Assignment 3:

❖ Q1. Can apriori mining algorithm handle convertible constraints? Justify.

- A convertible constraints is a special class of constraints, in which by proper data ordering, such constraints can be pushed deep into the iterative mining process and they may have the same pruning power as monotonic or antimonotonic constraints.

If the items in the itemset are arranged in a particular order, the constraint may become monotonic or antimonotonic with regard to the frequent itemset mining process. For example, in ascending order, the constraint may become antimonotonic, and in descending order, the constraint may become monotonic, because if the itemset satisfies the constraint and so on. So, Apriori mining cannot handle convertible constraints since they cannot be pushed deep into the an Apriori mining algorithm because no direct pruning based on the constraint can be made, but it can be pushed into frequent pattern growth framework.

[1]: Han, J., & Kamber, M. (2012). Data mining concepts and techniques, third edition (3rd ed., pp. 299-300). Waltham, Mass.: Morgan Kaufmann.

==========

❖ Q2. Discuss the relationship between colossal and core patterns.

- If we have a frequent pattern (colossal pattern) and there are subpatterns cluster tightly around it, and these subpatterns share a similar support set with the colossal pattern, we call this subpatterns (core patterns).

If the colossal patterns have far more core patterns than smaller patterns the colossal pattern comes more robust, so, if a small number of items are removed from the pattern, the resulting pattern would have a similar support set. Also, the larger the pattern size, the more prominent this robustness. This a robustness relationship between a colossal pattern and its corresponding core patterns can be extended to multiple levels, and the lower-level core patterns of a colossal pattern are called core descendants. [2]

[2]: Han, J., & Kamber, M. (2012). Data mining concepts and techniques, third edition (3rd ed., p. 305). Waltham, Mass.: Morgan Kaufmann.

==========

❖ Q3. What is boosting? State why it may improve the accuracy of decision tree induction?

- Boosting is one of the ensemble methods which can be used to increase overall accuracy by learning and combining a series of individual (base) classifier models. The idea is like when you are a patient, instead of consulting one doctor, you choose to consult several. So, Suppose you assign weights to each training tuple, a series of k classifiers is iteratively learned. Then the weights are updated to allow the subsequent classifier to “pay more attention” to the training tuples that were misclassified by the first classifier. The final boosted classifier combines the votes of each individual classifier, where the weight of each classifier’s vote is a function of its accuracy. This is the power of boosting. Besides, boosting assigns a weight to each classifier’s vote, based on how well the classifier performed. The lower a classifier’s error rate, the more accurate it is, which means the higher its weight for voting should be and that confirms that boosting tends to achieve greater accuracy. [3]

[3]: Han, J., & Kamber, M. (2012). Data mining concepts and techniques, third edition (3rd ed., pp. 380-383). Waltham, Mass.: Morgan Kaufmann.

==========

❖ Q4. Ensemble methods improve classification accuracy. How?

- These methods combines a series of k learned models (base classifiers), to create an improved composite classification model. We use a given data set to create k training sets which used to generate classifier. The base classifiers each vote by returning a class prediction, so, the ensemble returns a class prediction based on the votes of the base classifiers.

We can notice that an ensemble is more accurate than its base classifiers. If we have a tuple to classify, it collects the class label predictions returned from the base classifiers. So, the base classifiers may make mistakes, but the ensemble will misclassify the tuple only if over half of the base classifiers are in error. Ensembles give better results when there is significant diversity among the models. Ensemble methods are parallelizable since each base classifier can be allocated to a different CPU. Thus, ensemble methods generate a set of classification models and then given a new data tuple to classify, each

classifier “votes” for the class label of that tuple. The ensemble combines the votes to return a class prediction, and that increases the classifier accuracy.[4]

[4]: Han, J., & Kamber, M. (2012). Data mining concepts and techniques, third edition (3rd ed., p. 378). Waltham, Mass.: Morgan Kaufmann.

==========