KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLOMOUCI
www.inf.upol.cz

Jan Laštovička
https://www.inf.upol.cz/lide/jan-lastovicka
jan.lastovicka@upol.cz
17. listopadu 12, 771 46 Olomouc

Databáze ◇ poznámky k přednášce

11. Fulltextové vyhledávání

verze z 9. prosince 2024

Pro motivaci si představme, že chceme zachytit část světa vymezenou námi vlastněnými filmy. Vlastníme tři filmy označené čísly jedna, dva a tři. U každého filmu nás zajímá pouze jeho název. První film se jmenuje *The Fellowship of the Ring*, druhý *The Two Towers* a třetí *The Return of the King*.

1 Indexy

Dokument je textový řetězec nad anglickou abecedou se znakem pro mezeru. Dokumenty budeme uzavírat mezi dvojité uvozovky. Například `"The Fellowship of the Ring"`. **Token** je neprázdný textový řetězec nad malými písmeny anglické abecedy. Zapisujeme jej přímo, bez použití uvozovek. Například `the` nebo `fellowship`.

Dokumentu D přiřadíme posloupnost **tokenů** $t_1, \ldots, t_n$, kterou získáme tak, že znaky řetězce D převedeme na malá písmena a získaný řetězec rozdělíme na tokeny oddělené aspoň jednou mezerou. Například řetězec `"The Fellowship of the Ring"` rozdělíme na tokeny `the`, `fellowship`, `of`, `the`, `ring`. Zobrazení, které dokumentu přiřazuje posloupnost tokenů, označíme T. Tedy například:

$$\mathrm{T}(\texttt{"The Fellowship of the Ring"}) = (\texttt{the}, \texttt{fellowship}, \texttt{of}, \texttt{the}, \texttt{ring})$$

Volněji budeme tokeny dokumentu ztotožňovat s jeho slovy.

Budeme používat systém Elasticsearch. Návod na jeho instalaci naleznete v souboru `11_elasticsearch.pdf`. Dotazem:

```
POST /_analyze
{
  "analyzer": "standard",
  "text": D
}
```

získáme tokeny dokumentu D:

```
{
  "tokens": [
    {
      "token": t1,
```

```
    },
    ⋮
    {
      "token": t_n
    }
  ]
}
```

Například:

```
POST /_analyze
{
  "analyzer": "standard",
  "text": "   The Fellowship    of the Ring"
}

{
  "tokens": [
    {
      "token": "the",
    },
    {
      "token": "fellowship",
    },
    {
      "token": "of",
    },
    {
      "token": "the",
    },
    {
      "token": "ring",
    }
  ]
}
```

Záznam je dvojice $\langle d, D \rangle$, kde d je přirozené číslo a D je dokument. Číslo d se nazývá **identifikátor** záznamu. Například $\langle 1, \texttt{"The Fellowship of the Ring"} \rangle$ je záznam s identifikátorem 1.

Konečná množina záznamů $\{\langle d_1, D_1 \rangle, \ldots, \langle d_n, D_n \rangle\}$ s po dvou různými identifikátory se nazývá **index**. Například I_1:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"} \rangle,$$
$$\langle 2, \texttt{"The Two Towers"} \rangle,$$
$$\langle 3, \texttt{"The Return of the King"} \rangle\}$$

je index. **Vlastnost** (**záznamu**) je výroková forma $V(d, D)$ s dvěmi proměnnými d a D, kde doména proměnné d je množina přirozených čísel a doména proměnné D je množina všech dokumentů. Například $V(d, D)$ může být: „Film d má název D.“ Říkáme, že index I má **charakteristickou vlastnost** $V(d, D)$, jestliže $\langle d, D\rangle \in I$, právě když $V(d, D)$. Také pak říkáme, že index I je **určen** vlastností $V(d, D)$. Například index I_1 má charakteristickou vlastnost: „Film d má název D.“ Platí, že $I = \{\langle d, D\rangle \mid V(d, D)\}$.

