Acquisition, modélisation et mise en œuvre d'un entrepôt de données pour l'analyse d'informations issues de Twitter

Flavien Bouillot,

CONSERVATOIRE NATIONAL DES ARTS ET METIERS

CENTRE REGIONAL DE LANGUEDOC-ROUSSILLON

MEMOIRE
présenté en vue d'obtenir
le diplôme d'INGÉNIEUR CNAM
Spécialité : INFORMATIQUE
Option : SYSTEME D'INFORMATION

par
Flavien BOUILLOT

Acquisition, modélisation et mise en œuvre d'un entrepôt de données pour l'analyse d'informations issues de Twitter.

Soutenu le 5 juillet 2011

Jury
Président : M. Yves LALOUM (Responsable national CNAM)
Membres : M. Michel SALA (Responsable régional CNAM)
M. Pascal PONCELET (Tuteur pédagogique LIRMM)
M. Mathieu ROCHE (Tuteur pédagogique LIRMM)
M^{me}. Maguelonne TEISSEIRE (CEMAGREF)
M^{me}. Sandra BRINGAY (LIRMM)

Acquisition, modélisation et mise en œuvre d'un entrepôt de données pour l'analyse d'informations issues de Twitter. Mémoire d'ingénieur C.N.A.M., Montpellier 2011.

Résumé

Le succès des réseaux sociaux en général et de Twitter en particulier n'est plus à démontrer. Les tweets échangés sur Internet constituent une source d'information importante même si leurs caractéristiques les rendent difficiles à analyser (texte court de type SMS, méta-données disponibles, éléments de langage spécifiques comme les @ ou les #).

Dans ce mémoire nous proposons une solution pour pouvoir intégrer toutes les données qui peuvent être extraites des tweets dans un entrepôt de données. Cet entrepôt servira de support à une activité d'informatique décisionnelle sur les tweets.

Outre une agrégation classique sur le nombre de tweets, nous implémentons de nouvelles mesures pour analyser le contenu des messages permettant une analyse multidimensionnelle de tweets en lien avec un domaine d'application spécifique.

Mots-clefs:

Twitter, Tweet, informatique décisionnelle, analyse multidimensionnelle, TF-IDF, TF-IDF adaptatif

Summary

The success of social networks in general, and Twitter in particular needs no introduction. Tweets exchanged over the Internet represent an important source of information even if their characteristics make them dicult to analyze (short message with SMS's style, meta-information available, specific's part-of-speech such as @ or # etc.).

In this paper, we propose a solution in order to integrate all data that can be extracted from the tweets in a data warehouse that will support "business intelligence" activity on the tweets.

Besides a classical aggregation on the number of tweets, we implement new measures to analyze the message content to a multidimensional analysis of tweets related to a specific application domain.

Keywords

Twitter, tweet, social network,business intelligence, multidimensional analysis, TF-IDF, adaptative TF-IDF

Remerciements

Tout d'abord, je tiens à remercier Pascal Poncelet et Mathieu Roche pour votre accueil et la confiance qu'ils m'ont témoigné en me laissant une grande liberté durant la réalisation de ce travail. Merci aussi pour leur aide dans l'organisation et la rédaction de mon mémoire.

Ensuite je souhaite remercier toute l'équipe TATOO en général, pour son accueil et sa disponibilité.

Je souhaite également remercier Maguelonne Teisseire pour ses conseils lors de la rédaction de ce mémoire.

Merci au FONGECIF Languedoc-Roussillon d'avoir sélectionné mon dossier, rendant possible la réalisation de ce projet et la finalisation de mon cursus d'ingénieur commencé il y a 7 ans maintenant.

Enfin, je remercie Isabelle Gely (CNAM Languedoc-Roussillon) et Éléonore Gondeau (CNAM Rhône-Alpes) pour m'avoir accompagner lors de la mise en place de ce projet. Un grand merci pour leur patience et la rapidité de leurs réponses.

Table des matières

1 Synopsis....6
2 Chapitre 2 : Environnement et organisation du projet....7
2.1 LIRMM – Equipes TATOO et TAL....7
2.2 Vers un outil d'analyse de données issues d'un réseau social....9
2.3 Processus de développement....10
3 Chapitre 3 : État de l'art et définitions préliminaires....15
3.1 Les entrepôts de données....15
3.1.1 Architecture générale....15
3.1.2 Modélisation et conception d'un entrepôt de données....20
3.1.3 Navigation au sein d'un entrepôt....24
3.1.4 Intégration de sources de données....28
3.2 Twitter: un réseau social....30
3.3 Panorama des outils d'analyse de données "tweets"....37
3.4 Bilan des solutions vis à vis du projet....44
4 Chapitre 4 : Conception d'un entrepôt de données dédié aux tweets....46
4.1 Données "tweet"....46
4.2 Contexte : le domaine médical....49
4.3 Verrous liés aux données....51
4.3.1 Gestion des données textuelles....51
4.3.2 Gestion de la localisation....52
4.3.3 Uniformisation du MeSH....52
4.3.4 Gestion de la désambiguïsation....53
4.4 Modèle conceptuel....53
4.4.1 Solution générique....53
4.4.2 Solution adaptée au domaine d'application....54
5 Chapitre 5 : Architecture technique et mise en œuvre....58
5.1 Présentation des choix techniques....58
5.2 Phase 1: Acquisition des tweets....60
5.3 Phase 2: Gestion des contraintes....65
5.3.1 Normalisation du texte du tweet....66
5.3.1.1 Nettoyage du tweet....66
5.3.1.1.1 Détermination de la langue....68
5.3.1.2 Analyse morpho-syntaxique....69
5.3.2 Normalisation de la localisation....73
5.3.3 Gestion de la désambiguïsation....80
5.3.3.1 Motivations....80
5.3.3.2 Mesure de fouille du web....81
5.3.3.3 Mesure de Recherche d'Information....82
5.4 Phase 3: Alimentation du cube....92
6 Chapitre 6 : Restitutions et visualisation des analyses....97
6.1 Condition d'utilisation et définition du périmètre d'analyse....97
6.2 Comparaison des tweets en lien avec notre domaine d'application....98
6.2.1 Liens....98
6.2.2 Retweet....99
6.2.3 Destinataires @....99
6.2.4 Tags #....100
6.2.5 Composante géographique d'un tweet....101

6.2.6 Répartition de la langue....104
6.2.7 Analyse grammaticale....104
6.3 Analyse multidimensionnelle....106
6.4 Exemple de possibilité d'analyse en ligne des résultats....107
7 Chapitre 7 : Conclusions et perspectives....111
7.1 Bilan critique....111
7.1.1 Critiques globales du système....111
7.1.2 Limites constatées liées aux verrous....112
7.2 Perspectives d'amélioration....114
7.2.1 A court terme et à moyen terme....114
7.2.1.1 Gestion de l'historisation....114
7.2.1.2 La grammaire du tweet....117
7.2.2 A long terme....117
7.2.2.1 Amélioration de la désambiguïsation sur les textes courts....117
7.2.2.2 La navigation des sentiments....118
7.3 Bilan personnel....120
8 Bibliographie....121

1 Synopsis

Le succès des réseaux sociaux ne fait plus aucun doute et leurs taux d'activité ont atteint des niveaux sans précédent. De part sa facilité d'utilisation, Twitter permet la publication d'un grand nombre d'informations. La disponibilité des données, qui par défaut sont publiques et accessibles à tous, en font de véritables vecteurs d'information qui peuvent offrir aux décideurs de nouvelles connaissances utiles. Ainsi l'objectif de ce mémoire est de présenter une solution pour pouvoir intégrer toutes les données qui peuvent être extraites des tweets dans un entrepôt de données qui servira de support à des outils d'analyse ou de reporting.

Pour cela nous présentons dans le chapitre 2 le projet et son environnement. Nous présentons le laboratoire et les équipes du d'accueil (2.1), puis nous décrivons les besoins utilisateurs (2.2) avant de présenter le processus de développement retenu (2.3).

Nous présentons dans le chapitre 3 les définitions et les notions relatives à la conception de notre solution, la notion d'entrepôt de données (3.1) et la présentation du réseau social de Twitter (3.2). Nous effectuons alors un état de l'art des travaux existants autour de l'analyse de données associée à Twitter (3.3), avant de conclure sur l'adéquation des travaux existants avec nos besoins (3.4).

Dans le chapitre 4, nous détaillons la conception d'un entrepôt de données dédié à l'analyse des tweets, d'abord en présentant les données "tweets" (4.1) puis les données relatives au domaine d'application retenu (4.2). Nous présentons ensuite les verrous auxquels nous sommes confrontés (4.3). Pour conclure ce chapitre, nous présentons le modèle conceptuel de notre solution (4.4).

Dans le chapitre 5, nous présentons l'architecture de notre solution. Tout d'abord nous abordons les choix techniques de notre application (5.1), puis nous présentons les différentes étapes allant de l'extraction des tweets (5.2) à l'alimentation du cube (5.4) en passant par les traitements de normalisation et de transformation mis en place (5.3).

Dans le chapitre 6, nous livrons une ensemble d'analyses et de restitutions. Dans un premier temps nous définissons le périmètre de ces expérimentations (6.1), puis nous comparons un ensemble de statistiques entre les tweets en général et les tweets liés au domaine d'application du projet (6.2). Nous présentons ensuite deux exemples d'analyses via notre solution, en premier lieu une analyse multidimensionnelle (6.3) puis un exemple de navigation au sein du cube (6.4).

Dans le chapitre 7, nous effectuons un bilan critiques de notre solution (7.1) avant de proposer des pistes d'améliorations (7.2). Nous concluons ce chapitre et ce mémoire par un bilan personnel sur les apports de ce travail (7.3).

2 Chapitre 2 : Environnement et organisation du projet

Ce mémoire a été réalisé dans le cadre d'un accord avec le FONGECIF (FONds de GEstion des Congés Individuel de Formation) de la région Languedoc-Roussillon au sein du Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM) sous le tutorat de M. Pascal Poncelet (équipe TATOO) et M. Mathieu Roche (équipe TAL).
Dans ce second chapitre nous présentons tout d'abord le LIRMM , puis les équipes TAL et TATOO auxquelles l'auditeur est rattaché (2.1). Nous redéfinissons dans une seconde partie l'intérêt de ce travail et son adéquation avec les besoins des utilisateurs (2.2). Dans la troisième partie, nous nous intéressons aux aspects organisationnels liés à ce projet (2.3).

2.1 LIRMM – Equipes TATOO et TAL

Le Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM) est une Unité Mixte de Recherche de l'Université Montpellier 2 (UM2) et du Centre National de la Recherche Scientifique (CNRS). Créé en 1992, par une volonté scientifique commune du CNRS et de l'Université Montpellier 2, sur la base de la fusion de deux laboratoires (CRIM et LAMM), le LIRMM regroupe ainsi des informaticiens, des microélectroniciens et des roboticiens.

De l'information aux systèmes, de la technologie à l'humain et aux usages, les activités de recherche du LIRMM concernent la modélisation et la conception de systèmes logiciels et matériels (robots, circuits intégrés), les études en algorithmique, bioinformatique, bases de données et intelligence artificielle. Fort de cette diversité, le LIRMM renforce son originalité en alliant théorie, outils, expérimentations et applications dans tous ses domaines de compétence scientifique, en favorisant l'émergence de projets interdisciplinaires au sein du laboratoire (image, sécurité numérique, handicap,…) et à l'interface d'autres laboratoires et domaines scientifiques (mathématiques, sciences du vivant, santé, neurosciences,…).

Le LIRMM s'appuie sur plus de 365 collaborateurs au 1er mars 2011 (50% d'informaticiens, 25% de microélectroniciens, 25% de roboticiens) organisés en équipes-projet.
L'organisation du laboratoire en équipes-projet a amélioré la lisibilité des activités de recherche,

mais également donné l'impulsion nécessaire pour orienter les recherches aux frontières, dans un souci d'innovation et d'ouverture, permettant de couvrir un large champ de thématiques de recherche aussi bien en informatique (algorithmique, gestion de données et extraction de connaissances, génie logiciel, intelligence artificielle, interaction homme-machine) qu'en microélectronique (conception et test de systèmes et circuits intégrés microélectroniques, systèmes matériels-logiciels hétérogènes) ou en robotique (la rééducation et la suppléance fonctionnelle, la robotique médicale, la robotique de manipulation rapide, la vision pour la robotique, protection et analyse d'images, l'interaction haptique, la robotique humanoïde, la robotique d'exploration sous-marine et terrestre, les architectures de contrôle des robots).

Le Département d'Informatique du LIRMM regroupe actuellement 100 chercheurs permanents, 13 associés et plus de 70 doctorants. Parmi les équipes-projets le structurant, nous pouvons citer les équipes TATOO et TAL qui regroupent plus particulièrement les spécialistes en fouille de données et fouille de textes.

Le projet de recherche de l'équipe TATOO sur la fouille de données s'inscrit dans le domaine des grandes bases de données et plus particulièrement dans le domaine de l'Extraction de Connaissances. Il est dans la continuité des actions menées ces dernières années par les différents membres du projet et est au cœur des préoccupations de la communauté nationale et internationale puisqu'il concerne les différents axes suivants : fouille de données dans des bases de données complexes (e.g. données structurées, semi structurées, multidimensionnelles, qualitatives et quantitatives, textuelles etc.) et dynamiques, fouille de données approximatives et aide à la décision. En effet, étant donné la complexité et la vitesse à laquelle les données manipulées évoluent, il devient indispensable de proposer de nouvelles techniques de fouilles de données et, de manière générale, de repenser le processus d'Extraction de Connaissances.
Ces techniques doivent, comme précédemment, permettre d'extraire rapidement la connaissance (comment gérer des données qui arrivent de manière continue, e.g. données de capteurs, données boursières, news, …) mais également proposer à l'utilisateur une certaine flexibilité dans les connaissances extraites. En effet, la "rigidité" des approches classiques pénalise certaines applications pour lesquelles il est primordial d'adopter une méthode approximative. Par exemple, le fait de savoir qu'un utilisateur se comporte "plutôt" ou "peu" comme tel autre groupe de consommateurs offre de nouvelles informations fondamentales pour le décideur. Celui-ci est ainsi en demande d'environnement d'aide à la prise de décision qui au travers de la représentation de la connaissance extraite et de différents critères vont lui permettre de disposer de toutes les

informations nécessaires.

Depuis de nombreuses années, l'équipe TATOO collabore avec l'équipe TAL afin de concilier les approches de fouille de données et les nouvelles méthodes de traitement de données textuelles.

L'équipe TAL (Traitement Algorithmique du Langage) du LIRMM s'est investie dans de nombreux thèmes de recherche liés à la recherche et l'extraction d'informations dans les textes. Afin de mettre en œuvre des méthodes pertinentes, la représentation des données textuelles est une étape essentielle. Dans ce contexte, l'utilisation de descripteurs linguistiques (lexicaux, syntaxiques, sémantiques) et statistiques issues des données textuelles se révèle cruciale. Ces différents types de connaissances peuvent être combinés dans le but de répondre à des problématiques précises.

Parmi les travaux menés conjointement par les deux équipes, nous pouvons citer les cubes de textes, la détection d'opinion, la classification de documents, l'extraction d'information, l'analyse de dépêches en ligne, l'extraction de la terminologie d'un domaine. Outre le choix des descripteurs linguistiques les plus pertinents qui est une problématique complexe, un des verrous scientifiques de ce domaine tient à la caractérisation de fonctions d'agrégation originales. Ces dernières doivent être à la fois adaptées au contexte des entrepôts de données tout en étant pertinentes pour le traitement des données textuelles.

Ces dernières années, les équipes TAL et TATOO se sont intéressées aux informations véhiculées par les nouveaux outils de communication et notamment sur l'extraction de connaissances au travers des messages déposés via l'outil de microblogging Twitter.

2.2 Vers un outil d'analyse de données issues d'un réseau social

Le succès des réseaux sociaux ne fait plus aucun doute et leurs taux d'activité ont atteint des niveaux sans précédent. Des centaines de millions d'internautes sont inscrits dans ces réseaux. Ils échangent via des forums, maintiennent des blogs, racontent leurs dernières pensées, humeurs ou activités en quelques mots.
Le développement des outils mobiles tels que les téléphones portables, permettant de contribuer à ces réseaux de n'importe quel endroit, a favorisé l'émergence de ces nouvelles pratiques. Twitter est l'un de ces réseaux. Il permet aux internautes de "microblogguer", c'est-à-dire d'envoyer des messages courts, des "tweets" de 140 caractères uniquement et de lire les messages d'autres

utilisateurs.

Si certains attributs comme l'auteur, la source du message (web, smartphone, SMS,...), ou encore la localisation sont accessibles au travers de tags spécifiques, la multitude des thèmes abordés ainsi que le contenu même du message (140 caractères de texte libre) rendent difficile un traitement automatique. Le volume d'informations disponibles constitue une autre spécificité à intégrer (10 milliards de messages entre le lancement en 2006 et mars 2010, la barre des 20 milliards franchit 4 mois plus tard, le 31 juillet 2010).

Néanmoins les tweets sont de véritables mines d'informations pouvant offrir aux décideurs de nouvelles connaissances utiles. Des travaux récents, menés notamment au sein des équipes TATOO et TAL, se sont intéressés à l'analyse de tweets pour aider à acquérir rapidement des informations sur des catastrophes naturelles.

Toutefois, l'analyse de telles quantités de données nécessite de définir de nouveaux outils permettant, par exemple, d'extraire les tendances, les résurgences, les similitudes dans des thématiques différentes. Ainsi l'objectif est de pouvoir intégrer toutes les données qui peuvent être extraites des tweets dans un entrepôt de données qui servira de support à des outils d'analyse ou de reporting. De manière plus générale, l'objectif est de proposer pour les tweets une activité d'informatique décisionnelle regroupant une large catégorie de technologies pour collecter, stocker et analyser les données volumineuses pour permettre la prise de décisions stratégiques. Des applications liées au domaine biomédical seront privilégiées. De telles applications peuvent par exemple apporter des informations pertinentes et nouvelles sur le comportement lié à différentes maladies, virus, etc.

2.3 Processus de développement

Pour le développement de ce projet, un cycle de développement en spirale a été adopté (c.f. Figure 1). Nous savions que des analyses complémentaires réalisées au cours du projet généreraient de nouveaux besoins. Cette organisation nous a permis, par l'implémentation de versions successives, de proposer un produit de plus en plus complet.

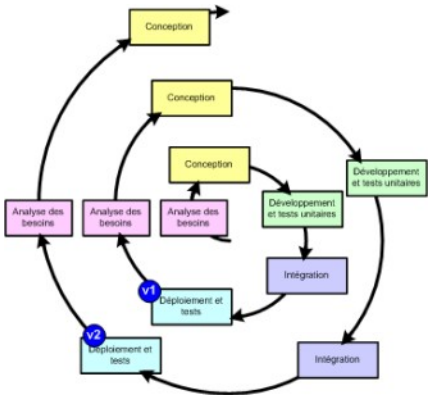


Figure 1: Cycle de développement en spirale

Un certain nombre de tâches sont définies comprenant tout d'abord une analyse de l'existant, puis une première définition des spécifications suivie des premiers développements. Les analyses réalisées nous permettent alors de définir ou d'affiner de nouveaux besoins (Tableau 1).

N° tâche	Identifiant tâche	Description tâche	Délai estimée JO	Charges estimées JH	Date de début prévue	Date de fin prévue
1	AST	Analyse des spécificités des tweets	10 jours	10	18/10/10	29/10/10
2	ATE	Analyse des travaux existants	10 jours	10	01/11/10	15/11/10
3	ATFT	Apprentissage des techniques de fouille de textes	35 jours	30	15/11/10	31/12/11
4	PDS	Première définition des spécifications	21 jours	13	03/01/11	31/01/11
5	ATDW	Apprentissage des techniques liées à l'utilisation d'entrepôts de données et des langages utilisés	41 jours	27	03/01/11	28/02/11
6	PDT	Premier développement et tests	23 jours	20	01/03/11	31/03/11
7	ARPA	Analyse des résultats et proposition d'amélioration	43 jours	40	01/04/11	31/05/11
8	ASMP	Amélioration des spécifications et mis en place d'un prototype complet intégrant l'analyse de données	43 jours	40	01/06/11	29/07/11
9	EARD	Expérimentations, améliorations, mise en place de l'architecture et rédaction des différentes documentations	44 jours	34	01/08/11	29/09/11
10	RRPA	Rédaction du rapport et prise en compte des améliorations proposées par les encadrants	32 jours	30	01/09/11	14/10/11

Tableau 1: Planning prévisionnel du projet

Le diagramme de GANTT représentant ce planning prévisionnel est illustré dans la figure 2.

