Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Matúš Cimerman

Analýza prúdu prichádzajúcich udalostí použitím rôznych metód pre analýzu údajov

Diplomová práca

Vedúci práce: Ing. Jakub Ševcech

máj, 2016

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Matúš Cimerman

Analýza prúdu prichádzajúcich udalostí použitím rôznych metód pre analýzu údajov

Diplomová práca

Študijný program: Informačné systémy
Študijný odbor: x.x.x Informačné systémy
Miesto vypracovania: Ústav informatiky a softvérového inžinierstva, FIIT STU Bratislava
Vedúci práce: Ing. Jakub Ševcech

máj, 2016

Abstract

Nowadays we can see emerging need for data analysis as data occur. Processing and analysis of data streams is a complex task, first, we particuraly need to provide low latency and fault-tolerant solution.

In our work we focus on proposal a set of tools which will help domain expert in process of data analysis. Domain expert do not need to have detailed knowledge of analytics models. Similar approach is popular when we want analyse static collections, eg. funnel analysis. We study possibilities of usage well known methods for static data analysis in domain data streams analysis. Our goal is to apply method for data analysis in domain of data streams. This approach is focused on simplicity in use of selected method and interpretability of results. It is essential for domain experts to meet these requirements because they will not need to have detailed knowledge from such a domains as machine learning or statistics. We evaluate our solution using software component implementing chosen method.

Abstrakt

Dnes môžeme pozorvať narastujúcu potrebu analyzovať dáta počas ich vzniku. Spracovanie a analýza prúdov dát predstavuje komplexnú úlohu, pričom je dôležité poskytnúť riešenie s nízkou odozvou, ktoré je odolné voči chybám.

V našej práci sa sústreďujeme na návrh súboru nástrojov, ktoré pomôžu doménovému expertovi počas analýzy dát.

Poďakovanie

Na prvom mieste vyslovujem poďakovanie vedúcemu mojej bakalárskej práce, Ing. Jakubovi Ševcechovy, za všetky jeho odborné rady, odovzdané skúsenosti a usmernenie pri tvorení práce.
Touto cestou taktiež vyslovujem poďakovanie všetkým výskumníkom zo skupiny PeWe, za prínosné diskusie a ich spätnú väzbu týkajúcu sa mojej práce.
V neposlednom rade ďakujem celej mojej rodine a priateľom.

Matúš Cimerman

Obsah

1. Úvod

V súčasnosti pozorujeme vysoký nárast záujmu v oblasti analýzy údajov. Vhodné použitie metód a techník na analýzu prináša hodnotné výstupy pre používateľa, ktoré môžu môžu byť použité pre strategické rozhodnutia v podnikoch. Najčastejší postup je aplikovaním metód ako napríklad lieviková analýza alebo rozhodovacie stromy nad statickou kolekciu dát. Avšak tento prístup neposkytuje okamžitý výsledok pre používateľa. Nedostatkom takýchto prístupov je nutnosť dáta najskôr zozbierať a uložiť, čo je dnes, kedy vznikajú milióny záznamov za deň, veľký problém.

TODO: Pridať vysvetlenie pojmu *analytika*.

Pri dolovaní v prúde dát čelíme niekoľkým výzvam: objem, rýchlosť (frekvencia) a rozmanitosť. Veľký objem dát, ktoré vznikajú veľmi rýchlo je potrebné spracovať v ohraničenom časovom intervale, často v takmer reálnom čase (závisí od kontextu problému). Objem dát sa neustále zvyšuje, potenciálne až do nekonečna. Identifikujeme niekoľko najviac zasiahnutých oblastí, ktoré sú zdrojmi týchto dát, a to senzory, počítačové siete, sociálne siete a Internet Vecí (angl. Internet of Things). Na informácie generované z takýchto zdrojov sa často pozeráme ako na neohraničené a potenciálne nekonečné prúdy údajov. Spracovanie a nasledujúca analýza týchto prúdov je komplexná úloha. Pre aplikácie je kritické spracovať údaje s nízkou odozvou, pričom riešenie musú byť presné, škálovateľné a odolné voči chybám. Nakoľko sú prúdy neohraničené vo veľkosti a potenciálne nekonečné, môžeme spracovať len ohraničený interval prúdu. Potom hovoríme, že dáta musia byť spracované tak ako vznikajú. Tradičné metódy a princípy pre spracovanie statickej kolekcii údajov nie sú postačujúce na takéto úlohy.

2. Otvorené problémy pri analýze prúdu údajov

Dolovanie a objavovanie znalostí, alebo tiež analýza a spracovanie prúdu dát čelí trom hlavným výzvam: *objem*, *rýchlosť* a *nestálosť* dát (Krempl et al., 2014).

2.1 Použitie tradičných metód dolovania dát a strojového učenia

2.2 Evaluácia modelov a algoritmov

2.3 Vizualizácia výsledkov používateľovi

2.4 Technické výzvy

3. Analytické úlohy v prúde udalostí

Spracovanie a dolovanie znalostí predstavuje výzvu, zvláštnu pozornosti si tieto úlohy vyžadujú pri spracovaní a analýze prúdu udalostí. Prúd udalostí je tiež často nazývaný *prúd dát* alebo *údajov*, či len skrátene *prúd*. V tomto texte budeme pre jednoduchosť používať najmä termín *prúd* (Tran et al., 2014). Avšak môžu sa vyskytnúť aj terminý ako: *prúd dát*, *prúd udalostí*, *sekvencia udalostí, či elementov*, pričom všetky termíny majú v tomto texte *rovnaký* význam.

Definícia 3.0.1 *Prúd je nekonečná sekvencia elementov $S = ()$*

Takmer každé odvetvie dnes generuje masívne množstvo dát, ktoré obsahujú hodnotné znalosti a poznanie. Vzhľadom na veľký objem vzniknutých dát, analytyci často strácajú schopnosť dolovať v celej sade dát. Stáva sa preto častým zvykom, že sa analyzuje len reprezentujúca vzorka, pretože to predstavuje menšiu časovú výzvu pre doménového experta (Hulten et al., 2001). Tvrdíme, že bude pre doménoveho experta vysokým prínosom možnosť vykonávať analýzy nad prúdom v reálnom čase. Výstupy z takejto analýzy sú na rôznej granularite a úrovni, pričom môžu byť neskôr použité na ďalšie spracovanie alebo na priame prezentovanie výsledkov.

