Chapter 5 : Part 2

**Association Rule Learning**

**Eclat**

Data Types, Numbers, Operators, Type Conversion, f-strings

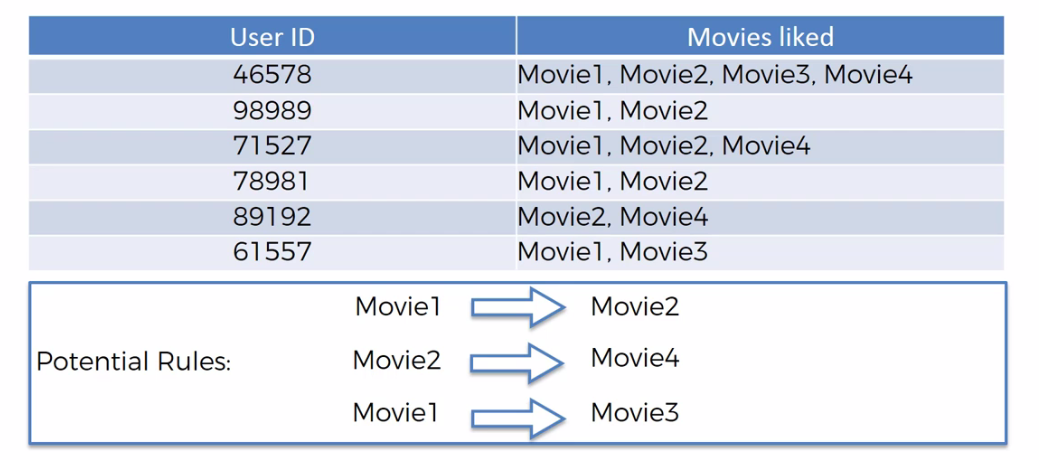
**5.2.1 Eclat algorithm**

***Eclat*** stands for ***Equivalence Class Clustering*** and ***Bottom-Up Lattice Traversal*** and it is an algorithm for association rule mining (which also regroups frequent itemset mining).

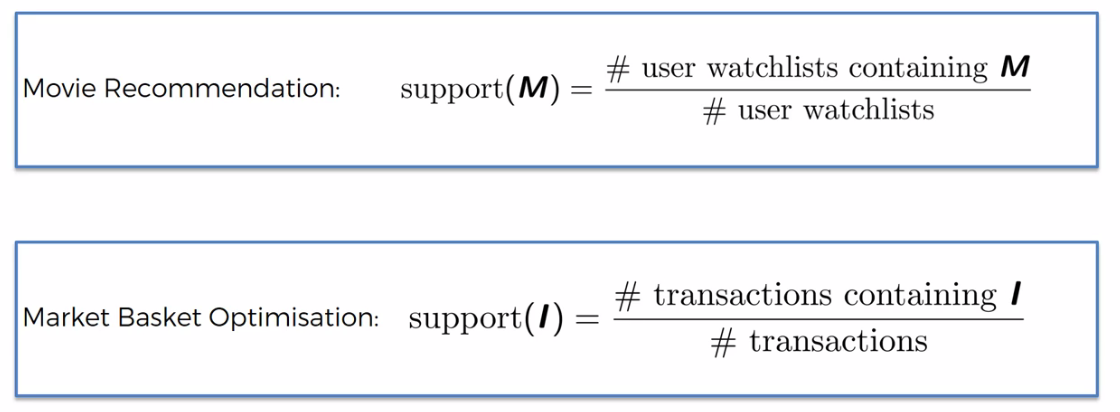
* Eclat: In this model, only *Support value* is used, which shows *how frequent a set of itmes occur*. Therefore, Eclat is a *simplified version of Apriori model*.
* Association rule mining and frequent itemset mining are easiest to understand in their applications for basket analysis: the goal here is to understand which ***products*** are often ***bought together by shoppers***.
* These association rules can then be used for example
* for *recommender* *engines* (in case of *online* *shopping*) or
* for *store* *improvement* for *offline* *shopping*.
* ECLAT for association rule: The ECLAT algorithm is ***not the first algorithm for association rule mining***. The foundational algorithm in the domain is the ***Apriori algorithm***. Since the Apriori algorithm is the first algorithm that was proposed in the domain, it *has been improved upon in terms of computational efficienc*y (i.e. they made *faster* *alternatives*).
* ECLAT vs FP Growth vs Apriori: There are two faster alternatives to the Apriori algorithm that are state-of-the-art:

1. one of them is FP Growth and
2. the other one is ECLAT.

