Rozšířený Abstrakt

Škálovatelný distribuovaný systém pro detekci podobných dokumentů

Ing. Ondřej Smola, Ing. Jindřich Žďánský, Ph.D.

Abstrakt

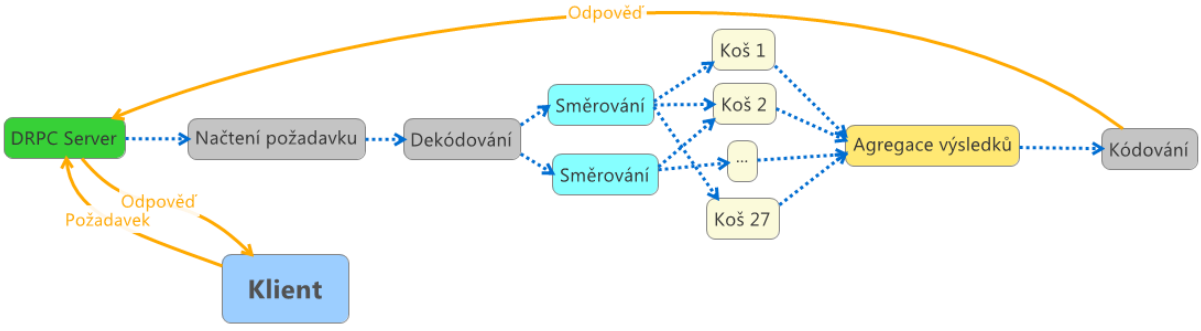
Práce se zabývá tvorbou škálovatelného distribuovaného systému pro online detekci podobných dokumentů. Systém je realizován s pomocí rámce Apache Storm a skládá se z topologie rámce Storm, DRPC serveru a databáze Redis. Výsledkem práce je systém škálovatelný dle plánovaného zatížení a množství porovnávaných dokumentů. Systém je i za použití pouze jednoho stroje s 4 jádry a 8GB paměti schopen zvládat desítky dotazů ze sekundu při historii několika milionů dokumentů. Navíc je vytvořený systém odolný vůči výpadků jednotlivých strojů a v případě kritické chyby zaručuje zachování historie dat dle konfigurovatelného intervalu zálohování.

Úvod

Cílem práce bylo vytvořit online systém pro detekci podobných dokumentů. Mezi hlavní požadavky na tento systém patřila jeho vysoká dostupnost v případě výpadku strojů clusteru, perzistentní historie a konfigurovatelná škálovatelnost podle očekávaného množství dokumentů v historii. Detekce podobnosti je důležitá například u dokumentů z mnoha různých zdrojů, u kterých je velká pravděpodobnost, že byl původní originální článek lehce upraven a použit v mnoha jiných zdrojích (regionální noviny, webové články). Takto získaná data mohou zkreslit následné výsledky zpracování textu. Ukázkovým příkladem může být například stavba jazykového modelu. Součástí tvorby jazykového modelu je zjištění počtu uspořádaných slovních n-tic (n-gramů) ze vstupního textu. Ty následně slouží pro výpočet pravděpodobnosti, že za sebou budou slova následovat. Podobné dokumenty mohou v tomto případě způsobit zkreslení modelu. Minimální požadavky na systém byly stanoveny na minimální zpracování 15 000 dokumentů denně při zachování týdenní historie.

Experiment a metody

Systém je realizován nad plánovačem Apache Mesos na clusteru NanoGrid, který nabízí volné zdroje rámci Apache Storm. Aplikace je realizována jako Storm topologie komunikující s DRPC serverem. Topologie Storm periodicky zálohuje data do databáze Redis. Ta běží díky rámci Marathon v módu vysoké dostupnosti a její transakční log je ukládán na distribuovaný souborový systém GlusterFS. Schéma topologie je znázorněno na obrázku 1.



Obrázek 1: Schéma Storm topologie pro detekci podobných dokumentů

Nejprve klient kontaktuje DRPC server s otiskem dokumentu a čeká na výsledek. Storm topologie odebírá požadavky klientů z DRPC serveru. Otisk dokumentu je reprezentován 256 bity a je vytvořen spočtením MinHash funkce z tokenů textu. Tokeny textu vzniknou převodem textu na ASCII znaky, malá písmena a rozdělením po mezerách. Otisky jsou v topologii rozděleny do 27 košů podle bitové kardinality otisku při normálním rozložení. Příchozí otisk je podle požadované minimální

podobnosti směrován jen do košů, kde se teoreticky mohou nacházet podobné otisky a uložen pouze do jednoho koše podle kardinality. Podobnost dokumentů je určena podle Jaccardovy podobnosti otisků. Z každého koše je vybrán otisk s největší podobností (pokud splňuje alespoň minimální požadovanou podobnost) a pokud je alespoň z jednoho koše vrácen neprázdný výsledek, je nalezen podobný otisk a výsledek navrácen klientovi. Klient poté může z databáze získat text podobného dokumentu.

Výsledky a diskuze

Systém byl otestován na přibližně 800MB dat (deníky, webové příspěvky). Při nastavení minimální podobnosti na 94%, bylo výsledkem přibližně 140MB textu obsahujícího velmi podobné dokumenty. Jelikož je srovnávání podobnosti velice subjektivní věc, není možné jednoduše vyhodnotit celkovou úspěšnost. I přesto podle manuální kontroly systém velice dobře detekuje běžné úpravy používané při kopírování: vynechání vět, změna pořadí slov, pravopisné chyby, vložené věty nebo i kusy vloženého HTML kódu. Odolnost systému byla otestována náhodným vypnutím strojů a systém se vždy dokázal obnovit do stavu plné funkcionality s pouze krátkodobým přerušením činnosti. Problematické je pouze zálohování historie, které je nastaveno na jednou za minutu. Z toho vyplývá, že v případě výpadku systém může ztratit až poslední minutu historie. Periodu zálohování je možné konfigurovat a interval určuje poměr mezi konzistencí a výkonem řešení. Propustnost řešení při zálohování nastaveném na 1 minutu je zobrazena v tabulce 1. Pokud bychom uvažovali původní zadání 15 000 nových dokumentů denně při týdenní historii (105 000 dokumentů), je systém schopný garantovat zpracování denní dávky za méně než 1 minutu. I při historii 5 let by systém zpracoval denní dávku do 15 minut.

Velikost historie	Požadavků/s
270 000	200
2 700 000	50
27 000 000	20

Tabulka 1: Propustnost realizovaného systému (alokováno 4 CPU, 8GB RAM)

Závěr

Vytvořený systém umožňuje online detekovat podobné dokumenty a jeho propustnost přesahuje plánované zatížení. Řešení je odolné vůči výpadku strojů a má plně konfigurovatelnou škálovatelnost, jednak podle množství přidělených zdrojů a také podle délky ukládané historie a intervalu zálohování. Systém byl otestován na 800MB testovacích dat a zjistil, že obsahují přibližně 140MB textu, který si je velmi podobný nebo duplikát.

Reference

[1] ANDERSON, Quinton. Storm real-time processing cookbook. Birmingham, England: Packt Publishing, 2013. ISBN 978-1-78216-443-2.
[2] SHRIVASTAVA, A. & Li, P. (2014), In Defense of MinHash Over SimHash, in 'Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)' .
[3] CARLSON, Josiah L. Redis in action. Shelter Island, NY: Manning, 2013. ISBN 9781617290855.