Meta Kaggle

GROUP 4

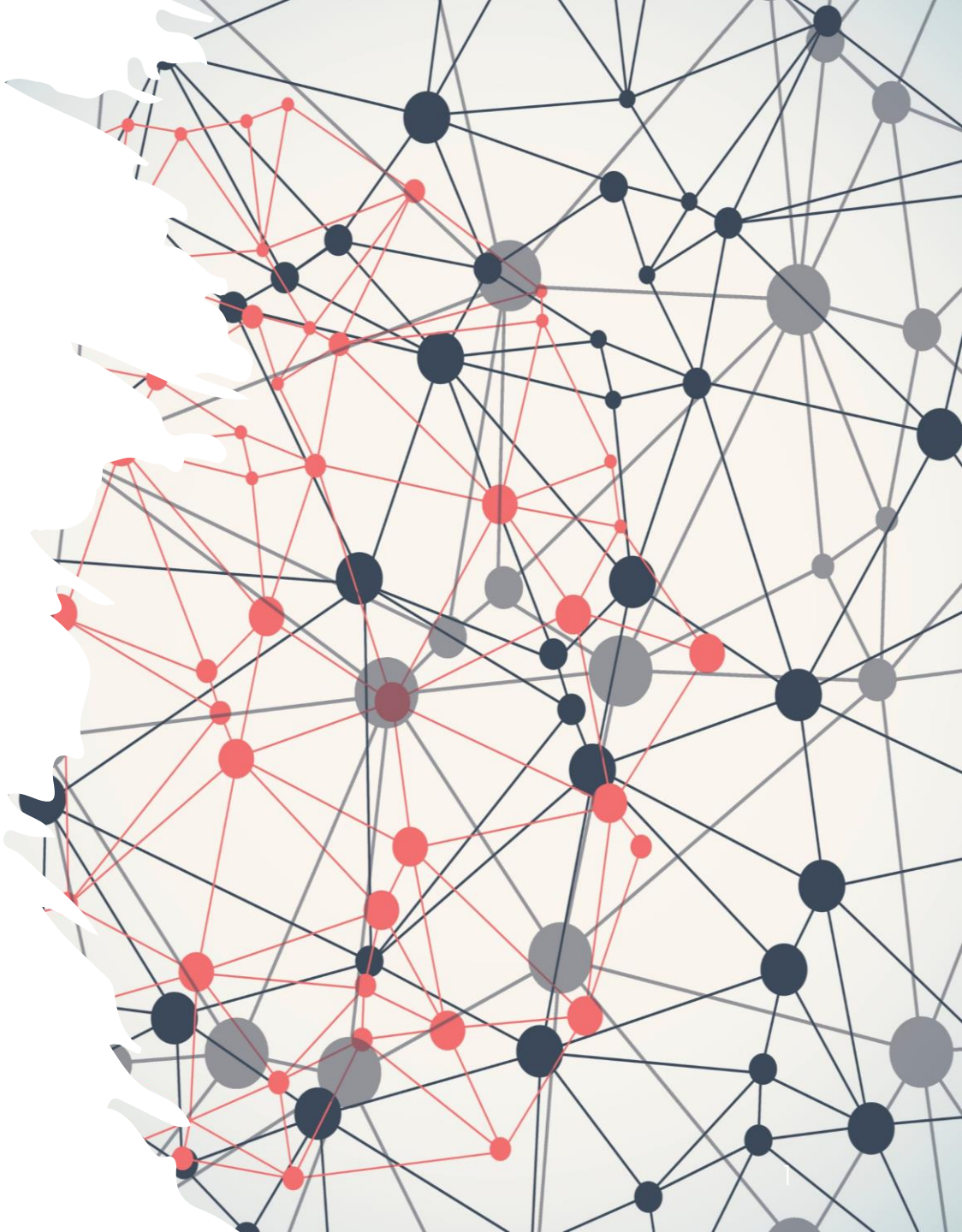
AAKRITI LNU

GOKULA NAVEEN

MUHAMMAD RAEES

PRAJJWAL MEHTA

CSCI 620



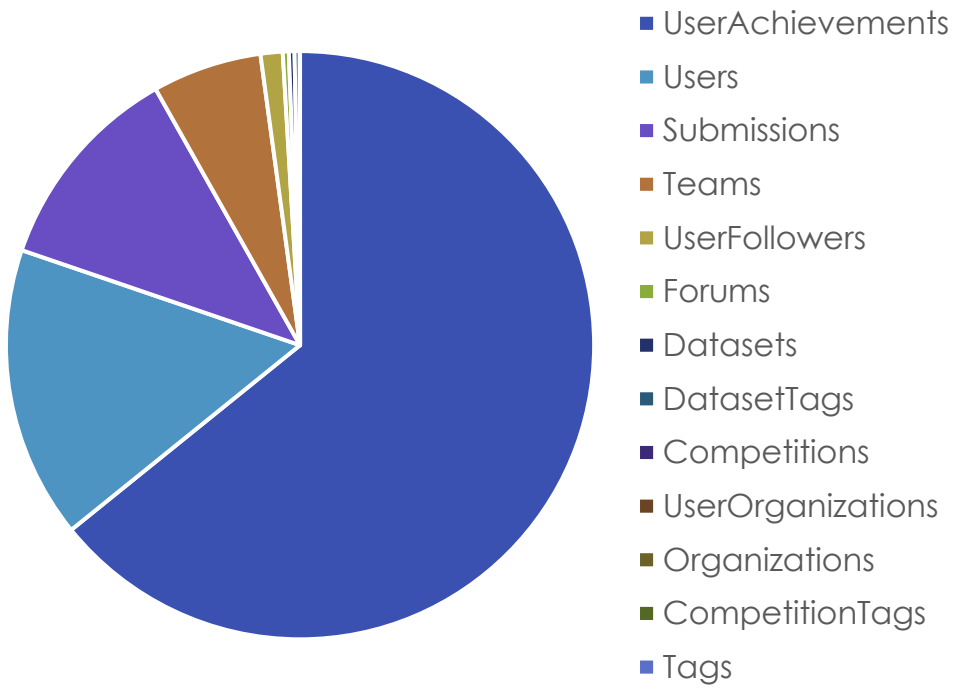
Project Overview

- This project explores the Meta Kaggle dataset
  - View of the Kaggle community and platform activities
  - Users, Competitions, Datasets, and interactions between them