3.1 Predspracovanie prúdu

Predspracovanie je azda najdôležitejším krokom v aplikáciach reálneho sveta a časovo najnáročnejšou úlohou pre každého analytika. Nakoľko dáta prichádzajú z nehomogénneho sveta, môžu byť zašumené, nekompletné,

duplicitné alebo často obsahovať hodnoty, ktoré sa značne líšia od ostatných. Predspracovanie prúdiach údajov je potrebné čo najviac automatizovať. Existuje potreba pre implementovanie metód strojového učenia, ktoré sa adaptujú v čase s meniacimi sa dátami. Avšak, tieto modely a metódy by mali byť synchronizované s prediktívnymi modelmi. Pri aplikovaní strojového učenia a prediktívnych modelov je nutnosť uvažovať historické dáta. Čím podstatne narastá zložitosť celého prístupu. Hlavné výskumné problémy rozdeľujeme do dvoch hlavých kategórií, *dátovo orientované* a *orientované na úlohu*. Dátovo orientované techniky používané pri spracovaní prúdiacich údajov:

- Vzorkovanie je proces výberu dátovej vzorky na spracovanie podľa pravdepodobnostného modelu. Problém pri vzorkovaní je v kontexte analýzy prúdiach dát je potenciálne nekonečný dataset, resp. nevedomosť jeho skutočnej veľkosti.
- Kontrolovanie zaťaženia je proces zahadzovania niektorých, potenciálne nepotrebných vzoriek dát. Opäť tu nástava problém v kontexte prúdu. Môže nastať moment, kedy práve zahodená vzorka dát je pre aktuálnu analýzu príznačná a dôležitá.
- Agregácia je proces vypočtu štatistických údajov nad prúdmi. Problém je pri veľkých tokoch v prúde, pričom je tiež potreba pohľadu do minulosti.

- Aproximačné algoritmy použitie aproximačných algoritmov má za následok podstatné zrýchlenie spracovania a analýzy prúdov za predpokladu istej chybovosti. Chybovosť je zvačsa ohraničená.
- Posuvné okno, tento prístup vznikol s potrebou analýzy definovaného časového okna z prúdiacih údajov.

3.2 Dolovanie a extrakcia informácií

Dolovanie z prúdiacih dát dnes predstavuje niekoľko výziev. Jednou z oblastí je bezpečnosť a dôvernosť v dolovaní dát. Návrh modelu môže byť navrhovaný

na pôvodných dátach, ale nasadenie by malo byť realizované na anonymizovaných dátach. Identifikujeme dve hlavné výzvy pre uchovanie bezpečnosti pri dolovaní v prúdoch. Prvou je nekompletnosť informácií, informácie často prichádzajú nekompletné alebo neaktuálne. Druhá výzva nadväzuje na prvú, a to že dáta sa môžu v čase meniť a vyvýjať. Môže sa meniť štruktúra, komplexita a ich presnosť. Preto, vopred definované bezpečnostné pravidlá a politiky nemusia byť časom aktuálne a pokrývať stanovenú oblasť. Medzi známe techniky dolovania v prúdoch údajov považujeme nasledujúce:

- Klastrovanie, existuje niekoľko výskumov, ktoré sa venovali špeciálne klastrovaniu implementovaním napríklad k-mediánu a inkrementálnych algoritmov.
- Klasifikácia, opäť existuje niekoľko známych výskumov, ktoré s venujú problému klasifikácie. Napríklad použitím dát z reálneho sveta a umelých dát, pričom implementujú algoritmy, ktoré triedia dáta na základe porovnaní medzi týmito dvoma vzorkami.
- Počítanie frekvencie a opakovaní, použitím posuvných okien a inkrementálnych algoritmov na detekciu vzorov v prúde.
- Analýza časových radov použitím symbolickej reprezentácie časových radov v prúde dát. Takáto reprezentácia nám umožňuje redukciu veľkosti prenášaných dát. Táto technika pozostáva z dvoch hlavných krokov, aproximácia po častiach a následná transformácia výsledku do diskrétnych veličín.

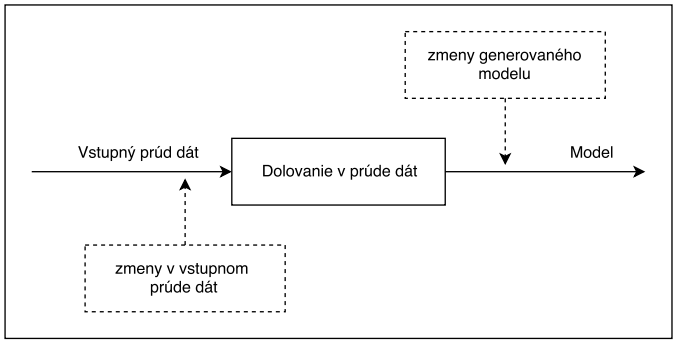
3.3 Detekcia zmien

Detekcia zmien zohráva, v dnešnom rýchlo sa meniacom svete, dôležitú úlohu. Zmeny nastávajú veľmi rýchlo a nečakane. Preto je preto stúpa potreba detekcie zmeny a následná správna reakcia, ktorá vyplynie z detekovanej zmeny. Na to aby sme boli schopný na tieto zmeny adekvátne reagovať je potrebné dáta spracovávať tak ako vznikajú a pozerať sa na ne ako na prúd udalostí. Tradičné metódy pre paralelné spracovanie uvažujú len statickú

kolekciu dát (Tran et al., 2014).

Detekcia zmeny predstavuje proces identifikácie zmeny aktuálneho stavu objektu voči predchádzajúcemu. Na tento objekt sa pozeráme v rôznom čase. Dôležitý rozdiel medzi zmenou a rozdielom je, že zmena hovorí o prechode objektu do iného stavu, zatiaľ čo rozdiel znamená nepodobnosť v atribútoch dvoch objektov. V kontexte prúdu detekovanie zmeny je proces segmentácie prúdu udalostí do rôznych segmentov a identifikovanie miest kde sa zmení dynamika prúdu (Ross et al., 2009). Metóda pre detekciu zmien musí riešiť nasledujúce úlohy (Tran et al., 2014): *detekcia zmeny* a *lokalizácia zmeny*. Ďalší rozdiel, ktorý je potrebné zadefinovať, je medzi rozdiel detekovaním posunu pojmu (angl. concept drift, ďalej len *concept drift*). Detekcia concept drift-u sa sústreďuje na označkované dáta, zatiaľ čo detekcia zmeny pracuje s označkovanými rovnako ako s neoznačkovanými dátami.

