

Sem vložte zadání Vaší práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Bakalářská práce

Systém pro analýzu proudu dat v reálném čase

David Viktora

Vedoucí práce: Ing. Adam Šenk

10. dubna 2016

Poděkování

Poděkování

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 10. dubna 2016

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2016 David Viktora. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Viktora, David. *Systém pro analýzu proudu dat v reálném čase*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

Klíčová slova Nahradte seznamem klíčových slov v češtině oddělených čárkou.

Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

Keywords Nahradte seznamem klíčových slov v angličtině oddělených čárkou.

Obsah

Úvod	1
1	3
1.1 Streamové zpracování dat	3
1.2 Big data a technologie pro práci s nimi	3
1.3 Streamové zpracování velkých objemů dat	5
1.4 Apache Spark	6
2 Analýza	7
2.1 Metody pro analýzu textu	7
2.2 Požadavky na systém	7
2.3 Dostupné technologie	7
2.4 Twitter streaming API	7
3 Návrh	9
3.1 Celkový pohled na systém	9
3.2 Analýza tweetů	9
3.3 Struktura databáze	9
3.4 Návrh API	9
4 Implementace	11
4.1 Použité technologie	11
5 Testování	13
5.1 Test API endpointů	13
5.2 ???	13
6 Nasazení	15
7 Zhodnocení výsledků	17

Závěr	19
A Seznam použitých zkratek	21
B Obsah přiloženého CD	23

Seznam obrázků

Úvod

Kratce o jednotlivých bodech zadání a struktuře práce, motivace

????

V současné době generujeme obrovská množství dat - podle některých odhadů to například v roce 2012 mohlo být až 2,5 exabajtů za den[?]. Od té doby se rychlost přibývání dat stále zvyšuje. Například nárůst objemu dat dostupných na internetu je způsoben jeho neustále větším rozšířením a rostoucí dostupností. V roce 2016 je k němu připojeno již XXX obyvatel planety[?]. Roste také míra využívání internetu. Oproti dřívějšímu na internetu trávíme nejen díky chytrým telefonům stále více času a využíváme například sociální sítě, internetové vyhledávání a další online služby. Během toho jsou nám zobrazována personalizovaná data a cílené reklamní nabídky. Také ve firemní i státní sféře výrazně roste stupeň využívání informačních technologií a v návaznosti na to objem produkováných dat. S přibývajícími daty nastávají problémy s jejich zpracováním. Jedním z nich je jejich zpracování v dostatečně krátkém, ideálně reálném čase. Tím druhým problémem je pak právě zpracování obrovských objemů dat.

1.1 Streamové zpracování dat

....

1.2 Big data a technologie pro práci s nimi

Klasické technologie používané v minulosti se stále častěji ukazují jako nedostatečně rychlé a je proto nutné sáhnout po nových technologiích určených pro práci s velkými objemy dat. Touto problematikou obecně se zabývá relativně nový a často skloňovaný obor Big Data. Bez technologií pro Big Data se v dnešní době neobejdou především internetové vyhledávače nebo sociální sítě[?]. Veliké možnosti jejich využití se ale nabízejí i v dalších oblastech a jejich rozvoj je předpokladem pro rozšíření tzv. Internet of Things[?].

Obecně můžeme říct, že o Big Datech hovoříme v případech, ve kterých je potřeba zpracovávat objemy dat v řádech gigabajtů a více. To je ale velice zjednodušený popis tohoto termínu - přesná definice však neexistuje a na celý problém se dá dívat různými způsoby. Rozšířený je například také popis který říká, že o Big Datech mluvíme zkrátka v těch případech, kdy klasické databázové a softwarové nástroje kvůli objemu těchto dat selhávají[?].

Přestože jednotná definice neexistuje, ustálilo se několik charakteristik, která mají Big Data společná. Jedná se o takzvaná 3+1V - Volume, Velocity, Variety a později přidaná vlastnost Veracity[?]. Volume popisuje samotný objem zpracovávaných dat, Velocity pak rychlost, jakou data přibývají. Charakteristikou Variety popisujeme různorodost dat a Veracity určuje úplnost a důvěryhodnost dat. Tyto charakteristiky spíše popisuje klasické problémy, se kterými se s práci s Big Daty setkáme - data však nemusí splňovat všechny tyto vlastnosti zároveň, abychom je mohli za Big Data označit.

