

# ALS1 – Přednáška 10

Metrickým stromem rozumíme jakoukoli stromovou datovou strukturu specializovanou na indexování dat v metrických prostorech. Metrické stromy využívají vlastnosti metrických prostorů, jako trojúhelníková nerovnost k efektivnějšímu přístupu k datům. Příklady jsou M-stromy, vp-stromy, cover stromy, MVP stromy, a bk stromy.

Metrika vzdálenosti  $d(x, y)$  pro metrický prostor je definována následovně

- i)  $d(x, y) = d(y, x)$
- ii)  $0 < d(x, y) < \infty, x \neq y$
- iii)  $d(x, x) = 0$
- iv)  $d(x, y) \leq d(x, z) + d(z, y)$  (trojúhelníková nerovnost)

*Near Neighbor Query:* Z dané množiny datových objektů  $X = \{X_1, X_2, \dots, X_n\}$  z metrického prostoru s metrikou  $d()$ , získaj všechny datové objekty, které jsou do vzdálenosti  $r$  od zadaného bodu  $Y$ . Výsledná množina bude

$$\{X_i \mid X_1, X_2, \dots, X_n\}.$$

Parametr  $r$  se obecně nazývá měření podobnosti nebo toleranční faktor.

Jsou možné i některé varianty near neighbor query:

- nearest neighbor query,
- k-nearest neighbor query
- farthest neighbor query,

## 1 vp-stromy (vantage point trees)

VP-stromy rozkládají množinu dat podle vzdáleností objektů vzhledem k referenčnímu bodu (vantage point). Medián těchto vzdáleností je použit jako oddělovač k rozkladu objektů do dvou vyvážených podmnožin, na které může být rekurzivně použita ta samá procedura.

Struktura binárního vp-stromu je velmi jednoduchá. Každý vnitřní uzel je ve tvaru  $(S_v, M, R_{ptr}, L_{ptr})$ , kde  $S_v$  je vantage point,  $M$  mediánová vzdálenost všech bodů (od  $S_v$ ) indexovaných pod tím uzlem,  $R_{ptr}$  a  $L_{ptr}$  jsou ukazatele na levý a pravý podstrom. Levý podstrom uzlu indexuje ty body, jejichž vzdálenost od  $S_v$  je menší nebo rovna  $M$ , a pravý podstrom uzlu indexuje ty body, jejichž vzdálenost od  $S_v$  je větší nebo rovna  $M$ . V listových uzlech jsou reference na datové body míst ukazatelů na podstromy.

Je-li dána konečná množina  $S = S_1, S_2, \dots, S_n$  objektů, a metrika  $d()$ , binární vp-strom  $V$  na  $S$  je konstruován následovně.

- 1) Pokud  $|S| = 0$ , vytvoř prázdný strom.
- 2) Jinak, necht'  $S_v$  je libovolný objekt z  $S$  (vantage point).  
 $M = \text{median}(\{d(S_i, S_v) \mid \forall S_i \in S\})$   
Necht'  $S_l = \{S_i \mid d(S_i, S_v) \leq M \text{ kde } S_i \in S \text{ a } S_i \neq S_v\}$   
 $S_r = \{S_j \mid d(S_j, S_v) \geq M \text{ kde } S_j \in S\}$  (kardinalita  $S_l$  a  $S_r$  by měly být zhruba rovny)  
Rekurzivně vytvoř vp-stromy na  $S_l$  a  $S_r$  jako levý a pravý podstrom.

Binární vp-strom je vyvážený, takže může být snadno stránkován pro uložení v sekundární paměti. Konstrukce vyžaduje  $\mathcal{O}(n \log_2(n))$  výpočtů vzdáleností.

Pro daný objekt  $Q$ , množina datových objektů, které jsou ve vzdálenosti  $r$  od  $Q$  je nalezena vyhledávacím algoritmem popsáním níže:

- 1) Pokud  $d(Q, S_v) \leq r$ , pak  $S_v$  je ve výsledné množině.
- 2) Pokud  $d(Q, S_v) + r \geq M$ , rekurzivně prohledej pravý podstrom.
- 3) Pokud  $d(Q, S_v) - r \leq M$ , rekurzivně prohledej levý podstrom.

(všimněte si, že 2) a 3) mohou nastat současně, pak se prohledávají oba podstromy).

## Zobecnění binárního vp-stromů na m-ární vp-stromy

Binární vp-strom může být snadno zobecněn na m-ární stromovou strukturu. Konstrukce vp-stromu řádu  $m$  je velmi podobná konstrukci binárního vp-stromu. Namísto hledání mediánu vzdáleností mezi vantage pointem a datovými body, jsou body seřazeny a rozloženy do  $m$  skupin o stejné kardinalitě. Hodnoty vzdáleností použité pro ten rozklad jsou zaznamenány v uzlu.