Metódy pre detekovanie zmien môžme klasifikovať do nasledujúcich prístupov (Liu et al., 2010): *metódy založené na stave*, *metódy sledujúce trend* a *prahové metódy*. Algoritmus pre detekciu zmien by mal spĺňať aspoň nasledovné požiadavky: *presnosť*, *rýchlosť* a *odpoveď v reálnom čase*. Algoritmus by tiež mal detekovať čo najmenej chybných zmien a čo najviac správnych presných miest zmeny. Algoritmy by mali byť prispôsobené reálnemu prostrediu a spracovaniu prúdov vysokých objemov a rýchlostí. Na obrázku 3.3 je zobrazený všeobecný diagram pre detekciu zmeny v prúde udalostí.



Obrázok 3.1: Všeobecný diagram zobrazujúci detekciu zmeny v prúde udalostí.

Prístupy k detekovaniu zmien v prúde dát (Tran et al., 2014):

- *model prúdu dát* môže byť jeden z nasledujúcich: model časových radov, pokladničný model a model turniketu. Podľa modelu prúdu dáť existujú príslušné algoritmy, ktoré boli vytvorené pre daný model.

- *charakteristika dát*, metódy pre detekciu zmien môžu byť klasifikované na základe charakteru dát, s ktorými pracujú. Najčastejšie môžme prúdu klasifikovať do kategorických alebo numerických prúdov.

- *kompletnosť štatistickej informácie*, mnoho aplikácií reálneho sveta nemá normálne rozdelenie dát, preto je potrebné aplikovať modely a metódy, ktoré sú parametrické alebo semi-parametrické. Ďalej je potrebné spomenúť neparametrické metódy, ktoré pracujú obvykle s posuvným oknom.

- *rýchlosť prúdiacich dát*, boli navrhnuté rámce a metódy pre detekciu zmeny na základe zmeny hustoty vznikajúcich dát v vopred používateľom zadefinovanom časovom okne.

3.4 Detekcia anomálií

Detekcia anomálií predstavuje proces identifikácie dát, ktoré sa významne odchyľujú (angl. deviate) od historických vzorov (Hodge and Austin, 2004). Anomálie môžu spôsobovať chyby v meraní senzorov, nezvyčajné správanie systému alebo chyba pri prenose dát, či zámerné vytváranie anomálií v používateľmi generovanom obsahu. Takže detekcia anomálií má veľa praktického použitia napríklad v aplikáciach, ktoré dohliadajú na kvalitu a kontrolu dát (Hill et al., 2007) alebo adaptívne monitorovanie sietí (Hill and Minsker, 2010). Tieto aplikácie často kladú požiadavku aby boli anomálie detekované v čase ich v vzniku, teda v reálnom čase. Potom metódy pre detekciu anomálií musia byť rýchle vo vykonávaní a mať inkrementálny charakter.

V minulosti sa obvykle anomálie detekovali manuálne s pomocou vizualizačných nástrojov, ktoré doménovým expertom pomáhali v tejto úlohe. Manuálne metódy avšak zlyhávajú pri detekcií anomálií v reálnom čase. Výskumníci navrhli niekoľko metód, ktoré majú myšlienku v prístupoch strojového učenia sa a automatizovaného štatistického vyhodnocovania (Hill and Minsker, 2010):

minimálny objem elipsoidu, *konvexný zvon*, *najbližší sused*, *zhlukovanie*, *klasifikácia neurónovou sieťou*, *klasifikácia strojom podporných vektorov* a *rozhodovacie stromy*. Tieto metódy sú pochopiteľne rýchlejšie než manuálna detekcia, avšak jeden význačný nedostatok, niesú vhodné pre prúdové spracovanie v reálnom čase.

Dátovo riadená metóda (angl. data-driven), ktorú navrhli (Hill and Minsker, 2010), využíva dátovo riadený jednorozmerný autoregresívny model prúdu dát a predikčný interval (ďalej len PI) vypočítaný z posledných historických dát na identifikáciu anomálií v prúde. Dátovo riadený model časového radu je použitý, pretože je jednoduchší na implementáciu a použitie v porovnaní s ostatnými modelmi časových radov. Tento model tiež poskytuje rýchle a presné prognózy. Dáta sú potom klasifikované ako anomálie na základe toho, či sú spadnú do zvoleného intervalu PI. Metóda teda poskytuje principiálny rámec pre výber hraničného prahu kedy majú byť anomálie klasifikované. Výhoda metódy je, že nevyžaduje žiadne vzorky dát, ktoré sú vopred označkované alebo klasifikované. Je veľmi dobre škálovateľná na veľké objemy dát a vykonáva inkrementálne počítanie tak ako dáta vznikajú. Metóda pozostáva z nasledujúcich krokov so začiatkom v čase t:

1. použi model na predikciu o krok vpred (angl. one-step-ahead), ktorý má ako vstup $D^t = \{x_{t-q+1}, ..., x_t\}$ q je rôzne meranie x v čase t a D^t je model predikcie. Tento model je použitý ne predikovanie hodnoty $\overline{x}_{t+1}$ ako očakávaná hodnota v čase *t+1*.

2. výpočet hornej a spodnej hranice kam by malo spadnúť pozorované meranie s pravdepodobnosťou p.

3. porovnaj pozorovanie v čase *t+1*, či spadá do určeného intervalu. Ak spadne mimo intervalu, objekt je klasifikovaný ako anomália.

4. (a) pri stratégii metódy detekcie anomálií a zmiernenia (angl. anomaly detection and mitigation) ADAM, ak je pozorovaný objekt klasifikovaný ako anomália, modifikuj D^t odstránením x_{t-q+1} z konca pozorovaného okna a pridaním $\overline{x}_{t+1}$ na začiatok okna, čím vytvoríme D^{t+1}.

