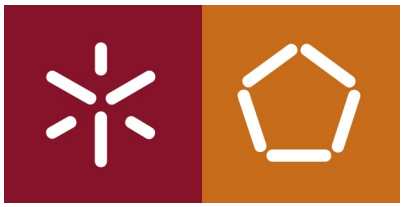


Universidade do Minho

MIEI - 4º Ano

Gestão de Grandes Conjuntos de Dados

Twitter
Serviço de Análise em Tempo Real

Trabalho Prático

Proposta de projeto

Luís Ferreira (A78607) Fábio Fontes (A78650) Ângela Amorim (PG38407)

Março 2019

Conteúdo

1 Objetivos

1.1 Descrição do Problema

> *Twitter data is the most comprehensive source of live, public conversation worldwide. Our REST, streaming, and Enterprise APIs enable programmatic analysis of data in real-time or back to the first Tweet in 2006. Get insight into audiences, market movements, emerging trends, key topics, breaking news, and much more.*

\- developer.twitter.com

Tal como referido no site do *twitter*, é possível analisar *tweets* que tenham sido publicados de forma a extrair informações em tempo real sobre movimentos nos mercados, tendências emergentes, novos movimentos ativistas, entre outros.
Este tipo de análise é bastante útil, e até possivelmente crítica, para negócios que queiram tomar decisões mais acertadas em relação a certo produto/serviço, pois conseguirão obter uma imagem mais nítida da futura aceitação (ou falta dela) por parte do seu público alvo.
Portanto, com a API do *twitter* é possível fazer uma análise de sentimento em relação a *tweets* sobre um dado tema e ver qual a opinião geral do seu público alvo. É possível também fazer um *tracking* das menções de dada empresa e/ou produto para determinar a popularidade de algo em tempo real. Ou seja, utilizando a API do *twitter* é possível auxiliar na tomada de decisão de vários problemas que surjam no contexto de uma empresa. É também possível, tal como se viu recentemente com o caso do facebook, utilizar dados de redes sociais (como é o caso do twitter) para avaliar e até manipular o rumo de eleições e perspetivas políticas.
As possibilidades de perguntas e respetivas extrospecções são imensas. Neste trabalho iremo-nos focar nas questões descriminadas na subsecção seguinte.

1.2 Requisitos do sistema

As principais questões a serem respondidas com a implementação do nosso sistema são:

- Análise sentimental:
  - Verificar se um termo, frase ou hashtag específicos têm conotação positiva ou negativa naquele momento;
  - Dentro de um tópico, ver quais são os termos que surgem mais frequentemente.
- Hashtags frequentemente mencionadas, filtradas por:

  País;

  Distrito;

  Cidade.
- Termos frequentemente mencionados, filtrados por:

  Global;

  Distrito;

  Cidade.
- Lugares populares, ou seja, lugares onde são publicados mais *tweets*.
- Análise de imagens: a cada momento queremos saber qual o tipo de imagens que estão a ser publicadas com mais frequência (e.g. imagens de cães,gatos,paisagens,...);

- *Não definitivos*:

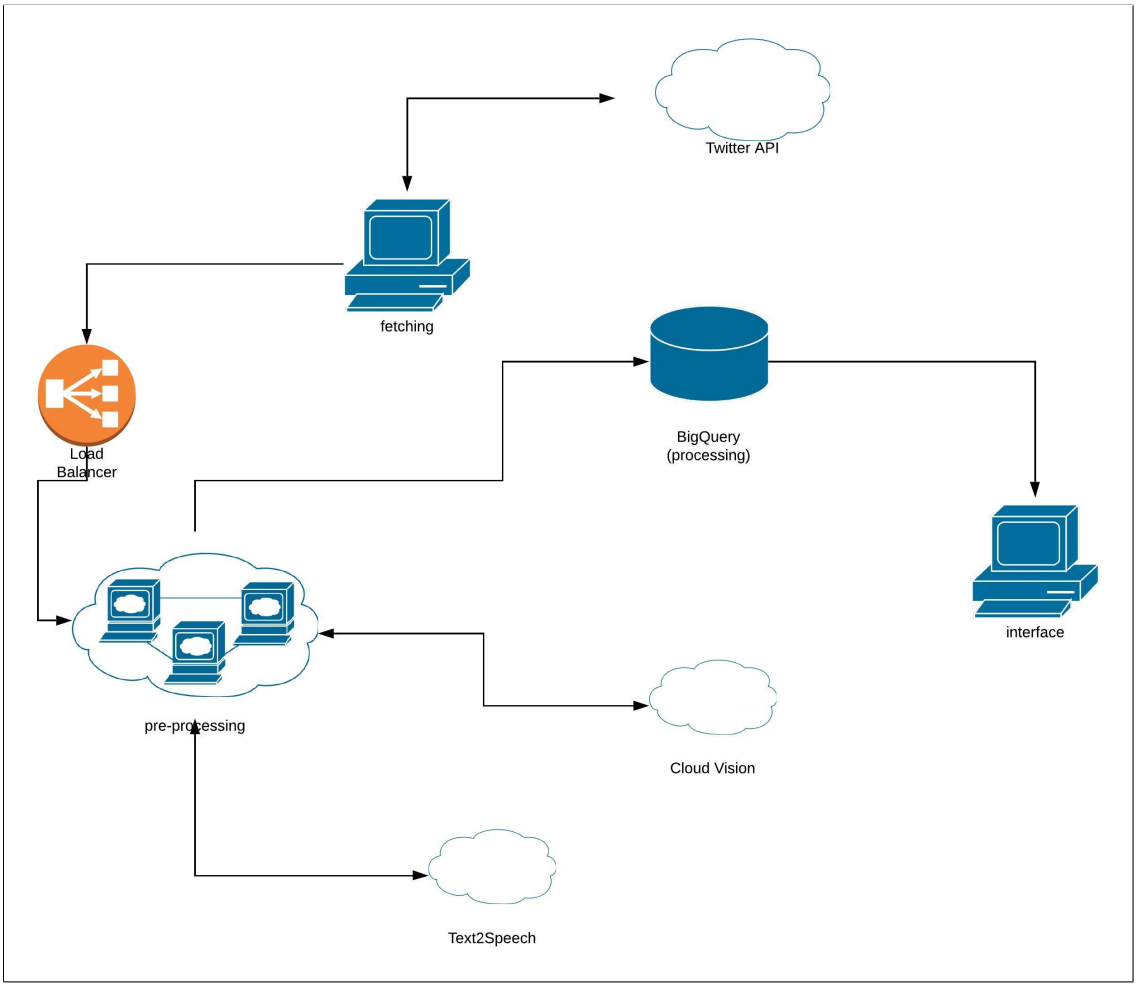
  Análise de voz (palavras que estão a surgir bastante neste momento);

