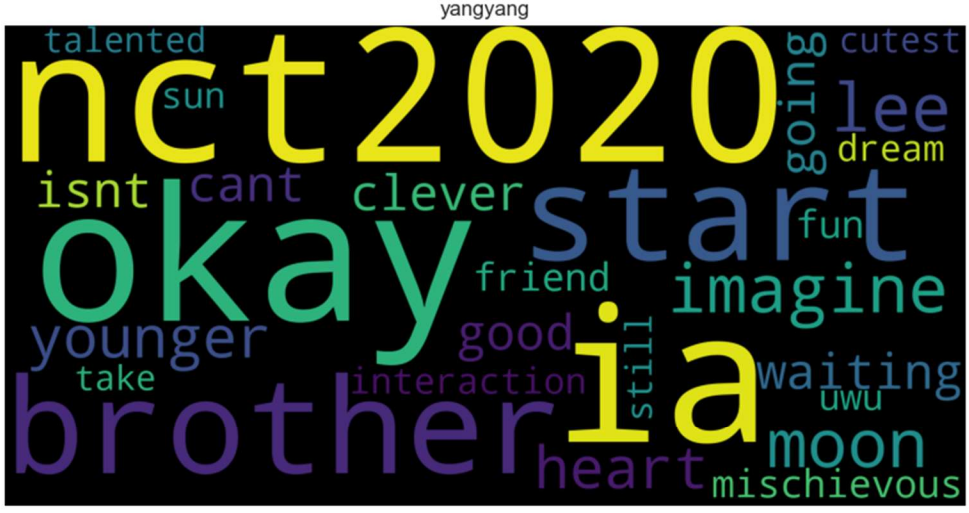


haechan

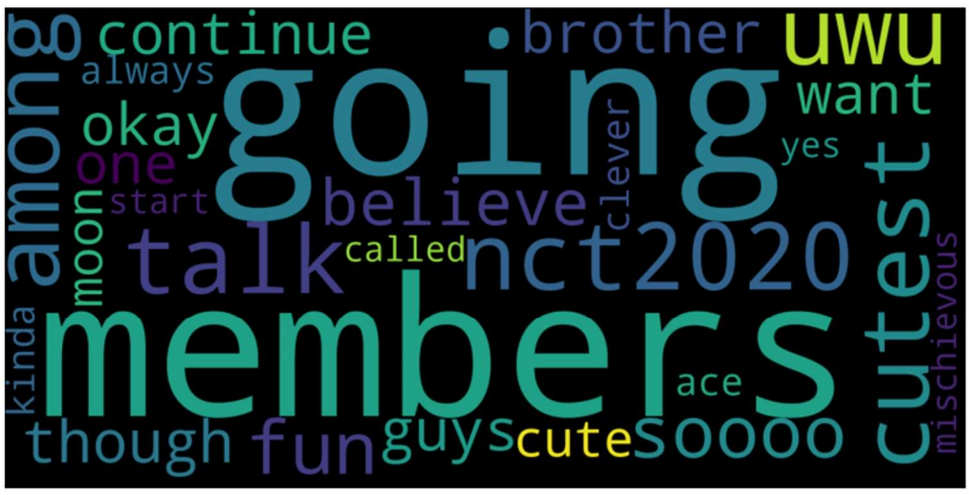


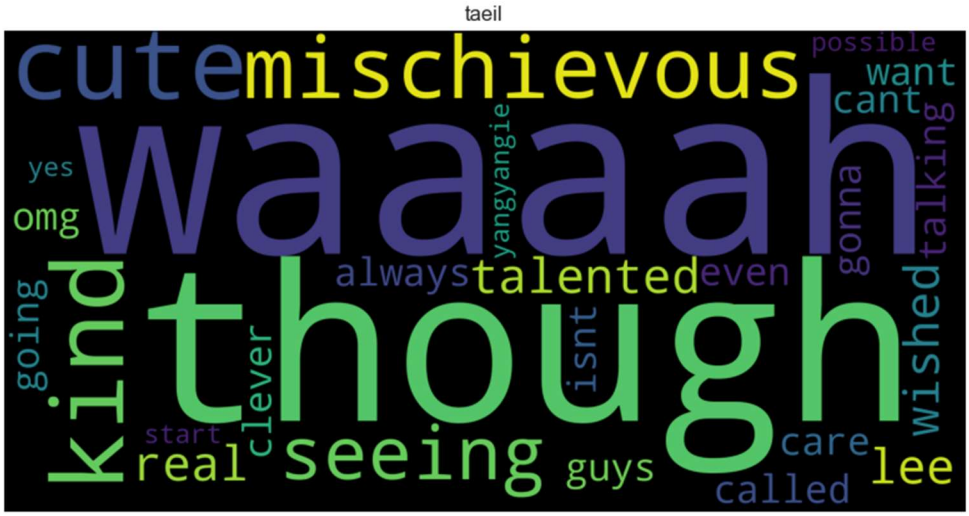
taeil

yangyang



haechan





Collaboration:

I did not collaborate with anyone for this project. I did reference the documentations quite frequently for all the libraries I imported, including the libraries we learned about in class, as well as previous lessons from class to remember syntax and methods we had used previously.

At one point, I needed a clean way to break out of a for loop to prevent an index out of bounds. Searching up google, I discovered the break statement and used this. I am stating this here as the concept was something not covered in our class. I, however, did not keep track, nor remember, exactly which resource helped me to understand this concept. I also did this for exceptions, as I have never learned about that before and needed someway to bypass when *langdetect* was unable to figure out what language the text was in and caused an error.

To create the API key I needed for the project I used the clear instructions from https://blog.hubspot.com/website/how-to-get-youtube-api-key.

From this page, which is a part of the YouTube Data API, https://developers.google.com/youtube/v3/docs/commentThreads/list, I scrolled to bottom of the "Try this API" section on the right, and then clicked showed code. I then went to the Python section and copied this code to use in my get_data.py module and altered it to fit it to my use.

Conclusion:

In conclusion, the Word2Vec model I used is accurate for finding keywords of the members over all the videos. It, however, does not do a good job of finding good keywords for members, per video. The methods I used to find associations between members and popularity seem to be very accurate as well. For future improvement to this project, instead of finding top keywords for members, per video, it may be more beneficial to come up with general keywords per video, not

associated to a specific member. It would also be interesting to look at the use of different languages in the comments and see what kind of audience these videos have reached.