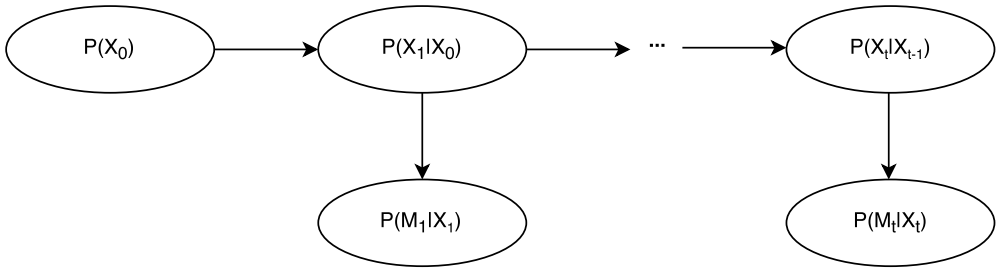
(b) pri jednoduchej stratégii detekcie anomálií (angl. anomaly detection) AD, modifikuj D^t odstránením x_{t-q+1} z konca okna a pridaj x_{t+1} na začiatok okna čím vznikne D^{t+1}.

5. opakuj kroky *1-4*

Metóda dynamických bayesových sietí (angl. Dynamic Bayesian Networks) (Hill et al., 2007) bola vytvorená pre detekciu anomálií v prúdoch zo senzorov, ktoré sú umiestnené v životnom prostredí. Bayesové siete predstavujú acyklický orientovaný graf, zobrazené na obrázku 3.4, v ktorom každý uzol obsahuje pravdepodobnostú informáciu v súvislosti k všetkým možným stavom, v ktorých sa môže premenná nachádzať. Táto informácia spolu s topológiou bayesovej siete, špecifikuje úplné spojenie distribúcie stavu premennej, pričom sada známych premmených môže byť použitá na odvodenie hodnoty neznámych premenných. Dynamické bayesové siete s topológiou, ktorá sa vyvýja v čase, pridáva nové stavové premenné pre lepšiu reprezentáciu stavu systému v aktuálnom čase t. Stavové premmné môžeme kategorizovať ako *neznáme*, ktoré predstavujú skutočný stav systému a *merané*, ktoré sú nedonalé merania. Tieto premenné môžu byť naviac diskrétne alebo spojité. Nakoľko sa veľkosť siete zväčšuje s časom, vytváranie záverov použitím celej siete by bolo neefektívne a časovo náročné. Preto boli vyvinuté aproximačné algoritmy ako *Kalmanové filtrovanie* alebo *Rao-Blackwellized častícové filtrovanie*.
Hill et al. navrhli v (Hill et al., 2007) dve stratégie pre detekovanie anomálií v prúde dát:

- *Bayesov dôveryhodný interval* (angl. Bayesian credible interval - BCI), ktorý sleduje viacrozmernú gausovskú distribúciu lineárneho stavu premennej, ktorý korešponduje s neznámym stavom systému a jej meraným náprotivkom.

- *Maximálne posteriori meraný status* (angl. Maximum a posteriori measurement status - MAP-ms) používa komplexnejšiu dynamickú bayseovú sieť. Princíp je rovnaký ako pri BCI, pričom MAP-ms metóda je naviac rozšírená o status (napr. anomália áno/nie), ktorý je reprezentovaný distribúciou diskrétnej premennej každého merania senzoru.



Obrázok 3.2: Štruktúra dnamickej bayseovej siete. Vektor X reprezentuje spojitú zložku, neznáme alebo tiež nazývané skryté premenné systému a vektory M predstavujú spojité pozorované premenné v čase t.

3.5 Detekcia trendov

Detekcia trendov predstavuje kritickú úlohu pre analytikov. Reagovať na vzniknutý trend v čase jeho vzniku môže mať kritické dopady na fungovanie spoločnosti. Preto existuje záujem detekovať trendy v čase ich vzniku a byť schopný adekvátne reagovat príslušnými akciami v reálnom čase. Ak hovoríme o trendoch v obsahu, ktorý je generovaný používateľmi, napríklad na sociálnej sieti, potom sú trendy typicky poháňané udalosťami, ktoré náhle vznikajú a používatelia javia o ne záujem (Mathioudakis and Koudas, 2010). Mathioudakis and Koudas navrhli a implementovali metódu na detekciu trendov na sociálnej sieti Twitter[1]. Metóda vykonáva detekciu trendov a ich následnú dodatočnú analýzu. Detekcia trendu pozostáva z dvoch krokov:

1. *detekcia nárazových kľúčových slov* identifikuje keď sa kľúčove slovo K začne vyskytovať v prúde s neobvykle vysokým podielom v prúde. Napríklad náhly nárast frekvencie kľúčového slova *NBA* môže byť spojený s prebiehajúcim dôležitým zápasom NBA. Pre detekciu nárazových kľúčových slov navrhli (Mathioudakis and Koudas, 2010) nový algoritmus *QueueBurst* s nasledujúcimi charakteristikami:

   (a) *jeden prechod* (angl. one-pass). Keďže ide o prúdové spracovanie, dáta môžu byť prečítané iba raz.

   (b) *spracovanie v reálnom čase*. Identifikácia nárazových kľúčových slov je vykonávané tak ako dáta vznikajú.

[1] https://twitter.com/

(c) *odolnosť voči falošným nárazovým kľúčovým slovám.* Niekedy sa stane, že kľúčové slovo začne nárazovo prúdiť, ale nemusí predstavovať prúd, môže sa vyskytnúť zhodou okolností.

(d) *odolnosť voči spam-u.* Existuje veľa automatických botov a používateľov, ktorí generujú spamujúce správy. Spam by mohol značne znížiť presnosť detekcie trendu.

2. *zoskupovanie nárazových kľúrových slov* po tom čo algoritmus *QueueBurst* identifikuje K_t kľúčových slov pre každý časový moment t, sú kľúčové slová $k \in K_t$ periodicky zoskupované do nesúvislých (angl. disjoint) podmnožín $K_t^i \in K_t$. Potom identifikovaný trend predstavuje podmnožina K_t^i. Zoskupovanie vykonáva algoritmus *GroupBurst*, ktorý posuďuje spoločný výskyt v posledných správach. Algoritmus je realizovaný lačnou stratégiou.