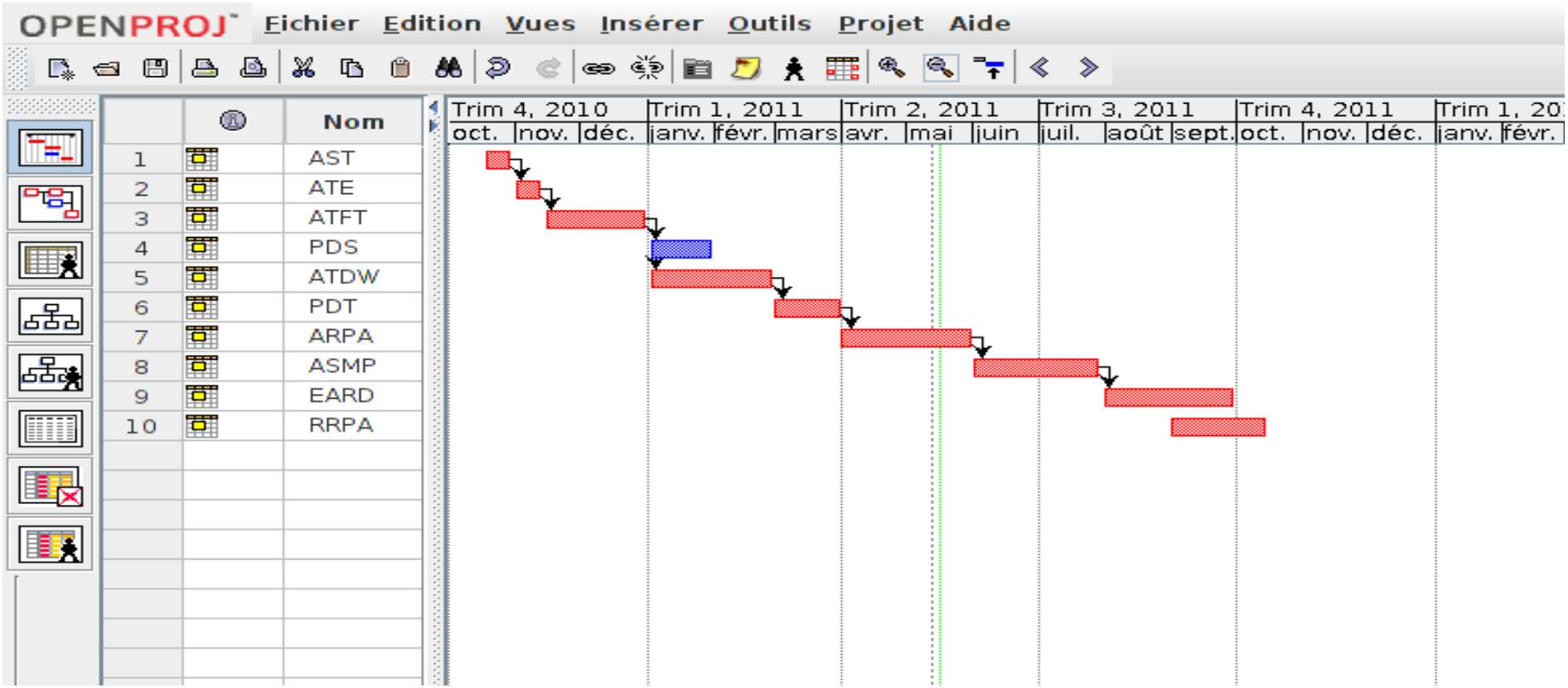


Figure 2: Diagramme de Gantt prévisionnel

Concernant les environnements techniques du projet, nous disposons de deux environnements distincts. Le premier que nous pouvons qualifier d'environnement de "développements" sert de support aux analyses, aux développements et aux tests unitaires. Le second que nous pouvons qualifier d'environnement de "production" sert de support aux tests en conditions et en volumétrie réelle. L'environnement de "production" héberge toujours une version stabilisée de la solution.

Les travaux présentés dans ce mémoire respectent dans les grandes lignes le planning établi en amont même si certains aléas couramment associés à la vie d'un projet ont nécessité de légères adaptations.

3 Chapitre 3 : État de l'art et définitions préliminaires

Nous souhaitons tout d'abord introduire les éléments nécessaires à la compréhension de ce travail. Dans ce troisième chapitre, nous commençons par répondre aux questions *"Qu'est-ce qu'un entrepôt de données?"* (3.1) et *"Qu'est ce que Twitter ?"* (3.2). Une fois ces définitions partagées, nous effectuons dans la section 3.3 un état de l'art des travaux et des applications exploitant les données transmises par Twitter. En conclusion, nous discutons de l'adéquation de ces différents travaux à notre besoin (3.4).

3.1 Les entrepôts de données

De plus en plus d'informations sont disponibles au sein des entreprises. Cette connaissance est utilisée pour prendre des décisions stratégiques. L'informatique décisionnelle (ou *Business Intelligence, BI)* est définie comme un système interprétant des données complexes permettant aux dirigeants d'entreprise de prendre les décisions pertinentes. Les bases de données de type relationnel sont inadaptées à de tels besoins décisionnelles. En effet, les requêtes décisionnelles, particulièrement complexes par principe, perturbent les traitements opérationnels lors de leurs exécutions. Un entrepôt de données (ou base de données décisionnelle, ou encore *data warehouse*) est une base de données utilisée pour collecter, ordonner, journalier et stocker des informations provenant de bases de données opérationnelles. L'entrepôt se trouve ainsi être au cœur d'une architecture décisionnelle dont l'objectif est de construire de l'information utile pour l'aide à la décision.

3.1.1 Architecture générale

Les systèmes de gestion des bases de données (SGBD), assurant la gestion et l'accès aux bases de données opérationnelles, et les entrepôts de données sont basés sur deux systèmes différents : OLTP et OLAP

OLTP (*On Line Transaction Processing*) ou traitement transactionnel en ligne est le modèle utilisé dans les systèmes de gestion de base de données. Le traitement transactionnel en ligne permet d'effectuer des modifications d'informations en temps réel. Ce type de traitement est utilisé dans des activités opérationnelles, par exemple lors des transactions commerciales (opérations bancaires, achats de biens, billets, réservations). Ces actions doivent pourvoir être effectuées très rapidement

par de nombreux utilisateurs simultanément. Chaque transaction travaille sur de faibles quantités d'informations, et toujours sur les versions les plus récentes des données.

Les entrepôt de données reposent quant à eux sur le système **OLAP** (*On Line Analytical Processing*). Ce système travaille en lecture seulement. Les programmes consultent d'importantes quantités de données pour procéder à des analyses. Les objectifs principaux sont le regroupement, l'organisation des informations provenant de sources diverses, leurs intégrations et leurs stockages. Ce système permet à l'utilisateur de retrouver et d'analyser les informations facilement avec des temps de réponse quasi-instantanés. Il dispose des données historisées mais ne dispose en revanche pas de mises à jour en temps réel. Il existe un décalage temporaire entre la réalité d'une situation retranscrite dans les bases de données opérationnelles et les entrepôts de données.
Ces bases de données sont souvent d'un ordre de grandeur nettement supérieur à celle des bases OLTP, du fait de la conservation de l'historique.

Les deux types d'applications peuvent être comparés selon différents aspects. Nous présentons le tableau comparatif inspiré de [TESTE, 2000], qui dresse les comparaisons d'un point de vue données et d'un point de vue utilisateur (c.f. Tableau 2).

	OLTP	OLAP
Données	Exhaustives, détaillées	Données Agrégées, résumées
	Courantes	Historiques
	Mises à jour	Recalculées
	Dynamiques	Statiques
	Orientées applications	Orientées sujets d'analyse
	De l'ordre des gigaoctets	De l'ordre des téraoctets
Utilisateurs	Agents opérationnels	Décideurs
	Nombreux	Peu nombreux
	Concurrents	Non concurrents
	Mises à jour et interrogations	Interrogations
	Requêtes prédéfinies	Requêtes imprévisibles
	Réponses immédiates	Réponses moins rapides
	Accès à peu d'informations	Accès à de nombreuses informations

Tableau 2: Comparatif OLTP/OLAP

En 1993, E. F. Codd, fondateur des bases de données relationnelles, définit le concept OLAP dans [CODD, 1993]. Il y décrit douze règles de conception du "modèle OLAP" (complétées par six autres règles en 1995 toujours par E. F. Codd) qui définissent dix-huit dispositifs pour la conception de système OLAP.

Les règles OLAP définies par E. F. Codd ont cependant été controversées. Nigel Pendse, l'initiateur de l'OLAP Report (un organisme normalisateur autour des notions OLAP[1]), considère même le terme OLAP comme peu explicite [PENSE et CREETH, 1997], il ne fournit pas une définition et ne permet pas de savoir si un outil relève ou non de cette technologie. Par ailleurs, douze règles ou dix-huit dispositifs, c'est, selon lui, une quantité trop importante pour être facilement retenue. Nigel Pendse suggère alors d'assimiler le terme OLAP à une définition comprenant cinq termes : *Fast Analysis of Shared Multidimensional Information* (le modèle FASMI[2]) traduit en français par "Analyse Rapide d'Information Multidimensionnelle Partagée". Cette définition correspond aux critères retenus pour simplifier les règles de E. F. Codd et pour faciliter l'évaluation des outils OLAP.

Ainsi, on parle généralement de système OLAP, sous-entendant ainsi un système d'information répondant aux règles évoquées par E. F. Codd ou aux critères FASMI. L'aspect analyse y est très important.

Techniquement, il existe deux modèles de stockage physique des données. Soit la base est structurellement multi-dimensionnelle comme le propose le modèle MOLAP soit la base est de type relationnelle mais utilisée comme une base multi-dimensionnelle comme le propose le modèle ROLAP.

La base MOLAP (*Multidimensional OLAP*) est l'application physique du concept OLAP. Il s'agit réellement d'une structure multidimensionnelle. Les bases MOLAP sont rapides et performantes. Elles proposent des fonctionnalités particulièrement évoluées. Les bases de type MOLAP restent limitées au gigaoctet.

La base ROLAP (*Relational OLAP*) est en fait une classique base relationnelle organisée pour fonctionner comme une base OLAP. Les bases ROLAP sont bien plus lentes et nettement moins performantes que les bases MOLAP. Mais, immense avantage, elles sont sans limite de taille.

Un troisième modèle, le modèle HOLAP (*hybride OLAP*), propose de cumuler les avantages des deux modèles précédents. Les données agrégées sont stockées sous formes multi-dimensionnelles,

1 http://www.bi-verdict.com/

2 http://www.bi-verdict.com/fileadmin/dl_temp/20b7a4a9b816565790291bcaa0ae4827/fasmi.htm

alors que les données détaillées sont stockées dans des structures relationnelles.
Ces modèles sont détaillés dans la figure 3.

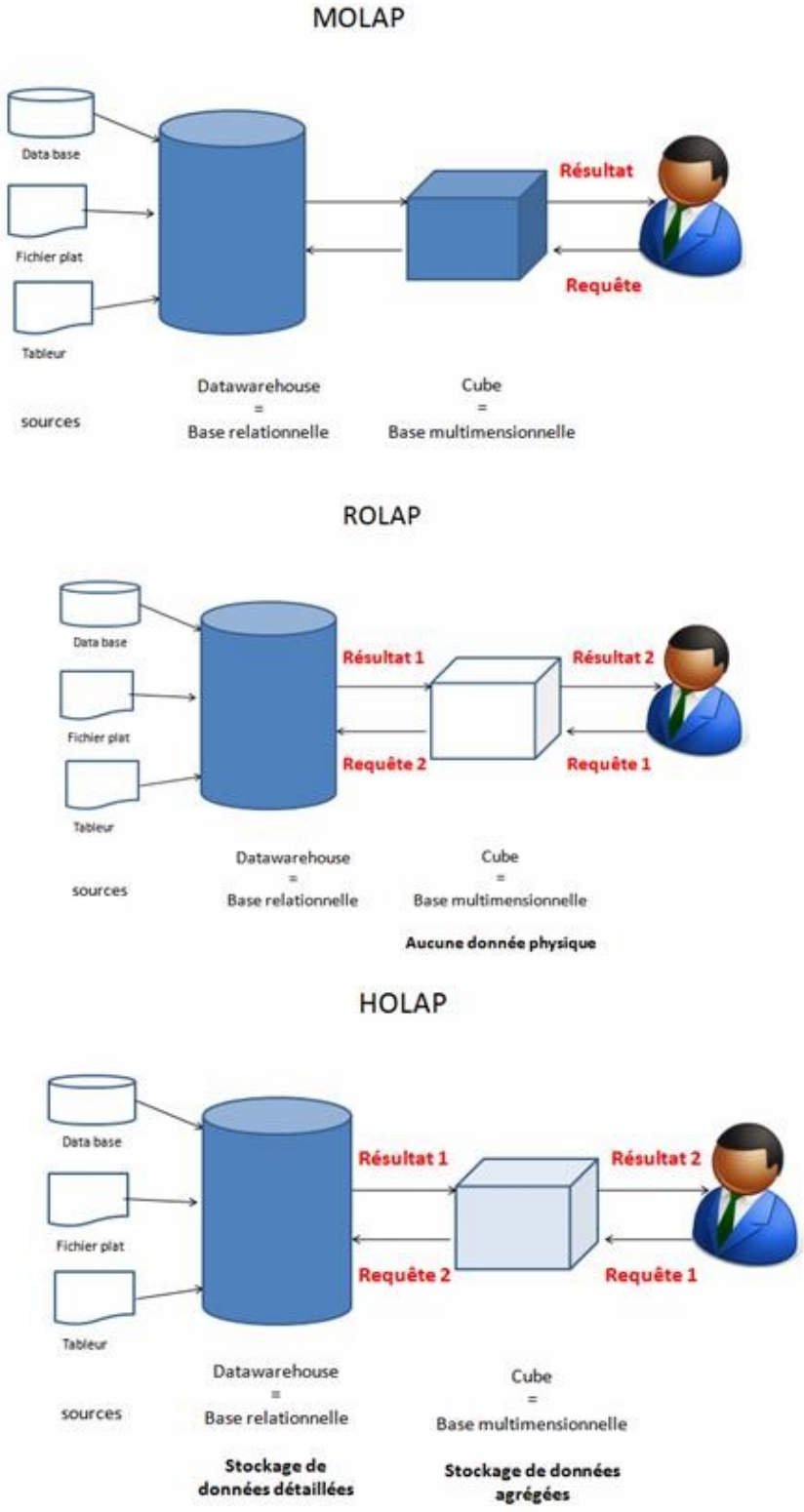


Figure 3: Illustration du MOLAP, ROLAP, HOLAP

L'entrepôt de données est le support nécessaire à la réalisation d'une architecture décisionnelle, qui va permettre, comme l'indique son nom, l'aide à la décision grâce au processus d'analyse qu'elle offre. Cette architecture décisionnelle, peut être représentée classiquement selon la Figure 4.

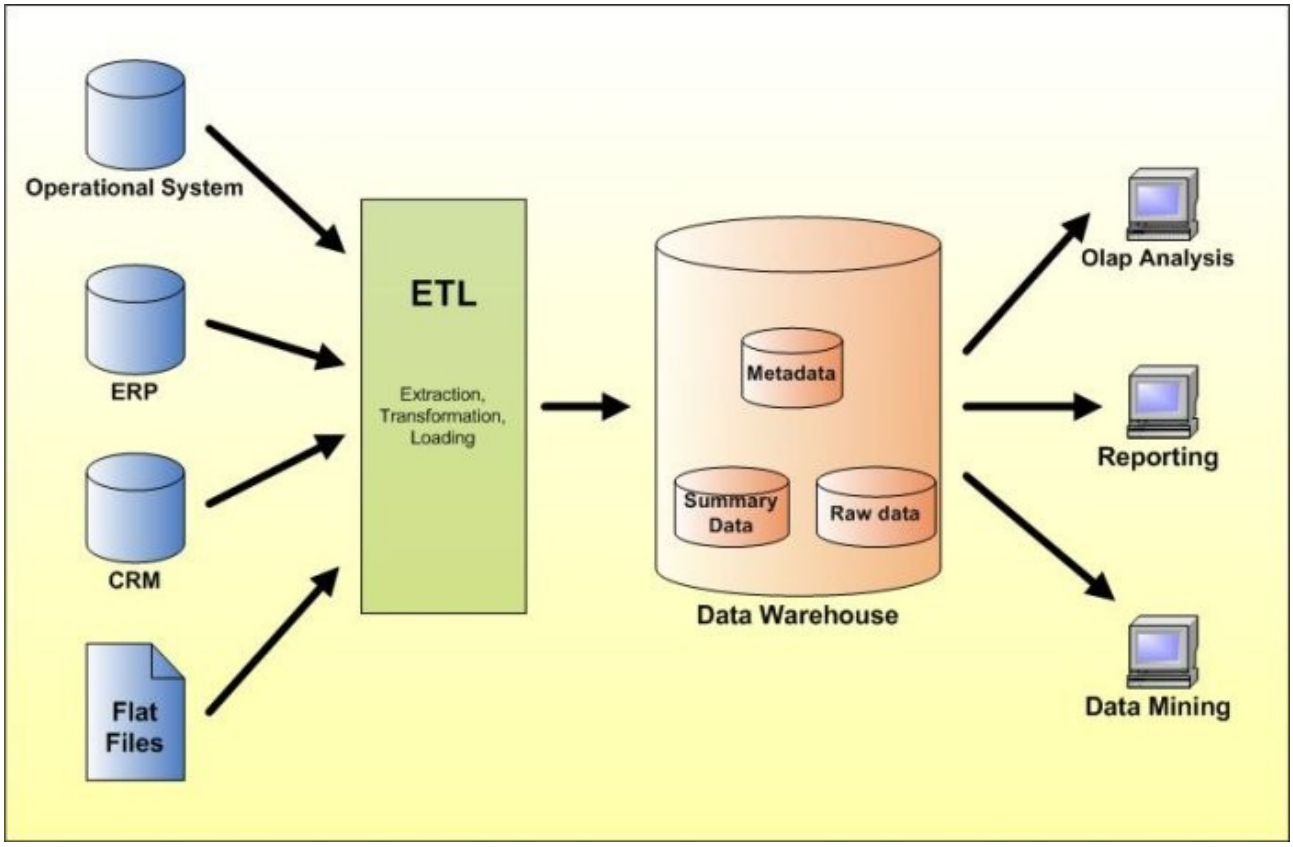


Figure 4: Une architecture décisionnelle

On peut y distinguer les parties sources, gestion/transformation des données et enfin analyse. On parle généralement d'architecture n-tiers en raison des différentes couches nécessaires pour gérer les données et réaliser des analyses.

Cette architecture met en avant deux phases caractéristiques qui sont l'intégration des données et l'analyse. L'ensemble du processus d'intégration de données dans un entrepôt peut être décomposé en trois étapes:

1. Extraction des données à partir de leurs sources
2. Transformation des données (structurelle et sémantique)
3. Chargement des données intégrées.

On parle de façon usuelle de processus ETL (*Extracting, Transforming and Loading*, Figure 5)

[CHAUDHURI ET DAYAL, 1997].

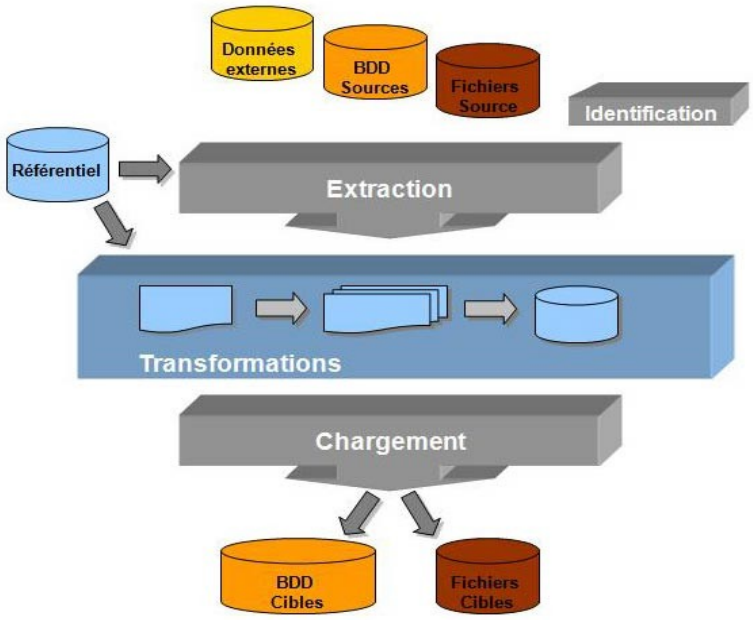


Figure 5: Extraction, Transformations et Chargement

Le processus d'ETL nécessite la mise en place d'une stratégie de rafraîchissement des données. Le rafraîchissement doit être réalisé périodiquement afin d'intégrer les dernières données dans l'entrepôt. La périodicité dépendra des caractéristiques des données sources et des besoins utilisateurs.

On trouve ensuite la phase d'analyse qui peut exploiter aussi bien des analyses statistiques, des tableaux de bord, de la fouille de données, ...
Notons que ces processus d'analyse peuvent s'opérer aussi bien sur l'entrepôt de données que sur les magasins de données (*DataMart,* ensemble de données ciblées, organisées, regroupées et agrégées pour répondre à un besoin spécifique à un métier ou un domaine donné).

3.1.2 Modélisation et conception d'un entrepôt de données

[INMON, 1996] définit un entrepôt de données comme étant une "collection de données orientées sujet, intégrées, non volatiles et historisées, organisées pour le support du processus d'aide à la décision". Les données sont "orientées sujet" dans la mesure où elles sont organisées autour des sujets majeurs et des métiers de l'entreprise. Il permet une vision transversale des différentes activités de l'entreprise. Le fait que les données soient "intégrées" exprime leur provenance de

sources différentes. Cette intégration nécessite une bonne connaissance des sources de données, des règles de gestion, de la sémantique des données, etc. En outre, les données sont "historisées" afin de rendre possible la réalisation d'analyses au cours du temps, nécessitant un recours à un référentiel temporel associé aux données.

De plus, les données sont dites "non volatiles". Une requête émise sur les mêmes données à différents intervalles de temps doit donner le même résultat. Cela doit permettre de conserver la traçabilité des informations et des décisions prises. Enfin, les données sont "organisées pour le support du processus d'aide à la décision" ; il s'agit en l'occurrence d'une organisation multidimensionnelle. Cette organisation est en effet propice à l'analyse et, en particulier, à l'agrégation.

