Automatická klasifikace dokumentů do tříd za použití metody Itemsets

Jiří HYNEK[1], Karel JEŽEK[2]

[1]*inSITE, s.r.o., Knowledge Management Integrator*
Rubešova 29, 326 00 Plzeň
jiri.hynek@insite.cz
[2]*Katedra informatiky a výpočetní techniky, ZČU Plzeň*
Univerzitní 22, Plzeň
jezek_ka@kiv.zcu.cz

Abstrakt. Motivací pro vznik této metody je snaha o automatizaci časově náročné klasifikace dokumentů v rámci digitální knihovny. Navrhovaná původní metoda je založena na množinách položek (itemsets), čímž rozšiřuje tradiční oblast aplikace Apriori algoritmu, který je v našem případě využit pouze pro generování vstupních dat. Je vhodná zejména pro automatickou klasifikaci krátkých dokumentů (abstraktů, anotací), ve kterých nelze předpokládat opakování slov opravňující použití metod založených na četnosti výskytu termů v dokumentu (metody TF×IDF). Příspěvek prezentuje základní principy metody a výsledky dosažené v praxi. Vysoká úspěšnost algoritmu dovoluje reálné nasazení této metody v komerčním prostředí. Metoda Itemsets je určena k integraci do rozsáhlého informačního systému podniku Západočeská energetika, a.s.

Klíčová slova: itemset, množina položek, klasifikace, Apriori algoritmus, podobnost dokumentů, elektronická knihovna, digitální knihovna

1 Úvod - stručně o digitální knihovně

Vytvoření digitální knihovny je finančně i časově nákladný úkol. Zařazované publikace jsou vesměs chráněny autorským právem a je nutné platit za jejich použití. Jelikož abstrakty odborných článků jsou obvykle na webu volně přístupné, případně je lze vytvořit, je možné levně vytvářet knihovnu abstraktů. Podle jejich obsahu pak uživatelé knihovny žádají o zakoupení plného textu.

Digitální knihovna použitá při implementaci navrhovaného klasifikátoru je reálnou knihovnou nasazenou v komerčním prostředí. Specializuje se na odborné články z oblasti elektroenergetiky a trhu s elektrickou energií. Pro srovnání s jinými metodami byla použita kolekce Reuters-21578, celosvětově hojně využívaná pro testování klasifikačních algoritmů. Testovaná česká kolekce i Reuters-21578 vykazují jistou podobnost, zejména v průměrné velikosti zařazovaných dokumentů. Tento parametr je pro nasazení metody itemsets nejvýznamnější.

Uspořádání témat v české knihovně odpovídá organizační struktuře uživatele knihovny a nebylo zvoleno tak, aby se automatická klasifikace dokumentů usnadnila. Některé tematické okruhy byly do knihovny doplňovány za běhu podle potřeby a dříve začleněné dokumenty již nebyly zpětně přeřazovány. Tato skutečnost degraduje konečnou kvalitu klasifikace, neboť do fáze učení klasifikátoru se zanáší nepřesnosti.

Při implementaci technické knihovny se vycházelo z třívrstvé architektury s tenkým klientem (obvyklý webový prohlížeč), aplikačním serverem (webový server Apache, skripty v jazyce PHP) a databázovým serverem (MySQL).

2 Klasifikace metodou Itemsets

2.1 Apriori algoritmus

Apriori algoritmus využíváme pouze pro generování vstupních dat pro klasifikátor. Vlastní klasifikační metoda Itemsets je původní.

Apriori algoritmus (Agrawal a kol.) představuje účinnou metodu dolování znalostí ve formě asociačních pravidel [2]. Stručný popis uvádíme i v [3] a [4]. Jeho praktické využití spatřujeme i v oblasti kategorizace dokumentů. Původní Apriori algoritmus se aplikuje na transakční databáze nákupních košů na trhu. Naší obdobou této databáze je pak množina dokumentů reprezentovaných množinami významových termů. V souladu s obvyklou terminologií budeme termy značit jako položky a množiny termů jako množiny položek (*itemsets*).

