Extraction de tendances avec *MapReduce*

Gary Benattar Yoann Couillec

24 novembre 2011

Résumé

Dans ce document, nous détaillons un procédé d'extraction de tendances dans des documents et plus précisément dans des opinions d'utilisateurs en utilisant des outils statistiques simples et des outils technologiques permettant le *passage à l'échelle* afin de pouvoir faire du calcul distribué ainsi que de la persistance de données sur un nombre toujours croissant de documents à analyser et à stocker.

1 Environnement

Nous avons lancé les *jobs* sur le *cloud* de Paris 6. Nous avons eu l'autorisation de lancer nos *jobs* sur un maximum de huit machines. Chacune de ces machines est équipée d'un processeur Intel(R) Core(TM) i7 2.93GHz 8 coeurs. Les machines sont reliées en réseau et l'espace disque est organisé avec NFS, ce qui nous permet de configurer à un seul endroit les paramètres de *Hadoop* et *HBase*.

2 Données

Les données en entrée sont :

- Un fichier CSV, contenant des opinions, récupérées sur Toluna.com (20 millions d'opinions en langue anglaise - environ 150 Go sur une période d'activité de 2 ans)
- Un corpus de données extrait par un script Python sur le forum français du jeu Trackmania (2 Go en langue française sur une période d'activité de 1 mois)

3 Outils statistiques

3.1 Statique

Nous avons besoin de calculer la moyenne des fréquences d'apparitions des mots. Mais également l'écart-type afin de déduire le *Z-Score*.

Moyenne : $\bar{x_n} = \frac{\Sigma_{i=1}^{n} x^i}{n}$
Écart-type : $\sigma = \sqrt{(\frac{1}{N}) \sum P(w_t) - \bar{x(w)}}$
Z-Score : $Z(w_t) = \frac{x_i - \bar{x}}{\sigma}$

3.2 Dynamique

Afin d'affiner les calculs et pour ne pas tous recalculer après l'ajout d'une valeur, nous avons décidé d'implémenter des outils statistiques incrémentaux, décrémentaux et par fenêtre. Tels que la *moyenne* et l'*écart-type*. Nous nous sommes inspirés du travail de Tony Finch [1], qui présente la *moyenne* incrémentale et l'*écart-type* incrémental, afin d'y ajouter les outils décrémentaux et par fenêtre.

moyenne incrémentale : $\overline{x_{inc,n+1}} = \overline{x_n} + \frac{x - \overline{x_n}}{n+1}$
moyenne décrémentale : $\bar{x_{dec,n-1}} = \frac{n \times \bar{x_n} - x}{n-1}$
moyenne fenêtre : $\bar{x_{win,n}} = \bar{x_{dec,n}}(\bar{x_{inc,n+1}}(\bar{x_n}, n, x_2), n+1, x_1)$
S-Term : $S_n = n\sigma_n^2$
S-Term incrémental : $S_{inc,n+1} = S_n + (x - \bar{x_n} \times (x - \bar{x_{inc,n+1}}(\bar{x}, n, x)))$
S-Term décrémental : $S_{dec,n-1} = S_{n+1} - (x - \bar{x_n} \times (x - \bar{x_{dec,n-1}}(\bar{x}, n, x)))$
Écart-type incrémental : $\sigma_{inc,n+1} = \sqrt{\frac{S_{inc,n+1}(S_n, \bar{x_n}, n, x)}{n+1}}$
Écart-type décrémental : $\sigma_{dec,n-1} = \sqrt{\frac{S_{inc,n-1}(S_n, \bar{x_n}, n, x)}{n-1}}$
Écart-type fenêtre : $\sigma_{win,n} = \sigma_{dec,n}(\sigma_{inc,n+1}(\sigma_n, \bar{x_n}, n, x_2), \bar{x_{inc,n+1}}(\bar{x_n}, n, x_2), n+1, x_1)$

4 Jobs

4.1 Extraction des mots communs

Prérequis :

1. lancement de l'*exporter* afin de polluer la table contenant nos documents *(date, id :value, opinions)* issus de notre fichier **.csv* de la forme *(Id, Timestamp, document)*.
2. en sortie de notre *job MapReduce*, une table dont chaque ligne est un couple *(date, word : w, timestamp, count)* ou *timestamp* $\in [0 - 86400[$ secondes.
3. création d'une seconde table qui va nous permettre de calculer le nombre de mots apparus chaque jour afin de pouvoir calculer $P(w_t)$

Extraction :

1. lancement d'un *job MapReduce* afin d'extraire les statistiques (*moyenne* et *écart type*) liés à chaque mot
2. détection des mots communs à l'aide d'un palier sur la moyenne (0.0014) et sur l'écart type (0.3) obtenu par *training* sur différents jeux de données

4.2 Détection des trends

1. Prérequis : lancement d'un *job MapReduce* afin de créer une table contenant 24 colonnes dont une ligne correspond au nombre d'occurrences d'un mot au jour donné et à une heure donnée.
2. Extraction : lancement d'un *job* pour le calcul de ($\bar{x}$,σ,z,x,n,p) pour un mot donné

4.3 Détection des collocations

1. Prérequis : Aucun
2. Extraction : pour une date d, une tendance t et un intervalle de timestamp $[secondeMin, secondeMax[$, sélection des documents apparus dans cet interval et extraction des collocations[2] contenant t
3. Élection : choix de la collocation la plus présente contenant t

4.4 Chainage des jobs

Prérequis : Import du fichier **.csv*

1. Comptage des fréquences des mots par jour
2. Comptage du nombre de documents par jour
3. Calcul de $\bar{x}$ et σ pour w à la date d
4. Extraction des mots communs
5. Comptage des fréquences des mots par heure
6. Extraction de ($\bar{x}$,σ,z,x,n,p) pour un mot donné
7. Extraction des tendances
8. Extraction des collocations pour une tendance t à une date d et une heure h
9. Election des tendances

4.5 Perspective

Nous souhaitons générer une interface graphique contenant des séries temporelles associées à chacun des mots ou l'utilisateur pourra observer les variations du mot. Nous pourrons de plus fournir un outil ou l'utilisateur pourrat analyser et extraire des tendances sur son propre fichier de données.

Références

[1] T. Finch *Incremental calculation of weighted mean and variance* University of Cambridge Computing Service (2009)

[2] Frank Smadja, *Retrieving Collocational Knowledge from Textual Corpora* PhD thesis, Computer Science Department, Columbia University, New York, NY, 1991.