Technologie pro Big Data je samozřejmě možné použít jak na data strukturovaná, tak především na ta nestrukturovaná. Jelikož je těch nestrukturovaných až XXXX krát více než těch strukturovaných[?], a protože je jejich zpracování často výpočetně náročnější, jsou právě technologie pro Big Data vhodnou volbou. Klasickým příkladem nestrukturovaných dat je lidská řeč v psané formě - ta rozhodně obsahuje spoustu informací, ale ve formě kterou je počítačově složité analyzovat. Jako příklad takových dat můžeme uvést právě příspěvky na sociální síti Twitter, kterými se tato práce zabývá.

Analýzou těchto dat se zabývají obory jako Data Mining či Natural Language Processing. Pomocí nich je často možné získat opravdu cenné informace. Například analýzou dat pohybu uživatele po webové stránce a jeho chováním můžeme odhalit nedostatky tohoto webu a jejich odstraněním zvýšit míru konverze. Hovoříme-li v kontextu sociální sítě Twitter, možnosti jsou ještě zajímavější. Na základě příspěvků a vyplněných informací jednotlivých uživatelů můžeme například odhadovat jejich volební preference nebo nabízet velice přesně cílenou reklamu.

První konkrétní technologie pro práci s Big Daty vznikaly na přelomu tisíciletí ve společnosti Google. Právě Google v roce 2004 zveřejnil článek o modelu MapReduce[?], která se stala stavebním kamenem pro většinu dalších technologií pro práci s velkými objemy dat. MapReduce vlastně popisuje dvě nezávislé funkce. První z nich je funkce Map, ve které jsou ze vstupních dat vygenerovány dvojice klíč a hodnota. Poté co je funkce Map dokončena, její výstup je použit jako vstup do funkce Reduce. Ta pak spojí vstupní data podle klíče[?]. Klíčovou vlastností MapReduce modelu je možnost paralelizace Map fáze na počítačovém clusteru. Jeden z počítačů v clusteru například přijme požadavek od uživatele. Tento počítač náhodně rozdělí vstupní data všem počítačům v clusteru a vyčká na provedení Map fáze jednotlivými počítači. Výsledná data pak sám master spojí v Reduce fázi a navrátí výsledek uživateli. Obecně distribuce co největšího množství operací a výpočtů po počítačovém clusteru je hlavním principem fungování technologií pro zpracování velkých

objemů dat.

Konkrétních frameworků pro zpracování Big Dat je několik, tím klíčovým je ale bezesporu open-source framework Hadoop, jehož první plná verze vyšla na konci roku 2011, ale byl využíván již přibližně od roku 2006 například ve společnosti Yahoo[?]. Je určen pro použití na počítačových clusterech složeným z řádově desítek až stovek relativně levných serverů. Hadoop se skládá ze tří hlavních komponent - Hadoop Distributed File System, Hadoop MapReduce a Hadoop YARN. HDFS neboli Hadoop Distributed File System je distribuovaný filesystem zajišťující rozprostření dat po jednotlivých počítačích v clusteru. Klade důraz na toleranci výpadků částí clusteru. Pro zabránění ztráty dat v případě takového výpadku dochází mimo jiné k jejich replikaci na více strojů. Hadoop MapReduce je konkrétní implementace MapReduce modelu zmiňovaného dříve. Hadoop YARN pak slouží především k řízení zdrojů v clusteru, tedy například rozdělování práce jednotlivým serverům v clusteru. Nevýhodou Hadoopu je fakt, že nepodporuje proudové zpracování dat, ale pouze zpracování batchové.

1.3 Streamové zpracování velkých objemů dat

Technologií umožňujících streamové zpracování velkých objemů dat je několik, ty nejzajímavější jsou ale Apache Spark, Apache Storm a Apache Flink. Každá z těchto technologií funguje na trochu jiném principu a je vhodná pro jiné využití.