Apriori vlastnost je základem procedury pro efektivní generování kandidátních množin položek. Označme T množinu klasifikačních tříd, $Ç_k$ množinu kandidátních k-množin položek a F_k množinu častých k-množin položek. Generování F_k z $Ç_k$ a potažmo F_{k-1} popisuje následující algoritmus:

```
// Pro 1-itemsety :
Ç1 :=  všechna významová slova;
F1 := ∅;
for ∀ Πi ∈ Ç1 do
  for ∀ tj ∈ T do
    if (podpora Πi je ve třídě tj větší než daný minimální práh)
      then begin
             přidej Πi do F1
             break; // další třídy není nutno zkoumat
       end;

// Pro k-itemsety, kde k ≥ 2:
for ∀ Πi ∈ Fk-1 do
  for ∀ Πj ∈ Fk-1, Πi ≠ Πj do
    if shoda(prvních k-2 položek v Πi a Πj) ∧ poslední položky se liší
      then begin
        c := Πi join Πj;
         // nutno zajistit „častost" všech podmnožin:
        if (∃ podmnožina s, s ⊂ c mající k-1 prvků, kde s ∉ Fk-1)
          then break;
           // c patří do Çk, nyní nutno prověřit „častost" kandidáta:
          else for ∀ tm ∈ T do
```

```
            if (podpora c je ve třídě tm větší než daný minimální práh)
              then begin
                přidej c do Fk;
                break;
              end;
        end;
```

2.2 Terminologie

V rámci tohoto článku budeme vycházet z následující notace:

Π_i	Množina (častých) položek	$D\Pi_i$	Množina dokumentů obsahujících Π_i
T	Téma (klasifikační třída)	$\|D\Pi_i\|$	Počet dokumentů obsahujících Π_i
D	Dokument	DT_i	Množina dokumentů zařazených do tématu T_i
$\overline{D}$	Množina významových termů obsažených v dokumentu D	$\|DT_i\|$	Počet dokumentů zařazených do tématu T_i
L	Počet tematických okruhů	C_i	Soubor množin položek charakterizujících téma T_i
N_i	Počet častých množin položek o mohutnosti *i*	$\|C_i\|$	Počet množin položek charakterizujících téma T_i

Na základě výše popsaného Apriori algoritmu definujeme časté množiny položek různé mohutnosti. Pro jednice: Π_1, Π_2, …, Π_{N1}, pro dvojice: Π_{N1+1}, Π_{N1+2}, …, Π_{N1+N2}, pro trojice: $\Pi_{N1+N2+1}$, $\Pi_{N1+N2+2}$, …, $\Pi_{N1+N2+N3}$ atd.

2.3 Úloha klasifikace

Úlohu klasifikace lze rozdělit na dvě části: *fáze učení* a *fáze vlastní klasifikace*. Fáze učení probíhá v následujících krocích:
(1) Nadefinování hierarchie (stromu) tematických oblastí – provádí odborník na danou problematiku. Nadefinuje se L klasifikačních tříd (L odpovídá počtu listových témat). (2) Ruční zařazení určitého počtu dokumentů do témat odborníkem. Jinými slovy, pro každou třídu klasifikace definujeme klasifikační atributy (trénovací množinu dat). Odborník kategorizuje všechny „trénovací“ dokumenty, které má k dispozici. Do každého tematického okruhu by měl spadat statisticky významný počet dokumentů. (3) Automatické vytvoření charakteristických množin položek pro každý tematický okruh. Během vlastní klasifikace využíváme charakteristických množin položek k zařazování dokumentů do příslušných témat.

2.4 Fáze metody Itemsets

Fáze učení: Pro každou množinu častých položek Π_j můžeme zjistit reprezentativní množinu dokumentů obsahujících Π_j. Označme tuto množinu dokumentů jako $D\Pi_j$. Je patrné, že mohutnost $D\Pi_j$ bude vyšší než jistá prahová hodnota, neboť množina Π_j byla vybrána jako častá.

Množině Π_1 odpovídá množina $D\Pi_1$, Π_2 odpovídá množina $D\Pi_2$, atd. Pracujeme-li pouze s jednicemi, dvojicemi, trojicemi a čtveřicemi, vytvoříme celkem N1 + N2 + N3 + N4 množin dokumentů.

Analogicky, pro každé téma T_i existuje charakteristická množina dokumentů zařazených do tohoto tématu. Označme tuto množinu jako DT_i. Tématu T_1 odpovídá množina DT_1, tématu T_2 pak množina DT_2, atd. Celkem vytvoříme L množin dokumentů.