Les possibilités d'analyse sont conditionnées par le schéma de l'entrepôt de données. Différents travaux ont proposé des méthodes pour déterminer le schéma de l'entrepôt [KIMBALL et al., 2000], [GOLFARELLI et al., 1998], [MOODY et KORTINK, 2000], [TRUJILLO et al., 2001]. La construction du schéma de l'entrepôt n'étant pas une tâche facile, plusieurs travaux ont proposé un automatisation partielle [SOUSSI et al., 2005], ou complète de cette tâche [KIM, 2003].

Du point de vue de la conception du schéma de l'entrepôt, nous distinguons trois grandes approches: celle guidée par les données, qualifiée également d'ascendante ; celle guidée par les besoins d'analyse, dénommée également descendante et l'approche mixte qui combine les deux précédentes [SOUSSI et al., 2005].

L'approche orientée données ignore les besoins d'analyse a priori. Cette approche consiste à construire le schéma de l'entrepôt à partir de ceux des sources de données et suppose que le schéma qui sera construit pourra répondre à tous les besoins d'analyse.

Les approches orientées besoins d'analyse, quant à elles, proposent de définir le schéma de l'entrepôt en fonction des besoins d'analyse et supposent que les données disponibles permettront la mise en œuvre d'un tel schéma.

Enfin, l'approche mixte considère à la fois les besoins d'analyse et les données pour la construction du schéma. Cette approche est celle qui fait l'objet de plus d'investigations aujourd'hui. L'idée générale est de construire des schémas candidats à partir des données (démarche ascendante) et de les confronter aux schémas définis selon les besoins (démarche descendante) [PHIPPS et DAVIS, 2002].

La modélisation des entrepôts de données s'appuie sur deux concepts fondamentaux : le **concept de fait** et le **concept de dimension**. Un fait représente un sujet d'analyse, caractérisé par une ou plusieurs mesures, qui ne sont autres que des indicateurs décrivant le sujet d'analyse. Ce fait est analysé selon des axes d'observation qui constituent également ses descripteurs.

Un entrepôt de données présente alors une modélisation dite "multidimensionnelle" puisqu'elle répond à l'objectif d'analyser des faits en fonction de dimensions qui constituent les différents axes d'observation des mesures. Ces dimensions peuvent présenter des hiérarchies qui offrent la possibilité de réaliser des analyses à différents niveaux de granularité (niveaux de détail).

Ces concepts de base ont permis de définir trois schémas classiques reconnus comme relevant d'un niveau logique de conception, en raison du recours à la notion de table (table de faits, table de dimension).

Le premier schéma est le schéma en étoile. Il se compose d'une table de faits centrale et d'un ensemble de tables de dimension (Figure 6).

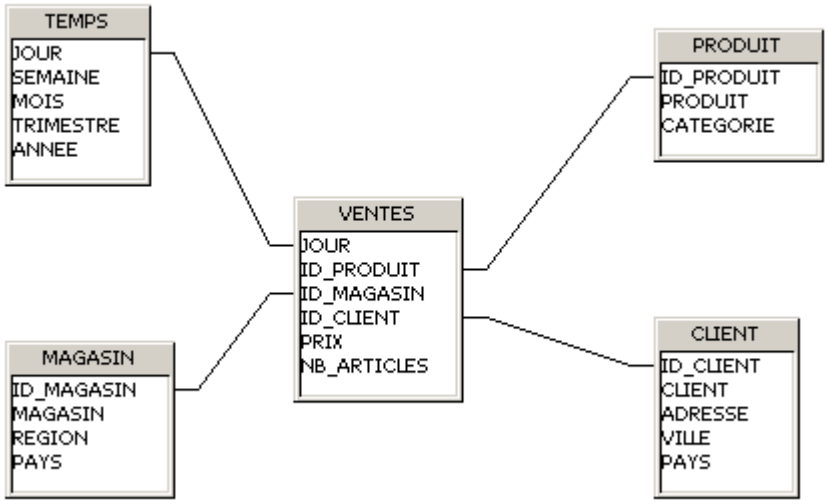


Figure 6: Exemple de modèle en étoile

Le deuxième schéma est le schéma en flocon de neige. Il correspond à un schéma en étoile dans lequel les dimensions ont été normalisées, faisant ainsi apparaître des hiérarchies de dimension de façon explicite. La normalisation permet un gain d'espace de stockage en évitant la redondance de données, mais engendre une dégradation des performances, dans la mesure où elle multiplie le nombre de jointures à effectuer pour l'analyse (Figure 7).

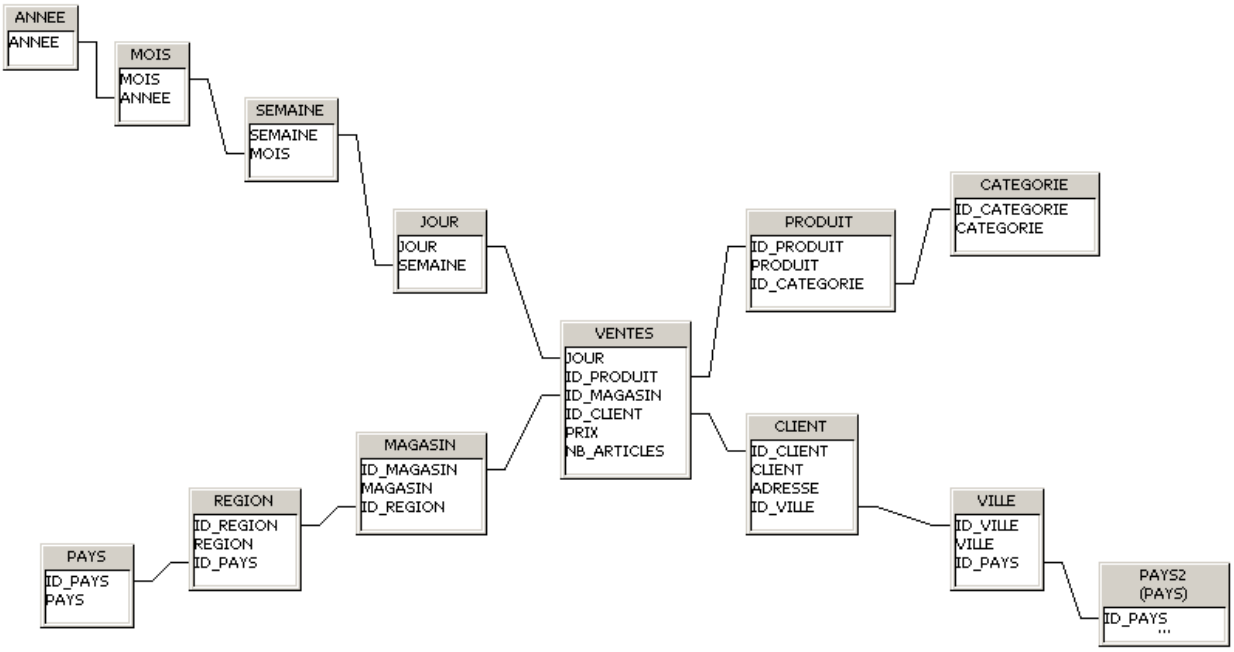


Figure 7: Exemple de modèle en flocon

Le troisième est le schéma en constellation, aussi appelé flocon de faits. Il fait coexister plusieurs tables de faits qui partagent ou pas des dimensions communes hiérarchisées ou non (c.f. Figure 8).

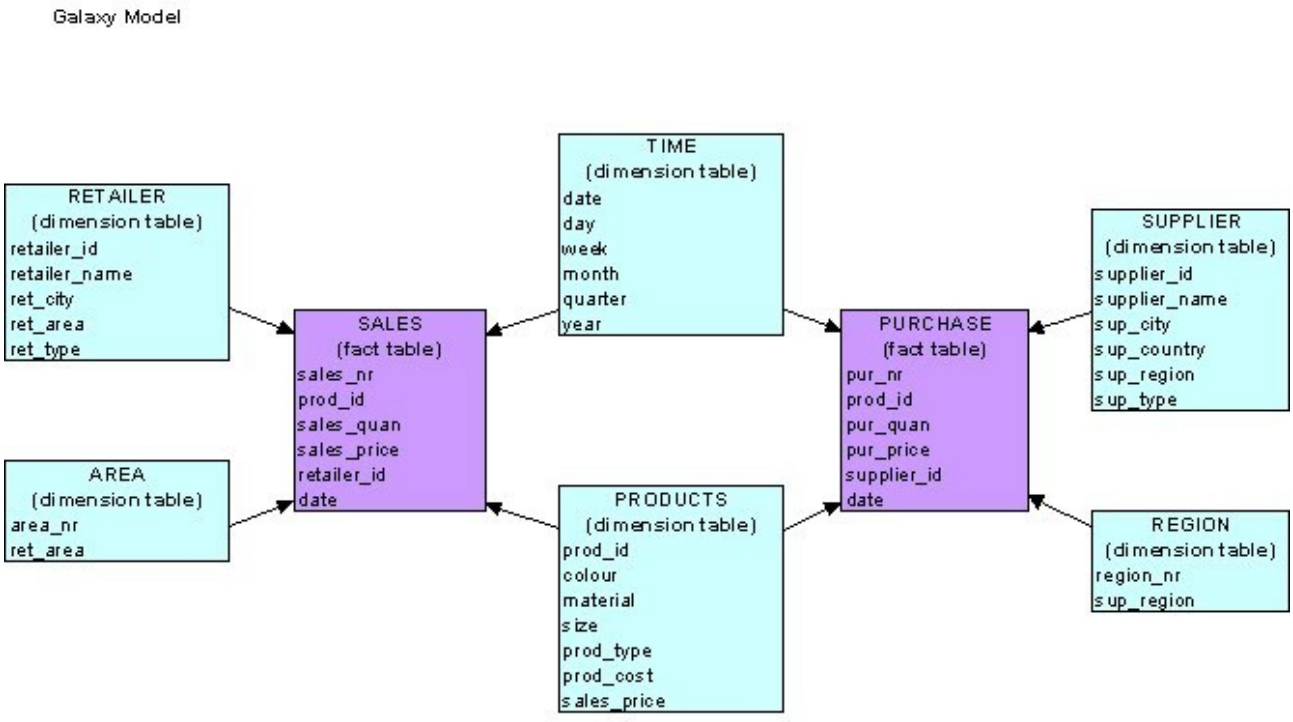


Figure 8: Exemple de modèle en constellation

3.1.3 Navigation au sein d'un entrepôt

Si un des objectifs de l'analyse en ligne est bien entendu la rapidité des temps de réponse, la richesse des possibilités d'analyse a également son importance. Cette richesse dépend du schéma de l'entrepôt et, plus particulièrement, des dimensions et de leur(s) hiérarchie(s). En effet, la navigation dans les données est conditionnée par cette organisation dimensionnelle des données.

Cette navigation se fonde entre autres sur l'agrégation des données. Celle-ci est soutenue par le concept de hiérarchie. En effet, dans les entrepôts de données, les hiérarchies vont permettre de représenter la manière avec laquelle les données sont agrégées. La hiérarchisation des données dans les modèles multidimensionnels permet des analyses à différents niveaux de détail. Classiquement, les hiérarchies sont représentées par des concepts qui sont reliés par des relations un à plusieurs. Autrement dit, une instance d'un niveau inférieur correspond à une seule instance du niveau supérieur et une instance du niveau supérieur correspond à plusieurs instances du niveau inférieur. Par exemple, dans le cas d'une dimension géographique, une ville appartient à pays, un pays contient plusieurs villes. Ainsi le niveau ville constitue le niveau inférieur et le pays le niveau supérieur dans la hiérarchie représentant notre dimension géographique. D'une façon générale, les hiérarchies correspondent à une réalité des données.

Elles peuvent ainsi être définies soit grâce à l'expression des besoins d'analyse des utilisateurs qui connaissent le domaine, soit au niveau des sources de données même puisque ces dernières renferment la réalité de ces données.

La figure 9 présente un cube de données défini sur trois dimensions, dont deux sont hiérarchisées (la géographie et le mode de communication) .

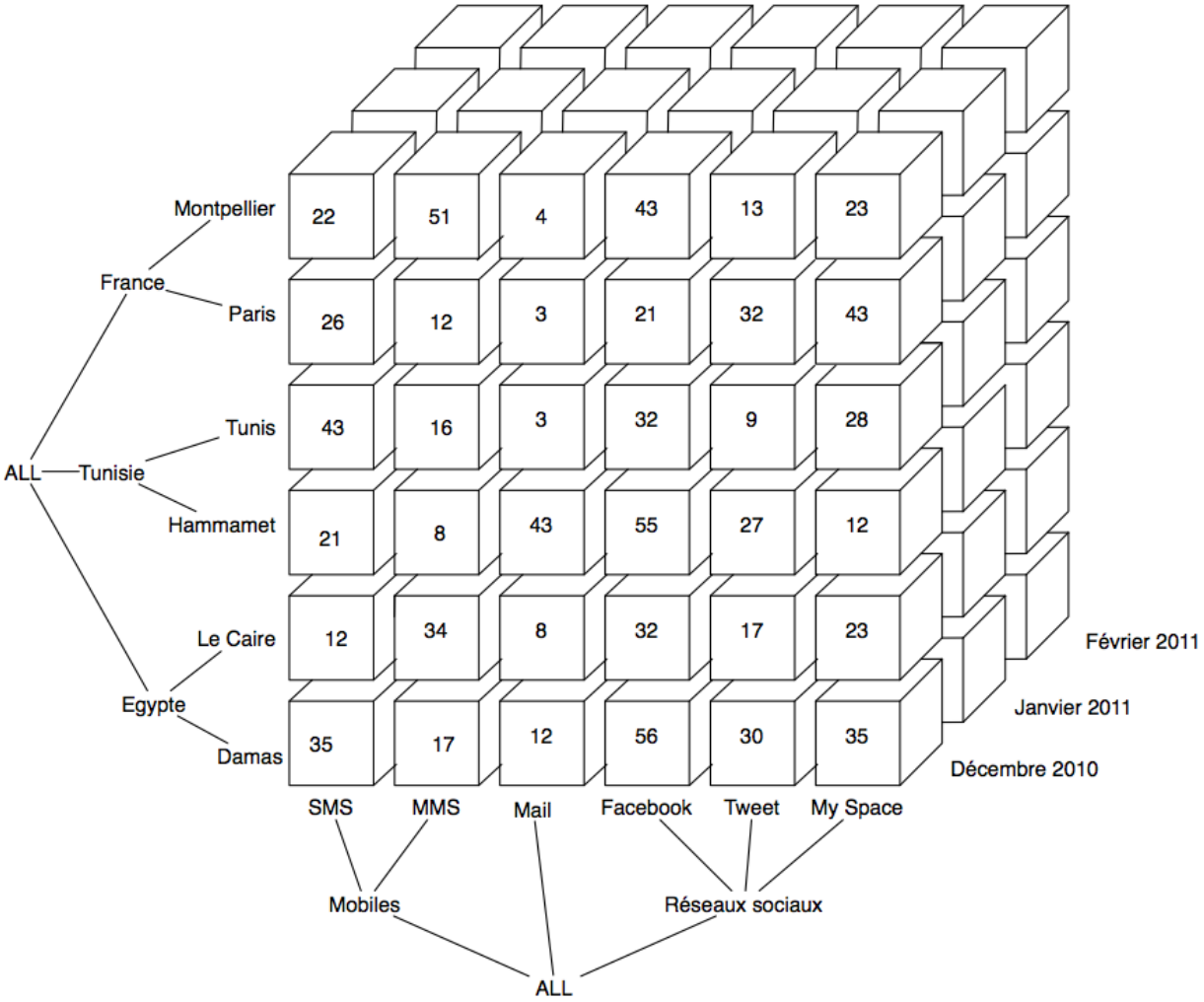


Figure 9: Représentation graphique d'un cube de données

L'entrepôt a pour objectif final l'analyse des données en vue de la prise de décision. Différents types d'analyse peuvent être réalisés comme des analyses statistiques ou des analyses en ligne des données. L'analyse en ligne des données consiste à naviguer dans les données. Cette analyse peut être qualifiée d'exploratoire. Le principe général est d'arriver au cours de la navigation à détecter des points intéressants que l'utilisateur essaye de décrire, d'expliquer en naviguant au sein même des données, par exemple en allant chercher davantage de détails ou en recoupant les informations.

Par exemple, un utilisateur peut décider d'observer le nombre de connexion d'un mode de communication donné par pays puis décider d'analyser plus finement ce nombre de connexion selon la ville d'un pays données afin de mieux comprendre ce qui se passe au sein de ce pays.

Le rôle de l'utilisateur est ici central puisque c'est lui qui réalise la navigation ; celle-ci nécessite une connaissance du domaine afin d'être en mesure de savoir si les valeurs des mesures sont intéressantes ou non.

Afin de réaliser la navigation, différents opérateurs s'appliquent au niveau d'un cube de données. Ils

peuvent être classés en deux catégories : les opérateurs liés à la structure et les opérateurs liés à la granularité. D'une manière générale, les opérateurs liés à la structure permettent la manipulation et la visualisation du cube.

Par exemple une rotation (*Rotate*) permet d´accéder aux données concernant un fait, selon un axe d'analyse (une dimension) différent (figure 10).

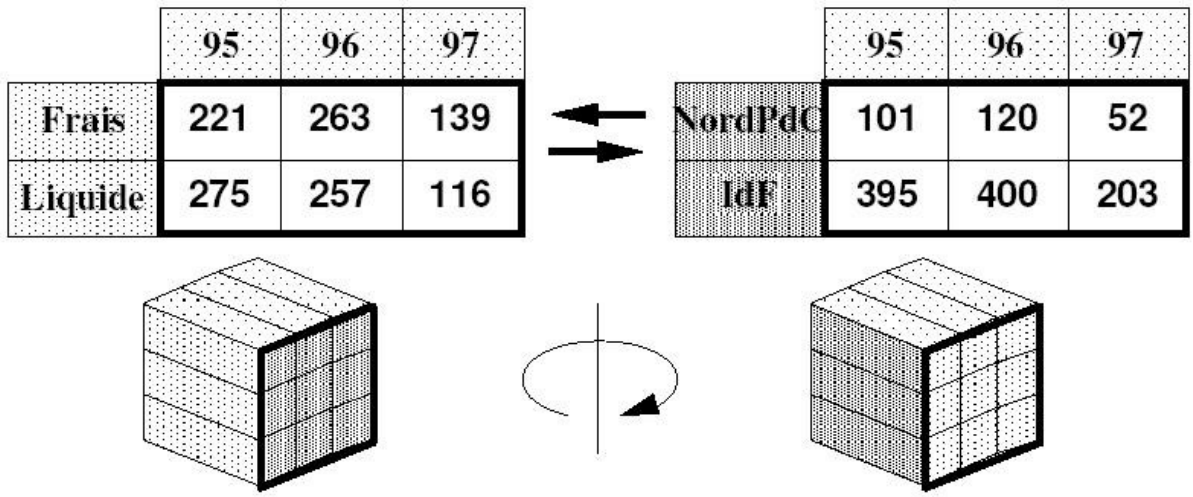


Figure 10: OLAP - Opération de rotation

Le découpage en tranche (*Slicing*) permet d´accéder aux données concernant un fait, selon une partie des axes d´analyse (des dimensions) choisis (figure 11).

		1995	1996	1997
Frais	IdF	220	265	284
	Province	225	245	240
Liquide	IdF	163	152	145
	Province	187	174	184

		1996
Frais	IdF	265
	Province	245
Liquide	IdF	152
	Province	174

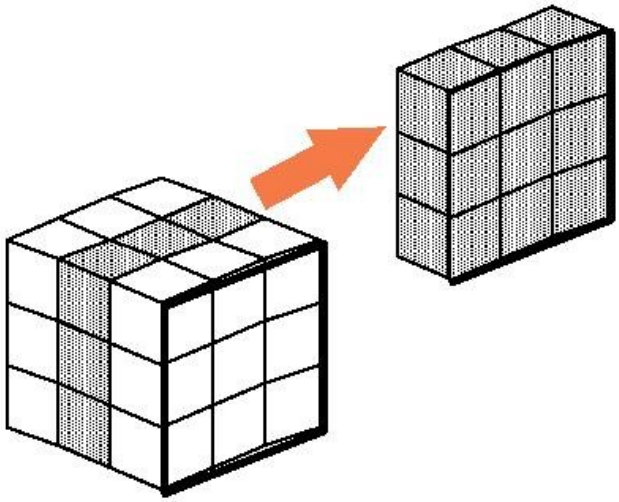


Figure 11: OLAP - Opération de découpage en tranche

La définition d´un contexte d´analyse (*Scoping*) permet d´accéder aux données concernant un fait, en choisissant le niveau de détail désiré, pour chaque axe d´analyse (dimension) choisi (c.f. Figure 12).

		1995	1996	1997
Frais	IdF	220	265	284
	Province	225	245	240
Liquide	IdF	163	152	145
	Province	187	174	184

		1995	1996
Frais	IdF	220	265
	Province	225	245

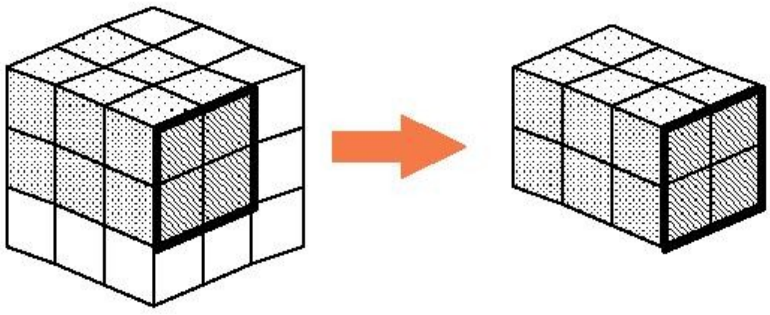


Figure 12: OLAP - Opération de définition d'un axe d'analyse

Les opérateurs liés à la granularité sont au nombre de deux. Ils s'appuient sur la hiérarchie de

navigation entre les différents niveaux. Les opérations liées à la granularité permettent d'agréger les données pour obtenir des données résumées et inversement. Un *roll-up* (Forage vers le haut) consiste à représenter les données du cube à un niveau de granularité supérieur conformément à la hiérarchie définie sur la dimension. Une fonction d'agrégation (somme, moyenne, etc.) en paramètre de l'opération indique comment sont calculées les valeurs du niveau supérieur à partir de celles du niveau inférieur. A l'inverse un *drill-down* (Forage vers le bas) consiste à représenter les données du cube à un niveau de granularité de niveau inférieur, donc sous une forme plus détaillée (c.f. Figure 13).