  - Executed in three phases

Dataset Overview

- ***Dataset:*** *Meta Kaggle*
  - *Kaggle's platform activities*
  - Data is provided in CSV files
  - *https://www.kaggle.com/datasets/kaggle/meta-kaggle*

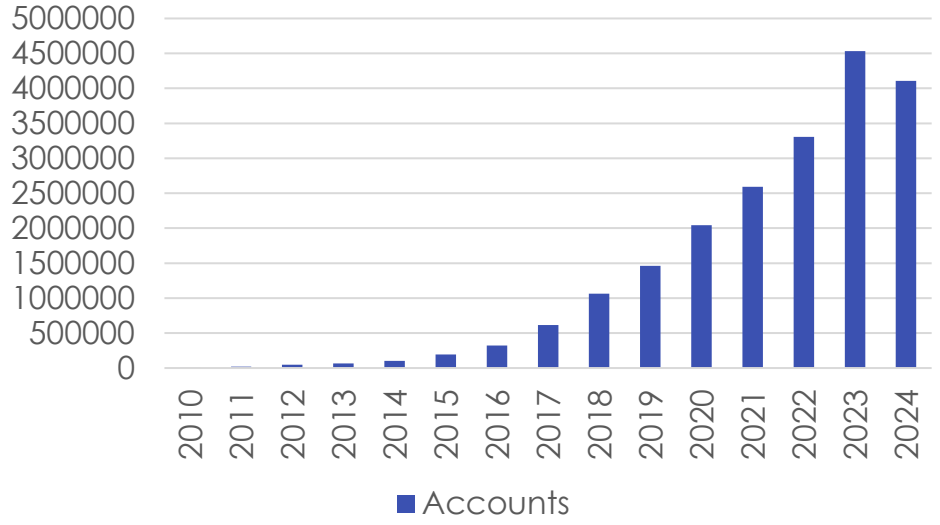
- ***Size:*** *Large (above billion)*
  - Subset **(127 *million)*** taken to make the project manageable