## 2 MVP-stromy (multiple vantage point trees)

MVP-stromy, rozkládají prostor do sférických řezů okolo vantage pointů (podobně jako vp-stromy), rozklady vytváří vzhledem k více než jednomu vantage pointu na každé úrovni a udržuje si informaci navíc v listech pro kvůli efektivnímu odfiltrování kandidátních bodů.

MVP-strom používá dva vantage pointy v každém uzlu. Na každý uzel v MVP-stromu může být nahlíženo jako na dvě úrovně VP-stromu (rodič a jeho přímí potomci) kde všechny uzly v potomcích na nižší úrovni používají ten samý vantage point. To umožňuje mít více odfiltrovaných dat v každém uzlu při vyhledávání a menší počet vantage pointů v nelistových úrovních.

MVP-strom má tři parametry:

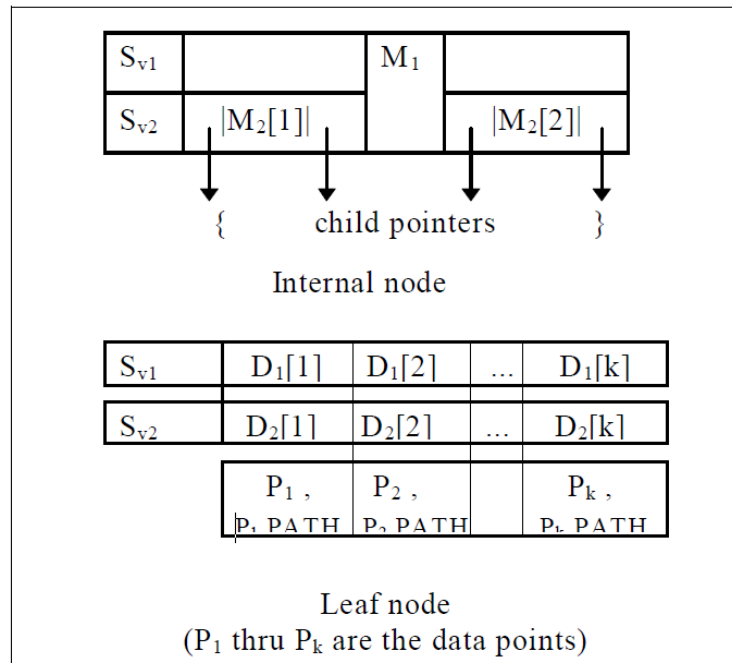
- počet tříd rozkladu vytvořený každým vantage pointem ( $m$ ),
- maximum dat pro listové uzly ( $k$ ),
- a počet vzdáleností pro datové body v listech ( $p$ ).

V binárních MVP-stromech rozděluje první vantage point (označme ho  $S_{v1}$ ) prostor na dvě části, a druhý vantage point (označme ho  $S_{v2}$ ) rozděluje každou z těchto částí na dvě. Máme tedy 4 potomky v binárním případě. Obecně počet potomků vnitřního uzlu je  $m^2$ .

V každém vnitřním uzlu je udržovány mediány  $M_1$  a  $M_2$  pro rozklad vzhledem k  $S_{v1}$  a  $S_{v2}$ .

V listových uzlech uchováváme přesné vzdálenosti mezi datovými body a vantage pointy toho listu.  $D_1[i]$  a  $D_2[i]$  ( $i = 1, 2, \dots, k$ ) jsou vzdálenosti z prvního a druhého vantage pointu, a  $k$  je počet dat v listech (může být zvoleno vyšší než  $m^2$ ).

Pro každý datový bod  $x$  v listech uchovává pole  $x.PATH[p]$  vypočtené vzdálenosti mezi datovým bodem  $x$  a prvními  $p$  vantage pointy na cestě z rootu do listového uzlu. Obrázek 1 ukazuje strukturu vnitřního uzlu a a listových uzlů v binárním MVP-stromu.



Obrázek 1. Struktura uzlu binárního MVP-stromu

### 2.1 Konstrukce MVP-stromu

Je-li dána konečná množina  $S = S_1, S_2, \dots, S_n$  objektů, a metrika  $d()$ , binární MVP-strom s parametry  $m = 2, k$  a  $p$  je konstruován na  $S$  následovně.

