GSP algorithm

From Wikipedia, the free encyclopedia

GSP algorithm (*Generalized Sequential Pattern* algorithm) is an algorithm used for sequence mining. The algorithms for solving sequence mining problems are mostly based on the *a priori* (level-wise) algorithm. One way to use the level-wise paradigm is to first discover all the frequent items in a level-wise fashion. It simply means counting the occurrences of all singleton elements in the database. Then, the transactions are filtered by removing the non-frequent items. At the end of this step, each transaction consists of only the frequent elements it originally contained. This modified database becomes an input to the GSP algorithm. This process requires one pass over the whole database.

GSP algorithm makes multiple database passes. In the first pass, all single items (1-sequences) are counted. From the frequent items, a set of candidate 2-sequences are formed, and another pass is made to identify their frequency. The frequent 2-sequences are used to generate the candidate 3-sequences, and this process is repeated until no more frequent sequences are found. There are two main steps in the algorithm.

- Candidate Generation. Given the set of frequent (k-1)-frequent sequences F(k-1), the candidates for the next pass are generated by joining F(k-1) with itself. A pruning phase eliminates any sequence, at least one of whose subsequences is not frequent.
- Support Counting. Normally, a hash tree–based search is employed for efficient support counting. Finally non-maximal frequent sequences are removed.

Contents

Algorithm

```
F1 = the set of frequent 1-sequence
k=2,
do while F(k-1)!= Null;
   Generate candidate sets Ck (set of candidate k-sequences);
      For all input sequences s in the database D
         do
      Increment count of all a in Ck if s supports a
      Fk = {a ∈ Ck such that its frequency exceeds the threshold}
```

```
        k = k+1;
    Result = Set of all frequent sequences is the union of all Fks
        End do
End do
```

The above algorithm looks like the Apriori algorithm. One main difference is however the generation of candidate sets. Let us assume that:

$$A \rightarrow B \text{ and } A \rightarrow C$$

are two frequent 2-sequences. The items involved in these sequences are (A, B) and (A,C) respectively. The candidate generation in a usual Apriori style would give (A, B, C) as a 3-itemset, but in the present context we get the following 3-sequences as a result of joining the above 2- sequences

$$A \rightarrow B \rightarrow C,\ A \rightarrow C \rightarrow B \text{ and } A \rightarrow BC$$

The candidate–generation phase takes this into account. The GSP algorithm discovers frequent sequences, allowing for time constraints such as maximum gap and minimum gap among the sequence elements. Moreover, it supports the notion of a sliding window, i.e., of a time interval within which items are observed as belonging to the same event, even if they originate from different events.

See also

Sequence mining

References

- *Data Mining Techniques Pujari, Arun K. (2001). Universities Press. (https://books.google.com/books?id=dH2KQhJboSYC&pg=PA256) ISBN 81-7371-380-4.* Missing or empty |title= (help) (pp. 256-260), p. 256, at Google Books

External links

- SPMF (http://www.philippe-fournier-viger.com/spmf/) includes an open-source implementation of the GSP algorithm as well as PrefixSpan, SPADE, SPAM, ClaSP, CloSpan and BIDE.

Retrieved from "https://en.wikipedia.org/w/index.php?title=GSP_algorithm&oldid=738247433"

Categories: Data mining algorithms

- This page was last edited on 7 September 2016, at 20:19.
- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.