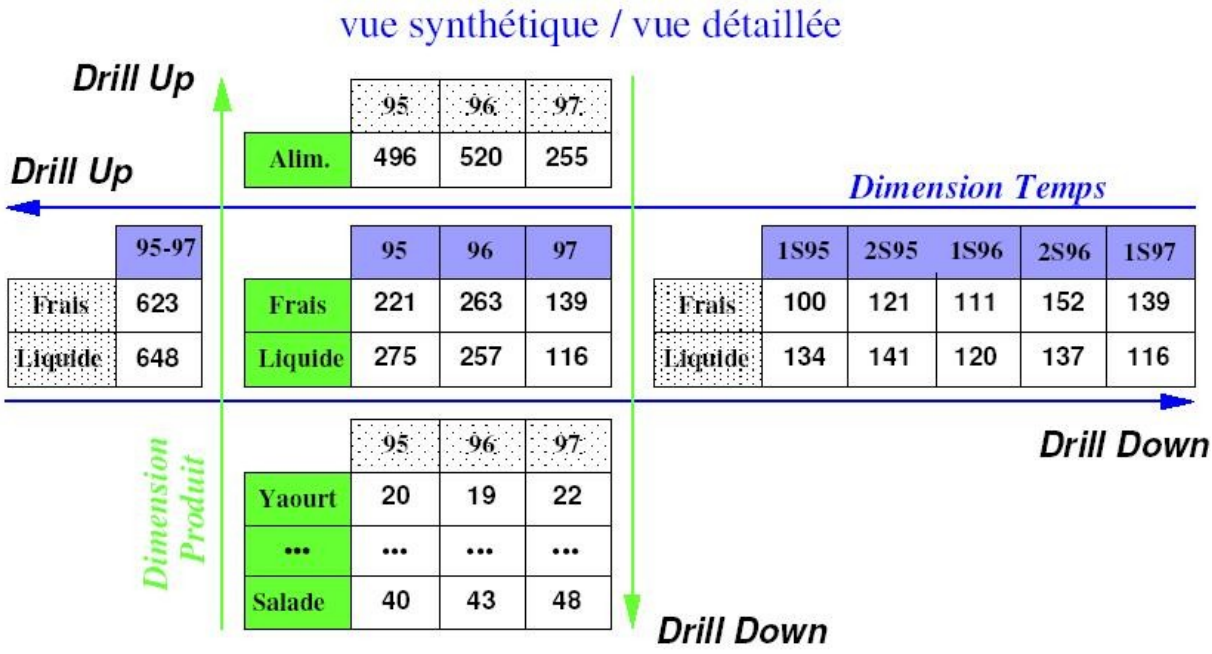


Figure 13: OLAP - Opérations Roll-Up et Drill-Down

3.1.4 Intégration de sources de données

Un entrepôt de données constitue avant tout une alternative pour l'intégration de diverses sources de données. Un système d'intégration a pour objectif d'assurer à un utilisateur un accès à des sources multiples, réparties et hétérogènes, à travers un point d'accès unique. Un tel système amène un utilisateur à se préoccuper davantage de ce qu'il veut obtenir comme informations plutôt que de se demander où est stockée l'information désirée. Cela le dispense aussi de rechercher et trouver les sources de données adéquates, interroger chacune des sources de données potentielles et enfin de synthétiser les différents résultats obtenus pour finalement disposer des informations recherchées.
Pour faire des recherches sur l'ensemble de ces sources, une intégration de celles-ci est nécessaire. Deux approches sont envisageables :

- migrer les requêtes vers les sources de données, on parle alors d'approche "virtuelle" ou d'approche "non matérialisée" ,
- où migrer les données pour les centraliser dans une source cible on parle alors d'une approche "matérialisée" ou d'approche "d'entreposage" [BOUSSAID et al., 2003].

Dans la première approche, les données restent au niveau des sources. Parmi les solutions "non matérialisées", nous pouvons citer les portails, les interfaces de programmation, ou encore la médiation. L'approche de médiation constitue une des approches les plus sophistiquées (Figure 14) [WIEDERHOLD, 1992].

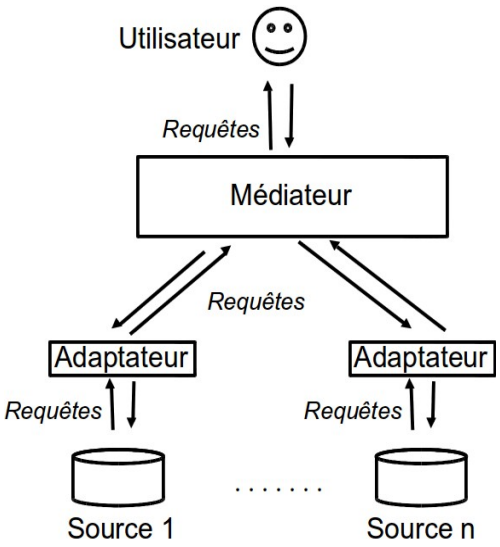


Figure 14: Approche virtuelle

A l'inverse, dans les approches matérialisées, les données sont extraites des différentes sources et combinées pour être stockées dans un entrepôt. Il s'agit ici d'une intégration des données. Toutes les données provenant des différentes sources sont organisées, coordonnées, intégrées pour être finalement stockées de manière centralisée (Figure 15). Cette centralisation physique des données permet à l'utilisateur d'avoir une vue globale des différentes sources en interrogeant directement l'entrepôt de données.

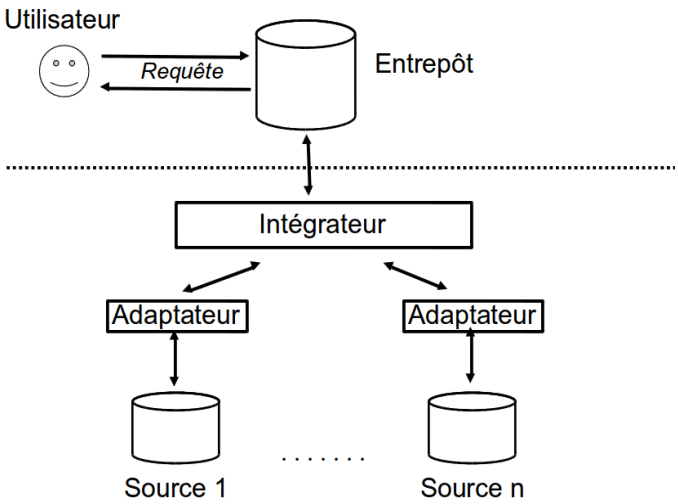


Figure 15: Approche matérialisée

Selon la fréquence de mise à jour des sources, leurs nombres et la capacité à prédire et anticiper les requêtes utilisateurs, où une approche virtuelle sera préférée à une approche matérialisée (Tableau 3).

	Matérialisé	Virtuelle
Connaissance des besoins	+	-
Performance	+	-
Historique	+	-
Volume	-	+
Actualisation temps réel	-	+
Ajout/suppression de sources	-	+

Tableau 3: Approche matérialisée ou virtuelle ?

Un volume de données très conséquent et une information en temps réel rendent difficile l'analyse des informations échangées. Les réseaux sociaux, qui présentent de tels caractéristiques, se prêtent au contexte des entrepôts de données. Nous présentons dans la section suivante un réseau social en particulier: Twitter.

3.2 Twitter: un réseau social

Twitter est un outil de réseau social et de microblogage qui permet à un utilisateur d'envoyer gratuitement des messages brefs, appelés tweets ("gazouillis") de 140 caractères.
Twitter a été créé à San Fransisco au sein de la startup Odeo Inc. Le slogan d'origine, *"What are you*

doing?" , le définissait comme un service permettant de raconter ce qu'on fait au moment où on le fait via SMS (c'est pour cette raison que les tweets sont limités à 140 caractères, un SMS étant limité à 160 caractères, les concepteurs ont estimé que 20 caractères étaient nécessaires pour y accoler le nom d'utilisateur). La première version s'intitulait stat.us puis twittr (en référence au site de partage de photos Flickr) puis Twitter, son nom actuel.
Le 21 mars 2006, un des membres du projet initial, Jack Dorsey (@jack), envoya le premier tweet[3].

Figure 16: Le premier tweet

Le service fut ouvert au public le 13 juillet 2006. Twitter supporte aujourd'hui de multiples canaux d'émission (Internet, messagerie instantanée, SMS, applications sur smartphone). L'utilisation est simple et gratuite, l'utilisateur dispose de 140 caractères pour diffuser un message à qui veut bien le recevoir si il spécifie son compte comme public ou à son réseau uniquement si il le défini en privé. Les Abonnés ou *Followers* sont les personnes qui suivent votre actualité. Parallèlement, vous choisissez les membres de Twitter dont vous voulez suivre les publications. Les Abonnements ou *Following* correspondent aux comptes Twitter que vous suivez. Il est possible d'envoyer un message directement à une personne (le message ne sera visible que par celle-ci) mais l'usage consiste à ouvrir ses discussions à l'ensemble de son réseau. Contrairement aux autres réseaux sociaux, Twitter n'invite pas les lecteurs à commenter les messages postés.

3 http://twitter.com/#!/jack/status/20

Au départ conçu pour communiquer sur ses activités en temps réel, de nombreux usages ont accompagné le développement de Twitter.

Voici quelques exemples d'utilisation:

- Faire partager à sa communauté sa vie au quotidien. Twitter peut se poser en alternative à un blog (on parle de *micro-blogging*).
- Alerter son réseau lors de mises à jour de son blog. Généralement le tweet contient le titre du billet et un lien vers son blog.
- Échanger comme une messagerie (sur des sujets assez légers et généralement non privés) à deux ou à plusieurs comme il arrive parfois dans un bar où toute personne à proximité peut écouter et participer s'il le souhaite.
- Communiquer professionnellement. Twitter propose une alternative au communiqué de presse traditionnel, plus facile à organiser et à maîtriser.
- Maîtriser sa visibilité dans les médias. Pour les hommes et femmes publiques (politique, *people*), Twitter permet de se maintenir aux yeux de tous, même lorsque leur actualité est de faible intensité.
- Permettre de faire suivre en direct des conférences ou des réunions (le *live-tweet*). On peut prendre l'exemple des *KeyNote* d'Apple (réunions de présentation des futurs produits) durant lesquelles de nombreux bloggeurs publient en direct de la salle les annonces du grand patron d'Apple.
- Indiquer une bonne table, un bon plan de manière éphémère. Le but est simplement de pouvoir communiquer à son réseau un site Internet, un blog, un article ou un service que l'on trouve intéressant.
- Interroger la communauté (si on recherche un bon restaurant par exemple). En interrogeant l'ensemble de son réseau, on multiplie les chances d'obtenir une réponse.
- Signaler sa présence dans un lieu.

Il existe aussi d'autres usages plus inédits comme :

- Servir de support aux révolutions modernes (Iran en 2010, en Égypte ou Tunisie en 2011)
- Permettre la publication d'un roman refusé par les éditeurs (Matt Stewart (@mjfstewart), l'auteur américain a publié son roman "The French Revolution" sur un compte Twitter @thefrenchrev[4] à raison de 3700 tweets pour arriver à un livre de 95 000 mots et de 480 000

4 http://twitter.com/#!/thefrenchrev

caractères. Le *buzz* généré lui a finalement permis de trouver un éditeur.

A noter deux performances techniques qui ont choisit Twitter comme illustration :

- Le 22 janvier 2010, le premier message envoyé depuis l'espace sans passer par une base terrestre. A l'aide d'une connexion internet spéciale développée pour l'occasion, l'astronaute de la NASA Timothy "TJ" Creamer (@Astro_TJ) a envoyé sans passer par la Terre le premier tweet de l'espace[5].

Figure 17: le tweet au delà des étoiles

- Le 6 mai 2011, le britannique Kenton Cool (@KentonCool) a envoyé un tweet depuis le sommet du mont Everest à 8848 m d'altitude[6].

5 http://twitter.com/#!/Astro_TJ/status/8062317551
6 http://twitter.com/#!/KentonCool/status/66298015043948544

Figure 18: le tweet au delà des nuages

Comme nous pouvons le constater, le but premier du service a vite été détourné pour des usages différents et de nouveaux continuent d'apparaître tous les jours.

De nombreux indicateurs montrent que Twitter croît à une vitesse vertigineuse selon les chiffres présentés sur le blog officiel de Twitter[7]:

Concernant les tweets :

- 3 ans, 2 mois et 1 jour : Le temps qu'il a fallu pour envoyer le premier milliard de tweets
- 1 semaine : Le temps qu'il faut aujourd'hui pour envoyer un milliard de tweets.
- 50 millions : La moyenne du nombre de tweets envoyés chaque jour, il y a un an.
- 140 millions : La moyenne du nombre de tweets envoyés chaque jour, le mois dernier.
- 177 millions : Le nombre de tweets envoyés le 11 mars 2011.
- 456 : Le nombre de tweets par second (TPS) envoyés lors de la mort de Michael Jackson le 25 juin 2009 (un record à l'époque).
- 6 939 : Le record actuel de TPS, fixé 4 secondes après minuit au nouvel an au Japon.

Concernant les Comptes utilisateurs :

7http://blog.fr.twitter.com/

- 543 000 : le record de comptes créés en une journée, fixé le 11 mars 2011.
- 460 000 : la moyenne des comptes créés chaque jour au cours du denier mois.
- 182%. Taux de croissance du nombre d'utilisateurs de notre service sur mobile en un an.

D'autres statistiques fournies par la société Karalys qui synthétise les chiffres clés à octobre 2010[8] :

- 145 millions d'utilisateurs (+80% en 1 an)
- 225 000 utilisateurs français estimés en août
- 7% des utilisateurs utilisent régulièrement le service (15 000 en France) et 8,15% intensivement (18 000 en France) soit 33 000 utilisateurs actifs en France
- 370 000 nouveaux inscrits chaque jour
- 90% des tweets sont publics, 25% contiennent un lien
- 10% des tweets renvoient vers un article et 4% sont du spams
- Les USA génèrent 62% (Royaume Uni 7.8% Canada 5.7%)du trafic, les Royaumes Unis 7,87% et la France en 33ème position 0,90%
- 90% des tweets sont générés par 10% des utilisateurs
- 92% des utilisateurs ont moins de 100 abonnés, 76% moins de 19 abonnés et 1,35% ont plus de 500 abonnés
- Il y a plus de femmes (53%) que d'hommes (47%)
- 57% des utilisateurs ont entre 20 et 34 ans, 66% moins de 25 ans et 81% moins de 30 ans
- 50% de fréquence occasionnelle ou rare (- de 1 fois par semaine), 15 % au moins 1 fois par jour
- 30% de récit personnel 27% de conversation privée 10% de lien vers des articles d'actualité ou de blog 6% de commentaires sur l'activité en live 4 % de spam 4% de publicité

Une des raisons de la réussite de Twitter vient du fait que les utilisateurs ont su s'approprier le service et en définir les règles et les coutumes.

8 http://www.blogkaralys.com/2010/10/twitter-les-principaux-chiffres-cles.html

Outre le contenu du message, un certain nombre de règles ou de tag ont été créées par les utilisateurs de Twitter pour régir la twittosphère.

Le "@" par exemple est toujours accolé au pseudo d'un compte Twitter et permet de faire savoir à son destinataire que vous lui adressez un message. Par exemple si un utilisateur Riri écrit dans son message "@Fifi" alors tout son réseau pourra lire le message mais les utilisateurs Loulou et Donald seront que ce message ne leur est pas destiné.

Un message contenant "RT" est un message déjà publié par une première personne et republié par une autre personne (RT pour retweet). Il est l'équivalent de la fonction transmettre/forward d'une messagerie classique. Le message est alors constitué de la manière suivante: "RT @auteurdutweet message". Par exemple si un utilisateur Fifi commence un message par "RT @Donald …" alors son réseau saura que le message initial provenait de l'utilisateur Donald.

Le "#" suivi d'un mot sans espace permet de définir de manière générale le sujet principal du tweet. Ils permettent de découvrir de nouvelles personnes qui parlent ou s'intéressent aux mêmes sujets. Lors d'un événement, il permet de suivre toutes les conversations sur Twitter relatives à cet événement, par exemple pour les Jeux Olympiques de Vancouver avec les tags #JO2010 ou #JO.

Certains tag spéciaux ont été créés comme #FF (le FollowFriday est un moyen de faire découvrir aux personnes qui vous suivent de nouveaux membres que vous appréciez et dont vous aimez suivre les tweets), #NSFW ("Not Safe For Work" qui peut être traduit par "Ne Pas Ouvrir Au Travail" et suit généralement un lien qu'il peut être préférable de ne pas ouvrir dans un espace public) ou encore #FOTD ("Find of the Day" qui pourrait être traduit en français par "Découverte du Jour". Dans le même esprit que le FollowFriday, il permet de faire découvrir des sites ou comptes Twitter à ses abonnés.).

En plus du message, un tweet est porteur d'un ensemble d'informations comme son identifiant, sa provenance (internet, téléphone, …) ou encore diverses informations sur l'émetteur. Ces données sont décrites dans la section 4.1.

Ces données sont accessibles grâce aux interfaces de programmation relativement complètes fournies par Twitter.
Twitter dispose officiellement de trois interfaces de programmation (*API - Application Programming Interface*) appelées REST API, STREAMING API et SEARCH API. Chacune de ces API délivre des informations spécifiques. Leur fonctionnement est expliqué avec précision à l'adresse http://dev.twitter.com/.
En résumé, la Rest API fournit des fonctionnalités basiques comme l'édition de tweets et le suivi

des abonnements (*followers, followings*). Il permet aux développeurs de récupérer certaines informations sur son réseau (par exemple: les *Followers*) ou sur son contenu (les 20 plus récents tweets retweetés par les autres utilisateurs, les 20 plus récents retweets faits par l'utilisateur,les 20 plus récents tweets faisant mention d'un utilisateur authentifié. ...).

La STREAM API permet d'obtenir, dans une approche quasi temps réel, un sous-ensemble de données publiques ou protégées de Twitter. Cette API se divise en 3 sous-API qui sont la STREAMING API (qui porte sur la gestion des statuts publics de tous les utilisateurs, filtrés de différentes manières : mots-clés, situation géographique), la USER STREAMS (qui fournit des informations relatives à un utilisateur) et enfin la SITE STREAMS (qui permet le suivi en temps réel de multiples comptes).

Enfin la SEARCH API permet de retourner des tweets qui correspondent à une recherche donnée sur un mot clef ou un utilisateur donné. Elle permet de limiter la recherche aux tweets les plus populaires ou aux tweets les plus récents.

Outre le contenu du tweet, les API fournissent des informations connexes au tweet (la date et l'heure du messages, la source (smartphone, site web, …), le lieu où le message a été posté, ...) et des informations sur l'utilisateur (pseudo, nom, date de création, site web, ...).

Grâce à ces interfaces de programmation, on trouve de nombreuses applications exploitant les tweets.

3.3 Panorama des outils d'analyse de données "tweets"

Nous pouvons distinguer deux types d'applications, soit celles présentes sur Internet, soit celles issues de travaux de recherche. Les applications Internet présentent l'avantage de pouvoir être manipulées mais il est difficile d'obtenir de l'information sur leur fonctionnement. A l'inverse, les applications issues de travaux de recherche font l'objet de documentation fournie, notamment au travers des publications, mais l'accès à l'application elle-même est généralement impossible.

On trouve sur Internet des applications permettant l'analyse des tweets (tendance, sentiments) ou des utilisateurs. Par exemple tweetsentiments[9] permet de suivre l'évolution du sentiment général au cours du temps.

9 http://tweetsentiments.com/

Figure 19: TweetSentiment, l'évolution du sentiment général

Twitter Emotion Graphs Twitter[10] montre l'état émotionnel de Twitter en temps réel selon différents sentiments (la joie, la tristesse, le dégoût, la peur, la surprise).

D'autres applications se focalisent sur l'analyse en temps réel des tweets.

"A world of tweets[11]" affiche une carte avec des zones rouges aux endroits où il y a eu le plus grand nombres de tweets depuis le chargement de la page.

10 http://davidguttman.com/twitter_emotion_graphs

11 http://aworldoftweets.frogdesign.com

Figure 20: A world of tweets, d'où tweete t'on ?

Toujours en temps réel, Trends Map[12] permet une visualisation géographique des mots les plus utilisés dans les tweets.

Figure 21: Trends Map, la représentation géographique du Buzz

12 http://trendsmap.com/

Twitter StreamGraph[13] montre l'utilisation dans le temps pour les mots les plus fortement associés dans les 1000 derniers tweets contenant un mot-clé (par exemple, les mots associés au mot "obama" dans la figure 22).

