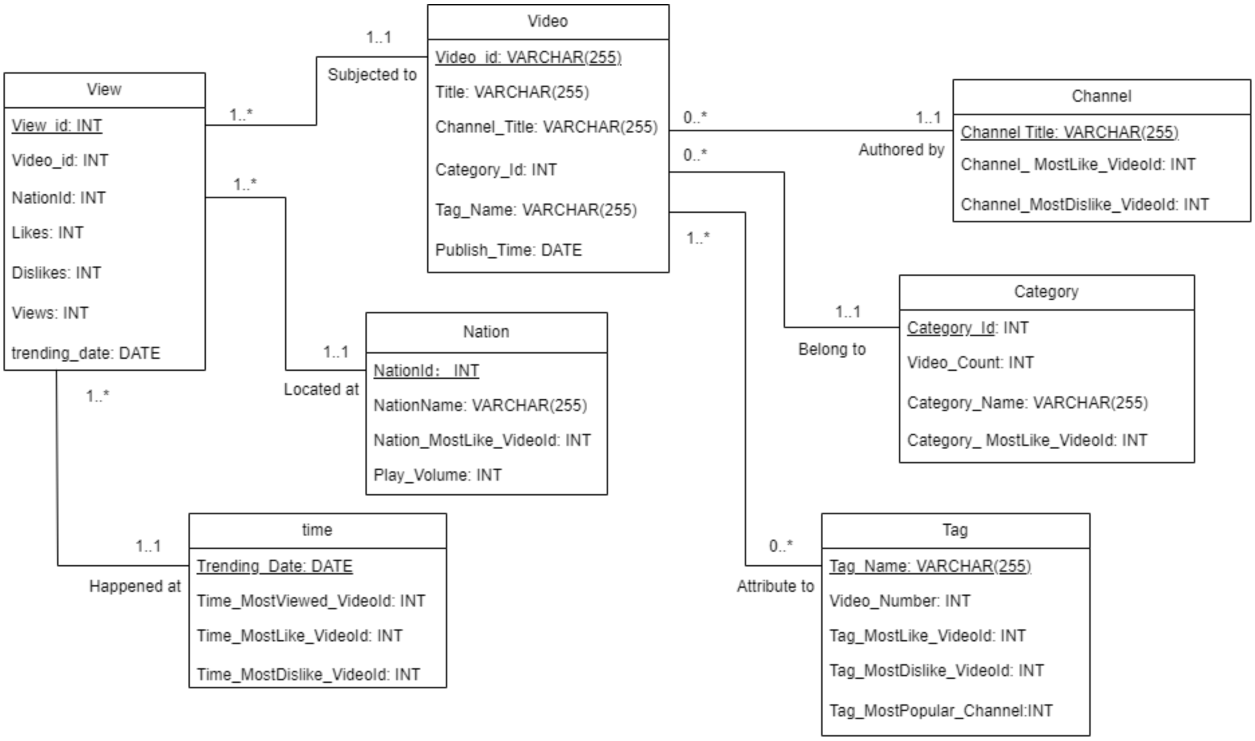
1. UML Diagram:



2. Normalization & show BCNF

In entity View, Regard View_Id as A’, Video_id as A, NationId as B’, Likes as C’,Dislikes as D’, Views as E’, trending_date as F’

In entity Video, Regard video_id as A,Title as B, Channel_Title as C, Category_Id as D, Tag_Name as E,Publish_Time as F

In entity time, regard Trending_date as F’,Time_mostviewed_videoid as A’’, Time_MostLike_videoId as B’’, Time_MostDislike_videoId as C’’

In entity Nation, regard NationId as B’, NationName as A’’’, Nation_MostLike_VideoId as B’’’, Play_Volume as C’’’.

In entity Channel, regard Channel_Title as C, Channel_MostLike_VideoId as A#,Channel_MostDislike_VideoId as B#

In entity Category, regard Category_Id as D, Video_Count as A*, Category_Name as B*, Category_MostLike_VideoId as C*.

In entity Tag, regard Tag_Name as E, Video_Number as A~, Tag_MostLike_VideoId as B~, Tag_MostDislike_VideoId as C~, and Tag_MostPopular_Channel as D~.