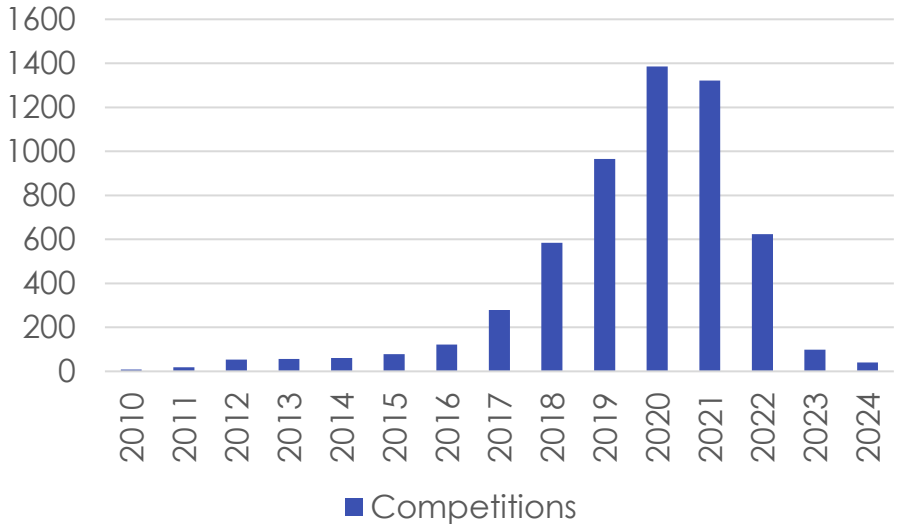


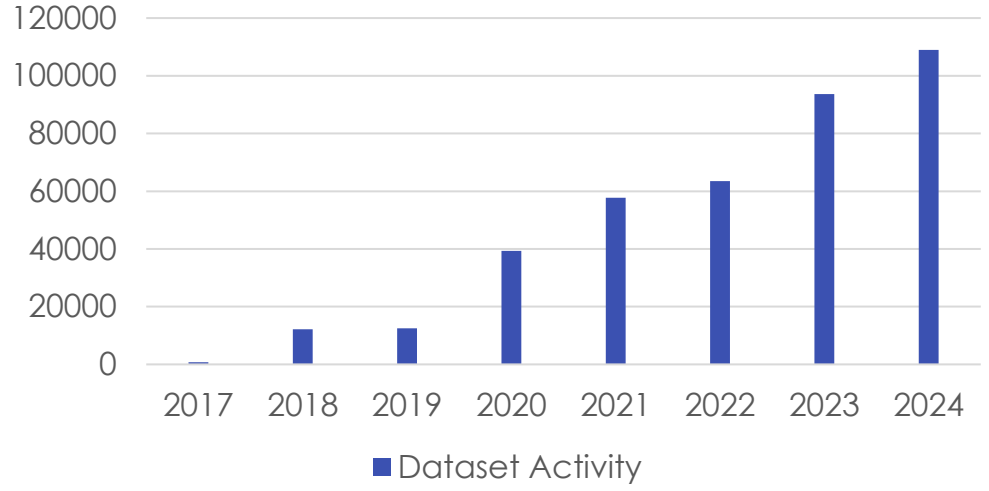
Dataset Overview

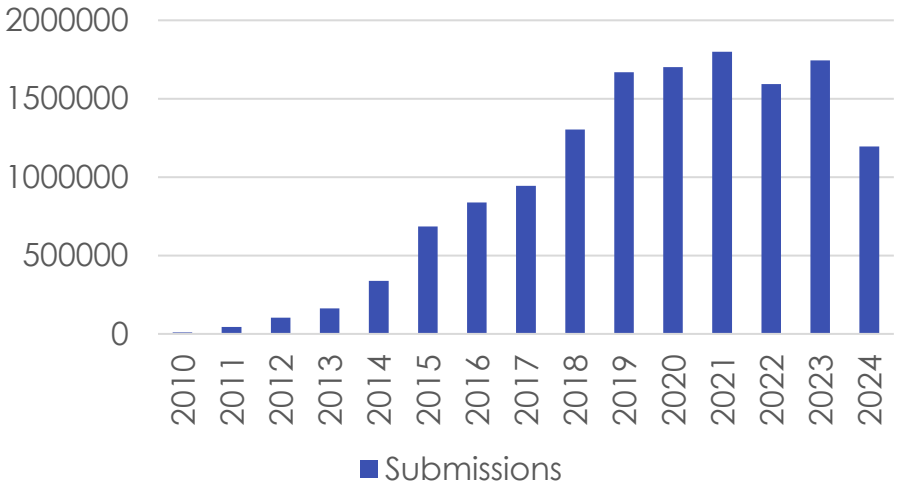
- **User Engagement**
  - User growth, competition participation, and submission trends
- **Competitions**
  - Competitions hosted, participants, and how rewards correlate with participation levels
- **Submissions**
  - Evaluate their timing and correlate them with user or team performance
- **Datasets**
  - Trends in dataset creation and usage to understand which datasets are most valuable to the community
- **Other Insights**
  - e.g., scores, algorithms, badges