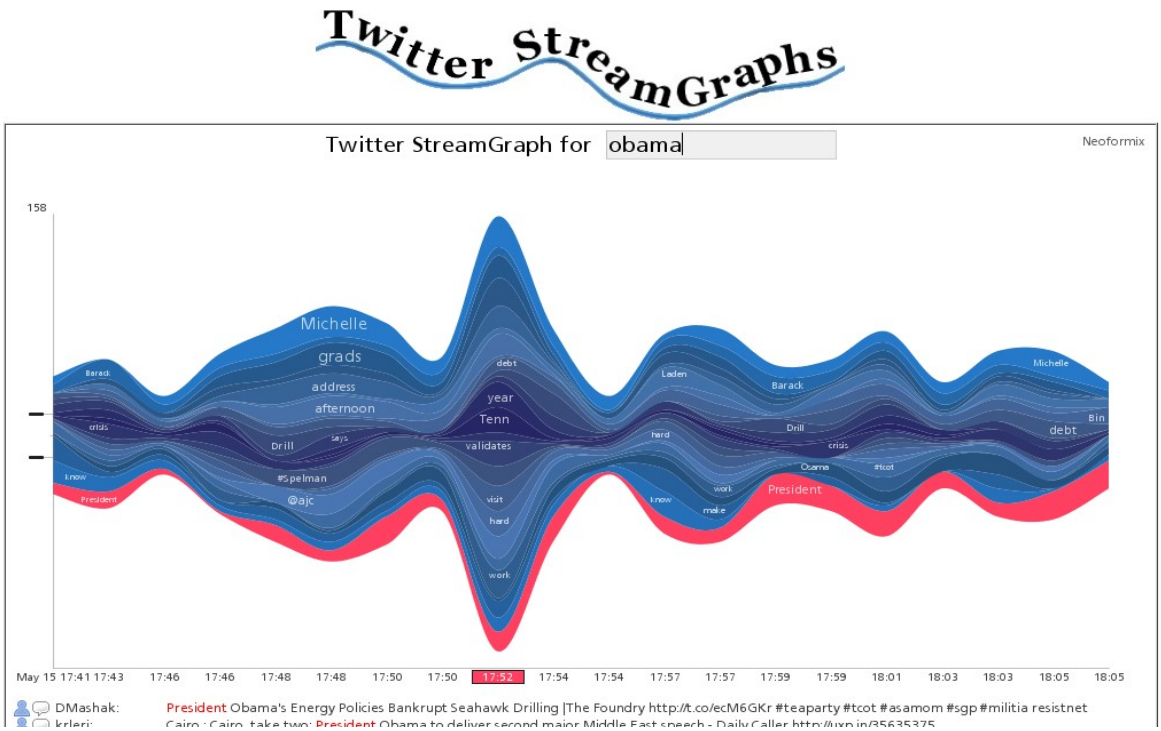


Figure 22: Twitter StreamGraph, l'association d'idées au travers des tweets

D'autres applications se concentrent sur les utilisateurs, Tweet Sentiment[14] propose aussi de décrypter le profil de l'auteur d'un tweet selon des critères de niveau d'éducation, d'extravagance, de niveau de langage, de genre et d'âge.

13 http://neoformix.com/Projects/TwitterStreamGraphs/view.php
14 http://tweetsentiments.com/

Figure 23: TweetSentiments: l'analyse de la personnalité au travers de ses tweets

TwitterGrader[15] permet, quant à lui, de vérifier la valeur d'un profil Twitter, il le compare avec les millions d'autres utilisateurs dont le profil a été mesuré.

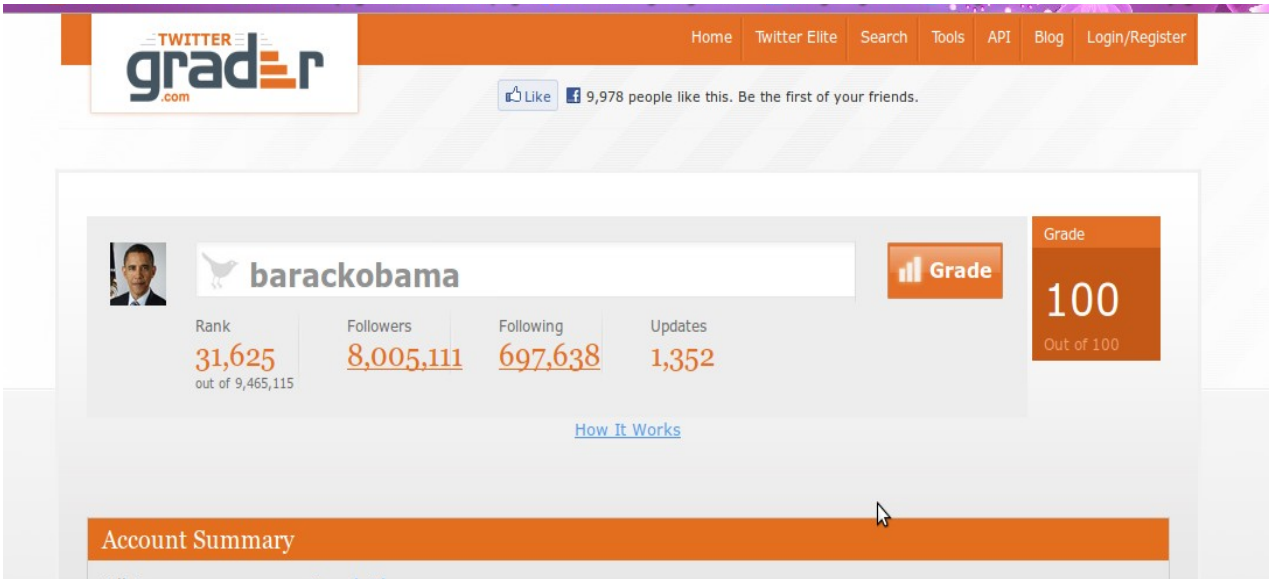
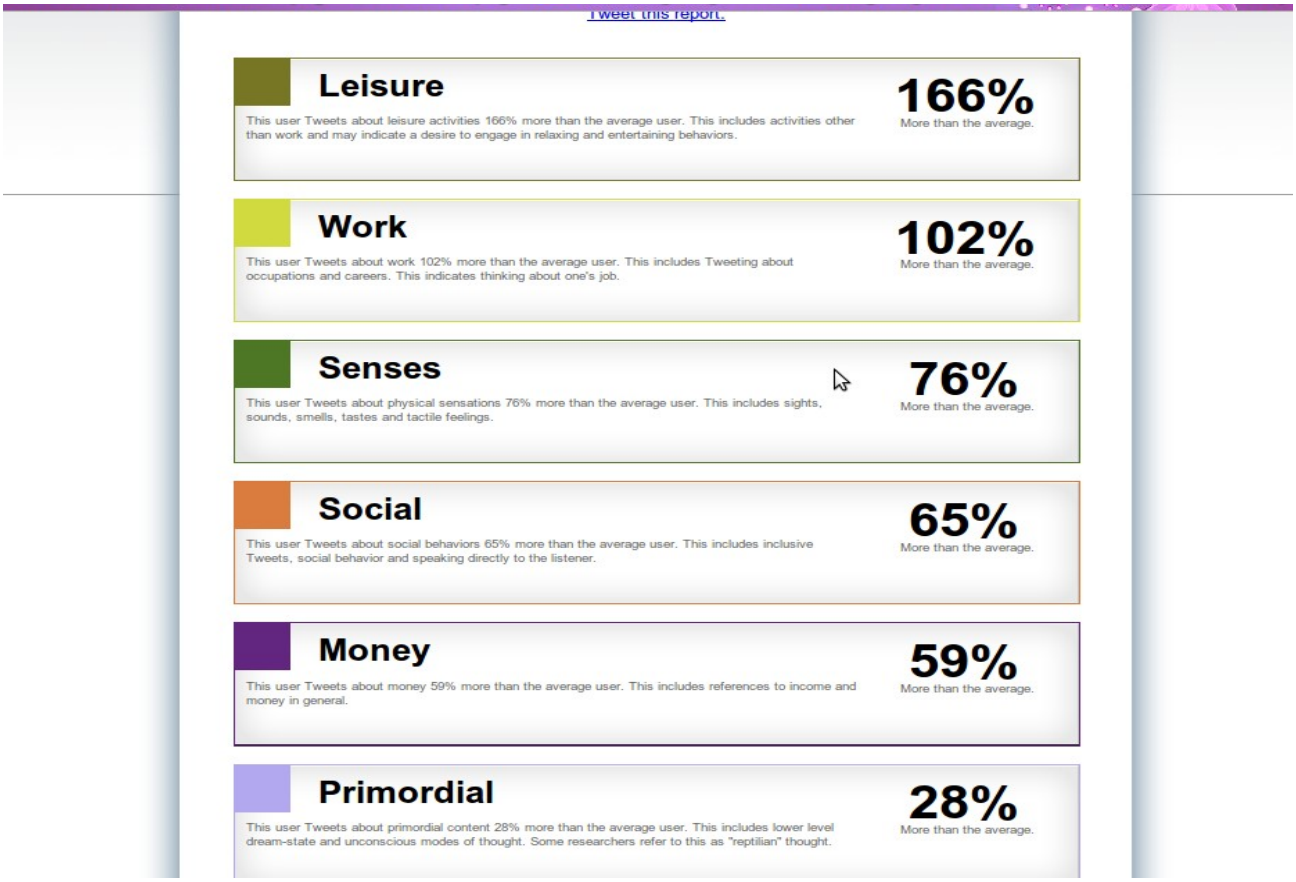


Figure 24: Twittergrader ou la valeur du profil

Autre approche TweetPsych[16] permet de déterminer le profil de l'utilisateur en sélectionnant différents thèmes qui caractériseront ces tweets.

15 http://twittergrader.com/
16http://tweetpsych.com/

Figure 25: TweetPsych, le "profiler" au service de la twittosphère



Il existe un grand nombre d'autres applications qui n'ont pas été présentées ici comme UKSnow (pour savoir où il neige au Royaume-Uni), WeFollow (un annuaire qui classe les personnes par tag), Twitterel (pour trouver des personnes à suivre en fonction de mots clés) et bien d'autres encore.

Certains travaux ont été réalisés ou sont en cours de réalisation au sein du LIRMM autour des problématiques liés aux tweets. Par exemple, une collaboration a été réalisée avec la société Web Report[17] pour développer un outil de veille stratégique pour détecter les informations avant même leur apparition dans les nouvelles des agences de presse. Dans une approche similaire à celle de Google qui a montré un lien entre les requêtes des internautes qui utilisent des termes liés à la grippe et le nombre de personnes présentant les symptômes de cette maladie [GINSBERG et al., 2009], le sujet de l'étude concernait la détection automatique de catastrophes à partir de ressources hétérogènes issues du web telles que les blogs, les tweets, les dépêches. L'objectif était de fouiller ces ressources pour détecter une catastrophe en temps réel.
Concrètement, l'objet de l'étude consistait à détecter automatiquement des signaux faibles à partir d'un ensemble de messages courts (tweet en français et en anglais).

17 http://www.webreport.fr/

L'analyse des données textuelles issues des tweets est un domaine de recherche actuel et de nombreuses propositions existent. Par exemple, dans [SAKAKI et al., 2010], les auteurs proposent d'analyser le contenu des tweets pour détecter en temps réel des alarmes lors d'apparitions de tremblement de terre. Les auteurs de TwitterMonitor [MATHIOUDAKIS et KOUDAS, 2010] présentent un système pour extraire automatiquement les tendances dans le flot des streams. Une approche assez similaire est proposée dans [BENHARDUS, 2010]. Cependant, à notre connaissance, la plupart des travaux existants proposent un traitement particulier des tweets et n'offrent pas d'outils généraux permettant au décideur, en fonction de ses besoins, de pouvoir manipuler l'information contenue dans les tweets. Ainsi, il n'existe que peu de travaux qui se soient intéressés à l'utilisation de cubes de données pour les tweets.

3.4 Bilan des solutions vis à vis du projet

Des applications récentes ont été proposées pour analyser l'information à partir des gros volumes de tweets produits au cours du temps, comme par exemple, le suivi de tendances, le repérage de buzz... Toutefois, il n'existe pas à notre connaissance d'approche exploitant leurs caractéristiques multi-dimensionnelles. De plus aucune des applications étudiées ne s'est focalisée précisément sur le domaine Médical. L'application twellow[18], qui propose un classement des tweets par catégorie, possède bien une catégorie "*Health*" mais elle ne propose ni de sous-catégorie suffisamment spécialisée pour répondre à notre besoin, ni d'approche multi-dimensionnelle. Nous souhaitons par exemple pouvoir connaître le nombre de tweets abordant le thématique du régime DUKAN en 2009 et 2010, repérer les tendances (par exemple, *quels sont les régimes qui sont de plus en plus utilisés ?*) ou encore pouvoir naviguer dans les données selon les types de maladie et pouvoir les croiser avec des informations spatio-temporelles.

Nous l'avons vu, la mise en place d'un entrepôt de données repose entre autres sur une bonne conception de son schéma, puisque c'est ce dernier qui va déterminer les possibilités d'analyse.
Les tweets sont associés à des méta-informations (l'utilisateur, la date, la géolocalisation...) , ils peuvent donc être représentés de manière muti-dimensionnelle en prenant en compte l'ensemble de ces méta-informations.
Nous souhaitons y associer une hiérarchie sur les mots spécifiques à notre domaine d'application pour permettre la contextualisation des tweets.

18http://www.twellow.com/

Dans ce chapitre nous avons évoqué les principaux concepts liés à la conception et à l'exploitation des entrepôts de données et présenté le service Twitter. La phase de conception va nous permettre de définir le modèle d'un entrepôt de données dédié à l'analyse des tweets.

4 Chapitre 4 : Conception d'un entrepôt de données dédié aux tweets

Comme nous l'avons vu au chapitre précédent, nous souhaitons mettre en place un outil d'analyse de tweets à partir des informations disponibles. Nous souhaitons aussi contextualiser les tweets en utilisant un thésaurus spécialisé du domaine médical. Dans ce quatrième chapitre, nous nous intéressons pour commencer aux informations et méta-informations disponibles, d'abord dans un tweet (4.1), puis dans le thésaurus spécialisé (4.2). L'analyse de ces données disponibles fait apparaître un certains nombre de verrous qui sont présentés dans la section 4.3. A partir de ces éléments, nous discutons du modèle conceptuel retenu pour notre application (4.4).

4.1 Données "tweet"

Twitter encourage la réutilisation du contenu grâce aux API présentées dans la section 3.2. Les données utilisées sont les données publiques disponibles via ces interfaces de programmation. Une personne ne souhaitant pas rendre publiques ses messages peut choisir de les rendre privés, visibles uniquement après validation d'une demande d'ajout à la liste des abonnés. Ce n'est pas le mode par défaut de Twitter et il n'est pas vraiment dans l'esprit de ce service (99% des utilisateurs rendent leurs messages publiques). Twitter recommande lui-même aux utilisateurs de placer leurs messages dans le domaine public[19].
En plus du message, chaque tweet est porteur d'un ensemble de meta-informations. Certaines concernent le tweet :

- identifiant unique
- date et heure de création du messages
- provenance du tweet
- si le tweet est une réponse à un autre, identifiant du tweet initial
- si le tweet est une réponse à un autre, identifiant de l'utilisateur initial
- nombre de fois où le tweet a été retweeté

D'autre se rapporte à l'utilisateur :

- identifiant unique de l'utilisateur
- nom

19 http://twitter.com/tos

- pseudonyme
- description renseignée par l'utilisateur
- emplacement géographique
- fuseau horaire
- date et heure de création du profil,
- image de l'avatar sous forme de lien
- nombre de messages, *de followers, de followings*

Une troisième catégorie concerne enfin les paramètres du compte Twitter :

- indication si le compte est privé ou public
- couleur de fond d'écran,
- couleur de police de caractère,
- image de fond d'écran sous forme de lien

De manière à mieux évaluer la composition réelle d'un tweet, nous avons réalisé en ensemble d'expérimentations.

Nous définissons tout d'abord la notion de "mots utiles" comme l'ensemble des noms, des verbes et des adjectifs auxquels nous retirons les "mots vides". Les "mots vides" (*stop words* en anglais) sont des mots non significatifs figurant dans un texte (par exemple avoir, être, le, la ou les). Ils sont trop communs pour être considérés comme des éléments discriminants et ils sont retirés des analyses. La liste des mots vides utilisés dans le projet est donnée en annexe 2.

Les noms, verbes et adjectifs sont identifiés grâce à des étiqueteurs grammaticaux. L'étiquetage grammatical est le processus qui consiste à associer aux mots d'un texte leur fonction grammaticale, grâce à leur définition et leur contexte (c'est-à-dire leur relation avec les mots adjacents dans un terme, une phrase ou un paragraphe).

Nous constatons une moyenne de 8 mots utiles par tweet repartie selon la distribution présentée dans le tableau 4, répartition constatée sur un échantillon de 30599 tweets de langue anglaise à partir de l'étiqueteur grammatical TreeTagger qui sera présenté ultérieurement dans ce document. Les tableaux 5, 6 et 7 mettent en évidence la composition moyenne d'un tweet en lien avec notre domaine d'analyse : 5 noms, 2 verbes et 1 adjectif.

Nombre de mots utiles	Pourcentage
1 mot	0,78%
2 ou 3 mots	7,35%
4 ou 5 mots	16,93%
6 ou 7 mots	20,74%
8 mots	17,63%
9 ou 10 mots	20,24%
11 ou 12 mots	11,19%
plus de 12 mots	5,15%

Tableau 4: Répartition du nombre de mots utiles

Nombre de noms utiles	Pourcentage
1 nom	2,67%
2 ou 3 noms	19,02%
4 ou 5 noms	28,91%
6 ou 7 noms	32,19%
8 noms	7,25%
9 ou 10 noms	7,78%
11 ou 12 noms	1,65%
plus de 12 noms	0,53%

Tableau 5: Répartition du nombre de noms utiles

Nombre de verbes utiles	Pourcentage
1 verbe	43,91%
2 ou 3 verbes	52,74%
4 ou 5 verbes	3,29%
6 ou 7 verbes	0,06%

Tableau 6: Répartition du nombre de verbes utiles

Nombre d'adjectifs utiles	Pourcentage
1 adjectif	55,60%
2 ou 3 adjectifs	39,44%
4 ou 5 adjectifs	4,66%
6 ou 7 adjectifs	0,18%
8 noms ou adjectifs	0,12%

Tableau 7: Répartition du nombre d'adjectifs utiles

Nous souhaitons ajouter une hiérarchie sur les mots. Il n'existe pas de hiérarchie logique comme il peut en exister sur une hiérarchie temporelle. Le but du projet étant lié à l'analyse des données dans un contexte médical, nous avons allons utilisé les hiérarchies disponibles de ce domaine.

4.2 Contexte : le domaine médical

Nous avons retenu les hiérarchies du MeSH[20] (Medical Subject Headings). Le MeSH de la National Library of Medicine aux Etats-Unis[21] est un thésaurus spécialisé dans le domaine médical disponible gratuitement et libre d'utilisation. Il est composé d'un ensemble de termes correspondant à des concepts du domaine. Ces derniers sont associés à une structure hiérarchique de 13 niveaux maximum pour permettre une classification des concepts.

En 2011, 52 561 concepts sont disponibles dans MeSH. Au niveau le plus général de la hiérarchie on trouve 16 concepts très généraux :

- *"analytical, diagnostic and therapeutic techniques and equipment"*
- *"anatomy"*
- *"anthropology, education, sociology and social phenomena"*
- *"chemicals and drugs"*
- *"disciplines and occupations"*
- *"diseases"*
- *"geographicals"*
- *"health care"*
- *"humanities"*
- *"information science"*
- *"named groups"*

20 http://www.nlm.nih.gov/mesh/MBrowser.html
21 http://www.nlm.nih.gov

- *"organisms"*
- *"phenomena and processes"*
- *"psychiatry and psychology"*
- *"publication characteristics"*
- *"technology, industry, agriculture"*

Au niveaux les plus bas se trouvent des concepts tels que "*herpesvirus 4, human*", "*influenza a virus, h7n7 subtype*", "*leukomia*","*turkey*".

Une analyse fine du MeSH met en évidence quatre spécificités propres à ce thésaurus.
Tout d'abord, le MeSH est un arbre aux hiérarchies déséquilibrées comme l'illustre le tableau 8.

Niveau	Nombre de concepts du niveau
niv0	16
niv1	117
niv2	1664
niv3	6907
niv4	13252
niv5	12712
niv6	8605
niv7	5013
niv8	2890
niv9	1040
niv10	239
niv11	92
niv12	14

Tableau 8: Nombre de concepts par niveau hiérarchique du MeSH

De même les différents concepts du thésaurus ne sont pas homogènes dans leur déclinaison. Par exemple on retrouve des concepts écrits au pluriel *("organisms", "humanities"*), d'autres au singulier (*"anatomy", "noise"*). Certains termes apparaissent même au singulier et au pluriel dans des concepts différents (*"mice, congenic" ,"mammary tumor virus, mouse"*). Une standardisation est nécessaire.

Ensuite 34 884 termes sont des termes composés (c'est-à-dire qui comportent au moins un séparateur espace) ce qui impose un processus de détermination des syntagmes. Les syntagmes ne sont pas des mots composés standards mais ce sont des groupes de mots qui forment un sens, nous pouvons citer comme exemple les concepts *"Eye Diseases", "Fish Diseases", "Political Systems" ou "Salt Gland"*. Enfin certains concepts comportent plusieurs termes séparés par des virgules. Par exemple le concept "*influenza"* n'est pas présent en lui même mais peut faire référence à 59 termes parmi lesquels :

- *"Influenza, Human"*
- *"Influenza A Virus, H1N1 Subtype"*
- *"Influenza A Virus, H2N2 Subtype"*
- *"Influenza A Virus, H3N2 Subtype"*

La probabilité de retrouver les termes "*Influenza, Human*" dans cette ordre dans un tweet est limitée alors que le terme "*human*" précise simplement le concept "*influenza*".

