UNIVERSITY OF LIVERPOOL

Automated Generation of Graphs from Relational Sources to Optimise Queries for Collaborative Filtering

Ahmad Shahzad And Frans Coenen
Presented By: Ahmad Shahzad
University of Liverpool United Kingdom
Email: ahmads@liverpool.ac.uk

Ahmad Shahzad

- Industry expert of Software Development
- Keen proponent of data driven development strategies
- Graph Theorist
- Technology Entrepreneur
- Helping organisations build platforms to aid in autonomous decision making solutions to keep them ahead of the competition
- Current PhD candidate in data mining and machine learning

The Problem

Graph abstraction is intuitive and effective for collaborative filtering

Transactional data is typically stored in relational databases

Export, Transform and Load process are developed to transform relational data to graph data

It requires domain knowledge expertise

Expensive to develop

Requires knowledge of relational database as well as graph database

Typically ETL process is not optimised for queries which will be operated in graph database.

The Solution

Automated transformation of relational data to graph data

Limitations for the solution

Data has to be in 5^{th} normal form.

Graph data must adhere to basic standards of Property Graph Model

Graphs and Recommender Engines

Graphs are features in range of application domains from simple path finding to collaborative filtering algorithms.

Index free adjacency property where nodes only store information about its neighbours only hence there is no global index.

Edge traversal is independent of size of data

Effective local analysis of the graph making it suitable for recommender engines in general and collaborative filtering in particular.

Relational databases feature slower runtimes when there are multiple joins where as graph databases are inherently faster in modelling and identifying associations between entities.

Collaborative Filtering Algorithm

MEMORY BASED ALGORITHMS

Pearson Correlation Coefficient or Cosine Vector Similarity, both of which use the inner product.

Computationally efficient for real time analysis.

Uses entire data sets.

Moderately effective quality of recommendations produced.

MODEL BASED ALGORITHMS

Singular Value Decomposition (SVD), Latent Semantic Analysis, Bayesian Networks, deep learning and Association Rule-based methods.

Computationally inefficient for real time processing.

Uses larger part of data for training and smaller part for testing.

Highly effective quality of recommendation produced.

Cold Start Problem

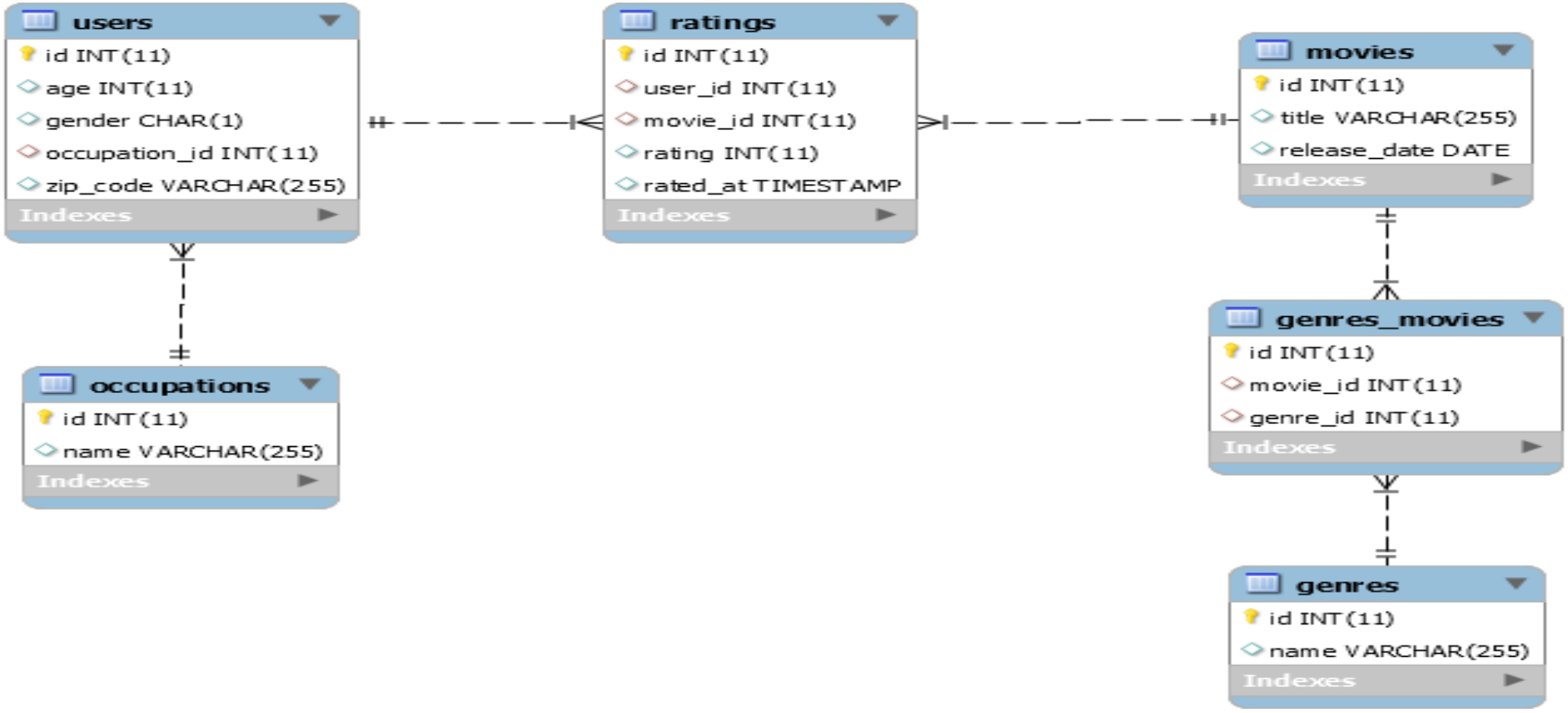
Recommendation Engines typically take advantage of past history to make predictions.