Prázdným indexem myslíme prázdnou množinu. Množinu všech identifikátorů $\{d_1, \ldots, d_n\}$ záznamů v indexu I značíme $\mathrm{Dom}(I)$. Například $\mathrm{Dom}(I_1) = \{1, 2, 3\}$. Index I je zobrazení, které číslu z $\mathrm{Dom}(I)$ přiřazuje dokument. Například $I_1(2) =$ `"The Two Towers"`.

Indexová proměnná je proměnná, jejíž hodnota je index. Hodnotu indexové proměnné ***index*** označujeme I_{index}. Indexové proměnné ***index*** je přiřazena **charakteristická vlastnost** $V(d, D)$, což je vlastnost, která je vždy charakteristickou vlastností její hodnoty.

Indexovou proměnnou ***index*** deklarujeme příkazem:

```
PUT /index
```

Hodnotou deklarované indexové proměnné je prázdný index. Například:

```
PUT /movie
```

Hodnota $I_{\texttt{movie}}$ proměnné `movie` je prázdný index $\emptyset$. Charakteristická vlastnost indexové proměnné `movie` je: „Film d má název D.“

Pro index I a přirozené číslo d index

$$\{\langle d', D'\rangle \in I \mid d' \neq d\}$$

označíme $\mathrm{Del}(I, d)$. Například pro výše uvedený index I_1 je $\mathrm{Del}(I_1, 2)$ rovno:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle, \\ \langle 3, \texttt{"The Return of the King"}\rangle\}$$

Pokud $d \notin \mathrm{Dom}(I)$, pak $\mathrm{Del}(I, d) = I$. Například $\mathrm{Del}(I_1, 4) = I_1$.

Příkaz:

```
PUT /index/_doc/d
{
  "text": D
}
```

nastaví hodnotu proměnné ***index*** na $\mathrm{Del}(I_{index}, d) \cup \{\langle d, D\rangle\}$.

Například:

```
PUT /movie/_doc/1
{
  "text": "The Fellowship of the Ring"
}
```

nastaví hodnotu proměnné `movie` na:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle\}$$

Po vykonání příkazů:

```
PUT /movie/_doc/2
{
  "text": "The Two Towers"
}

PUT /movie/_doc/3
{
  "text": "The Return of the King"
}
```

se hodnota proměnné `movie` změní na:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle,\\ \langle 2, \texttt{"The Two Towers"}\rangle,\\ \langle 3, \texttt{"The Return of the King"}\rangle\}$$

Příkaz:

```
DELETE /index/_doc/d
```

kde $d \in \mathrm{Dom}(I_{index})$ nastaví hodnotu proměnné ***index*** na $\mathrm{Del}(I_{index}, d)$. Například příkaz:

```
DELETE /movie/_doc/2
```

nastaví hodnotu proměnné `movie` na:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle,\\ \langle 3, \texttt{"The Return of the King"}\rangle\}$$

Záznam $\langle d, D\rangle$ zapíšeme řetězcem:

```
{
  "_id": d,
  "_source": {
    "text": D
  }
}
```

Například záznam $\langle 1, \texttt{"The Fellowship of the Ring"}\rangle$ zapíšeme řetězcem:

```
{
  "_id": 1,
  "_source": {
    "text": "The Fellowship of the Ring"
  }
}
```

Hodnota výrazu:

```
GET /index/_doc/d
```

kde $d \in \mathrm{Dom}(I_{index})$ je $\langle d, I_{index}(d)\rangle$. Například:

```
GET /movie/_doc/1
```

```
{
  "_id": 1,
  "_source": {
    "text": "The Fellowship of the Ring"
  }
}
```

Index $\{\langle d_1, D_1\rangle, \ldots, \langle d_n, D_n\rangle\}$ zapíšeme řetězcem:

```
{
  "hits": {
    "hits": [
      ⟨d_1, D_1⟩,
      ⋮
      ⟨d_n, D_n⟩
    ]
  }
}
```