4.3 Verrous liés aux données

En analysant les données disponibles dans le tweet, nous avons aussi été confrontés à certaines contraintes qu'il nous faut prendre en compte dans la mise en place notre modèle.

4.3.1 Gestion des données textuelles

Tout d'abord, un tweet est avant tout un message écrit par un être humain dans un langage courant généralement peu soigné. La limite des 140 caractères ne favorise pas une rédaction de qualité. Nous allons donc être confronté à tout ce que le langage de type SMS peut apporter comme contrainte dans le cadre d'une analyse textuelle par exemple :

- des abréviations spécifiques (lol, omg),
- des fautes volontaires ou involontaires (orthographes, grammaire et conjugaison),
- des émoticones (smiley),
- des signes de ponctuations inappropriés (enchaînement de point d'exclamation par exemple !!!!!!!!!)
- des mots inventés dérivés d'expressions réelles (superrrrrrrrrrrr).

De plus un tweet peut être rédigé dans n'importe quelle langue, plusieurs langues pouvant même être utilisées au sein du même message.

Enfin comme il a été présenté dans la section 3.2, un ensemble d'éléments spécifiques à la

Twittosphère sont portés par le message (@destinataire, tag, Re-Tweet).
Toutes ces spécificités doivent être prises en compte dans notre conception.

4.3.2 Gestion de la localisation

Comme expliqué dans la section 3.4, nous souhaitons mettre en place une analyse géographique. La notion de localisation dans Twitter n'est cependant pas normalisée. Elle peut être :

- soit définie automatiquement au travers d'un point d'accès Internet
- soit calculée automatiquement avec des coordonnées si une connexion depuis un téléphone mobile est utilisée
- soit renseigné manuellement par l'utilisateur. Dans ce cas il s'agit de texte libre ce qui laisse la porte ouverte à un nombre infini de propositions de localisation, du plus sérieux ("Paris, France","USA") au plus surprenant ("Auprès d"une chèvre" (sic), "Planète Terre").
- Il est aussi possible qu'aucune information (utile ou non) ne soit transmise.

Il existe une autre information permettant de déterminer la localisation du tweet. Twitter fournit aussi le fuseau horaire de l'utilisateur. Cette indication est définie automatiquement par le service mais peut être modifiée manuellement par l'utilisateur selon une liste fermée de choix. Le fuseau horaire est représenté par une ville (une capitale généralement) par exemple Paris, Lima, Quito ou encore Rome. Ainsi un utilisateur français sera rattaché au fuseau horaire Paris et son homologue italien au fuseau horaire Rome bien que le décalage horaire par rapport au méridien de Greenwich soit le même (+1 heure) pour la France et l'Italie. Cette information peut être qualifiée de fiable pour déterminer le pays d’où le tweet a été émis.

4.3.3 Uniformisation du MeSH

Enfin, comme expliqué précédemment dans la section 4.2, nous avons décidé d'utiliser le thésaurus du MeSH comme hiérarchie de mot. Cela impose une uniformisation du thésaurus selon les axes définis ci-dessous.
Premièrement, nous décidons d'équilibrer les hiérarchies du MeSH en reportant sur les niveaux inférieurs la valeur du dernier niveau renseigné. Ainsi chaque branche de notre arbre se compose de 13 niveaux.
Deuxièmement, nous décidons de lemmatiser l'ensemble des termes du MeSH afin de s'affranchir des différentes déclinaisons possible d'un concept. La lemmatisation regroupe les différentes formes

que peut revêtir un mot. La lemmatisation d'une forme d'un mot consiste à ne retenir que sa forme canonique soit pour un verbe, son infinitif, pour les autres mots, le mot au masculin singulier.
Troisièmement une recherche spécifique des syntagmes du MeSH doit être réalisée.
Quatrièmement, nous considérons le(s) premier(s) terme(s) avant la virgule comme le terme principal du concept et les autres termes comme des concepts servant à préciser le contexte du terme principal.

4.3.4 Gestion de la désambiguïsation

L'un des problèmes principaux avec l'utilisation de ce thésaurus comme hiérarchie de mots est que différents termes peuvent apparaître à plusieurs niveaux de la hiérarchie. Parmi les 52 561 termes, on comptabilise 22 922 termes uniques. Cette ambiguïté pose le problème de l'utilisation des opérateurs de type *roll-up* ou *drill-down* pour naviguer dans le cube.

Par exemple le terme *Pharyngitis* fait référence à :

- *Diseases >> Stomatognathic Diseases >> Pharyngeal Diseases >>Pharyngitis*
- *Diseases >> Respiratory Tract Diseases >> Respiratory Tract Infections >> Pharyngitis*
- *Diseases >> Otorhinolaryngologic Diseases >> Pharyngeal Diseases >>Pharyngitis*

Ces contraintes sont traitées dans les chapitres suivants.
Nous possédons maintenant les informations nécessaires à la constitution d'un modèle conceptuel.

4.4 Modèle conceptuel

4.4.1 Solution générique

Nous avons choisi un modèle avec trois dimensions (une dimension temporelle, une dimension géographique et une dimension mot).
Nous adoptons un schéma en étoile (c.f. Figure 26) puisque il nous paraît cohérent avec notre approche et une approche matérialisée puisque nous ne pouvons accéder aux bases de données de Twitter et qu'une normalisation des données est nécessaire (c.f. section 4.3).

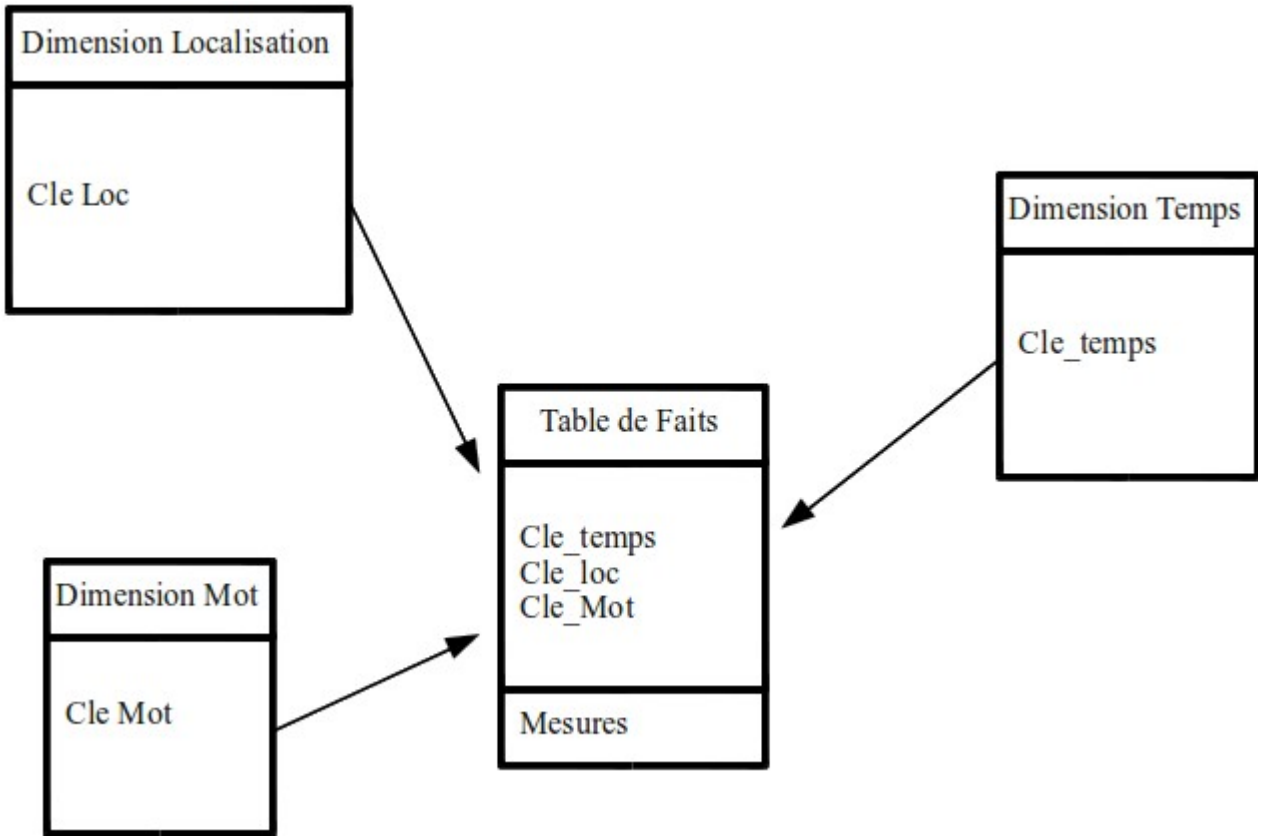


Figure 26: Modèle conceptuel générique

Nous instancions ce modèle générique en l'adaptant à notre domaine d'application.

4.4.2 Solution adaptée au domaine d'application

Nous retenons pour la dimension liée au Temps une dimension classique pour les données temporelles (Jour > Mois > Semestre > Années). Pour la dimension liée au mot du MeSH, nous retenons les 13 niveaux hiérarchiques du MeSH (Niveau 0 > Niveau 1 > … > Niveau 12). Enfin pour la dimension liée à la localisation, nous faisons le choix de retenir le découpage administratif suivant Ville > État > Pays, la notion d'État variant selon le pays (par exemple, une région en France, une province au Canada, un état aux Etats-Unis, ...). La représentation de ce modèle conceptuel est donnée dans la figure 27.

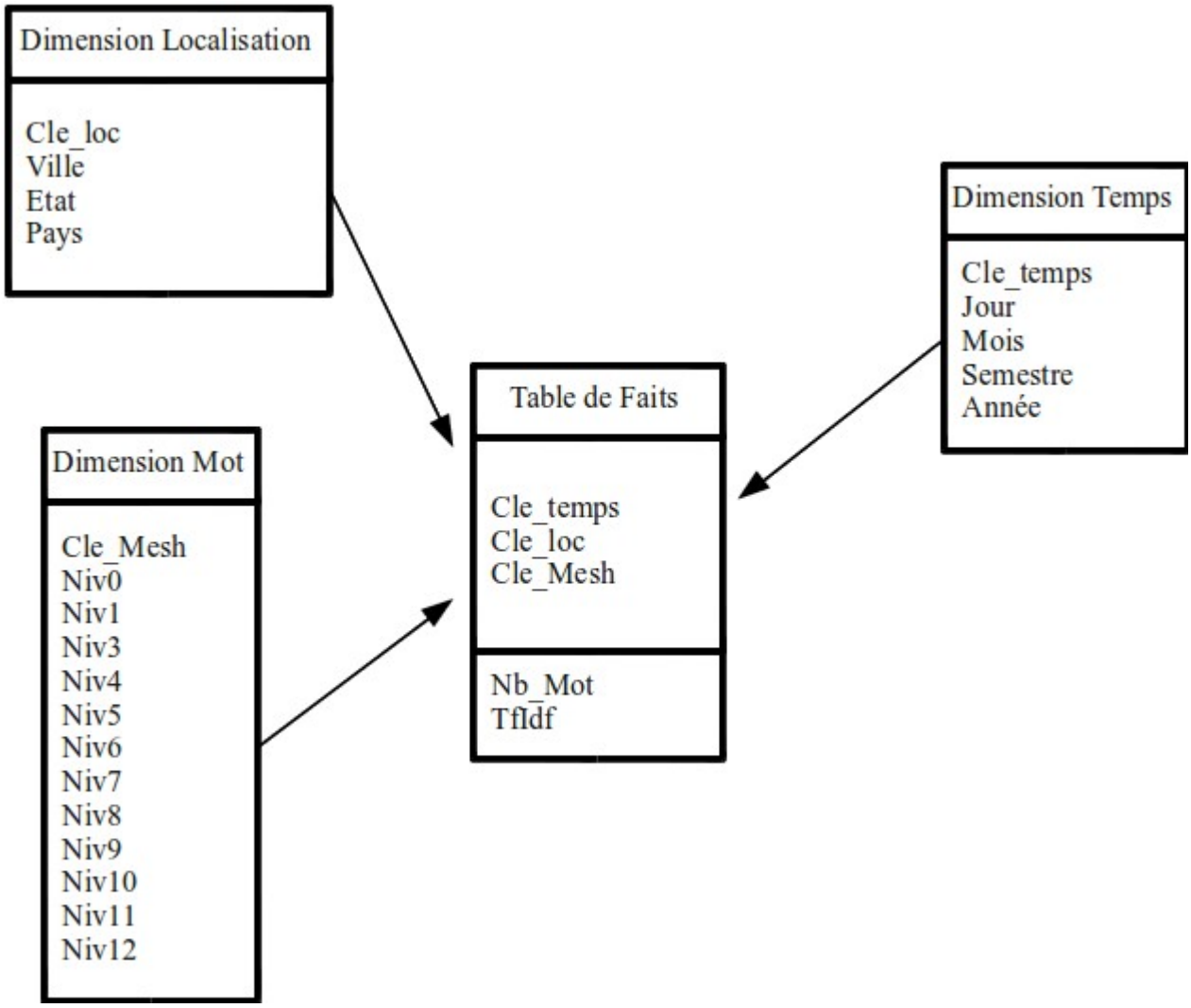


Figure 27: Modèle conceptuel de données retenu

Dans la table de faits, différentes mesures peuvent être utilisées. Il peut s'agir de mesures simples comme le nombre de tweets ou plus complexe comme le TF-IDF.

Le TF-IDF (*Term Frequency-Inverse Document Frequency*) est une méthode de pondération souvent utilisée en Recherche d'Information et en particulier en fouille de textes. Cette mesure statistique permet d'évaluer l'importance d'un terme contenu dans un document, relativement à une collection ou un corpus. Le poids augmente proportionnellement au nombre d'occurrences du mot dans le document. Il varie également en fonction de la fréquence du mot dans le corpus. Il est calculé à partir de la fréquence d'un terme et de la fréquence inverse de document.

La fréquence d'un terme (*term frequency*) est le nombre d'occurrences de ce terme dans le document considéré, normalisée par la somme des nombres d'occurrences de tous les termes du document. Le nombre d'occurrence peut rendre compte de "l'importance" d'un terme dans un document. La normalisation du nombre d'occurrences d'un terme rend possible la comparaison de deux documents

de longueurs différentes.

Le TF correspond au nombre d'occurrences de ce terme dans le document considéré. Ainsi, pour le document d_j et le terme t_i, la fréquence du terme dans le document est donnée par l'équation suivante :

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Avec $n_{i,j}$ le nombre d'occurrences du terme t_i dans d_j. Le dénominateur correspond au nombre d'occurrences de tous les mots dans le document d_j.

La fréquence inverse de document (*inverse document frequency*) est une mesure de l'importance du terme dans l'ensemble du corpus. Dans le schéma TF-IDF, elle vise à donner un poids plus important aux termes les moins fréquents, considérés comme plus discriminants. Elle consiste à calculer le logarithme de l'inverse de la proportion de documents du corpus qui contiennent le terme.

$$IDF_i = \log_2\left(\frac{|D|}{|\{d_j : t_i \in d_j\}|}\right)$$

Avec $|D|$ représentant le nombre total de documents dans le corpus et $|\{d_j : t_i \in d_j\}|$ le nombre de documents dans lesquels le terme t_i apparaît.

Le TF-IDF s'obtient en multipliant les deux mesures.

$$TF-IDF_{i,j} = TF_{i,j} \times IDF_i$$

D'autres mesures peuvent également être prises en compte. Par exemple, la mesure d'agrégation décrite dans [BRINGAY et al., 2011a] est une adaptation du TF-IDF classique qui permet de mettre en évidence les mots discriminants par niveau hiérarchique. La formule du TF est identique mais la formule de l'IDF devient:

$$IDF_i^k = \log_2\left(\frac{|E^k|}{|\{e_j^k : t_i \in e_j^k\}|}\right)$$

$|E^k|$ représente le nombre total d'éléments de type k (par exemple, k = {Ville ,Région , Pays}) qui correspond au niveau de la hiérarchie que le décideur souhaite agréger.

$|\{e_j^k : t_i \in e_j^k\}|$ est relatif au nombre d'éléments de type k dans lequel le terme t_i apparaît.

Cette dernière mesure permet d'identifier les mots les plus significatifs selon le niveau des hiérarchies du cube (dans notre exemple via la dimension localisation).

En effet les mots les plus significatifs d'une ville, ne seront pas forcement les mots les plus

significatifs de la région. Par exemple le tableau 9 présente les 12 mots les plus significatifs (selon le TF-IDF adaptatif) au cours du mois de janvier 2011 selon que l'on considère la ville (Chicago), l'état (l'Illinois) ou le pays (les États-Unis).

Etats Unis	Illinois	Chicago
wart	*risk*	*risk*
pneumonia	*vaccination*	*wart*
vaccination	*wart*	*pneumonia*
risk	*pneumonia*	*wood*
lymphoma	*wood*	*colonoscopy*
common cold	*colonoscopy*	*x-ray*
disease	*x-ray*	*death*
meningitis	*encephalitis*	*school*
infection	*death*	*vaccination*
vaccine	*school*	*eye infection*
life eye	*infection*	*patient*
hepatitis	*man*	*russia*

Tableau 9: Illustration TF-IDF adaptatif selon la dimension géographique

Ainsi dans cette partie, nous avons présenté tour à tour les données et les contraintes et nous avons défini le modèle conceptuel.

Nous pouvons maintenant proposer une solution en tenant compte des analyses réalisées dans ce chapitre.

5 Chapitre 5 : Architecture technique et mise en œuvre

La solution que nous proposons repose sur les éléments présentés dans les chapitres 3 et 4.

Dans ce chapitre nous présentons tout d'abord les choix techniques de la solution (5.1). Nous proposons ensuite une approche qui se décompose en trois phases distinctes pour permettre à terme d'alimenter notre cube de données depuis Twitter (c.f. Figure 28). La première étape concerne le processus d'acquisition des tweets et d'acquisition des utilisateurs depuis Twitter (5.2), la seconde porte sur la gestion des contraintes mises en avant lors de la phase de conception afin de préparer les données pour l'alimentation du cube (5.3). Enfin la troisième étape se concentre sur l'alimentation du cube (5.4).

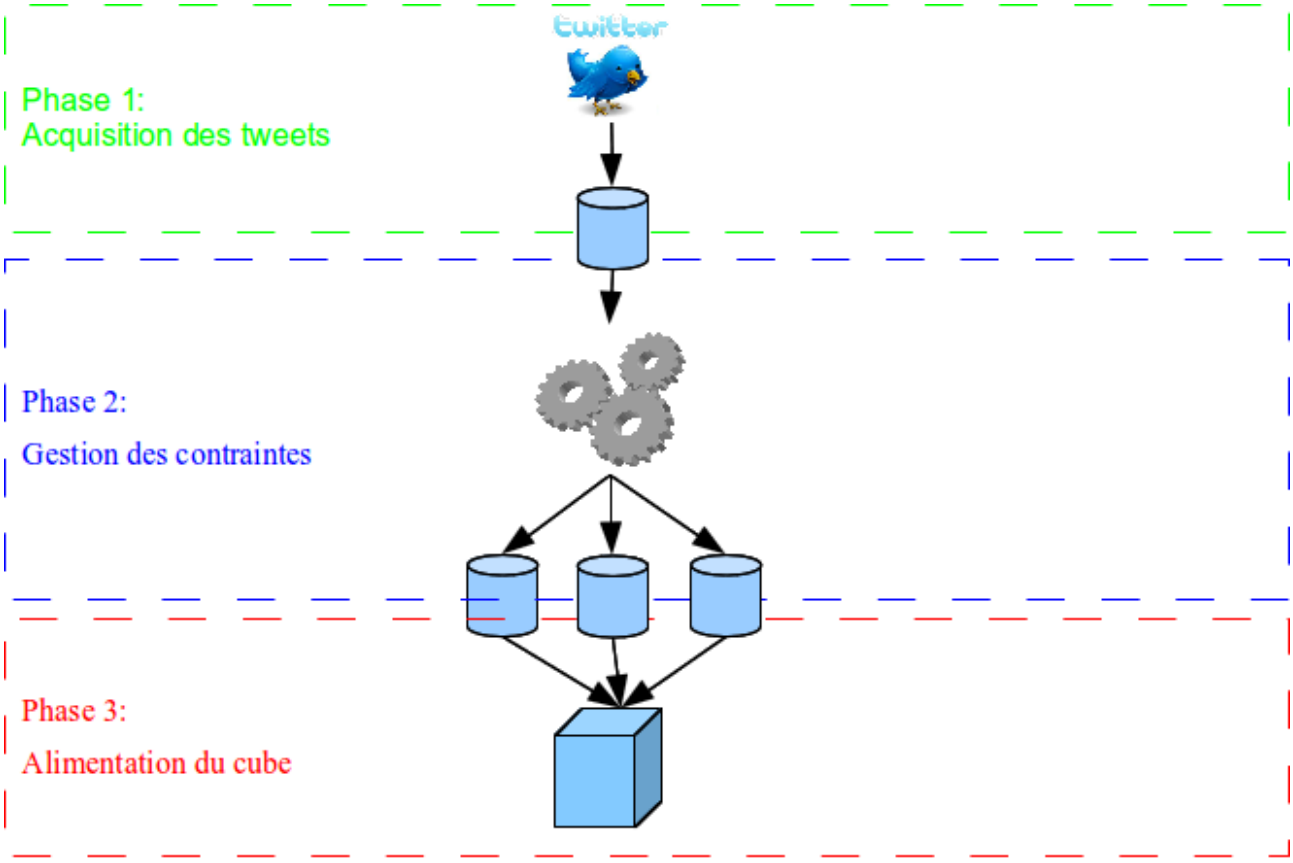


Figure 28: Schéma global de la solution proposée

Chacune de ces trois étapes est indépendante des autres. Leur périodicité et leur mode de fonctionnement sont aussi différents.