User with little or no history cannot be provided with recommendations?

Use of auxiliary information

Identifying user with a “User Group” based on extra information

Movie Lens Data Set



Transformation Rules

If there is only foreign key in a relation then the two entities are linked together with an edge with foreign key as property of the property.

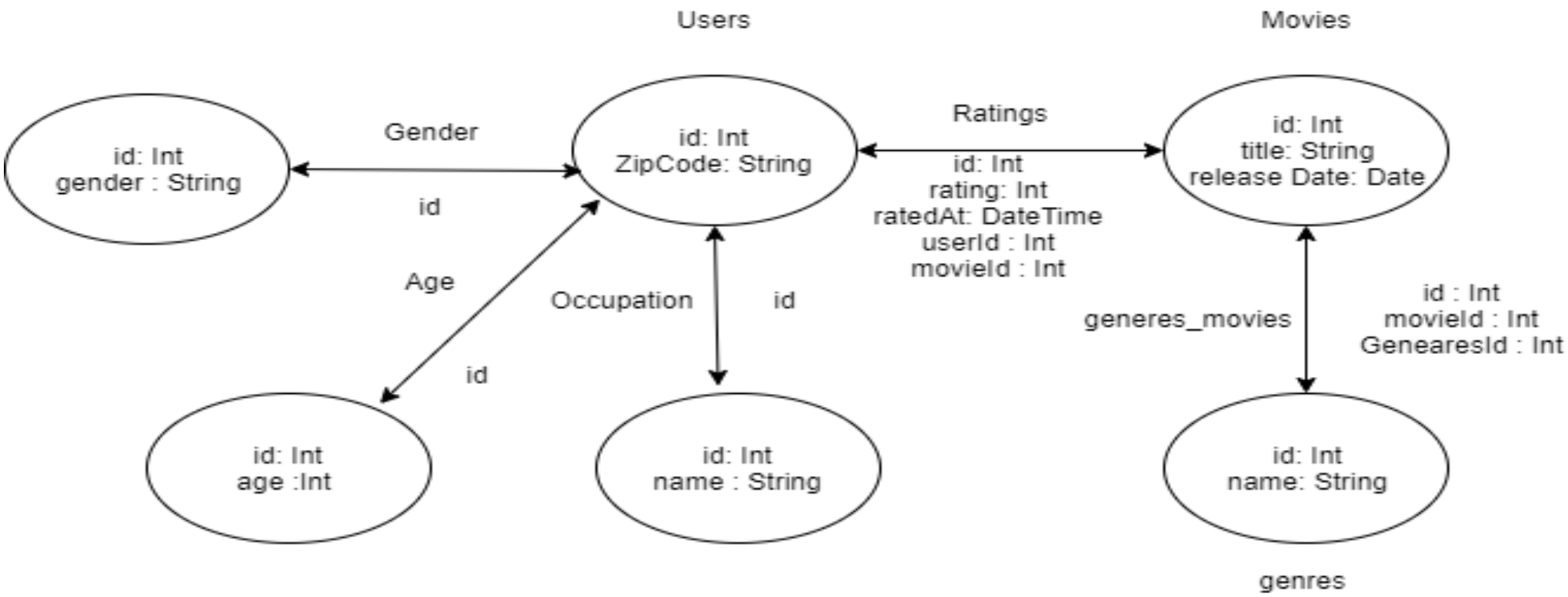
If there are two foreign keys then it created two vertices and all attributes of the table belong to the edge properties.

