**Association rule mining**

Association rules are conditional statements that help in finding relationship between random objects in a database or information repository. It gives information that how strongly two objects are related to each other which is impossible to detect with naked eyes.

It consists of two parts, an antecedent (if) and a consequent (then).  An antecedent is an item found in the data. A consequent is an item that is found in combination with the antecedent.

The core part of association rule is **apriori algorithm**. It is based on support and confidence. *Support* is an indication of how frequently the items appear in the database. *Confidence* indicates the number of times the if/then statements have been found to be true.

It is widely used now days in industries to find out hidden trend and pattern.

For example “*If a person buys milk then there is 60% of chances that he will buy eggs too*”. They play important role in determining customer behavior in grocery store, in shopping basket data analysis, product clustering, catalog design and store layout. Programmers use association rules to build programs capable of Machine Learning. Machine learning is a type of artificial intelligence that seeks to build programs with the ability to become more efficient without being explicitly programmed.

**Background for our research**

A number of previous studies have been done on large open projects such as Jedit, Eclipse, Apache Ant, and Mozilla Firefox etc. to detect the coupling between different java files, class and methods. In past Annie Ying has developed an approach which is based on association rule mining to find the recommendation, She generally focused on the files. For example if a developer changes a file then her approach will recommend him other files too that need to be changed for newer version of software. Similarly Zimmermann and his associates developed a tool named rose that too works on association rule mining of CVS. Their main concern was to find the finer grained entities, which was absent in ATT Yings work. They used Association rule mining with minimum support and confidence. Their tool seems to be powerful in terms of suggesting and predicting next likely to be change code elements. It also prohibits and warns about incomplete changes. It can also used for identifying coupling between different code elements.

**Dataset**

We have taken five major releases of Apache ant ranging from 1.5 to 1.9 for our study. We have considered only valid commits, commits that were actually made to code. Neglected commits made for signature changes, comments update etc. We have divided this data into 1:5 ratio and taken 80% one for training and rest for testing. We have taken changes made in one commit as one transaction for association rule mining.

**Methodology**

Those code elements that were changed in single commits are taken as one transaction and similarly we