5.1 Présentation des choix techniques

Concernant le langage de programmation, le langage PERL a été adopté. PERL est un langage interprété, polyvalent, et particulièrement adapté au traitement et à la manipulation de données textuelles, notamment du fait de l'intégration des expressions régulières dans la syntaxe même du langage. Ce choix est un élément important pour la maintenance et la ré-utilisabilité à moyen terme de la solution.

Concernant la base de données utilisées nous avons opté, en accord avec les équipes, pour une solution reposant sur des solutions libres.
La base de données Postgresql est souvent présentée comme un des meilleurs SGBD existants, notamment en termes de stockage et d'insertions massives de données. Néanmoins, pour éviter de lier notre solution à un seul SGBD, nous avons décidé de ne pas utiliser de *trigger* puisque le langage utilisé est propre à chaque produit (PL/pgSQL pour PostgreSql, PL/SQL pour Oracle, Transact-SQL pour Sql Server). Postgresql ne permettant pas l'analyse multidimensionnelle, nous avons retenu Mondrian pour y ajouter les fonctionnalités OLAP.
Mondrian est un moteur OLAP écrit en Java qui permet la conception, la publication et le requêtage de cubes multidimensionnels (c.f. Figure 29). Mondrian permet l'exécution de requêtes en langage MDX (langage de requête adapté à l'OLAP) sur des entrepôts de données s'appuyant sur des bases de données relationnelles. Mondrian est considéré comme la référence open source des systèmes ROLAP.

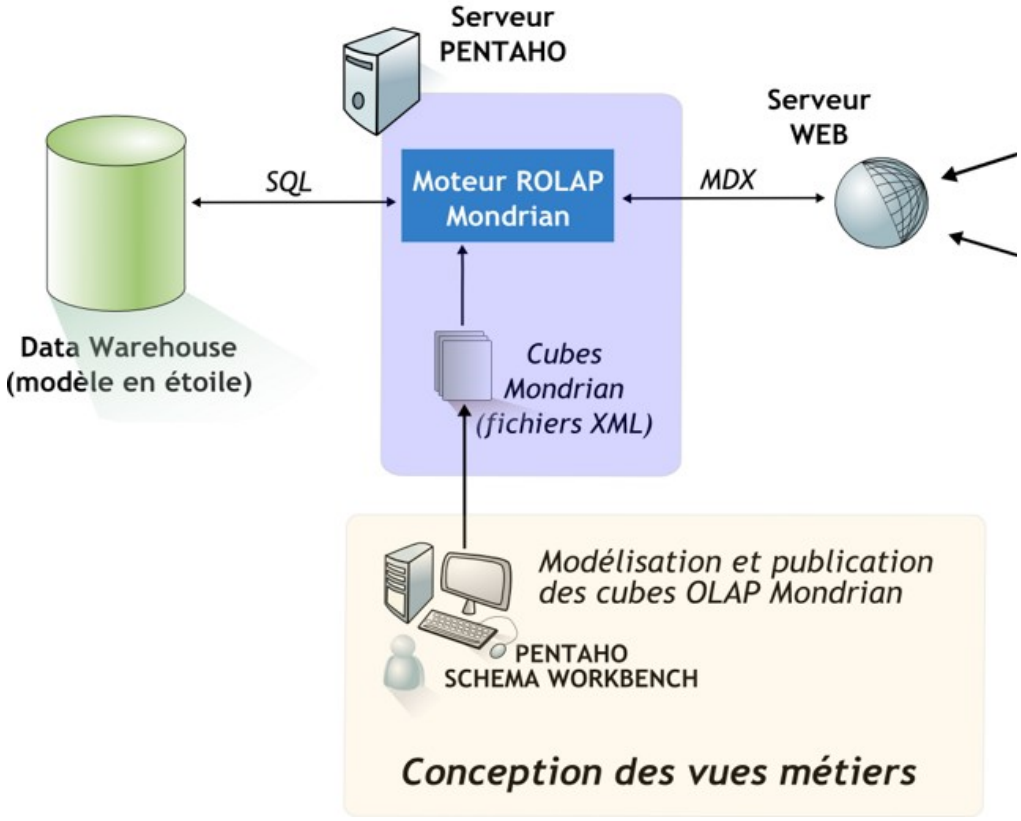
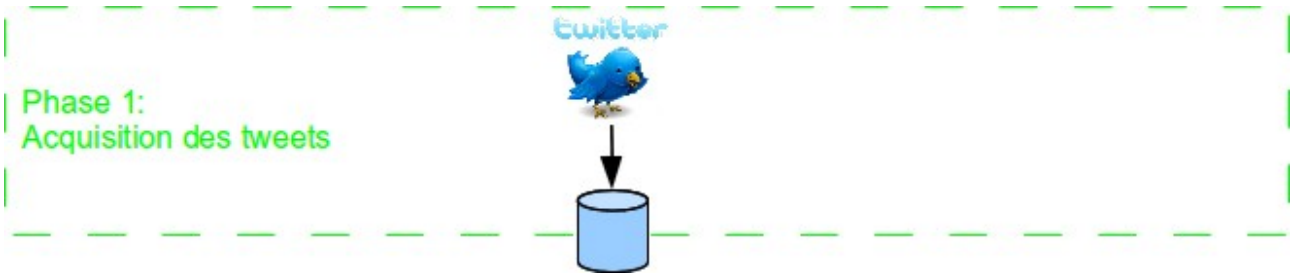


Figure 29: L'architecture Mondrian en image (source: http://www.osbi.fr/)

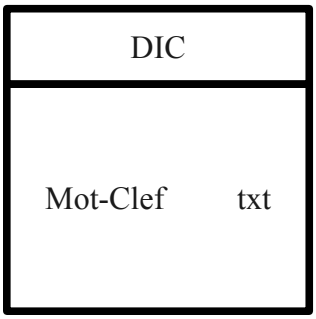
5.2 Phase 1: Acquisition des tweets



Les données sources que nous traitons sont issues de Twitter (c.f. section 3.2). Pour récupérer les informations nous utilisons les API fournies par Twitter. Deux des API à disposition permettent d'acquérir des tweets en ciblant des mots clef précis, SEARCH API et STREAM API. SEARCH API présente l'avantage de pouvoir obtenir des données anciennes mais elle ne peut remonter au

delà de 10 jours. De plus, elle se doit d'être relancée continuellement pour récupérer les derniers tweets. Cependant Twitter n'autorise qu'un certain nombre (non communiqué[22]) de connexion à l'API depuis un même compte sur une période donnée. Nous avons atteint cette limite lors de nos tests et nous avons été confronté à des suspensions temporaires de compte.

L'API STREAM, en revanche, est développée pour recevoir les tweets dans un flux continu plutôt que de faire des appels réguliers aux autres API. Cela a deux avantages : le premier étant d'obtenir les derniers tweets presque instantanément après leur publication, le second de libérer des ressources pour les autres API permettant ainsi de meilleures performances pour la plupart des clients. Elle impose simplement d'être connecté en permanence car il n'est pas possible de récupérer un tweet publié dans le passé.
Nous filtrons les tweets récupérés à partir d'une liste de mot-clef, liste limitée à 200 éléments par l'API. Nous prenons le parti de stocker les mots-clef dans une relation *(DIC)* pour faciliter la gestion de cette liste. Cette relation est simplement constituée d'un champ texte qui est aussi la clef primaire de la table pour garantir l'existence et l'unicité du mot-clef.



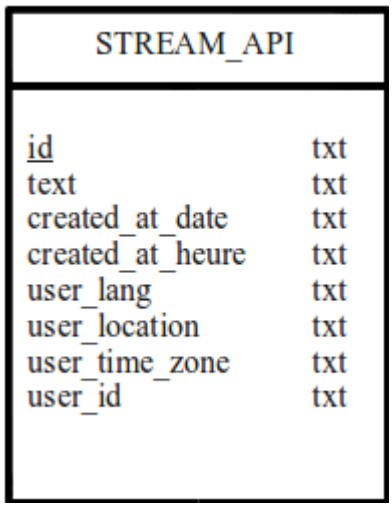
Relation 1: DIC

Champs	Commentaires
Mot-Clef	Mot cherché dans les tweets

Description 1: Relation DIC

En sortie, le traitement alimente deux relations, une contenant les tweets et l'autre les utilisateurs. Les tweets sont stockés à la volé dans la relation *STREAM_API* dont la clef primaire est constituée de l'identifiant du tweet (champ *id*).

22 http://dev.twitter.com/pages/rate-limiting

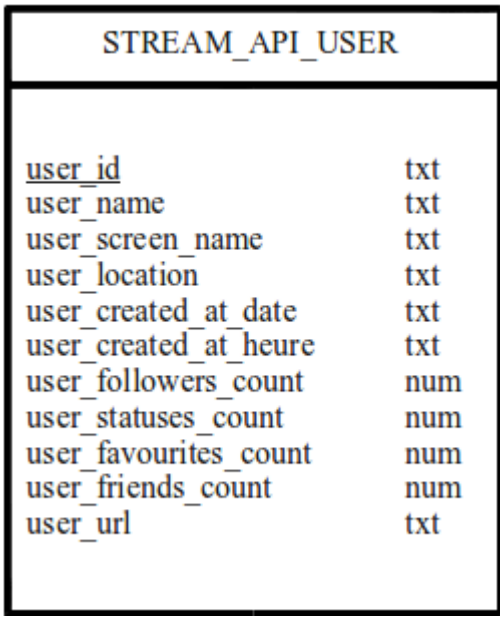


Relation 2: STREAM_API

Champs	Contenu
id	Identifiant du tweet
text	Tweet lui même (message)
created_at_date	Date de création du tweet
created_at_heure	Heure de création du tweet
user_lang	Langue définie par l'utilisateur
user_location	Localisation de l'utilisateur au moment où le tweet est écrit
user_time_zone	Fuseau horaire défini par l'utilisateur
user_id	Identifiant de l'utilisateur

Description 2: Relation STREAM_API

Les informations relatives aux utilisateurs sont portées dans chaque tweet. Afin de limiter le volume de la base, nous décidons de ne stocker que les dernières informations relatives à l'utilisateur avec un mécanisme de mise à jour (selon une logique de suppression puis d'insertion plutôt que de mise à jour pour des raisons de performance). La clef primaire de cette relation *STREAM_API_USER* est constituée de l'identifiant de l'utilisateur (champ *user_id*).



Relation 3: STREAM_API_USER

Champs	Contenu
user_id	Identifiant de l'utilisateur
user_name	Nom de l'utilisateur
user_screen_name	Pseudonyme de l'utilisateur
user_location	Dernière localisation de l'utilisateur
user_created_at_date	Date de création de l'utilisateur
user_created_at_heure	Heure de création de l'utilisateur
user_followers_count	Statistiques: Nombre de followers
user_statuses_count	Statistiques: Nombre de tweets
user_favourites_count	Statistiques: Nombre de favoris
user_friends_count	Statistiques: Nombre d'amis
user_url	Lien Web vers le compte Twitter

Description 3: Relation STREAM_API_USER

Toutes les données fournies par l'API ne sont pas intégrées, seules celles pertinentes et renseignées régulièrement ont été conservées. Par exemple les informations relatives à l'aspect esthétique du compte (avatar, couleur de fond d'écran, ...) , bien que pouvant présenter un intérêt dans un autre contexte, n'ont pas été retenues dans notre approche.

Dans cette phase les traitements annexes ont été limités au maximum afin de ne pas perturber le flux de données en provenance de Twitter et pouvoir traiter un grand volume de tweets.

Parallèlement aux relations *STREAM_API* et *STREAM_API_USER*, deux relations temporaires sont alimentées afin de préparer les données à traiter lors de la seconde phase. Il s'agit des relations *TWEET_A_TRAITER_ANN et TWEET_A_TRAITER_LOC* qui seront présentées dans la section suivante.

La figure 30 récapitule la phase d'acquisition.

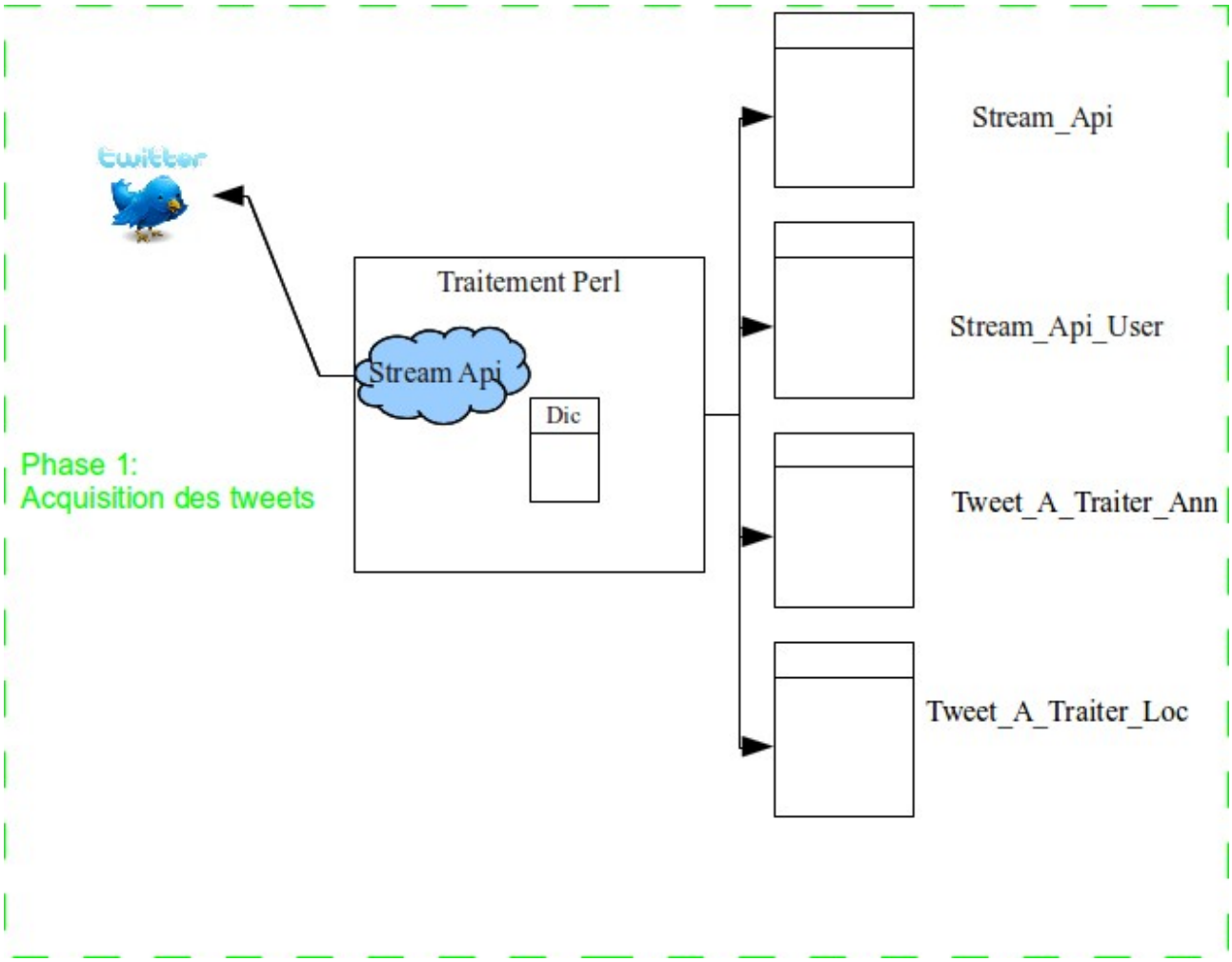
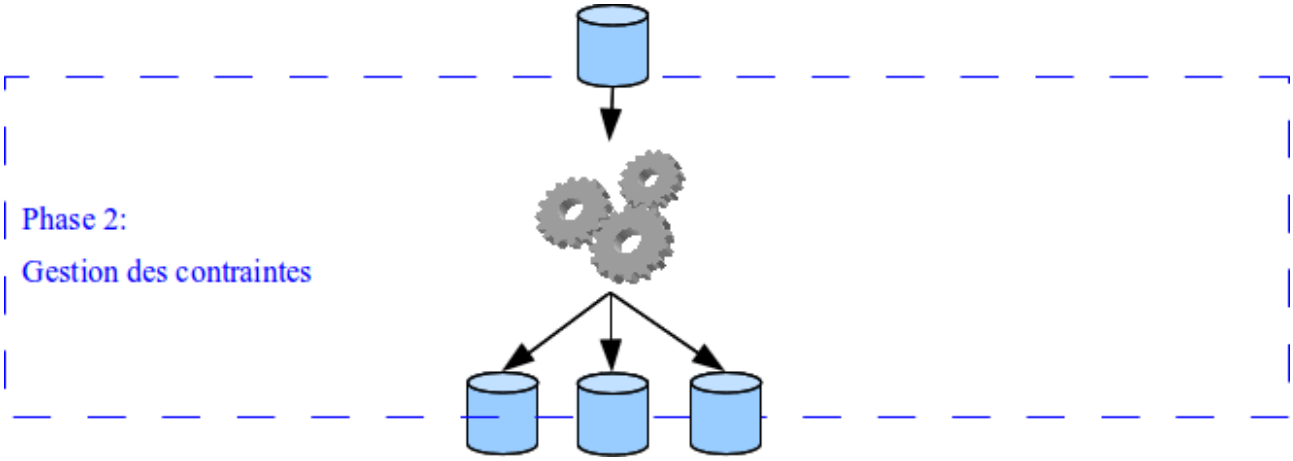


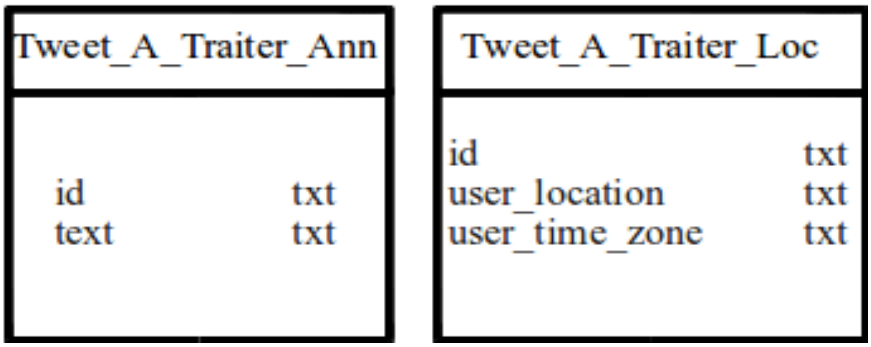
Figure 30: Phase d'acquisition des tweets

A l'issue de cette phase, nous pouvons intégrer un flux de Tweet. Comme nous l'avons vu au chapitre 4 lors de la conception de notre solution, certaines informations doivent être affinées avant de pouvoir intégrer les données dans le cube.

5.3 Phase 2: Gestion des contraintes



Les contraintes sont gérées indépendamment les unes des autres. Ces processus traitent les tweets sous forme de lots. Pendant qu'un ensemble de tweets est traité par un processus, un autre lot se constitue lors de la phase d'acquisition des tweets. Une fois le premier ensemble traité, le processus traite alors le lot nouvellement constitué et pendant ce temps un nouveau lot se constitue durant la phase d'acquisition. Elle est gérée au travers des relations *TWEET_A_TRAITER_ANN* et *TWEET_A_TRAITER_LOC* pour la normalisation du texte et la normalisation de la localisation.



Relation 4: TWEET_A_TRAITER_ANN et TWEET_A_TRAITER_LOC

Champs	Commentaires
id	Identifiant du tweet
text	Message contenu dans le tweet
user_location	Contenu du champ user_location retourné par l'API Twitter
user_time_zone	Contenu du champ user_time_zone retourné par l'API Twitter

Description 4: Relations TWEET_A_TRAITER_ANN et TWEET_A_TRAITER_LOC

Cette mécanique présente l'avantage de ne jamais perturber l'absorption du flux lors de la phase d'acquisition des tweets, et ce même en cas de problème rencontré, puisque les systèmes sont indépendants.

La gestion des tweets à désambiguïser fonctionne sur le même principe à la différence que la relation *TWEET_A_TRAITER_DESAMB* (qui contient le mot en attente de désambiguïsation) est alimentée lors du traitement de la normalisation du texte et non lors de la phase d'acquisition des tweets. Elle fera l'objet d'une présentation dans la section suivante.
Dans cette partie nous étudions chacun des processus mis en place pour lever les ambiguïtés en commençant par la normalisation du tweet.