3. Posledným krokom je analýza identifikovaného trendu K_t^i. Prvým krokom je identifikovať ďalšie kľúčové slová, ktoré sa spájajú s trendom K_t^i. Toto je dosiahnuté algoritmami na extrakciu kontextu, ktoré sú spustené na nedávnej histórií správ. Algoritmus vráti kľúčové slová, ktoré najviac korelujú s identifikovaným trendom K_t^i. Navyše, trendy na sociálnej sieti často pozostávajú z komentárov na aktuálne správy a novinky vo svete (napr. nytimes.com). Preto má zmysel ďalej extrahovať aj príslušné hypertextové odkazy a prideliť ich k trendu K_t^i. Posledný krok tejto metódy je zobrazenie priebehu identifikovaného trendu pomocou vizualizácie pre používateľa.

3.6 Rozpoznanie pocitu a nálady z používateľom generovaného obsahu

Analýza pocitu alebo nálady (angl. sentiment analysis) može byť chápaný ako problém klasifikácie. Úlohoou je to klasifikovať správy (najčastejšie v kontexte sociálnych sietí) do dvoch kategórií na zákalde ich pozitívnych alebo negatívnych dojmov. Ak by sme pracovali s dátami zo sociálnej siete Twitter, je možné použiť na označkovanie správ. pomerne dobre, extrahovaním emotikonov, ktoré vyjadrujú pocity používateľa (Bifet and Frank, 2010).

Bifet and Frank publikovali v práci (Bifet and Frank, 2010) tri metódy a ich overenie na rozpoznanie nálady a pocitov z používateľom generovaného obsahu na sociálnej sieti Twitter. Experimentovali s tromi inkrementálnymi metódami, ktoré sú vhodné na spracovanie prúdu dát.

Multinomiálny Naive Bayes je klasifikátor najčastejšie používaný na klasifikáciu dokumentov, ktorý obvykle poskytuje dobré výsledky aj čo sa týka presnosti výsledku aj rýchlosti. Túto metódu je jednoduché aplikovať v kontexte prúdu dát (Bifet and Frank, 2010). Multinomiálny naivný Bayes sa pozerá na dokument ako na zhluk slov. Pre každú triedu c, $P(w|c)$, pravdepodobnosť, že slovo w patrí do tejto triedy je odhadovaná z trénovacích dát jednoducho vypočítaním relatívnej početnosti každého slova v trénovacej sade pre danú triedu. Klasifikátor potrebuje naviac nepodmienenú pravdepodobnosť $P(c)$. Za predpokladu, že n_{wd} je počet výskytov slova w v dokumente d, pravdepodobnosť triedy c z testovacieho dokumentu je nasledovaná:

$$P(c|d) = \frac{P(c)\prod_{w\in d} P(w|c)^{n_{wd}}}{P(d)}$$

Kde $P(d)$ je normalizačný faktor. Aby sme sa vyhli problému kedy sa trieda nevyskytuje v datasete ani jeden krát, je bežné použitie Laplacovej korekcie a nahradenie nulových početností jednotkou, resp. inicializovať početnosť každej triedy na 1 namiesto 0.

Stochastický gradientný zostup (angl. Stochastic Gradient Descent, SGD). Bifet and Frank v ich práci použili implementáciu tzv. vanilla stochastický gradientný zostup s pevnou rýchlosťou učenia, optimalizujúc stratu s L_2 penalizáciou. L_2 penalizácia je často používaná pri podporných vektorových strojoch (angl. support vektor machines). Lineárny stroj, ktorý je často aplikovaný na problémy klasifikácie dokumentov, optimalizujeme funkciu straty nasledovne:

$$\frac{\lambda}{2}\left\|w\right\|^2 + \sum[1-(yxw+b)]_+$$

kde w je váhovaný vektor, b je sklon, λ regulačný parameter a označenie triedy y je z intervalu $\{+1,-1\}$.

Hoeffdingov strom (angl. Hoeffding tree) je najznámejšia implementácia rozhodovacích stromov v použití prúdového spracovania. Hoeffdingov algoritmus implementuje stratégiu pred-prerezávania, ktorá je založená na Hoeffdingovom ohraničení. Toto umožňuje inkrementálne budovanie rozhodovacieho stromu. Uzol stromu je rozvinutý hneď ako obsahuje dostatočne silnú štatistickú informáciu. Existujú ďalšie sofistikované implementácie Hoeffdingových stromov, ktoré implementujú rýchlejšie a efektívnejšie algoritmy, napr. VFDT - Very Fast Decision Trees (Domingos and Hulten, 2000) alebo CVFDT - Concept adapting Very Fast Decision Trees (Hulten et al., 2001)

3.7 ETL proces a kvalita dát

3.8 Zhodnotenie

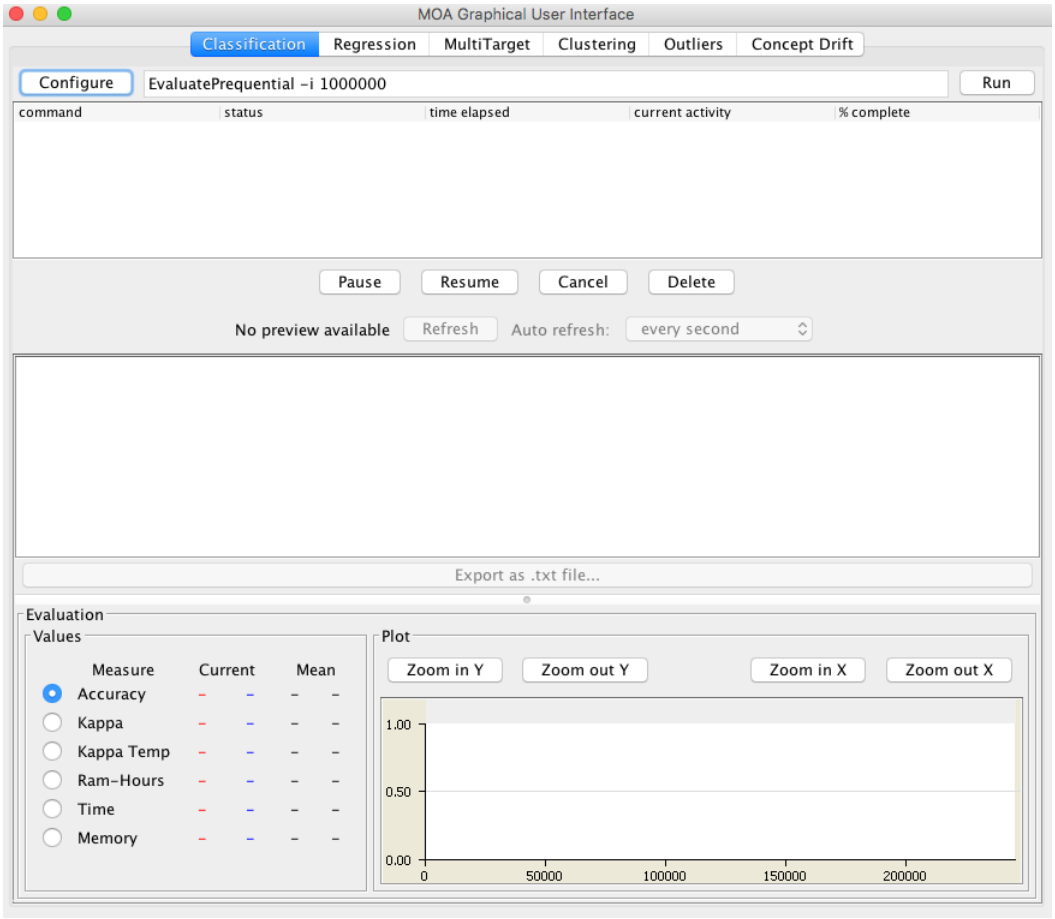
TODO: zhodnotenie analytických uloh, aké sú problémy atd.

