Unidad 2: Minería de elementos frecuentes

Algoritmos de pasadas limitadas

Blanca Vázquez y Gibran Fuentes-Pineda

Octubre 2020

Para un número de elementos d grande

- Si no se puede tener en memoria principal la base de datos y los contadores, se necesitan múltiples pasadas

- Muestreo
  - Muestreo aleatorio simple
  - Algoritno Savasere, Omuiecinski y Navathe
  - Algoritmo de Toivonen

Muestreo aleatorio simple

- Se buscan los conjuntos frecuentes a partir de una muestra aleatoria de la base de datos
- Se muestrea pasando por cada entrada de la base de datos y eligiéndola con una probabilidad p
  - Si la base de datos es de tamaño n, tendríamos aproximadamente $p \cdot n$ muestras
- Se aplica el algoritmo apriori, PCY, multietapa o multihash a la muestra

Evitando errores en muestreo aleatorio

- Los conjuntos con soporte muy cercano al umbral pueden o no ser frecuentes en la muestra aleatoria
- No produce todos los conjuntos que son frecuentes (falsos negativos)
  - Se pueden evitar verificando que sean frecuentes en la base de datos completa
- Produce algunos conjuntos no frecuentes (falsos positivos)
  - Se pueden reducir si disminuimos el soporte mínimo al considerar un conjunto como frecuente en la muestra; por ej. $0.9 \cdot \mathit{minsup}$ lo cual equivale a $0.9 \cdot p \cdot \mathit{minsup}$ entradas

Algoritmo de Savasere, Omiecinski y Navathe (SON)

- Se divide la base de datos en trozos y se aplica el algoritmo apriori, PCY, multietapa o multihash a cada trozo

- Se toman como candidatos los conjuntos frecuentes encontrados en todos los trozos
  - Si un conjunto no es frecuente en ningún trozo, su soporte es menor a *minsup* en todos los trozos y en general su soporte sería menor a $\frac{1}{p} \cdot p \cdot minsup = minsup$ en la base de datos completa
  -

- En una segunda pasada se descartan los candidatos que no sean frecuentes

Algoritmo de SON en el modelo mapeo-reducción

- Cada trozo se procesa en paralelo y se combinan los conjuntos frecuentes de cada uno para formar los candidatos

- En el modelo mapeo-reducción
  1. Mapeo 1: Toma el subconjunto de transacciones asignadas y encuentra los conjuntos frecuentes
  2. Reducción 1: Produce los conjuntos frecuentes que aparecen al menos una vez como candidatos
  3. Mapeo 2: Recibe todos los candidatos y cuenta sus ocurrencias en una porción de la base de datos
  4. Reducción 2: Toma cada candidato y suma sus ocurrencias en distintas porciones para producir su soporte en la base de datos completa

Algoritmo de Toivonen

- Se selecciona una muestra de la base de datos para encontrar candidatos usando un soporte mínimo más pequeño (por ej. $0.9 \cdot p \cdot minsup$)

- Buscamos la frontera negativa: colección de conjuntos que no son frecuentes en la muestra pero cuyos subconjuntos inmediatos son frecuentes

- Se realiza un conteo de los conjuntos frecuentes en la base de datos completa
    1. Ningún miembro de la frontera negativa es frecuente, por lo que los conjuntos frecuentes son los que
    2. Algunos miembros son frecuentes y no es posible dar una respuesta