If there are 3 or more than 3 foreign keys then there will be N number of vertices and they edge properties will be the corresponding foreign keys.

If for any attribute there is a value less than threshold corresponding to the primary key of relation then a new vertex will be generated and all the tuples will use the new vertex for transformation.

If vertex already exists then it will not be created again as vertex Ids will be same as db ids.

Movie Lens Generated Graph



Why 5th normal form is required

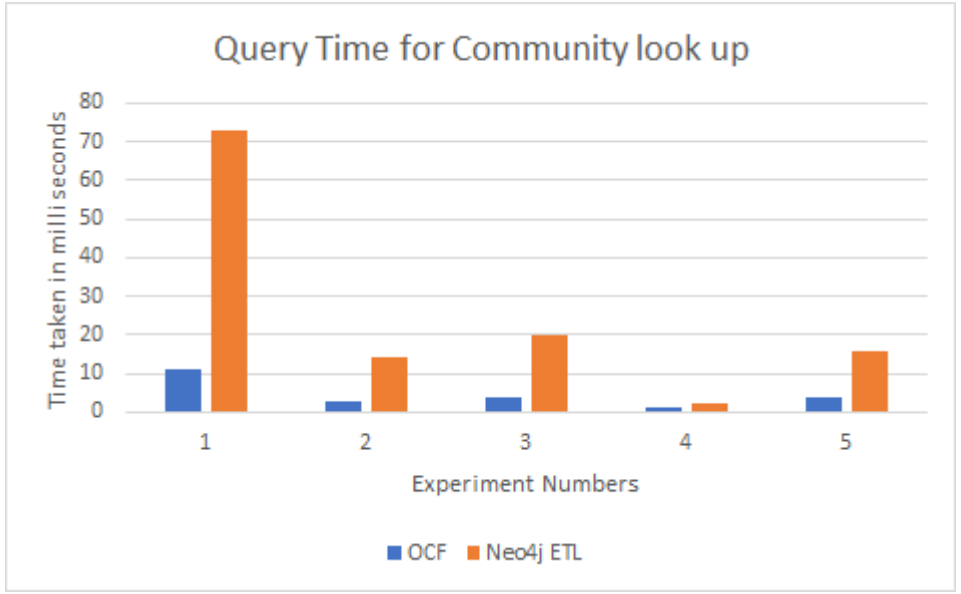
Graph database has to filter and scan all nodes if there is filter on a node property.

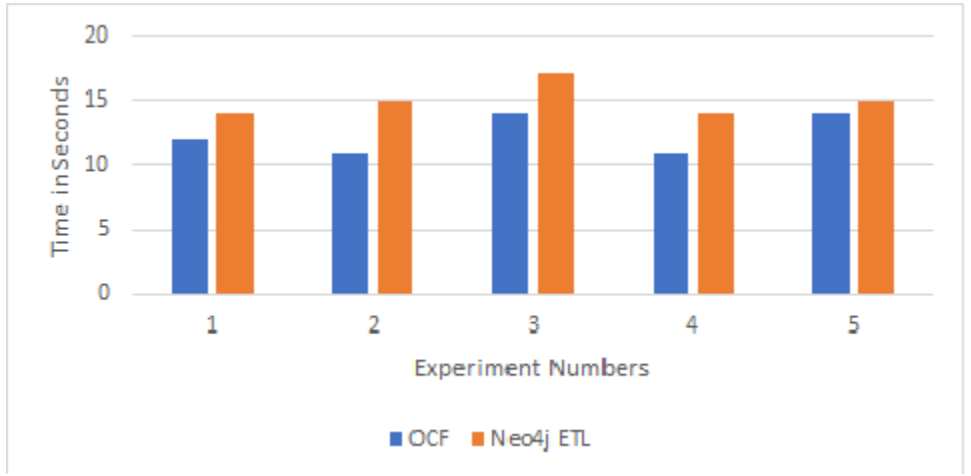
It will be far more efficient if there was a separate node type for that filter so that the number of nodes which need to be scanned can be drastically reduced.

It can only possible if the data is in 5th normal form.

A lot of databases are naturally in 5th normal form if they are already in 3rd normal form.

Experimental Results





Question & Answer