Гибридная рекомендательная система на основе графа:

Пусть – набор n пользователей,

– набор каких-то объектов (в нашем случае фильмы)

– матрица размером на , где каждая (если оценки нету, дефолтное значение берётся = 0).

– матрица размером на содержащая дополнительную информацию о пользователях (пол, страна проижвания, возраст и т.д.)

– матрица содержащая дополнительную информацию об объектах

Алгоритм:

**Вход:**  ,

**Выход:** приближённые значения пар пользователь-объект

Для каждого пользователя в :  
 Для каждого пользователя в :

= процент объектов со схожими рейтингами между двумя пользователями ()

= Посчитать аггрегированное сходство пользователей

= (: Similarity Graph, пользователи - узлы)

= Построить граф фич () (строит на основе графа похожести пользователей граф фич , который представляет собой матрицу размером n на g, содержащую для каждого пользователя такие характеристики как Degree Centrality (метрика сколько связей имеет узел), Closeness Centrality (мера близости узла ко всем остальным узлам в сети), Average Neighbor Degree (средняя степень соседа) и т.д.)

= Обработать и классифицировать граф характеристик пользователей () (возвращается матрица размером n на sc)

= Объединить матрицы (, ) (возвращается матрица размером n на g + sc, содержащая для каждого пользователя как его фичи, так и его характеристики)

= Пропустить через автоэнкодер () (извлекает матрицу из фич-веторов для каждого пользователя (уменьшает количество измерений фич до k <= g + sc))

= k-Means Clustering (, оптимальное количество кластеров())

= сгенерировать матрицу принадлежностей пользователей кластерам ()

= Приблизить рейтинги фильмов в кластерах ():

Для каждого кластера из :

Для каждого фильма из :

Если есть пользователи, которые оценили фильм в :

[] = average(users rates())

Либо если есть схожие фильмы фильму , которые имеют оценку в кластере :

[] = average(users rates(similar items(), ))

Либо:

[] = average(all users rates())

= Приблизить рейтинги пользователей ()

Return

 