První ze zmiňovaných, Apache Spark, kterým se tato práce zabývá, je zaměřený především na batchové zpracování dat. Jedna z jeho komponent nabízí i možnost zpracování proudů dat, narozdíl od zbývajících dvou však nejde o plnohodnotné proudové zpracování, ale o takzvané micro-batchové zpracování. Spark nereaguje na každý nový datový objekt samostatně, ale nejdříve data po zadanou dobu střeďá a poté je víceméně klasicky batchově zpracuje[?]. Přestože Spark při zpracování proudů dat obvykle nedosahuje takových rychlostí jako zbylé dvě technologie, i s ním je možné dosáhnout zpracování dat v téměř reálném čase[?]. Spark je oproti zbylým dvěma projektům v pozdější fázi vývoje a je v rámci Apache Software Foundation nejaktivněji vyvíjeným projektem[?].

Apache Flink v relativně velké míře konkuruje Sparku. Podporuje oba způsoby zpracování dat, primárně je ale zaměřený na streamy. Jak již bylo řečeno, narozdíl od Sparku se jedná o streamové zpracování v pravém slova smyslu a je v něm tak možné dosahovat kratších dob zpracování dat. I právě proto se předpokládá, že v oblasti zpracování proudů dat v budoucnu předstihne Spark[?]. Jeho další výhodou je možnost využití existujících programů pro Storm či MapReduce.

Apache Storm jako jediný ze zmiňovaných frameworků nabízí pouze streamové zpracování. I on nabízí právě streamové zpracování. Jeho API je oproti

Flinku více nízkoúrovňové a vývoj v něm tak může být pracnější. Obecně ale funguje na podobném principu jako právě Flink. Oproti zbývajícím dvěma technologiím Storm postrádá například tzv. Streaming Windows a další funkce[?].

Obecně se tedy v kontextu streamového zpracování Spark hodí pro ty případy užití, kdy není nutné co nejrychlejší zpracování a řádově několikasekundové zpoždění nehraje roli. Je pak momentálně oproti zbylým technologiím vhodnou volbou díky svojí vyspělosti a jednoduchosti. Pokud je nutné opravdu real-time zpracování, je třeba volit mezi Flinkem a Stormem. Ty fungují na podobném principu, nicméně Flink má některé funkcionality, které ve Stormu chybí. Nabízí vysokoúrovňové API a do budoucna se jeví jako perspektivnější framework.

1.4 Apache Spark

Apache Spark je relativně novou technologií. Navazuje na již zmínovaný Hadoop a může používat některé jeho komponenty,

Analýza

2.1 Metody pro analýzu textu

Sentiment, ...

2.2 Požadavky na systém

Funkční a nefunkční (co konkrétně měří?, ...)

2.3 Dostupné technologie

- dostupné technologie

2.4 Twitter streaming API

Jak vypadá API, tweet, ze API nenabízí všechny příspěvky - gnip.com

Návrh

3.1 Celkový pohled na systém

jednotlive casti, propojeni, diagram

3.2 Analýza tweetů

jak bude vypadat program ve sparku

3.3 Struktura databáze

schema, ...

3.4 Návrh API

definice endpointů atd.

Implementace

4.1 Použité technologie

4.1.1 Analýza proudu dat

4.1.2 Databáze

4.1.3 API webserver

4.1.4 Webová aplikace

Testování

5.1 Test API endpointů

da se web castecne povazovat jako otestovani api?

5.2 ???

Nasazení

Zhodnocení výsledků

Závěr

Zaver

Seznam použitých zkratek

Item1 foo

Item2 bar

Obsah přiloženého CD

	readme.txt.....	stručný popis obsahu CD
	exe	adresář se spustitelnou formou implementace
	src	
	impl.....	zdrojové kódy implementace
	thesis	zdrojová forma práce ve formátu L ^A T _E X
	text	text práce
	thesis.pdf	text práce ve formátu PDF
	thesis.ps	text práce ve formátu PS