Data Insights









Users

- Username (Display Name)
- Registration Date
- Organizations
  - Join Date
- Followers
  - Following Date
- Achievements
  - Achievement Date
  - Points
  - Rankings
  - Badges (Gold, Silver, Bronze)

Competitions and Datasets

- Competitions
  - Tag
  - Forum
  - Creation and Deadline
  - Prizes
  - Teams
  - Submissions
  - Evaluation
- Datasets
  - Tag
  - Forum
  - User
  - Creation and Last Activity
  - Views
  - Download
  - Votes

Teams, Submission, Tags, and Forums

- Teams
  - Name
  - Team Leader
  - Competition
- Submission
  - Submitter
  - Date
  - Public and Private Score
- Tags
  - Tag Name
  - Parent Tag
- Forums
  - Forum Name
  - Parent Forum

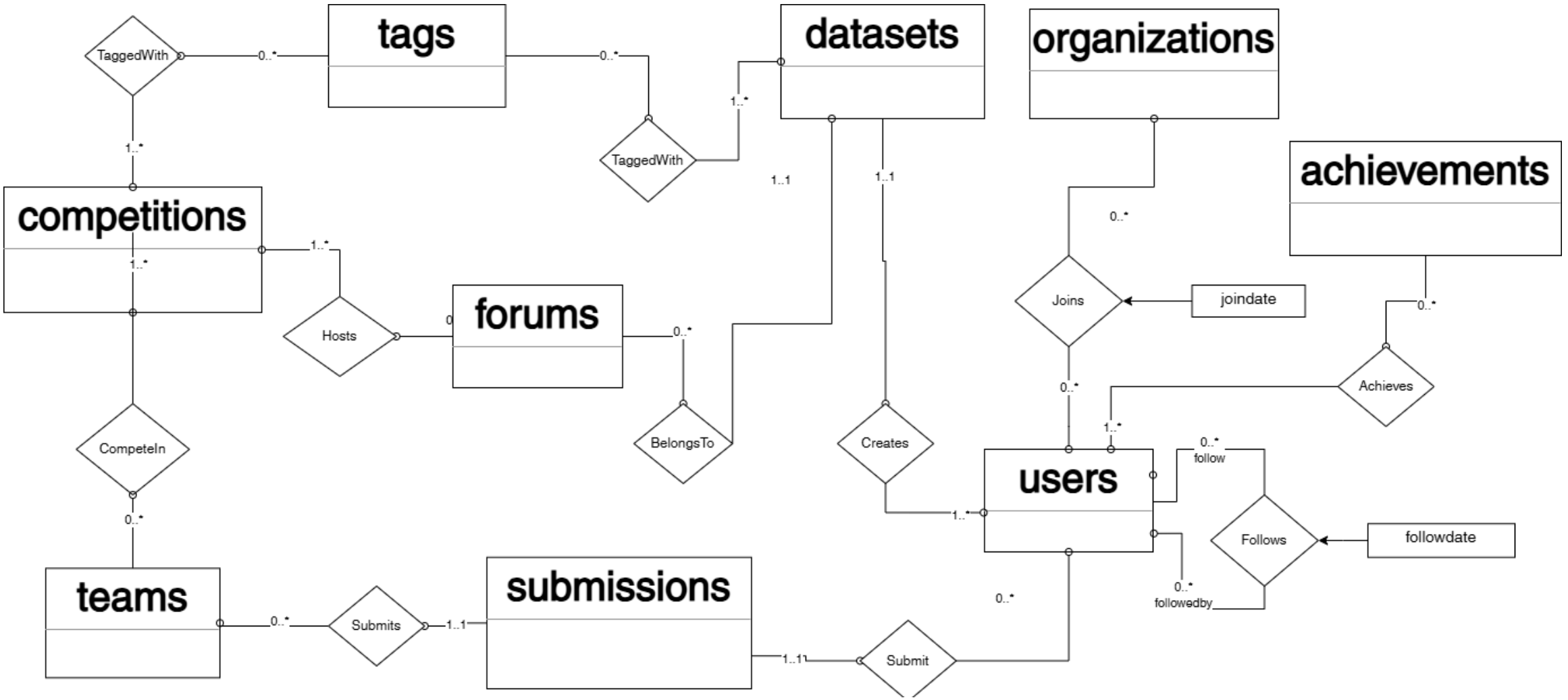
Phase I

RELATIONAL MODEL

Main Entities

- Users
- Achievements
- Forums
- Teams
- Competitions
- Submissions
- Tags
- Datasets
- Organizations