- 1) Pokud  $|S| = 0$ , vytvoř prázdný strom a skonči.
- 2) Pokud  $|S| \leq k + 2$  pak
  - 2.1) Vyber libovolný objekt z  $S$  ( $S_{v1}$ , první vantage point)
  - 2.2)  $S := S - \{S_{v1}\}$

- 2.3) Vypočítej všechny  $d(S_i, S_{v_1})$  kde  $S_i \in S$ , a ulož je v poli  $D_1$ .
- 2.4) Nechť  $S_{v_2}$  je nejvzdálenější bod od  $S_{v_1}$  (druhý vantage point).
- 2.5)  $S := S - \{S_{v_2}\}$
- 2.6) Vypočítej všechny  $d(S_j, S_{v_1})$  kde  $S_j \in S$ , a ulož je v poli  $D_2$ .
- 2.7) skonči.
- 3) Jinak (pokud  $|S| > k + 2$ )
  - 3.1) Vyber libovolný objekt z  $S$  ( $S_{v_1}$ , první vantage point)
  - 3.2)  $S := S - \{S_{v_1}\}$
  - 3.3) Vypočítej všechny  $d(S_i, S_{v_1})$   
Pokud ( $level \leq p$ )  $S_i.PATH[l] = d(S_i, S_{v_1})$ .
  - 3.4) Uspořádej objekty v  $S$  vzhledem k jejich vzdálenosti od  $S_{v_1}$ .  
 $M_1 =$  medián z  $\{d(S_i, S_{v_1})\}$ , rozlož objekty do dvou seznamů,  $SS_1$  a  $SS_2$ , stejné délky (podle  $M_1$ ).
  - 3.5) Nechť  $S_{v_2}$  je libovolný bod z  $SS_2$  (druhý vantage point).
  - 3.6)  $S := S - \{S_{v_2}\}$
  - 3.7) Vypočítej všechny  $d(S_j, S_{v_2})$  kde  $S_j \in SS_1 \cup SS_2$ .  
Pokud ( $level \leq p$ )  $S_j.PATH[level + 1] = d(S_j, S_{v_2})$ .
  - 3.8)  $M_2[1] =$  medián z  $\{d(S_j, S_{v_2}) \mid \forall S_j \in SS_1\}$   
 $M_2[2] =$  medián z  $\{d(S_j, S_{v_2}) \mid \forall S_j \in SS_2\}$
  - 3.9) Rozpul seznam  $SS_1$  do dvou seznamů stejné délky podle  $M_2[1]$ . Podobně pro  $SS_2$ .  $level+ = 2$ , rekurzivně zpracuj všechny čtyři množiny.

## 2.2 Vyhledávání v MVP-stromech

Pro daný objekt  $Q$ , množina datových objektů, které jsou ve vzdálenosti  $r$  od  $Q$  je nalezena vyhledávacím algoritmem popsáným níže:

- 1) Vypočítej vzdálenosti  $d(Q, S_{v_1})$  a  $d(Q, S_{v_2})$ .
  - pokud  $d(Q, S_{v_1}) \leq r$ , pak  $S_{v_1}$  je ve výsledné množině;
  - pokud  $d(Q, S_{v_2}) \leq r$ , pak  $S_{v_2}$  je ve výsledné množině.
- 2) pokud je aktuální uzel list,
  - Pro všechny datové body ( $S_i$ ) v uzlu,
    - 2.1) Najdi  $d(S_i, S_{v_1})$  a  $d(S_i, S_{v_2})$  v polích  $D_1$  a  $D_2$ .
    - 2.2) pokud  $[d(Q, S_{v_1}) - r \leq (S_i, S_{v_1}) \leq d(Q, S_{v_1}) + r]$  a současně  $[d(Q, S_{v_2}) - r \leq (S_i, S_{v_2}) \leq d(Q, S_{v_2}) + r]$ , pak pokud pro všechna  $i = 1 \dots p$  platí  $(PATH[i] - r \leq S_i.PATH[i] \leq PATH[i] - r)$ , vypočítej  $d(Q, S_i)$ .  
Pokud  $d(Q, S_i) \leq r$ , pak  $S_i$  je ve výsledné množině.
- 3) jinak, (pokud je aktuální uzel list interní)
  - 3.1) pokud ( $l \leq p$ ),  $PATH[l] = d(Q, S_{v_1})$ ,  
pokud ( $l < p$ ),  $PATH[l + 1] = d(Q, S_{v_2})$ .
  - 3.2) pokud  $d(Q, S_{v_1} + r \leq M_1$ , pak
    - \* pokud  $d(Q, S_{v_2}) + r \leq M_2[1]$ , rekurzivně prohledej první podstrom s  $l = l + 2$ .
    - \* pokud  $d(Q, S_{v_2}) - r \geq M_2[1]$ , rekurzivně prohledej druhý podstrom s  $l = l + 2$ .
  - 3.3) pokud  $d(Q, S_{v_1} - r \geq M_1$ , pak
    - \* pokud  $d(Q, S_{v_2}) + r \leq M_2[1]$ , rekurzivně prohledej třetí podstrom s  $l = l + 2$ .
    - \* pokud  $d(Q, S_{v_2}) - r \geq M_2[1]$ , rekurzivně prohledej čtvrtý podstrom s  $l = l + 2$ .