Functional Dependency:
A->BCDE
A’->AB’C’D’E’F
F’->A’’B’’C’’
B’->A’’’B’’’C’’’
C->A# B#
D->A*B*C*
E->A~B~C~D~

Minimal Basis:
A->B, A->C, A->D, A->E
A’->A, A’->B’, A’->C’, A’->D’, A’->E’, A’->F
F’->A’’, F’->B’’, F’->C’’
B’->A’’’, B’->B’’’, B’->C’’’
C->A#, C->B#
D->A*, D->B*, D->C*
E->A~, E->B~, E->C~, E->D~

So in all tables of our UML, there is no data element in the table that depends on other non-keyword data elements in other tables. So our UML satisfies 3NF. Moreover, the relation in our database also satisfies the condition that given a relation, X is a superkey for every functional dependency (FD) X->Y. And since our relation is in 3NF, so it satisfies conditions of BCNF.

The reason why we choose BCNF: BCNF can eliminate unnecessary functional dependencies existing in 3NF, so it will reduce the redundancy of data, decreasing update error and enhance the consistency and completeness of data.

3. Relational Schema Definition

View (
    View_Id: INT [PK],
    Video_Id: INT [FK to Video.Video_Id],
    Nation_Id: INT [FK to Nation.Nation_Id],
    Likes: INT,
    Dislikes: INT,
    Views: INT,

```
        Trending_Date: DATE() [FK to Time.Trending_Date]
)

Nation (
        Nation_Id: INT [PK],
        Nation_Name: VARCHAR(255),
        Nation_MostLike_VideoId: INT,
        Play_Volume: INT
)

Time (
        Trending_Date: DATE() [PK],
        Time_MostViewed_VideoId: INT,
        Time_MostLike_VideoId: INT,
        Time_MostDislike_VideoId: INT
)

Video (
        Video_Id: INT [PK],
        Title: VARCHAR(255),
        Channel_Title: VARCHAR(255) [FK to Channel.Channel_Title],
        Category_Id: INT [FK to Category.Category_Id],
        Tag_Name: VARCHAR(255) [FK to Tag.Tag_Name],
        Push_Time: DATE()
)

Channel (
        Channel_Title: VARCHAR(255) [PK],
        Channel_ MostLike_VideoId: INT,
        Channel_MostDislike_VideoId: INT
)

Category(
        Category_Id: INT [PK],
        Video_Count: INT,
        Category_Name: VARCHAR(255),
        Category_ MostLike_VideoId: INT
)

Tag(
```

Tag_Name: VARCHAR(255) [PK],
Video_Number: INT,
Tag_MostLike_VideoId: INT,
Tag_MostDislike_VideoId: INT,
Tag_MostPopular_Channel: INT,
)

AttributeTO(
Tag_Name: VARCHAR(255) [PK],
Video_Id: INT [PK]
)

4. Description and Assumption of Relations

We have tables: View, Nation, Time, Video, Channel, Category and Tag.

Assumption:

1. We assume one view includes one video and corresponds to one nation, who views it.
2. We assume that one nation corresponds to multiple views because one nation's people could watch multiple videos.
3. We have one time corresponding to multiple views because the trending date for different videos and nations could be the same.
4. We assume one video corresponds to multiple views because one video could be watched by multiple nations.
5. We assume one video corresponds to only one channel, but one channel could have multiple videos.
6. We assume one video only has one category, but one category could contain multiple videos.
7. We assume that one video could have zero to multiple tags.

Description of relations:

1. View & Nation: One nation's people can view many videos and have one or more views. And one view includes one video which is viewed. In the Nation entity, it includes some data about the nation.
2. View & Time: One trending time can correspond to one or more videos or nations. But each view, each pair of nation and video, could only have one trending time. In the Time entity, it includes some data on that date.
3. View & Video: One video can be viewed by many nations and has one or more views. And one view includes one nation, who views it. In the Video entity, it includes some data about the video.
4. Video & Channel: The Channel entity contains channel information for each video. Each video has one channel. For each channel, it includes title and other information.
5. Video & Category: The Category entity contains category information for each video. Each video has one category. For each category, it includes some information, for example, name, total count of videos in this category.

6. Video & Tag: The Tag entity contains tag information for videos. One Tag must contain one or more videos, however, some videos may not have tags. For each Tag, it includes some information about this tag, for example, name, total count number of videos, most liked video in this tag.

Description of each entity:

View: entity View describes some property of each view, for example, view Id, which video is viewed, people in which country view it and time when viewing and so on.

Nation: entity Nation includes some property, for example, country Id name, what video is popular in this country and so on. We can directly get the information about one country directly.

Time: entity Time will tell you what happens in a particular time. Which video is most popular in that time.

Video: Entity Video is important, since it includes all properties for each video, like title, channel, published time.

Channel: Entity Chennel will tell you information for a channel. For example, you will know what is most popular in this channel.

Category: Like channel, entity category will also introduce some information for a category. You can know about particular category from it.

Tag: like channel, entity Tag has some information for each tag. For example, the most popular video in this tag, the total video number.