**Boolean Model:**

In the standard Boolean model, the documents are represented by keywords and it is based on set theory in which each term in the query is represented as a 1 or 0. Even the query is connected by the connectors “AND” “OR” “NOT”

Most popular search is phrase search but partial implementation of the Boolean operators. The options in the Boolean search is used by the expert searchers but not all the people and also not for the people who uses to improve the performance of the search engine

The main goal is to limit the document number retrieved and also to rank the resulted documents that matches the user query

Mainly end-users face the problem of this disadvantage because they don’t use the system frequently to form a query in a perfect manner

**Extended Boolean Model:**

Extended Boolean model came Into theory to avoid the drawbacks of the Boolean model.

Some of the disadvantages of Boolean model is : it doesn’t consider the term weight of the words in the query. Because of this there will be result of huge collection of documents or either the result will be too small.

Extended Boolean model idea is to combine term weight and the matching in the vector space model. This new model will have characteristics of vector space model and also Boolean algebra and also the ranking which represent the similarity between query and the documents

This will result in the retrieval of the documents which are relevant with the queries.

It will basically the combination of both vector and the Boolean models

The term weight I is associated with the document D and it is also normalized by the term frequency. If there are more number of frequent words, then the document is more important

If the term frequency is I and it is present in the j document, then it should be represented by fij

If we want to represent the document frequency of term I and to represent the number of time I in a documents is dfi

There will be an inverse document i.e an index that will map back the terms and also the documents where it is present

Representation of frequency of term I on a inverse document is log2(n/dfi) = idfi

Finally, the weight of the document is given as

Wi,j = fij \* idfi /max idf

So now the extended Boolean model design has ranking of the documents

We can derive formula for the query using OR operator then the similarity between the query and the document is

*Sim (Qor, Dj)* **= ((w1p + w2p +…+ wkp)/k) 1/p** where 1 ≤ p ≤ ∞.

If we use AND operator, then the similarity between the query and the document is

*Sim (Qand, Dj)* **= 1 - (((1-w1)p + (1-w2)p +…+ (1-wk)p)/k) 1/p** where 1 ≤ p ≤ ∞.