Toivonen's Algorithm

Stephane Moore

October 13, 2010

1] Background

Toivonen's algorithm is a heuristic algorithm for finding frequent itemsets from a given set of data. For many frequent itemset algorithms, main memory is considered a critical resource. This is typically because itemset counting over large data sets results in very large data structures that quickly begin to strain the limits of main memory.
For those not familiar with immediate supersets and immediate subsets, informal definitions of these terms are provided here:

Definition 1.1. Given a set S, an *immediate subset* of S is defined to be any set $\{s|s \subset S, |s| = |S| - 1\}$. The empty set has no immediate subset.

Definition 1.2. Given a set S, an *immediate superset* of S is defined to be any set $\{s|s \supset S, |s| = |S| + 1\}$.

Toivonen's algorithm can be explained using the concept of a *negative border*, which is defined as follows:

Definition 1.3. Given a data set D, an itemset T is in the *negative border* of D if T is not frequent in D, but all immediate subsets of T are frequent in D.

In order to ensure the discovery of all frequent itemsets, the algorithm's terminating conditions make use of the following theorem:

Theorem 1.1. Given a data set D and a sample set $S|S \subseteq D$, if there is an itemset T that is frequent in D but not frequent in S, then there is an itemset T' that is frequent in D and is in the negative border of S.

Proof. (by Contradiction.) Assume that if there is an itemset T that is frequent in D but not frequent in S, then there are no itemsets that are frequent in D in the negative border of S. Consider a data set D, a sample set $S|S \subseteq D$, and an itemset T that is frequent in D, not frequent in S, and not present in the negative border of S. Let T' be the smallest subset of T that is not frequent in S. It is trivial to show that any subset of a frequent itemset is also frequent. Hence T' is frequent in D because T is frequent in D and $T' \subseteq T$. If there exists an immediate subset of T' that is not frequent in S, then T' violates its definition as the smallest subset of T that is not frequent in S. Hence, by definition, all immediate subsets of T' are frequent in S and T' is not frequent in S. Thus it follows that T' is in the negative border of S. Hence the existence of the itemset T, which is frequent in D but not frequent in S implies the existence of the itemset T', which is frequent in D and appears in the negative border of S. This contradicts our assumption.

An important consequence of this conclusion, derived through *modus tollens*, is that for a data set D and a sample set $S|S \subseteq D$, if there are no itemsets in the negative border of S that are frequent in D, then there are no itemsets that are frequent in D but not frequent in S.

It is also clear that, given a data set D and a sample set $S|S \subseteq D$, if there is an itemset T that is frequent in D and is in the negative border of S, then there is an itemset that is frequent in D but not frequent in S. This holds from the fact that itemsets in the negative border of a set are defined as not being frequent in that set.

Theorem 1.2. Given a data set D and a sample set $S|S \subseteq D$, if there is an itemset T that is frequent in D and is in the negative border of S, then there is an itemset that is frequent in D but not frequent in S.

Proof. By definition 1.3, any itemset in the negative border of S is not frequent in S. Hence T is frequent in D but not frequent in S.

2] Toivonen's Algorithm

Let D be our data set.
First Pass: Acquire a sample set $S|S \subseteq D$. Use an existing frequent itemset algorithm [1] to acquire F, the set of all itemsets that are frequent in S. Also acquire N, the set of all itemsets in the negative border of the sample S.
Second Pass: Count all itemsets in $F \cup N$ over the data set D. If any itemset $T \in N$ is frequent in D, we can assume by theorem 1.2 that there is an itemset that is frequent in D but not frequent in S; in such a case, we are forced to restart the algorithm as we have already failed to discover at least one itemset that is frequent in D. From theorem 1.1 and *modus tollens* we know that if there are no itemsets in the negative border of S that are frequent in D, then there are no itemsets that are frequent in D but not frequent in S. Hence if we find no itemset $T \in N$ that is frequent in D, we are permitted to terminate the algorithm as we have discovered all the frequent itemsets of D.

3] Reflections

Toivonen's algorithm presents an interesting approach to discovering frequent itemsets in large data sets. The algorithm's deceptive simplicity allows us to discover all frequent itemsets through a sampling process. For certain data sets, this ability proves invaluable.
In practice, numerous optimizations and approximations can be made to improve the algorithm's performance on data sets with particular properties. For instance, the set of frequent itemsets in the sample set can be generated under a slightly lowered threshold. This subtle modification aims to minimize the omission of itemsets that are frequent in the entire data set as such omissions result in additional passes through the algorithm. However, the support threshold should also be kept reasonably high so that the counts for the itemsets in the second pass fit in main memory. Keep in mind that there are a multitude of optimizations that can be considered as well as additional state that can be maintained to improve the algorithm's performance.
Toivonen's algorithm is a powerful and flexible algorithm that provides a simplistic framework for discovering frequent itemsets while also providing enough flexibility to enable performance optimizations directed towards particular data sets.

[1] Variants of the Apriori algorithm are commonly used to discover frequent itemsets in the first pass of Toivonen's algorithm.