  Análise de padrões de cores mais comuns nas imagens publicadas nos *tweets*, a partir dos quais se pode também derivar uma análise de sentimento; Indicação do desvio temporal entre o momento em que o último *tweet* processado pelo sistema foi publicado e o momento em que o resultado é mostrado ao utilizador (atraso em relação ao tempo real).

Os itens definidos como não definitivos poderão ou não ser implementados, dependendo dos serviços disponibilizados pela plataforma *google cloud*, saldo disponível e tempo de implementação dos requisitos anteriores. Poderão também ser adicionadas funcionalidades adicionais ao sistema não mencionadas acima, se houver tempo para as implementar e o grupo considerar que são relevantes para o sistema a ser desenvolvido.

2 Arquitetura

Na figura abaixo podemos ver a arquitetura proposta para podermos solucionar o problema previamente descrito.



As imagens com computadores azuis representam instâncias de máquinas virtuais (daqui adiante designadas por VM's) na plataforma *google cloud*.
Os dados serão obtidos a partir da API do twitter, por intermédio da VM designada por **fetching**. De seguida, estes serão passados a um serviço de balanceamento de carga (**load**

balancer) que os irá distribuir por diferentes VM's responsáveis por efetuar o pré-processamento dos dados. Nesta fase, pretendemos manipular os tweets capturados, eliminando informação supérflua, de forma a minimizar a quantidade de dados a ser armazenada. Para além disso, pretende-se também recorrer aos serviços **Cloud Vision** e **Text2Speech**, oferecidos pela plataforma, para efetuar a análise de imagens e áudio.
A informação resultante do pré-processamento será depois armazenada num sistema de armazenamento de dados de grande escala. Como primeira opção escolhemos o *data warehouse* oferecido pela *google cloud*, **BigQuery**. No entanto, se durante a implementação do projeto for encontrada uma solução melhor, essa poderá vir a substituir a atual. Uma vez que a quantidade de dados estará sempre a aumentar, num contexto realista, seria necessário eliminar informação que estivesse no sistema de armazenamento mais do que uma quantidade pré-definida de tempo.
Finalmente, a VM **interface** irá disponibilizar uma interface web que irá efetuar *queries* ao serviço *BigQuery* para responder aos problemas apresentados na secção de requisitos.
O objetivo desta arquitetura é manter uma separação de componentes com diferentes funcionalidades de forma a manter a solução organizada e escalável.

3 Escalabilidade

A arquitetura foi pensada com o intuito de ser escalável. Uma vez que o volume de dados gerados pelo Twitter é bastante elevado (12 GB/s apenas em texto) esta característica é especialmente relevante.
Assim, é possível dividir os *tweets* publicados por regiões geográficas (com maior ou menor granularidade) e atribuir cada uma destas regiões a um conjunto *fetching+load balancer+pre-processing* diferente. Poderão também ser criados datasets separados no serviço *BigQuery* para cada região.
Finalmente, dentro de cada região será ainda possível ajustar o número de VM's de pré-processamento de acordo com a carga específica da região.

4 Dificuldades esperadas

No contexto de elaboração deste projeto, obviamente não será possível processar todos os *tweets* que são publicados em tempo real, devido à sua grande dimensão, a possíveis restrições impostas pelas APIs do *Twitter* e *Google Cloud*, etc. Como alternativa, iremos processar apenas *tweets* que são publicados em Portugal. Se esta medida for insuficiente, para além da restrição geográfica, iremos limitar-nos ao número máximo de *tweets* que a API permita obter ou que o nosso sistema consiga processar.
Em ambos os casos, o intervalo de tempo em que irá ocorrer a fase de recolha (*fetching*) provavelmente terá que ser curto.

5 Avaliação do sistema

Para avaliar a performance do sistema, iremos focar-nos em tempos de resposta e *throughput*.
Relativamente aos tempos de resposta, queremos que quando é dada uma resposta a um utilizador, o desfasamento entre o momento de publicação do twitter mais recentemente processado e o tempo real seja mínimo. Queremos também que o tempo de consulta ao serviço *BigQuery* seja o mínimo possível, para evitar que o cliente tenha que esperar muito tempo para obter um resultado. No caso da medida de desvio em relação ao tempo real, esta medida poderá não ser relevante no contexto da nossa solução, devido às limitações esperadas.
Quanto ao *throughput*, poderemos medir quantos *tweets* o sistema consegue "pré-processar"por segundo.