Naším úkolem je stanovit určitý soubor charakteristických množin položek pro každý tematický okruh, přičemž každá množina položek je asociována s podmnožinou množiny všech témat. Množina položek Π_j je asociována s těmi tématy T_i, kde hodnoty váhy $w_{\Pi j}$ přesahují určenou prahovou hodnotu (tato váha vyjadřuje, do jaké míry množina Π_j charakterizuje třídu T_i). Jako efektivní se ukázal následující výpočet kvalitativní váhy $w_{\Pi j}$:

$$w_{\Pi j} = \frac{\left|D\Pi_j \cap DT_i\right|}{\left|DT_i\right| \times \left[1 + \left|D\Pi_j\right| - \left|D\Pi_j \cap DT_i\right|\right]} \qquad i = 1, 2, \ldots$$

Jmenovatel vzorce normalizuje váhy s ohledem na počet dokumentů patřících do témat T_i a také s ohledem na to, zda se množina položek nevyskytuje příliš často také v jiných tématech. Důležitost termů, které se často vyskytují v jiných dokumentech než DT_i, je tak potlačena (pozornosti neunikne analogie s principem metody TF×IDF). Původně jsme k výpočtu $w_{\Pi j}$ používali vzorec ve tvaru

$$w_{\Pi j} = \frac{\left|D\Pi_j \cap DT_i\right|}{\left|DT_i\right|} \qquad i = 1, 2, \ldots, L$$

Tento vztah udává podporu množiny položek ve třídě, tj. hodnotu udávající v kolika procentech dokumentů třídy se množina položek vyskytuje. Tento fakt však nijak nezohledňuje skutečnost, že množina položek se může často vyskytovat i v ostatních třídách. Pokud by se množina položek stala charakteristickou pro všechny třídy, nepředstavuje její výskyt v dokumentu pro nás žádnou informační hodnotu a na klasifikaci se může projevit spíše nepříznivě. Vzhledem k blízkosti tematických okruhů neumožňovaly v tomto případě množiny položek vyhovující rozlišení. Například významový term „energie“ se vzhledem k charakteru české knihovny vyskytuje ve více než 1/3 všech dokumentů – a je tedy asociován s poměrně velkým počtem tematických okruhů. Potažmo i časté k-tice obsahující tento term spadají do velkého počtu témat, a tudíž přinášejí do klasifikátoru jen velmi malou informační hodnotu.

Po asociaci množin položek s konkrétními tématy na základě výše uvedeného vzorce získáme soubory množin položek C_i reprezentující konkrétní téma T_j.

Fáze klasifikace: Při klasifikaci dokumentů musíme brát do úvahy mohutnost množin položek, abychom rozlišili např. shodu ve dvojicích od shody ve trojicích, neboť statistická významnost těchto množin je různá. Za tímto účelem definujeme váhový faktor odpovídající mohutnosti množiny položek. Pro jednice použijeme wf_1, pro dvojice wf_2, pro trojice wf_3, atd.

Nyní můžeme přistoupit k zařazování dokumentu do tematických okruhů. Předpokládejme, že soubor C_j obsahuje položky Π_1, Π_2, …, $\Pi_{|Cj|}$. Vypočteme váhu odpovídající přesnosti asociace dokumentu D s tématem T_j:

$$W_{T_j}^{D} = \sum_{i=1}^{|C_j|} wf_{|\Pi_i|} \times w_{\Pi_i} \quad kde\ (\Pi_i \in C_j) \wedge (\Pi_i \subseteq \overline{D})\ pro\ všechna\ j = 1, 2, \ldots, L$$

Jinými slovy, váha klasifikace je určena součtem součinů vah $w_{\Pi i}$ s váhovými faktory $wf_{|\Pi i|}$ pro všechny množiny položek daného tématu, které jsou zároveň obsaženy v zařazovaném dokumentu. Důsledkem použití $w_{\Pi i}$ je zdůraznění takových množin položek, které téma T_j charakterizují nejlépe.

Dokument D bude zařazen do tématu T_j, jemuž odpovídá nejvyšší váha $W_{T_j}^D$. Můžeme přirozeně chtít asociovat dokument s větším počtem témat. V takovém případě zařadíme dokument *D* do všech tříd, pro které váha $W_{T_j}^D$ přesáhne θ % maximální dosažené $W_{T_j}^D$ (tj. θ % váhy pro třídu, do které dokument spadá nejlépe). Jako efektivní se ukázala volba θ = 75 (viz výsledky). S rostoucí hodnotou θ klesá přesnost klasifikace, úplnost naopak roste. Měníme-li dynamicky hodnotu parametru θ, přesnost a úplnost se vždy pohybují opačným směrem.