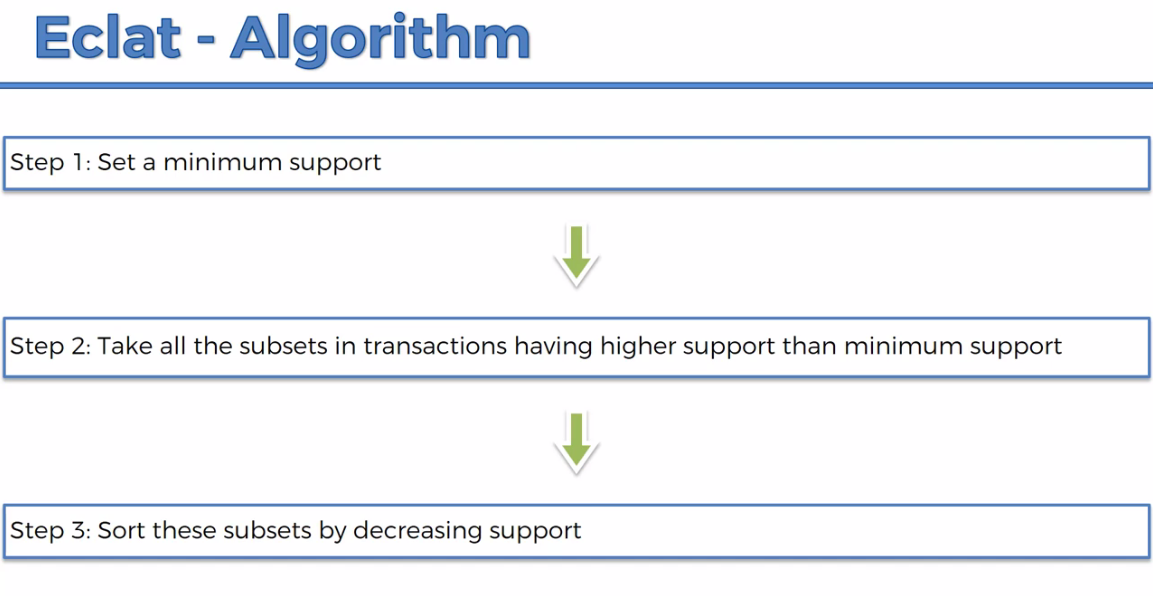
* Between ***FP*** ***Growth*** and ***ECLAT*** there is no obvious winner in terms of execution times: it will *depend on different data and different settings* in the algorithm.
* It also talks about the ***"people who bought also bought …"***. So it's kind of like a ***recommender system*** and similar to what we had in the Apriori algorithm.



* Eclat Algorithm: *Eclat algorithm* stands for ***Equivalence Class Transformation***. This algorithm uses a *depth-first search* technique to find *frequent* *itemsets* in a *transaction* database. It performs *faster* *execution* than *Apriori* Algorithm.
* In Apriori we worked with "***rules***" and these ***rules*** will have different strengths, and based on the Lift we could judge the strength of a *rule*.
* But in Eclat we are *not* actually going to be *talking* about *rules*. Here we are going to be talking about sets.
* In here we aren't judging the rules with their strength. We're not selecting rules and we're just saying *what could potentially be* and then the Eclat model is responsible for actually going *through all of these combinations* and telling us what we should focus on.
* In Eclat model we only have support*: In following* ***M*** *and* ***I*** *stands for* Set of Movies *or* Set of Items*. That is there will be more than one item in the set.*



* So, how often does this happen - ***Some people are watching a certain combinations of movies***. So in Eclat model, we don't have the ***confidence*** and the ***lift*** ***factors*** we're only looking at ***support*** and ***how frequently does this set of items occur***.
* Example: We check that how many people watched ***Ex-Machina*** and ***Intersteller*** both in given data. And then when we get the new data we find the people who watched either ***Ex-Machina*** or ***Intersteller*** and we recommend them the other movie (I.e watched one movie and recommend other related movie).
* Steps:



* It's much faster and the steps involved are set a minimum support. So you want to *set* up a *support level* then you take all the subsets in transactions *having higher support* and then you set the *subset* in *decreasing* *support*.
* And basically at the top you will have the most the strongest combinations of items which you should look at (e.g. end up with *top 10* or *top five*).