4. Existujúce nástroje pre analýzu prúdu udalostí

4.1 MOA

Masívny online analyzátor (z angl. Massive Online Analysis - MOA, ďalej len MOA) je softvérové prostredie pre implementáciu algoritmov a vykonávanie experimentov pre online učenie sa z vyvýjajúcich sa prúdov dát (Bifet et al., 2010). MOA pozostáva z kolekcie offline a online metód a tiež nástrojov pre evaluáciu týchto metód. MOA implementuje metódy a algoritmy pre klasifikáciu, zhlukovanie prúdu, detekciu inštancií, ktoré sa vymykajú prahovým hodnotám a tiež odporúčacie systémy. Presnejšie MOA implementuje napríklad nasledujúce: stupňovanie (angl. boosting), vrecovanie (angl. bagging) a Hoeffdingove stromy, všetky metódy s a bez Naive Bayes klasifikátorom na listoch. MOA podporuje obojsmernú interakciu s nástrojom WEKA, ktorý je detailne opísaný v nasledujúcej kapitole.

MOA je implementovaná v programovacom jazyku Java. Za hlavný benefit implementácie v Jave považujú autori jej platformová nezávislosť. MOA obsahuje tiež generár prúdu dát, vie dobre modelovať concept drift. V aplikácií je možné definovať pravdepodobnosť, že inštancia prúdu patrí do nového concept drift-u. Sú dostupné nasledovné generátory prúdu (Bifet et al., 2010): *Random Tree Generator, SEA Concepts Generator, STAGGER Concepts Generator, Rotating Hyperplane, Random RBF Generator, LED Generator, Waveform Generator, and Function Generator.*



Obrázok 4.1: Hlavná obrazovka GUI nástroja MOA.

4.2 WEKA

The Wakaito Enviroment for Knowledge Analysis (ďalej len Weka) vznikol s jednoduchým cieľom poskytunúť výskumníkom unifikovanú platformu pre prístup k state-of-the-art technikám strojového učenia sa (Hall et al., 2009). Weka vznikla na University of Waikato na Novom Zélande v roku 1992, pričom je aktívne vyvýjaná posledných 16 rokov. Weka poskytuje kolekciu algoritmov strojového učenia sa pre úlohy dolovania v dátach. Algoritmy môžu byť priamo aplikované na datasety prostredníctvom aplikácie alebo použité vo vlastných aplikáciach volaním Java kódu. Weka obsahuje tieť nástroje na predspracovanie dát, klasifikáciu, regresiu, zhlukovanie, asociačné pravidlá a vizualizáciu. Nástroj je tieť vhodný pre návrhovanie a vývoj nových schém pre strojové učenia sa v kontexte dolovania dát. Zaujímavosťou je tiež, že

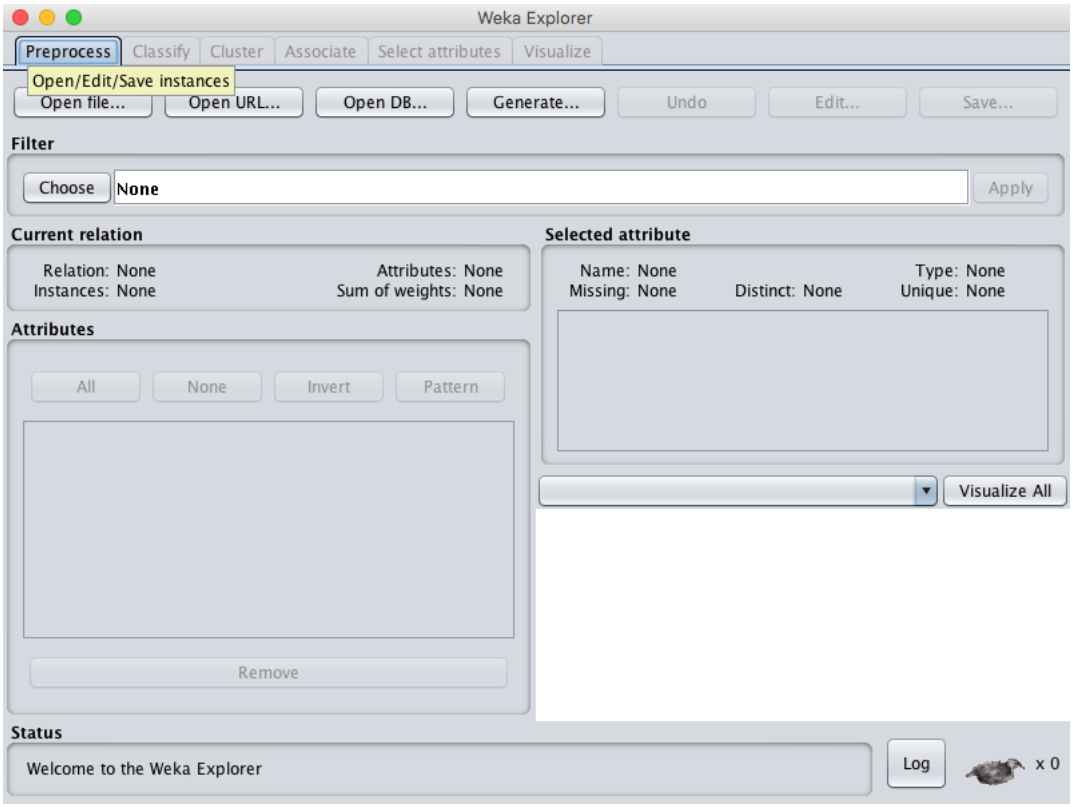
Weka je nelietajúci vták, ktorý žije len na ostrove Nového Zélandu.