Například index:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle, \\ \langle 3, \texttt{"The Return of the King"}\rangle\}$$

zapíšeme řetězcem:

```
{
  "hits": {
    "hits": [
      {
        "_id": "1",
        "_source": {
          "text": "The Fellowship of the Ring"
        }
      },
      {
        "_id": "3",
        "_source": {
          "text": "The Return of the King"
        }
      }
    ]
  }
}
```

Hodnota výrazu:

```
GET /index/_search
{
  "query": {
    "match_all": { }
  }
}
```

je I_{index}, což je hodnota proměnné ***index***. Například:

```
GET /movie/_search
{
  "query": {
    "match_all": { }
  }
}

{
  "hits": {
```

```
    "hits": [
      {
        "_id": "1",
        "_source": {
          "text": "The Fellowship of the Ring"
        }
      },
      {
        "_id": "3",
        "_source": {
          "text": "The Return of the King"
        }
      }
    ]
  }
}
```

Příkaz:

```
DELETE /index
```

zruší proměnnou ***index***. Například:

```
DELETE /movie
```

2 Podmínky

Zavedeme zobrazení Set, které posloupnosti termů $(t_1, \ldots, t_n)$ přiřadí množinu termů $\{t_1, \ldots, t_n\}$, které se v posloupnosti vysyktují. Například:

$$\text{Set}(\texttt{the}, \texttt{fellowship}, \texttt{of}, \texttt{the}, \texttt{ring}) = \{\texttt{the}, \texttt{fellowship}, \texttt{of}, \texttt{ring}\}.$$

Podmínka je výraz, pro který lze u každého dokumentu rozhodnout, zda ji splňuje. Podmínce θ můžeme přiřadit její výrokovou formu $V_\theta(D)$ danou „Dokument D splňuje podmínku θ.“ Například podmínka `fellowship` má výrokovou formu „Dokument D obsahuje slovo `fellowship`.“.

Term t je podmínka, kterou dokument D splňuje, jestliže

$$t \in \text{Set}(\text{T}(D)).$$

Tedy dokument D splňuje t, jestliže se term t nachází mezi termy dokumentu D. Například dokument `"The Fellowship of the Ring"` splňuje podmínku `fellowship`.

Vezměme podmínku θ a index I. Pak index

$$\sigma_\theta(I) = \{\langle d, D\rangle \in I \mid D \text{ splňuje } \theta\}$$

nazýváme **restrinkcí** I podle θ. Pokud $V_1(d, D)$ je charakteristická vlastnost indexu I, pak charakteristická vlastnost indexu $\sigma_\theta(I)$ je $V_2(d, D)$: „$V_1(d, D)$ a $V_\theta(D)$.“ Například pro index I_1:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle,\\ \langle 2, \texttt{"The Two Towers"}\rangle,\\ \langle 3, \texttt{"The Return of the King"}\rangle\}$$

daný: „Film d má název D.“ je $\sigma_{\texttt{of}}(I_1)$ rovno

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle,\\ \langle 3, \texttt{"The Return of the King"}\rangle\}.$$

a má charakteristickou vlastnost: „Film d má název D, který obsahuje slovo `of`.“

Vezměme podmínky $\theta_1, \ldots, \theta_n$. Pak

$$\theta_1 \vee \cdots \vee \theta_n$$

je podmínka, kterou dokument splňuje, jestliže splňuje aspoň jednu z podmínek $\theta_1, \ldots, \theta_n$. Například $\sigma_{\texttt{two}\vee\texttt{king}}(I_1)$ je rovno:

$$\{\langle 2, \texttt{"The Two Towers"}\rangle,\\ \langle 3, \texttt{"The Return of the King"}\rangle\}.$$

a je dáno: „Film d má název D, který obsahuje slovo `two` nebo `king`.“

Vezměme dokument Q, pak řetězec:

```
{
  "match": {
    "text": Q
  }
}
```

zapisuje podmínku $q_1 \vee \cdots \vee q_n$, kde $\mathrm{T}(Q) = (q_1, \ldots, q_n)$. Například řetězec

```
{
  "match": {
    "text": "Two   King"
  }
}
```

zapisuje podmínku `two` $\vee$ `king`.