ERD


tags
datasets
organizations
achievements
competitions
forums
users
teams
submissions
TaggedWith
TaggedWith
Hosts
BelongsTo
Creates
Joins
joindate
Achieves
CompeteIn
follow
Follows
followdate
followedby
Submits
Submit
0..*
1..*
1..1

Relations

Relation	Description
Users	Kaggle users, including their performance tiers and registration data
User Achievements	Achievements of Kaggle users, including rankings, points, and medal counts
User Followers	Tracks who follows whom
Organizations	Organizations on Kaggle
User Organizations	Links users with their affiliated organizations
Teams	Teams for competitions, such as membership and medals won
Submissions	Submissions to competitions, including scores and submission dates
Datasets	Dataset information, including total downloads, views, and votes
Competitions	Competitions, including deadlines, rewards, and evaluation methods
Tags	Tags applied to competitions, datasets, kernels, and forums
Dataset Tags	Datasets with tags for categorization purposes
Competition Tags	Competitions with tags for categorization purposes
Forums	Forum posts, including title and parent relationships

Challenges

- Size of the data
  - Basic analysis – data types, size (or length), completeness check
  - Basic filtering – removing some empty columns
  - Chunked insertion – to avoid memory over-runs
- Referential constraints
  - Enforced some constraints by checking inserted values
  - Ignored the type mis-match and missing cases

Phase II

DOCUMENT MODEL

Document Model

- Users
  - Organizations
  - Followers
  - Achievements
- Datasets
  - Tags, Forums
- Competitions
  - Tags, Forums
- Teams
  - Submissions

- Organizations
- Tags
- Forums

Collection Users


```
_id: ObjectId('672d28147810bd3e39c68d9d')
Id : 368
UserName : "antgoldbloom"
DisplayName : "Anthony Goldbloom"
RegisterDate : 2010-01-20T00:00:00.000+00:00
PerformanceTier : 2
Country : "United States"
▾ Organizations : Array (1)
  ▾ 0: Object
      UserId : ObjectId('672d28147810bd3e39c68d9d')
      OrganizationId : ObjectId('672d27f17810bd3e39c0027a')
      JoinDate : 2020-03-15T00:00:00.000+00:00
▾ Followers : Array (6)
  ▾ 0: Object
      UserId : ObjectId('672d28147810bd3e39c68d9d')
      FollowingUserId : ObjectId('672d28147810bd3e39c77bfb')
      CreationDate : 2018-08-07T00:00:00.000+00:00
  ▸ 1: Object
  ▸ 2: Object
  ▸ 3: Object
  ▸ 4: Object
  ▸ 5: Object
▾ Achievements : Array (4)
  ▾ 0: Object
      UserId : ObjectId('672d28147810bd3e39c68d9d')
      AchievementType : "Competitions"
      Tier : 1
      TierAchievementDate : "07/15/2016"
      Points : 43
      CurrentRanking : NaN
      HighestRanking : 75
      TotalGold : 0
      TotalSilver : 0
      TotalBronze : 0
```

Issues

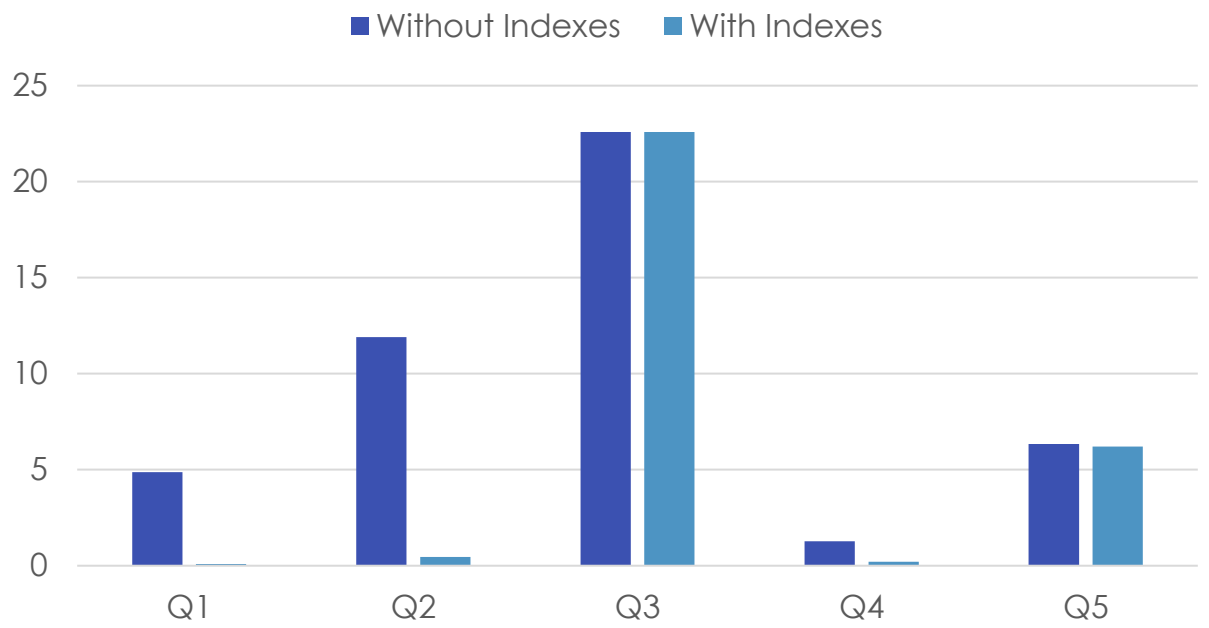
- Referential Constraint Mapping
  - Maping IDs to Object IDs
- 2.5X Slower Insertion
  - Find and Insert (bulk insertion)
- Slower Query Operations
  - Nested Structure