Vieľom nástroja je poskytnúť pracovný nástroj pre výskumníkov. Poskytuje napríklad (nástroj ich obsahuje omnoho viac, vymenované sú len vybrané) tieto algoritmy určené pre klasifikáciu dát:

- *Bayesová logistická regresia* (angl. Bayesian logistic regression), pre kategorizáciu textu s Gausovským a Laplacovým apriori.
- *Najlepší prvý rozhodovací strom* (angl. Best-first decision tree), konštrukcia rozhodovacieho stromu so stratégiou najlepší prvý.
- *Hybridná rozhodovacia tabuľka a naivný Bayes* (angl. Decision table naive Bayes hybrid) hybridný klasifikátor, ktorý kombinuje rozhodovacie tabuľky a metódu Naivný Bayes.
- *Funkčné stromy* sú rozhodovacie stromy s lomeným rozdelením a lineárnymi funkciami v listoch.

Weka poskytuje tiež nástroje pre predspracovanie dát, zoznam niektorých filtrov (vymenované sú len vybrané základné filtre):

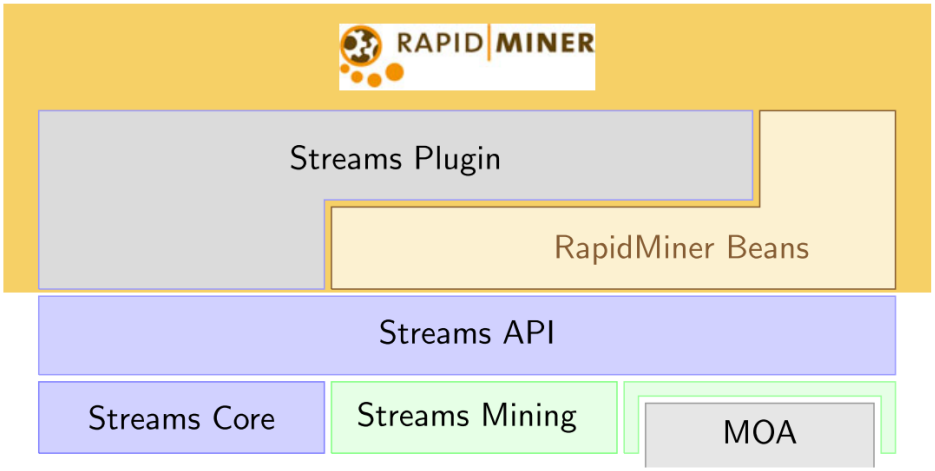
- *Pridanie klasifikátora*, pridá predikcie klasifikátora do datasetu.
- *Pridanie ID* ako nového atribútu pre každý záznam datasetu.
- *Pridanie hodnoty* chýbajúcim hodnotám z poskytnutého zoznamu.
- *Preskupenie atribútov* preusporiadanie poradia atribútov.
- *Numerické hodnoty na nominálne*, konverzia numerických hodnôt na nominálne.



Obrázok 4.2: Hlavná obrazovka GUI nástroja WEKA.

4.3 RapidMiner Streams-Plugin

Streams plugin poskytuje operátory RapidMiner-u pre základné budovanie blokov Streams API použitím obaľovača (angl. wrapper) na priame použitie implementácie, ktorú poskytuje Streams balík. Operátory Streams Plugin-u sú automaticky vytvorené pomocou procesora a použitím knižnice RapidMiner Beans (Bockermann and Blom, 2012). Architektúra Streams Plugin-u je postavená na Streams API, ktoré bolo navrhnuté v práci Bockermanna a Bloma.



Obrázok 4.3: Architektúra RapidMiner Stream Plugin-u a ďalších potrebných častí.

4.4 StreamBase

StreamBase[1] je platforma pre spracovanie udalostí, ktorá poskytuje vysokovýkonný softvér pre budovanie a nasadanie systémov, ktoré analyzujú a reagujú (napr. akciami) na prúdiace dáta v reálnom čase. StreamBase poskytuje prostredie pre svižný vývoj, server pre spracovanie udalostí s nízkou odozvou a vysokou priepustnosťou a zároveň integráciu do podnikových nástrojov, napríklad pre spracovanie historických údajov. Server analyzuje prúdiace dáta a poskytuje výsledky a odpovede v reálnom čase s extrémne nízkou odozvou. Toto je dosiahnuté maximalizáciou využitia hlavnej pamäte a ostatných prostriedkov servera, zatiaľ čo sa eliminujú závislosti na ostatné aplikácie. Integrované vývojové prostredie - StreamBase Studio umožňuje programátorom jednoducho a rýchlo vytvoriť, testovat a debugovať StreamSQL aplikácie použitím grafického modelu toku vykonávania. StreamBase aplikácie sú potom skompilované a nasadené za behu servera.