Vezměme podmínku θ. Pak hodnotou výrazu:

```
GET /index/_search
{
  "query": θ
}
```

je restrikce $\sigma_\theta(I_{index})$. Například pro indexovou proměnnou `movie` danou: „Film d má název D.“ s hodnotou I_1 uvedenou výše je:

```
GET /movie/_search
{
  "query": {
    "match": {
      "text":  "Two King"
    }
  }
}
```

rovno:

```
{
  "hits": {
    "hits": [
      {
        "_id": "2",
        "_source": {
          "text": "The Two Towers"
        }
      },
      {
        "_id": "3",
        "_source": {
          "text": "The Return of the King"
        }
      }
    ]
  }
}
```

s charakteristickou vlastností: „Film d má název D, který obsahuje slovo `two` nebo `king`.“

3 Kvalita splnění podmínky

Vezměme index I. Například index I_1:

$$\{\langle 1, \texttt{"The Fellowship of the Ring"}\rangle,\\ \langle 2, \texttt{"The Two Towers"}\rangle,\\ \langle 3, \texttt{"The Return of the King"}\rangle\}.$$

Dále vezměme term t. Pak číslo

$$n(I,t) = |\{\langle d, D\rangle \in I \mid t \in \mathrm{Set}(\mathrm{T}(D))\}|$$

udává **počet dokumentů** v I **obsahující term** t. Například $n(I_1, \texttt{the}) = 3, n(I_1, \texttt{of}) = 2, n(I_1, \texttt{king}) = 1, n(I_1, \texttt{three}) = 0$. **Počet záznamů v indexu** I označíme $N(I) = |I|$.

Číslo

$$IDF(I,t) = ln\left(1 + \frac{N(I) - n(I,t) + 0.5}{n(I,t) + 0.5}\right)$$

udává **vzácnost termu** t v indexu I, kde IDF je zkratkou za *inverse document frequency*. Například:

$$\begin{aligned} IDF(I_1, \texttt{of}) &= ln\left(1 + \frac{3 - 2 + 0.5}{2 + 0.5}\right) \\ &= ln\left(\frac{4}{2.5}\right) = ln(1.6) = 0,470\ldots \end{aligned}$$

Vezměme dokument D a označme $\mathrm{T}(D) = (t_1, \ldots, t_n)$. Pak číslo

$$f(t,D) = |\{i \mid t_i = t \text{ a } 1 \leq i \leq n\}|$$

je **počet výskytů termu** t v **dokumentu** D. Například:

$$f(\texttt{the}, \texttt{"The Return of the King"}) = 2.$$

Počet prvků posloupnosti termů $\mathrm{T}(D) = (t_1, \ldots, t_n)$ dokumentu D značíme $dl(D) = n$ a nazýváme **délka dokumentu** D. Například:

$$dl(\texttt{"The Return of the King"}) = 5.$$

Průměrná délka dokumentu v indexu I je dána vzorcem

$$avgdl(I) = \frac{\sum_i^n dl(D_i)}{N(I)}$$

kde $I = \{\langle d_1, D_1\rangle, \ldots, \langle d_n, D_n\rangle\}$. Například:

$$avgdl(I_1) = \frac{5 + 3 + 5}{3} = 4.\overline{3}$$

Frekvence termu t v dokumentu D vzhledem k indexu I je dána vzorcem

$$tf(I,t,D) = \frac{f(t,D)}{f(t,D) + k_1 \cdot \left(1 - b + b \cdot \frac{dl(D)}{avgdl(I)}\right)}$$

kde konstanta $k_1 = 1.2$ (*term saturation parameter*) a $b = 0.75$ (*length normalization parameter*). Například:

$$tf(I_1, \texttt{the}, \texttt{"The Return of the King"}) = \frac{2}{2 + 1.2 \cdot \left(1 - 0.75 + 0.75 \cdot \frac{5}{4.3}\right)} \approx 0.6$$

Definujeme zobrazení *score*, které přiřadí **kvalitu splnění** $score(I, D, \theta)$ podmínky θ dokumentem D z indexu I. Zobrazení definujeme zvlášť pro různé typy podmínek.