Phase II

DATA QUERYING

Querying Relational Model

- Our dataset was already normalized
- Querying provided interesting insights
  - Indexed key columns to improve query performance



Querying Relational Model

1. Top Competition Tags by User Medals
   - **Tables Used:** CompetitionTags, Tags, Competitions, UserAchievements
2. Top Users by Followers and Achievements
   - **Tables Used :** UserFollowers, Users, UserAchievments
3. User Achievements and Dataset Creation
   - **Tables Used :** Users, UserAchievements, Submissions, CompetitionTags, Tags, Datasets
4. Competition Tags with Highest Engagement
   - **Tables Used :** CompetitionTags, Tags, Competitions, Submissions
5. Dataset Engagement by High-Achieving Users
   - **Tables Used :** UserAchievements, Users, Datasets, DatasetTags, Tags

Query 1

- Top Competition Tags by User Medals

```sql
SELECT
    T.Name AS TagName,
    COUNT(DISTINCT UA.UserId) AS ActiveUsers,
    SUM(UA.TotalGold) AS GoldMedals,
    SUM(UA.TotalSilver) AS SilverMedals,
    SUM(UA.TotalBronze) AS BronzeMedals,
    SUM(UA.TotalGold + UA.TotalSilver + UA.TotalBronze) AS TotalMedals
FROM competitiontags CT
INNER JOIN tags T ON CT.TagId = T.Id
INNER JOIN competitionscleaned C ON CT.CompetitionId = C.Id
INNER JOIN userachievements UA ON UA.UserId = C.Id
GROUP BY T.Name
ORDER BY TotalMedals DESC, ActiveUsers DESC
LIMIT 10;
```

tagname	activeusers	goldmedals	silvermedals	bronzemedals	totalmedals
image-data	103	8	13	58	79
animals	14	1	4	44	49
automobiles	5	6	6	9	21
tabular-data	149	3	7	8	18
text-data	42	0	8	5	13
internet	15	0	6	3	9
binary-classification	48	1	2	4	7
nlp	19	0	2	4	6
biology	13	1	0	3	4
audio-data	8	2	2	0	4

Query 2

- Top Users by Followers and Achievements

```sql
WITH FollowerCounts AS (
    SELECT
        UF.UserId,
        COUNT(UF.FollowingUserId) AS FollowerCount
    FROM userfollowers UF
    GROUP BY UF.UserId
    HAVING COUNT(UF.FollowingUserId) > 100
)
SELECT
    U.DisplayName AS UserName,
    FC.FollowerCount,
    COALESCE(SUM(UA.TotalGold), 0) AS GoldMedals,
    COALESCE(SUM(UA.TotalSilver), 0) AS SilverMedals,
    COALESCE(SUM(UA.TotalBronze), 0) AS BronzeMedals,
    COALESCE(SUM(UA.TotalGold + UA.TotalSilver + UA.TotalBronze), 0) AS TotalMedals
FROM FollowerCounts FC
INNER JOIN users U ON FC.UserId = U.Id
LEFT JOIN userachievements UA ON U.Id = UA.UserId
GROUP BY U.DisplayName, FC.FollowerCount
ORDER BY FC.FollowerCount DESC, TotalMedals DESC
LIMIT 10;
```

username	followercount	goldmedals	silvermedals	bronzemedals	totalmedals
Santiago Mota	3902	2	12	72	86
Márcio Santos	3369	0	2	45	47
Yasir Hussein Shakir	3303	1	13	286	300
PAVAN KUMAR D	1766	60	42	692	794
asaniczka	1720	37	26	1082	1145
Vitaliy Lyalin	1210	0	0	1	1
V.B.	999	16	6	239	261
Firat Gonen	989	76	42	711	829
ARPAN CHOUDHURY 98	899	0	0	0	0
OH SEOK KIM	898	23	20	1300	1343