2.5 Modifikace metody – posuvné okénko

Jak jsme již uvedli, metoda nezohledňuje frekvenci výskytu množin položek v dokumentu. Podstatné je pouze to, zda se množina položek v dokumentu vůbec vyskytuje. Jednotlivé položky které tvoří víceprvkové množiny položek, se navíc nemusejí vyskytovat vedle sebe. Nemusí tvořit skutečnou frázi, či slovní spojení, ale stačí pouze fakt, že se společně vyskytují dostatečně často v dokumentech některé třídy.

Tato volnost při hledání častých množin položek má své opodstatnění, neboť naším zájmem není hledat fráze, ale skutečně pouze množiny společně se vyskytujících termů. I přes tento fakt je vhodné ověřit, jak se bude metoda chovat v případě, kdy omezíme vzdálenost mezi jednotlivými termy tvořícími víceprvkovou množinu položek.

Do implementace metody bylo přidáno restriktivní opatření zajišťující akceptaci pouze těch k-množin položek (pro $k \geq 2$), které se vyskytují v rámci okénka (angl. *sliding window*). Rozměr okénka je nastavitelný, a je tedy možné nastavit v jakém rozpětí se může množina položek vyskytnout, aby byla ještě akceptována. Pokud nastavíme při hledání dvojic rozměr okénka roven dvěma, nebo při hledání trojic roven třem, atd., budou se akceptovat pouze množiny položek složené z termů, jež se vyskytují vedle sebe. Nastavíme-li rozměry okénka na příliš vysokou hodnotu (větší nebo rovnou délce nejdelšího dokumentu), bude se metoda chovat stejně jako v případě, kdy se žádné okénko neuvažuje.

Níže uvedený algoritmus popisuje kontrolu výskytu množiny položek v rámci okénka uvnitř dokumentu. Uvažujeme zde fakt, že můžeme určit pozici termu v dokumentu. Pozice udává pořadí konkrétního slova v rámci dokumentu. Dokument vždy začíná slovem s pozicí 1 a končí slovem s pozicí rovnou délce dokumentu.

```
nalezni první výskyty všech termů tvořících množiny položek uvnitř dokumentu;
while (1)  begin
  MIN := pozice termu s nejnižší pozicí;
  MAX := pozice termu s nejvyšší pozicí;
  if ((MAX – MIN) < VELIKOST_OKÉNKA)
    return 1; // OK množina položek se vyskytuje v rámci okénka
  else begin
    nalezni další výskyt termu s nejnižší pozicí;
```

```
    if (další výskyt již není)
      return 0; // množina položek se v rámci okénka nevyskytuje
  end;
end;
```

Při vlastní realizaci jsme implementovali složitější verzi algoritmu, která umožňuje specifikovat kolik výskytů množiny položek se musí v dokumentu v rámci okénka nalézt, aby byl její výskyt akceptován. Zpracovávané dokumenty jsou však tak krátké, že není vhodné si vícenásobný výskyt množiny položek v dokumentu vynucovat.

3 Složitost algoritmu

Hodnocení časové a paměťové náročnosti klasifikace na datové kolekci Reuters-21578 (10 202 dokumentů, 10 vhodných tříd) se odvíjí od testů uvedených v závěru článku, kde porovnáváme jednotlivé metody mezi sebou. Naindexování textové kolekce Reuters před natrénováním vyžaduje 23,8 MB RAM a přibližně 15 sekund. Natrénování klasifikátoru si vyžádá necelých 10 MB RAM a 10 sekund, vlastní klasifikace cca 5 MB RAM a 8 sekund. Lze tedy počítat s časem v řádu sekund na natrénování jedné třídy (v konfiguraci PII-466, OS Windows 98). Spolehlivé natrénování tohoto objemu dat při generování 2-množin položek nebo větších vyžaduje alespoň 128 MB RAM. Doba výpočtu při použití Apriori algoritmu je do značné míry závislá na prahové hodnotě pro výběr častých množin položek. S narůstající mohutností častých množin položek rostou i časové nároky. Je to způsobeno kombinováním množin položek nižší mohutnosti ve smyslu Apriori algoritmu.

