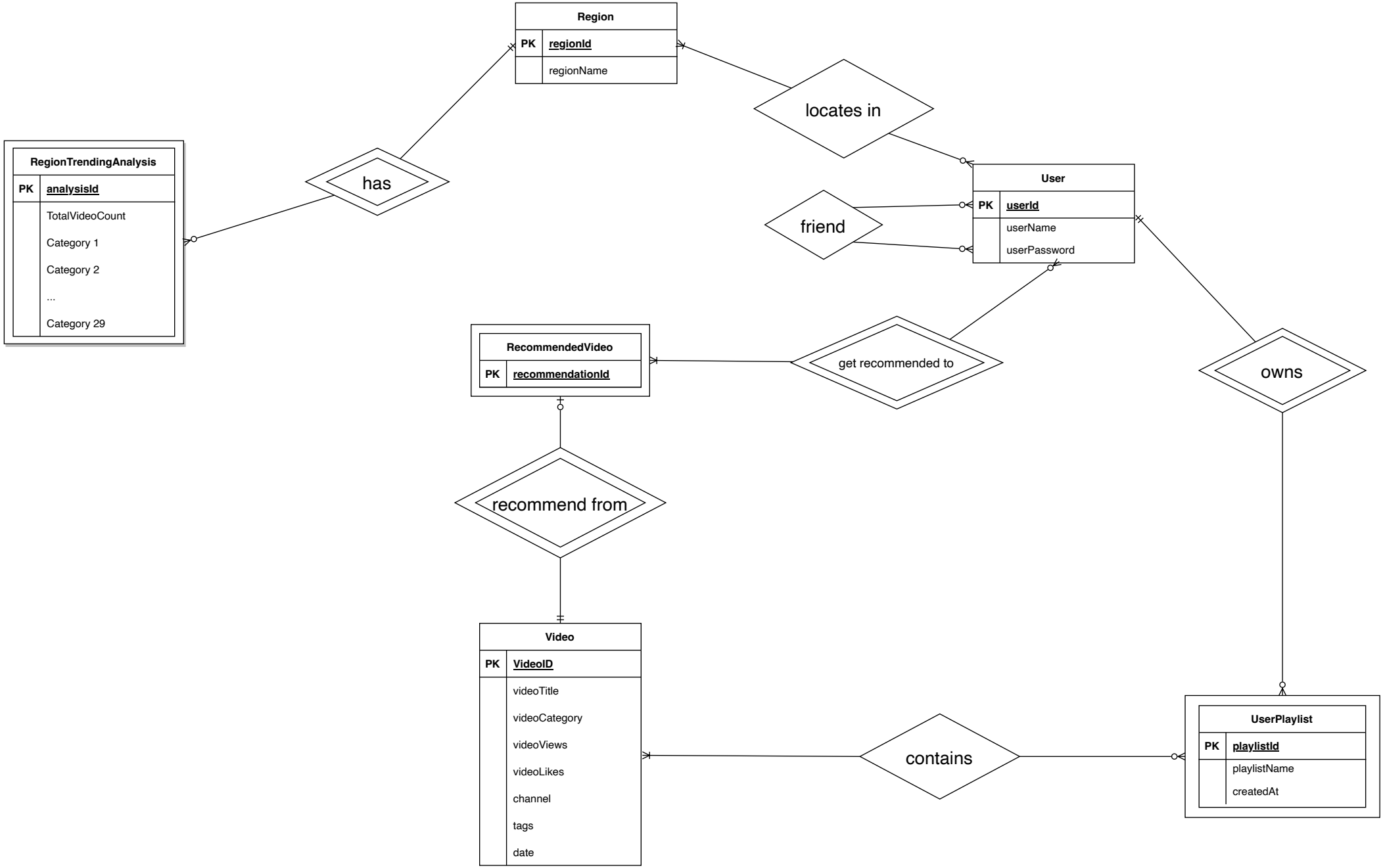
Region
PK
regionId
regionName
locates in
RegionTrendingAnalysis
PK
analysisId
TotalVideoCount
Category 1
Category 2
...
Category 29
has
User
PK
userId
userName
userPassword
friend
RecommendedVideo
PK
recommendationId
get recommended to
owns
recommend from
Video
PK
VideoID
videoTitle
videoCategory
videoViews
videoLikes
channel
tags
date
contains
UserPlaylist
PK
playlistId
playlistName
createdAt

BCNF

Our original diagram was already in BCNF. This is because all functional dependencies present in the relations are purely determined by the primary keys of those relations, meaning the left side is always a super key. Formally, for each relation:

User: userID -> userName, userPassword

UserPlaylist: playlistID -> playlistName, userID

Video: videoID -> videoTitle, videoCategory, videoView, videoLikes, channel, tags, date

RecommendedVideo: recommendationID -> videoID, userID

Region: regionID -> regionName

RegionTrendingAnalysis: analysisID -> regionID, totalVideoCount, category1, category2, ..., category29

Friend: Only two attributes

Contains: Only two attributes

GetRecommendedTo: Only two attributes

Entity Descriptions

User: Contains user login information.

Region: Specifies which regions to pull trending videos from.

User Playlist: A customized playlist containing the videos user put inside. This is a weak entity because each playlist is uniquely identified by playlistID and the userId of the user who created it.

Video: Instance of a single video. Can be recommended, and added to user playlists. User can also see more detailed information like channel, likes about the video.

Recommended Video: Instance of single recommended video that has been recommended to the user. This is a weak entity because each recommended video is uniquely identified by its recommendationID, userID of the user being recommended to, and a videoID of the video being recommended.

RegionTrendingAnalysis: Instance of single region trending analysis that been regenerated for each region. This is a weak entity because each analysis video is uniquely identified by its analysisId, and regionID of the region being analyzed.

Relationship Assumption

(* = any number of)

- Each user can have 0 to * Friends to other users
- Each user can have 0 to * user playlists and each user playlist belongs to exactly 1 user
- Each user playlist contains 1 to * videos and each trending video can belong to 0 to * user playlists
- Each video is recommended to 0 to 1 recommended video and each recommended video is recommended from exactly 1 video
- Each recommended video corresponds to 1 to * users and each user get recommend to 1 to * recommended videos
- Each user locates in exactly 1 region and each region has 0 to * users
- Each region has exactly 1 region trending and each region trending corresponds to exactly 1 region

Relational Schema

Entities:

- User(userID INT [PK], userName VARCHAR(20), userPassword VARCHAR(20))

- UserPlaylist(playlistID INT [PK], playlistName VARCHAR(30), userID INT [PK][FK to User.userID],)
- Video(videoID VARCHAR(11) [PK], videoTitle VARCHAR(50), videoCategory INT, videoView INT, videoLikes INT, channel VARCHAR(50), tags VARCHAR(100), Date DATE)
- RecommendedVideo(recommendationID INT [PK], VideoID VARCHAR(11) [PK][FK to Video.videoID], userID INT [PK][FK to User.userID])
- Region(regionID INT [PK], regionName VARCHAR(20))
- RegionTrendingAnalysis(analysisID VARCHAR(20) [PK]), regionID INT [PK][FK to Region.regionID], TotalvideoCount INT, Category1 INT, Category2 INT, …… , Category29 INT,)

Many-to-Many Relationships:

- Friend(userId INT, userId INT)
- Contains(VideoID VARCHAR(11), playlistID INT)
- GetRecommendTo(userID INT, recommendationID INT)