Query 3

- User Achievements and Dataset Creation Patterns by Competition Topic.

```sql
SELECT
    U.DisplayName AS UserName,
    COUNT(DISTINCT C.Id) AS CompetitionsParticipated,
    COUNT(DISTINCT DC.Id) AS DatasetsCreated,
    T.Name AS CompetitionTopic,
    SUM(UA.TotalGold + UA.TotalSilver + UA.TotalBronze) AS TotalMedals,
    AVG(DC.TotalViews) AS AvgDatasetViews,
    AVG(DC.TotalDownloads) AS AvgDatasetDownloads,
    AVG(DC.TotalVotes) AS AvgDatasetVotes
FROM users U
INNER JOIN userachievements UA ON U.Id = UA.UserId
INNER JOIN submissionscleaned S ON U.Id = S.SubmittedUserId
INNER JOIN competitionscleaned C ON S.TeamId = C.Id
INNER JOIN competitiontags CT ON C.Id = CT.CompetitionId
INNER JOIN tags T ON CT.TagId = T.Id
INNER JOIN datasetscleaned DC ON U.Id = DC.CreatorUserId
GROUP BY U.DisplayName, T.Name
HAVING SUM(UA.TotalGold + UA.TotalSilver + UA.TotalBronze) > 1  AND COUNT(DISTINCT DC.Id) > 1
ORDER BY TotalMedals DESC, AvgDatasetViews DESC;
```

username	competitionsparticipated	datasetscreated	competitiontopic	totalmedals	avgdatasetviews	avgdatasetdownloads	avgdatasetvotes
Konrad Banachewicz	1	166	biology	4254912	2573.77	254.753	14.1884
Psi	1	11	audio-data	39894	972.182	19.2727	9.18182
Eu Jin Lok	1	3	time-series-analysis	11169	82325.7	20323.3	124.667

Query 4

- Competition Tags with Highest Engagement by Submissions.

```sql
SELECT
    T.Name AS TagName,
    COUNT(DISTINCT C.Id) AS NumberOfCompetitions,
    COUNT(SC.Id) AS TotalSubmissions,
    ROUND(AVG(SC.PublicScoreLeaderboardDisplay)::numeric, 2) AS AvgPublicScore,
    ROUND(AVG(SC.PrivateScoreLeaderboardDisplay)::numeric, 2) AS AvgPrivateScore
FROM competitiontags CT
INNER JOIN tags T ON CT.TagId = T.Id
INNER JOIN competitionscleaned C ON CT.CompetitionId = C.Id
INNER JOIN submissionscleaned SC ON C.Id = SC.TeamId
GROUP BY T.Name
HAVING COUNT(SC.Id) > 100
ORDER BY TotalSubmissions DESC;
```

tagname	numberofcompetitions	totalsubmissions	avgpublicscore	avgprivatescore
tabular-data	75	997	278.21	10805.4
multiclass-classification	33	643	93.65	468.53
image-data	57	488	122.22	616.14
internet	11	255	0.86	0.89
binary-classification	22	200	0.61	50000.5
geography	1	162	0.47	0.48
text-data	18	162	1.54	1.61
time-series-analysis	7	139	0.76	0.78
card-games	1	132	0.89	0.89
marketing	3	114	0.47	0.48