4 Vyhodnocení a porovnání s jinými metodami klasifikace

Následující tabulka shrnuje výsledky klasifikátoru dosažené na kolekci Reuters-21578. Parametr "minimální počet dokumentů ve třídě" byl nastaven na 50. Pro každou třídu se používá pevný počet charakteristických n-tic.

PC1 (Práh asociace charakteristického termu) … Každou třídu může charakterizovat pouze několik termů s nejlepší váhou. Term bude uvažován jako charakteristický pro třídu tehdy, nabývá-li příslušná váha (míra s jakou je třída slovem charakterizována) hodnoty alespoň „PC1" procent nejvyšší dosažené váhy.

MPD (min. počet dokumentů třídy s výskytem termu) … Term bude považován za častý, pokud se vyskytuje alespoň v „MPD" procentech dokumentů nějaké třídy.

PAT (práh asociace třídy) … Dokument *D* zařadíme do všech tříd, pro které vypočtená váha $w_{T_j}^D$ přesáhne „PAT" % maximální dosažené $w_{T_j}^D$ (tj. „PAT" % váhy pro třídu, do které dokument spadá nejlépe).

P_{trn}, R_{trn}, Q_{trn}, M_{trn}…přesnost, úplnost, jejich průměr a počet dokumentů, jež se musí klasifikovat ručně při klasifikaci trénovacích dokumentů.

P_{tst}, R_{tst}, Q_{tst}, M_{tst}…přesnost, úplnost, jejich průměr a počet dokumentů, jež se musí klasifikovat ručně při klasifikaci testovacích dokumentů

PC1	MPD [%]	PAT [%]	P_{trn} [%]	R_{trn} [%]	Q_{trn} [%]	M_{trn} [%]	P_{tst} [%]	R_{tst} [%]	Q_{tst} [%]	M_{tst} [%]
20	10	40	88,79	95,37	92,08	1,64	85,59	93,52	89,56	2,61
20	10	65	91,15	92,29	91,72	1,64	89,05	90,46	89,75	2,61
20	10	90	92,36	89,26	90,81	1,64	90,21	87,70	88,96	2,61
20	20	40	82,93	96,15	89,54	0,00	82,33	95,43	88,88	0,00
20	20	65	88,18	91,63	89,90	0,00	87,86	91,58	89,72	0,00
20	20	90	89,36	86,68	88,02	0,00	90,13	87,88	89,01	0,00
20	30	40	79,20	94,09	86,64	0,00	79,09	94,85	86,97	0,00
20	30	65	84,83	89,72	87,27	0,00	84,87	91,15	88,01	0,00
20	30	90	87,69	86,54	87,11	0,00	86,82	85,78	86,30	0,00
40	10	40	87,08	97,48	92,28	0,00	84,03	95,79	89,91	0,00
40	10	65	92,07	94,11	93,09	0,00	90,03	93,18	**91,61**	0,00
40	10	90	93,01	90,25	91,63	0,00	91,75	89,48	90,61	0,00
40	20	40	82,01	97,03	89,52	0,00	80,90	95,86	88,38	0,00
40	20	65	88,54	93,29	90,92	0,00	88,49	92,24	90,36	0,00
40	20	90	90,54	87,93	89,24	0,00	90,08	88,32	89,20	0,00
40	30	40	79,19	94,47	86,83	0,00	78,85	94,99	86,92	0,00
40	30	65	84,87	89,68	87,28	0,00	85,18	91,22	88,20	0,00
40	30	90	87,94	86,75	87,35	0,00	87,08	85,99	86,54	0,00

Rekordní hodnotou (**∅ P a R**) dosaženou při dalším ladění parametrů bylo **96,59 %**. V tomto případě byly však klasifikovány výhradně dokumenty v předchozím použité pro natrénování, navíc ve 47 % případů nenalezl algoritmus žádnou třídu, tj. práci nechal na knihovníkovi[1]. Uvažujeme-li raději prakticky využitelné výsledky, tj. klasifikují se nově příchozí dokumenty nepoužité k natrénování a knihovník nezasahuje vůbec (pro nás priorita), dosáhli jsme stále vynikajících **91,61 %** (P = 90,03 %, R = 93,18 %). Jakýkoli další malý nárůst ∅ P a R vede k výraznému nárůstu počtu dokumentů, jež je nutno zařadit ručně.