Pokud je podmínka term t, pak

$$score(I, D, t) = boost \cdot IDF(I, t) \cdot tf(I, t, D)$$

kde konstanta *boost* je $k_1 + 1 = 2.2$.

Pokud má podmínka tvar $\theta_1 \vee \cdots \vee \theta_n$, pak

$$score(I, D, \theta_1 \vee \cdots \vee \theta_n) = \sum_{i=1}^{n} score(I, D, \theta_i).$$

Definujeme zobrazení

$$\Sigma_\theta(I, I_G) = \{\langle \langle d, D \rangle, score(I_G, D, \theta) \rangle \mid \langle d, D \rangle \in I\},$$

které záznamu $\langle d, D \rangle \in I \subseteq I_G$ přiřadí kvalitu splnění podmínky θ dokumentem D z indexu I_G. Prvek zobrazení $\langle \langle d, D \rangle, s \rangle$, který se nazývá **zásah**, zapíšeme řetězcem:

```
{
  "_id": "d",
  "_score": s,
  "_source": {
    "text": D
  }
}
```

Celé zobrazení $\Sigma_\theta(I, I_G) = \{\langle \langle d_1, D_1 \rangle, s_1 \rangle, \ldots, \langle \langle d_n, D_n \rangle, s_n \rangle\}$ zapíšeme řetězcem:

```
{
  "hits": {
    "hits": [
      ⟨⟨d1, D1⟩, s1⟩,
```

```
      ⋮
      ⟨⟨d_n, D_n⟩, s_n⟩
    ]
  }
}
```

kde pro $1 \leq j, k \leq n$ platí, že pokud $s_j > s_k$, pak $j > k$. Zásahy jsou tedy uspořádány sestupně podle kvality.

Hodnotou výrazu:

```
GET /index/_search
{
  "query": θ
}
```

je zobrazení $\Sigma_\theta(\sigma_\theta(I_{index}), I_{index})$.

Například:

```
GET /movie/_search
{
  "query": {
    "match": {
      "text":  "Two King"
    }
  }
}

{
  "hits": {
    "hits": [
      {
        "_id": "2",
        "_score": 1.1220688,
        "_source": {
          "text": "The Two Towers"
        }
      },
      {
        "_id": "3",
        "_score": 0.9227538,
        "_source": {
          "text": "The Return of the King"
        }
      }
```

```
    ]
  }
}
```

Hodnotou výrazu:

```
GET /index/_search {
  "query": θ,
  "explain": true
}
```

navíc k zásahu přidáme položku `_explanation`, vyjadřující postup při výpočtu kvality zásahu, která má hodnotu:

```
{
  "value": score(I, D, θ),
  "details": [
    {
      "details": [
        {
          "value": boost · IDF(I, q1) · tf(I, q1, D),
          "details": [
            { "value": boost },
            {
              "value": IDF(I, q1),
              "details": [
                { "value": n(I, q1) },
                { "value": N(I) }
              ]
            }, {
              "value": tf(I, q1, D),
              "details": [
                { "value": f(t, D) },
                { "value": k1 },
                { "value": b },
                { "value": dl(D) },
                { "value": avgdl(I) }
              ]
            }
          ]
        },
        ⋮
      ]
    }
  ]
}
```