Query 5

- Dataset Tag Engagement by High-Achieving Users.

```sql
SELECT
      T.Name AS DatasetTag,
      COUNT(DISTINCT DC.Id) AS TotalDatasets,
      COUNT(DISTINCT U.Id) AS HighAchievingUsers,
      AVG(DC.TotalViews) AS AvgViews,
      AVG(DC.TotalDownloads) AS AvgDownloads,
      AVG(DC.TotalVotes) AS AvgVotes
   FROM userachievements UA
   INNER JOIN users U ON UA.UserId = U.Id
   INNER JOIN datasetscleaned DC ON U.Id = DC.CreatorUserId
   INNER JOIN datasettags DT ON DC.Id = DT.DatasetId
   INNER JOIN tags T ON DT.TagId = T.Id
   WHERE UA.TotalGold + UA.TotalSilver + UA.TotalBronze >= 5  -- Only consider users with 5 or more medals
   GROUP BY T.Name
   HAVING COUNT(DISTINCT DC.Id) > 3  -- Only include tags with more than 3 datasets
   ORDER BY AvgViews DESC, AvgDownloads DESC, AvgVotes DESC;
```

datasettag	totaldatasets	highachievingusers	avgviews	avgdownloads	avgvotes
tidyverse	33	14	64760.8	9569.16	146.337
linguistics	275	123	44307.3	5779.94	95.0301
simulation-games	101	41	37879.8	2333.64	935.918
web-sites	180	114	36834.7	5720.86	108.714
aviation	115	92	32029.5	4925.8	66.2918
popular-culture	224	94	31604	4501.2	77.7548
diabetes	209	188	30376.1	5853.79	66.4902
healthcare	840	429	29211.9	3978.57	65.0896
multiclass-classification	564	365	28458.6	3846.86	63.2067
bigquery	118	35	28179.5	197.744	82.3406

Phase III

DATA CLEANING

Data Cleaning

- Duplicate Check
  - No duplicate data found
- Missing Values
  - Empty
  - NaN
- Incorrect Format (Data types)
  - Numbers
  - Dates

Missing Values

- Display Name (314/20M)
  - Solved by replacing with username
- Country (19M/20M), Rankings
  - Cannot do much about it
- Team Name (25K/7M),
  - Removed from the dataset
- Some missing values ignored
  - tag and organization descriptions
  - Scores, Rankings

Referential Validity

- We enforce all referential constraints
  - Removed missing references: Team Leaders, Submitter (users), Parent Tags
- We ensure all data type (and format) constraints
  - SubmittedUserId – float to integer
  - TeamLeaderId – float to integer
  - SubmissionDate – text to date (ignoring empty)
- Some fields had no solutions
  - Forum -> Parent Forum

Data Cleaning

Attribute	Table	Cleaning Action	Results	Insights
ParentTagId	Tags	Converted FLOAT to INTEGER and removed invalid references	10 nulls processed; 3 invalid references removed	Ensured valid self-references and consistent data types for hierarchical tags.
Country	Users	Retained null values.	19,034,320 nulls kept out of 20,485,253 records.	Optional field; does not impact core analysis.
FollowingUserId	UserFollowers	Removed records with invalid references.	29,375 records removed out of 1,555,414 total.	Ensured valid relationships between followers and followed users.
CreatorUserId	Datasets	Removed invalid references.	9,091 records removed out of 397,793 total.	Ensured dataset creators are valid and linked to Users table.
TeamLeaderId	Teams	Converted FLOAT to INTEGER and removed invalid references.	26,306 nulls removed out of 7,675,480 records.	Ensured valid leader references while retaining most data.
HighestRanking	UserAchievements	Retained null values.	81,417,236 nulls kept out of 81,940,708 records.	High null count; assumed default for users without rankings.

Phase III

ITEM SET MINING

Mining

- Implemented Apriori algorithm for 3 distinct analyses
  - Competition Tags (440 competitions)
  - Dataset Tags (171,990 datasets)
  - User Organizations (2,066 relationships)
- Transaction Definition
  - Each entity (competition/dataset/user) = 1 transaction
  - Items = associated tags or organizations

Key Association Rules

- Competition Tags
  - Beginner → Tabular Data (conf=1.0)
  - Banking → Tabular Data (conf=0.91)
- Dataset Tags
  - Data Cleaning + Marketing Analytics (lift=16.75)
  - Business + Finance (lift=18.56)
- Organizations
  - IIT KHARAGPUR ↔ SPARK4AI (conf=1.0, lift=39.73)

Key Implications

- Platform Insights
  - Tabular data dominates competition (40%)
  - Pre-trained models common in datasets (14.2%)
  - Strong educational institution partnerships
- Practical Applications
  - Tag recommendation systems
  - Competition difficulty prediction
  - Organizational partnership opportunities

Conclusions

- Data is Highly Normalized
  - Relational model is an effective choice
  - Growing collection size increases query time substantially
- Ensuring referential constraints is difficult without cleaning
  - Yet, most of the data is clean and valid
- User engagement with datasets and competitions shows interesting patterns across the Kaggle platform

Thank You

Questions