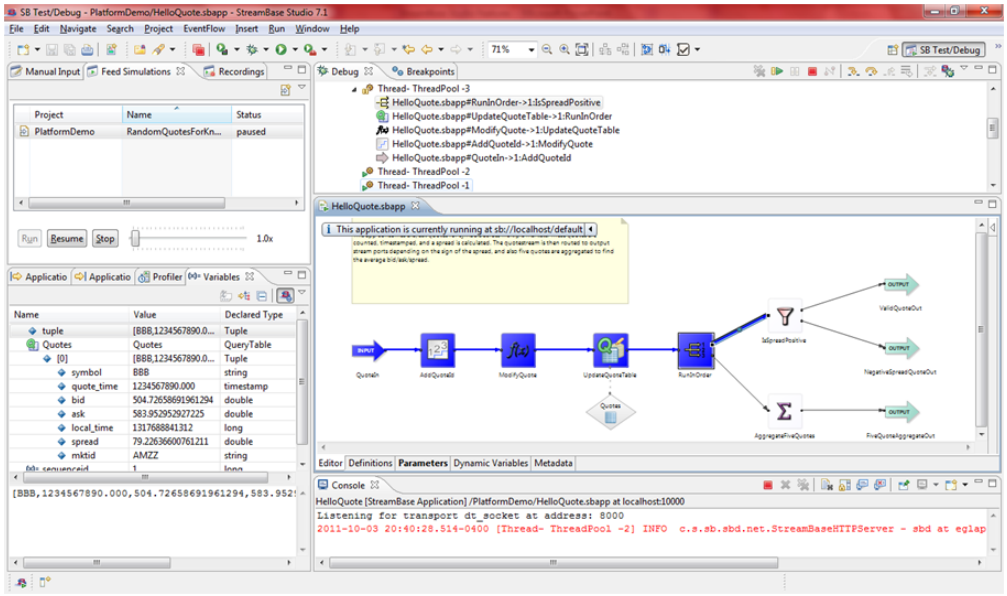
StreamSQL je dopytovací jazyk, ktorý rozširuje štandard SQL. StreamSQL umožňuje spracovanie prúdov v reálnom čase a dopytovanie sa do nich. Základná myšlienka jazyka SQL je možnosť dopytovať sa do uložených

[1] http://www.streambase.com/

statických kolekcií dát, StreamSQL umožňuje to isté, ale do prúdov dát. Teda, StreamSQL musí zvládnuť spracovať kontinuálny prúd udalostí a časovo orientované záznamy. StreamSQL zachováva schopnosti jazyka SQL zatiaľ čo pridáva nové možnosti ako napríklad: bohatý systém posuvných okien, možnosť miešania prúdiacich dát a statických dát a tiež možnosť pridať vlastnú logiku vo forme analytických funkcií.

StreamBase EventFlow je jazyk pre prúdove spracovanie vo forme tokov a operátorov ako grafických elementov. Používateľ má možnosť spájať tieto grafické elementy a vytvárať tak jednoducho topológiu pre prúdové spracovanie bez nutnosti programovania. EventFlow integruje všetky možnosti StreamSQL.

Použitie StreamBase sa výborne hodí pre štrukturované aplikácie "reálneho času", ktoré majú za cieľ rýchle spracovanie spolu s rýchlym prototypovaním a nasadením nových funkcionalít.



Obrázok 4.4: Vývojové prostredie nástroja StreamBase.

4.5 InforSphere Streams (IBM)

5. Metóda pre detekciu trendov v prúde udalostí

Navrhujeme metódu detekovanie trendov v prúde udalostí s polo-automatickým výberom vhodnej metódy. Našim cieľom je odbremeniť doménového experta od náročného výberu správnej metódy pre detekciu trendu v prúde. Mohli by sme sa zamerať na detekciu trendov v doménovo špecifickej oblasti, ale tento problém je z veľkej miery preskúmaný. Preto sa zameriavame na rôzne domény, či ide o senzorové syntetické dáta, dáta z energetickej alebo počítačovej siete, či používateľmi vytváraný obsah na sociálnych sietiach a mikro-blogových službách. Cieľom je teda: použiteľnosť a jednoduchosť nami navrhovanej metódy pre doménového experta bez nutnosti detailnej znalosti o fungovaní metódy a modelu, ktoré vytvára. Teda, kladieme nasledujúce hypotézy:

Hypotéza 5.0.1 *Naše riešenie detekuje trendy v prúde dát s istou pravdepodobnosťou použitím polo-automatického výberu vhodnej metódy pre detekciu trendov a zároveň poskytuje výsledky v reálnom čase*

Hypotéza 5.0.2 *Metóda je ľahká na použitie a interpretované výsledky sú jednoduché na pochopenie pre doménového experta bez detailnej znalosti o fungovaní modelu.*

Pri detekcií trendu bude nutné zohľadniť nasledujúce elementy:

- anomálie, chyby a spam,
- sezónnosť dát,
- concept drift a zmeny,

6. Zhodnotenie a budúca práca

Literatúra

Bifet, A. and Frank, E. (2010). Sentiment knowledge discovery in twitter streaming data. In *Discovery Science*, pages 1–15. Springer.

Bifet, A., Holmes, G., Kirkby, R., and Pfahringer, B. (2010). MOA: massive online analysis. *Journal of Machine Learning Research*, 11:1601–1604.

Bockermann, C. and Blom, H. (2012). Processing data streams with the rapidminer streams-plugin. In *Proceedings of the 3rd RapidMiner Community Meeting and Conference*.

Domingos, P. and Hulten, G. (2000). Mining high-speed data streams. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 71–80. ACM.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Hill, D. J. and Minsker, B. S. (2010). Anomaly detection in streaming environmental sensor data: A data-driven modeling approach. *Environmental Modelling & Software*, 25(9):1014–1022.

Hill, D. J., Minsker, B. S., and Amir, E. (2007). Real-time bayesian anomaly detection for environmental sensor data. In *Proceedings of the Congress-International Association for Hydraulic Research*, volume 32, page 503. Citeseer.

Hodge, V. J. and Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126.

Hulten, G., Spencer, L., and Domingos, P. (2001). Mining time-changing data streams. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 97–106. ACM.

Krempl, G., Žliobaite, I., Brzeziński, D., Hüllermeier, E., Last, M., Lemaire, V., Noack, T., Shaker, A., Sievi, S., Spiliopoulou, M., et al. (2014). Open challenges for data stream mining research. *ACM SIGKDD Explorations Newsletter*, 16(1):1–10.

Liu, X., Wu, X., Wang, H., Zhang, R., Bailey, J., and Ramamohanarao, K. (2010). Mining distribution change in stock order streams.

Mathioudakis, M. and Koudas, N. (2010). Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158. ACM.

Ross, G. J., Tasoulis, D. K., and Adams, N. M. (2009). Online annotation and prediction for regime switching data streams. In *Proceedings of the 2009 ACM symposium on applied computing*, pages 1501–1505. ACM.

Tran, D.-H., Gaber, M. M., and Sattler, K.-U. (2014). Change detection in streaming data in the era of big data: models and issues. *ACM SIGKDD Explorations Newsletter*, 16(1):30–38.

Príloha A