Například:

```
GET /movie/_search
{
  "query": {
    "match": {
      "text": {
        "query": "Towers"
      }
    }
  },
  "explain": true
}
```

```
{
  "hits": {
    "hits": [
      {
        "_id": "2",
        "_score": 1.1220688,
        "_source": {
          "text": "The Two Towers"
        },
        "_explanation": {
          "value": 1.1220688,
          "details": [
            {
              "details": [
                {
                  "value": 1.1220688,
                  "details": [
                    { "value": 2.2 },
                    {
                      "value": 0.98082924,
                      "details": [
                        { "value": 1 },
                        { "value": 3 }
                      ]
                    },
                    {
                      "value": 0.52000004,
                      "details": [
                        { "value": 1.0 },
                        { "value": 1.2 },
                        { "value": 0.75 },
                        { "value": 3.0 },
                        { "value": 4.3333335 }
```

```
                    ]
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Otázky a úkoly na cvičení

1. Přiřaďte posloupnost tokenů k následujícím dokumentům.

   (a) `The Life And Opinions Of Tristram Shandy`

   (b) `Emma`

   (c) `Nightmare Abbey`

   (d) `One Day in the Life of Ivan Denisovich`

   (e) `Life After Life`

   Porovnejte výsledky s Elasticsearch.

2. Vytvořte indexovou proměnnou `book` s charakteristickou vlastností „Kniha d má název D.“ a přidejte do ní dokumenty z předchozího úkolu.

3. Změňte název druhé knihy na `Frankenstein` a odstraňte z indexu knihu `Nightmare Abbey`.

4. Zjistěte hodnotu indexové proměnné `book`.

5. Určete hodnoty následujících restrikcí a ověřte výsledky systémem Elasticsearch.

   (a) $\sigma_{\texttt{life}}(I_{\texttt{book}})$

   (b) $\sigma_{\texttt{emma}}(I_{\texttt{book}})$

   (c) $\sigma_{\texttt{the}\vee\texttt{of}}(I_{\texttt{book}})$

   (d) $\sigma_{\texttt{falcon}\vee\texttt{opinions}\vee\texttt{shandy}}(I_{\texttt{book}})$

   Určete charakteristické vlastnosti získaných indexů.

6. Spočítejte následující hodnoty.

   (a) $n(I_{\texttt{book}}, \texttt{the})$

(b) $n(I_{\texttt{book}}, \texttt{life})$

(c) $N(I_{\texttt{book}})$

(d) $IDF(I_{\texttt{book}}, \texttt{the})$

(e) $IDF(I_{\texttt{book}}, \texttt{life})$

(f) $f(\texttt{life}, \texttt{"Life After Life"})$

(g) $f(\texttt{life}, \texttt{"Nightmare Abbey"})$

(h) $dl(\texttt{"The Life And Opinions Of Tristram Shandy"})$

(i) $avgdl(I_{\texttt{book}})$

(j) $tf(I_{\texttt{book}}, \texttt{life}, \texttt{"Life After Life"})$

(k) $tf(I_{\texttt{book}}, \texttt{life}, \texttt{"One Day in the Life of Ivan Denisovich"})$

(l) $tf(I_{\texttt{book}}, \texttt{life}, \texttt{"The Life And Opinions Of Tristram Shandy"})$

(m) $tf(I_{\texttt{book}}, \texttt{life}, \texttt{"Nightmare Abbey"})$

(n) $score(I_{\texttt{book}}, \texttt{"Life After Life"}, \texttt{life})$

(o) $score(I_{\texttt{book}}, \texttt{"One Day in the Life of Ivan Denisovich"}, \texttt{life})$

(p) $score(I_{\texttt{book}}, \texttt{"One Day in the Life of Ivan Denisovich"}, \texttt{life} \vee \texttt{the})$

7. U následujících výrazů určete zásahy v jejich hodnotě a pořadí zásahů.

(a)
```
GET /book/_search
{
  "query": {
    "match": {
      "text":  "Life"
    }
  }
}
```

(b)
```
GET /book/_search
{
  "query": {
    "match": {
      "text":  "the"
    }
  }
}
```

(c)
```
GET /book/_search
{
  "query": {
    "match": {
      "text":  "The   LIFE"
    }
  }
}
```

Porovnejte vaše zásahy i pořadí s hodnotami výrazů vypočítanými systémem Elasticsearch. Nechte si systémem zobrazit výpočet kvality splnění zásahů.