Pro porovnání s jinými metodami jsme využili výsledky uvedené v literatuře [2], kde se k testování různých klasifikátorů využívá též kolekce Reuters-21578. Porovnávané metody využívají 12 902 dokumentů, které byly klasifikovány do 118 tříd, přičemž výsledky se uvádí jen pro 10 největších (obsahují téměř 75 % všech dokumentů). Proto jsme stejně jako ostatní autoři použili jen třídy obsahující alespoň určitý počet dokumentů - v tomto případě 180 ks, přičemž rozsah mohutnosti jednotlivých tříd činil 212 až 2 779 dokumentů. Dokumenty jsme rozdělili na trénovací a testovací v poměru 3:1, stejně jako u porovnávaných metod.

Klasifikační metoda	**Itemsets**	NBayes	BayesNets	Linear SVM
∅ P a R	**91,36**[2]	81,50	85,00	92,00

[1] Průměr P a R zde přirozeně nezahrnuje ty dokumenty, které systém nebyl schopen automaticky zařadit. Parametry vedoucí k dosažení tohoto výsledku pro stručnost neuvádíme, neboť výsledek není prakticky využitelný.

[2] Testujeme-li klasifikátor na dokumentech použitých k natrénování, dosahujeme hodnoty (P+R)/2 = 91,90 %.

5 Závěr

Malá průměrná délka dokumentů (abstraktů) by v případě metod typu TF×IDF činila klasifikátoru problémy, v našem případě je naopak prospěšná. Při klasifikaci se totiž nevychází z opakování výskytu termů, avšak ze současného výskytu určitých dvojic, trojic (obecně n-tic) termů, které se použijí jako charakteristické. S rostoucí délkou dokumentů by se pouze zavlékaly další obecné termy, které právě pro svoji obecnost nelze chápat jako charakteristické pro třídu, do které byl dokument ručně zařazen.

Z dosažených výsledků je zřejmé, že vysoká úspěšnost klasifikátoru dovoluje jeho nasazení v reálném prostředí. Časové i výkonové nároky algoritmu jsou velmi rozumné, přičemž k čerpání absolutní většiny výpočetních zdrojů dochází ve fázi učení, která může probíhat off-line. Vlastní klasifikace je nesmírně rychlá.

Další výzkum bude zaměřen na automatickou tvorbu anotací k odborným článkům, volbu parametrů s dopady na úspěšnost klasifikace (délka dokumentů, objem trénovacích dat, velikost tříd), modifikaci naivního Bayesova klasifikátoru o prvky metody Itemsets a rovněž využití zde popisované metody Itemsets ke shlukování dokumentů.

Literatura

1. Agrawal et al.: Advances in Knowledge Discovery and Data Mining, MIT Press 1996, 307-328
2. Dumais S., Platt J., Heckerman D., Sahami M.: Inductive Learning Algorithms and Representations for Text Categorization, CIKM 98, Bethesda MD, U.S.A.
3. Hynek J., Ježek K., Rohlík O.: Short Document Categorization – Itemsets Method – sborník mezinárodní konference PKDD 2000, Lyon – Francie, září 2000
4. Hynek J., Ježek K.: Document Classification Using Itemsets – sborník mezinárodní konference MOSIS 2000, Rožnov pod Radhoštěm, květen 2000
5. Mladenic D., Grobelnik M.: Word Sequences as Features in Text Learning, Proc. Seventh Electrotechnical and Computer Science Conf. (ERK 98), IEEE Region 8, Slovenia Section IEEE, 1998, pp. 145 – 148

Příspěvek vznikl za částečné podpory výzkumného záměru MSM 235200005.

Annotation:

Automatic Document Classification Using Itemsets

The essential point of this paper is to develop a method for automating time-consuming document classification in a digital library. The original method proposed in this paper is based on itemsets, extending traditional application of the Apriori algorithm. It is suitable for automatic classification of short documents (abstracts, summaries) impeding usage of repeated occurrence of terms, such as in term-frequency-based methods. The paper presents basic principles of this method as well as results of its practical use. High success rate of the classification algorithm allows its usage in real-life environment. The method will become an integral part of the information system of a regional utility company.