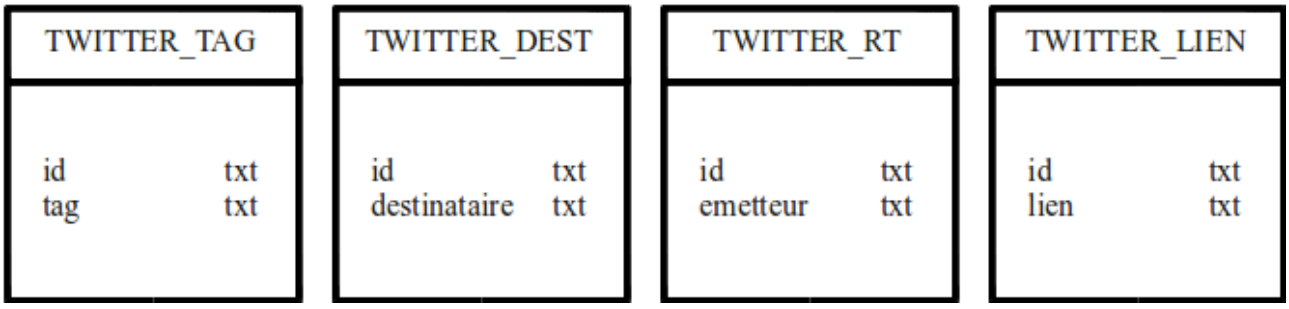
5.3.1 Normalisation du texte du tweet

5.3.1.1 Nettoyage du tweet

Au delà des problématiques liées à l'analyse de texte libre (abréviations, smiley, fautes d'orthographe et de grammaire), un tweet répond à un ensemble de règles propre au service Twitter. Ces spécificités (#, rt, @, …) présentées dans la section 3.2 ne sont pas compréhensibles par les outils de traitement automatique du langage actuel, puisque qu'ils ne possèdent pas de règles spécifiques adaptées au traitement des tweets. Avant de pouvoir les analyser, une première étape consiste à nettoyer le message du tweet de ces éléments.
Les éléments spécifiques aux tweets sont les sujets (#), les destinataires (@) et les re-tweet (RT).
Les informations transmises au travers des tag @ et RT sont toutes stockées afin de pouvoir les analyser mais le texte final, sur lequel portera l'analyse, sera épuré de ces éléments.
Concernant le tag #, il est souvent composé d'un mot qui est porteur de sens. Un sujet est souvent utilisé comme un mot puisqu'il ne perturbe pas la compréhension d'une message et permet ainsi de diminuer le nombre de caractères utilisés. Par exemple, un utilisateur écrit "*I have a #pneumonia*" plutôt que "*I have a pneumonia #pneumonia*".
Il a été décidé de retirer le symbole # mais de conserver le mot qui le suit dans le texte final.
Concernant les liens, ils sont généralement soumis à des racourcisseurs de lien type bit.ly ou goo.gl. A l'origine, ces raccourcisseurs d'URL étaient conçus pour transmettre plus facilement des adresses

de pages Web par e-mail, la plupart des clients de courrier électronique ne pouvant pas afficher de longues suites de caractères sans y insérer des retours à la ligne. Les adresses courtes ont aussi leur utilité lorsqu'il s'agit d'imprimer des URL sur papier, ce qui évite au lecteur d'avoir à taper une longue adresse pour accéder à une page. Mais ces services ont surtout connu une deuxième jeunesse avec le développement de Twitter : les messages étant fortement limités, les raccourcisseurs sont nécessaires pour pouvoir y publier aisément des liens. Ainsi le lien http://catalogues-formation.cnam.fr/recherche-avancee-241235.jsp de 66 caractères est équivalent au lien http://goo.gl/JzaVR qui ne mesure lui que 27 caractères.
D'un point de vue de l'analyse textuelle, si le premier lien pouvait apporter une information, le second quant à lui est inintéressant. Pour cette raison les liens sont aussi stockés et retirés des messages finaux.

Les éléments nettoyés sont stockés dans quatre relations différentes, toutes créées selon le même schéma : *id,element_nettoyé*, afin de pouvoir être analysées (c.f Relation 5).



Relation 5: TWITTER_TAG, TWITTER_DEST, TWITTER_RT et TWITTER_LIEN

Champs	Commentaires
id	Identifiant du tweet
tag	Valeur du tag (#mot)
destinataire	Valeur du destinataire (@destinataire)
emetteur	Valeur de l'émetteur (RT emetteur)
lien	Valeur du lien (http://)

Description 5: Relations TWITTER_TAG, TWITTER_DEST, TWITTER_RT et TWITTER_LIEN

De manière à mieux appréhender cette phase de nettoyage, considérons l'exemple suivant:
"*rt @mannix1000: @kathyireland hi please help with a retweet. #meningitis awareness, my son died age 3 months. http://goo.gl/JzAgT...*"

En appliquant les règles énumérées ci dessus nous obtenons :

"*~~rt @mannix1000:~~ ~~@kathyireland~~ hi please help with a retweet. ~~#~~meningitis awareness, my son died age 3 months. ~~http://goo.gl/JzAgT~~...*"

Le message qui sera pris en compte sera le suivant :

"*hi please help with a retweet. meningitis awareness, my son died age 3 months.…*"

Nous obtenons à l'issue de cette phase, une phrase susceptible d'être traitée par des outils de traitement automatique de texte.

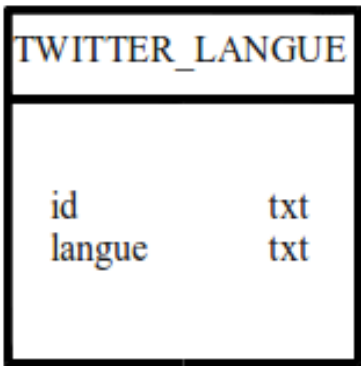
5.3.1.1.1 Détermination de la langue

L'étape suivante consiste à déterminer la langue du tweet. Les outils de traitement automatique de texte requièrent généralement la connaissance préalable de la langue puisque les méthodes appliquées diffèrent d'une langue à une autre. Nous avons retenu TextCat qui permet l'identification de 69 langues. TextCat présente l'avantage d'être un logiciel libre (distribué sous Licence publique générale limitée GNU[23]) et d'être développé en PERL. TextCat repose sur la méthode dite des n-grams. Cette technique consiste à analyser le texte et à répertorier tous les n-grams (séquences de n caractères consécutifs) et de compter la fréquence d'apparition de ces n-grams dans le texte à détecter. Par exemple, le trigramme "*the"* est assez spécifique de l'anglais.

Le résultat de ce comptage est comparé à un référentiel établi au préalable pour chacune des langues sur un corpus de documents. La fréquence de correspondance calculée amène à l'attribution d'un score de probabilité d'appartenance à une langue.

Ainsi dans notre tweet exemple, la langue identifiée est (par ordre de probabilité) soit de l'anglais, soit de l'écossais, soit de l'afrikaans. Nous décidons de considérer uniquement la première langue retournée par TextCat. Cette valeur est stockée dans une relation *TWITTER_LANGUE.*

23 http://fr.wikipedia.org/wiki/Licence_publique_g%C3%A9n%C3%A9rale_limit%C3%A9e_GNU



Relation 6: TWITTER_LANGUE

Champs	Commentaires
id	Identifiant du tweet
langue	Langue évaluée par TextCat

Description 6: Relation TWITTER_LANGUE

Nous focalisons notre attention sur les tweets de langue anglaise. Tout d'abord parce que nous utilisons un thésaurus de langue anglaise. Ensuite Twitter est plus utilisé sur le continent américain et principalement au États-Unis que dans le reste du monde. Enfin l'anglais représente la deuxième langue parlée dans le monde derrière le mandarin. L'accès à Twitter en Chine étant des plus aléatoires et nos compétences en mandarin étant limitées, nous avons choisi de ne traiter que les tweets de langue anglaise.

5.3.1.2 Analyse morpho-syntaxique

A partir des tweets identifiés comme étant de langue anglaise ou apparentée (Ecosse, Irlande, Pays de Galle, Gaelic), nous effectuons une analyse morpho-syntaxique. Une analyse morpho-syntaxique consiste à évaluer dans un segment de discours, d'une part la forme (morphologie flexionnelle), d'autre part la fonction (syntaxe) de ses éléments constitutifs. A cette fin, nous avons étudié deux outils qui sont TreeTagger et l'étiqueteur de Brill.

Développé au sein du projet TC[24] (*"textual corpora and tools for their exploration")* à l'institut de

24http://www.ims.uni-stuttgart.de/projekte/tc/

linguistique computationnelle de l'université de Stuttgart, TreeTagger est un système d'annotation de catégories morpho-syntaxiques permettant d'étiqueter des textes en anglais, français, allemand, italien, grec, et ancien français. Il est possible d'adapter l'étiqueteur à d'autres langues, à condition de disposer d'un lexique et d'un corpus manuellement annoté.
TreeTagger est proche des étiqueteurs n-grams traditionnels. Les deux systèmes déterminent la probabilité de l'annotation d'une séquence de mots. Cependant, contrairement à la plupart des étiqueteurs qui recourent aux modèles de Markov pour résoudre le problème des "données clairsemées" (*sparse data*), TreeTagger utilise un arbre de décision binaire pour calculer la taille du contexte à utiliser pour estimer les probabilités de transition.
L'étiqueteur de Brill est fondé sur les travaux de [BLOOMFIELD, 1933] et [HARRIS, 1954]. Reposant sur l'idée que l'étude d'une langue peut se fonder sur l'observation de faits linguistiques et indépendamment d'une théorie linguistique particulière, l'étiqueteur doit, pour fonctionner, être entraîné sur un corpus de taille restreinte étiqueté manuellement et à partir duquel il infère des règles d'étiquetage (distribution "extensionnelle"). Les mots inconnus sont traités à partir d'une hypothèse naïve sur la structure du langage. Enfin, une analyse de la distribution est effectuée afin de réduire les erreurs d'étiquetage.

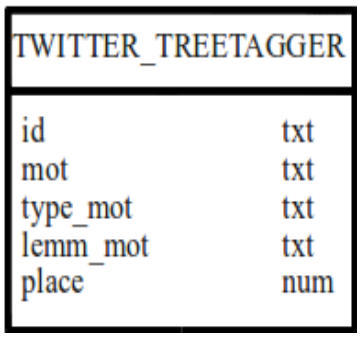
Les deux approches fournissent des résultats similaires dans un contexte de tweet. TreeTagger, contrairement à Brill, fournit la forme lemmatisée du mot. C'est pour cette raison que nous avons choisi TreeTagger.

Considérons de nouveau notre exemple (*"hi please help with a retweet. meningitis awareness, my son died age 3 months.…"*), en appliquant TreeTagger nous obtenons le résultat suivant (c.f. Tableau 10) :

Mot	Tag	Correspondance du tag	Mot lemmatisé
hi	UH	Interjection	*hi*
please	UH	Interjection	*please*
help	VB	Verbe	*help*
with	IN	Préposition ou conjonction de subordination	*with*
a	DT	Déterminant	*a*
retweet	NN	Nom singulier	*<unknown>*
.	SENT	Ponctuation	.
meningitis	NN	Nom singulier	*meningitis*
awareness	NN	Nom singulier	*awareness*
,	,		,
my	PP$	Pronom personnel	*my*
son	NN	Nom singulier	*son*
died	VBD	Verbe au passé	*die*
age	NN	Nom singulier	*age*
3	CD	Nombre	*3*
months	NNS	Nom pluriel	*month*
….	JJ	Adjectif	*<unknown>*

Tableau 10: Analyse morpho-syntaxique d'un tweet avec TreeTagger

L'utilisation de la forme lemmatisée du mot nous permettra par la suite de considérer par exemple les mots *months* et *month* comme un même mot. Le résultat est stocké dans la relation *TWITTER_TREETAGGER*.

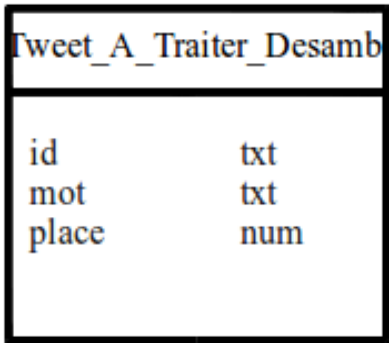


Relation 7: TWITTER_TREETAGGER

Champs	Contenu
id	Identifiant du tweet
mot	Mot présent dans le tweet
type_mot	Genre du mot (verbe, nom, adjectif, pronom, ...)
lemm_mot	Forme lemmatisée du mot
place	Position du mot dans la phrase

Description 7: Relation TWITTER_TREETAGGER

En parallèle, chacun des mots lemmatisés présent dans le thésaurus du MeSH est stocké dans la relation *TWEET_A_TRAITER_DESAMB* pour pouvoir être traité lors de la phase de désambiguïsation si nécessaire (c.f. sous-section 5.3.3).



Relation 8: TWEET_A_TRAITER_DESAMB

Champs	Contenu
id	Identifiant du tweet
mot	Forme lemmatisée du mot présent dans le tweet
place	Position du mot dans la phrase

Description 8: Relation TWEET_A_TRAITER_DESAMB

La phase de normalisation du tweet est résumée dans la figure 31:

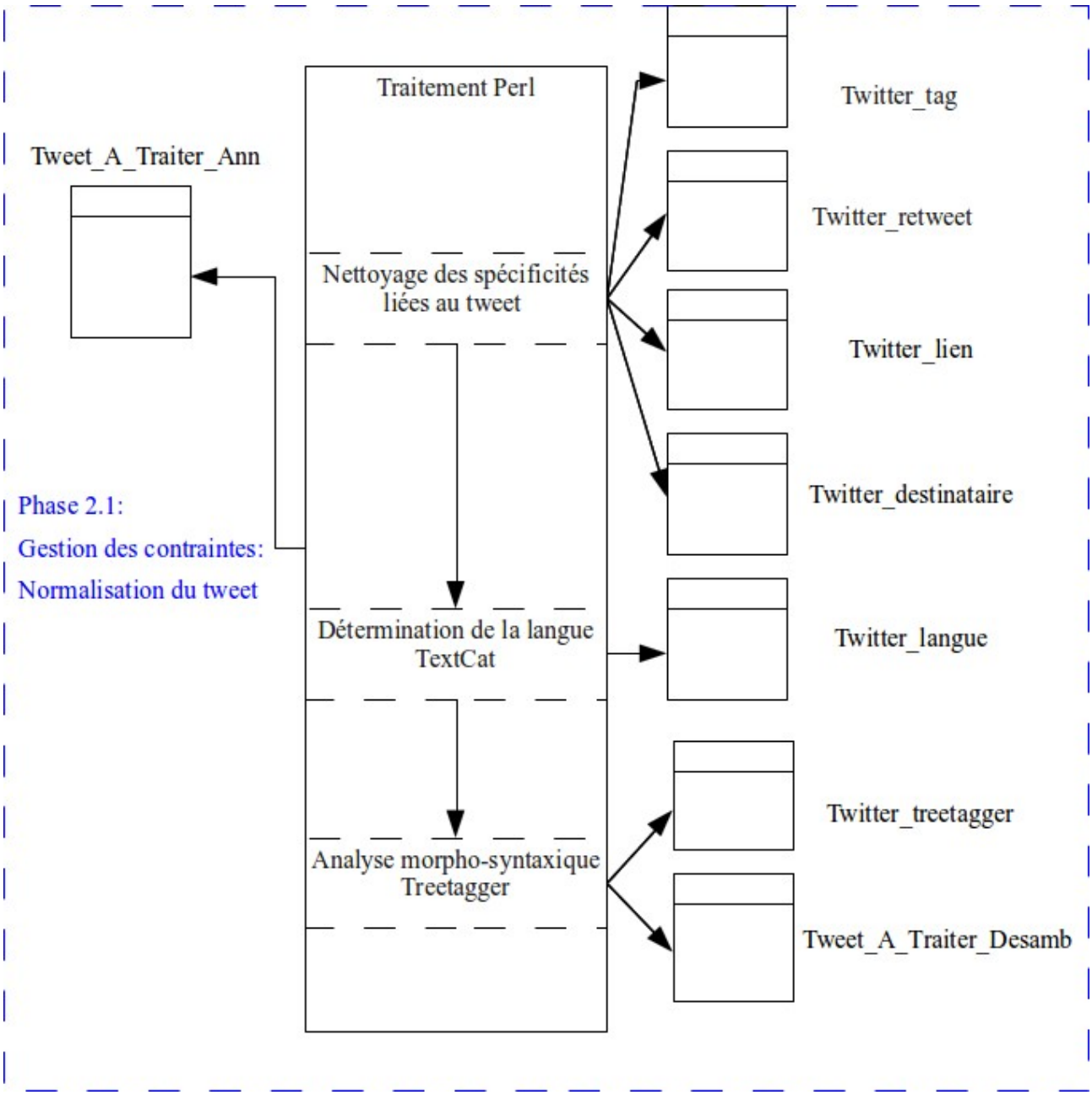


Figure 31: Phase de normalisation du tweet

Dans cette partie, nous sommes partis d'un "tweet" afin de constituer un ensemble de mots de forme canonique.

5.3.2 Normalisation de la localisation

Comme nous l'avons vu dans la sous-section 4.3.2, les informations relatives à la localisation ne peuvent être intégrées sans pré-traitement. L'objectif est de pouvoir rattacher le tweet à une localisation dont nous connaissons la hiérarchie pour préparer l'analyse multi-dimensionnelle.

Même dans le meilleur des cas, il reste à définir à quel niveau de cette hiérarchie (Paris → Ville, Usa → Pays) la localisation obtenue fait référence. De plus un traitement à partir des coordonnées géographiques (le plus intéressant car le plus précis) nécessite une approche différente car deux personnes éloignées de 100 mètres ne posséderont pas les même coordonnées.

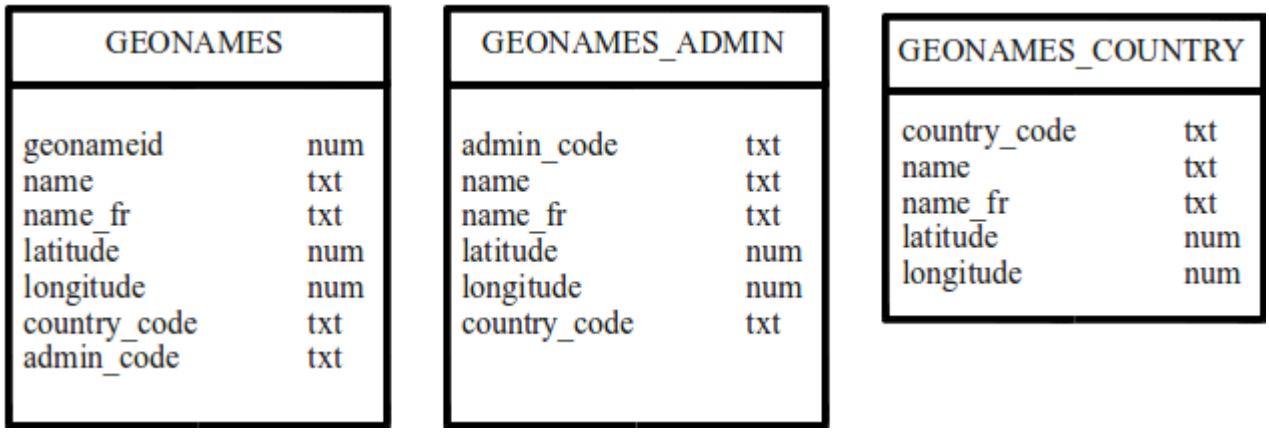
Nous avons choisi comme référence la base Geonames[25] qui est une base de données géographiques gratuite et accessible par Internet sous une licence Creative Commons.

La base de données contient plus de 8 millions de noms géographiques qui correspondent à plus de 6,5 millions de lieux existants. Ces noms sont classés en 9 catégories et 645 sous-catégories. Des données comme la latitude, la longitude, l'altitude, la population, la subdivision administrative, le code postal sont disponibles en plusieurs langues pour chaque emplacement. Les données sont téléchargeables sous forme de données textuelles. Nous avons intégré les villes ayant une population supérieure à 1000 personnes afin de limiter le volume de notre base de référence. De même en cas d'homonymie (par exemple Paris capitale de la France ou Paris dans l'Illinois aux États-Unis), nous avons pris comme hypothèse de ne retenir que la ville ayant la population la plus nombreuse. Une phase de désambiguïsation géographique sera présentée dans la sous-section 7.1.2.

Aux 88574 villes intégrées, nous avons ajouté sept abréviations américaines courantes ("Washington" pour "Washington DC", "JAX" pour "Jacksonville", "OKC" pour "Oklahoma City", "NYC","NY", "New York" pour "New York City", "LA" pour "Los Angeles"). Nous sommes conscients qu'il en existe d'autre mais nous nous sommes arrêtés sur ces sept abréviations les plus fréquemment rencontrées. Enfin cinq alias référençant les fuseaux horaires ont été intégrés avec les coordonnées des villes principales correspondant à leur zone ("*Central Time (US & Canada)*" à "Chicago", "*Eastern Time (US & Canada)*" à "New York", "*Mountain Time (US & Canada)*" à "Tucson",""*Pacific Time (US & Canada)*" à "Los Angeles" et "*Atlantic Time (US & Canada)*" à "Fredericton").

Ainsi la relation des villes (nommée *GEONAMES*) contient 88586 enregistrements. Deux relations annexes ont été créées, la première (*GEONAMES_ADMIN*) pour contenir le découpage administratif des pays concernés (l'état aux Etats-Unis ou en Australie, la province au Canada, "home nations" pour le Royaume-Unis) et la seconde pour les informations relatives au pays (*GEONAMES_COUNTRY*). Les champs *names_fr* permettent de faire la correspondance entre un nom abrégé et sa traduction (LA → Los Angeles, UK → United-Kingdom) ou entre un nom de pays en langue officielle et sa traduction en anglais (España → Spain). Ont été retenues comme latitude et longitude d'un pays (resp. d'un état) les coordonnées de la capitale du pays (resp. de l'état).

25 http://www.geonames.org/



Relation 9: GEONAMES, GEONAMES_ADMIN et GEONAMES_COUNTRY

Champs	Contenu
geonameid	Identifiant numérique de la ville fourni par Geonames
name	Ville ou état/province ou pays
name_fr	Ville ou état/province ou pays traduits ou abrégés
latitude	Coordonnée latitude
longitude	Coordonnée longitude
country_code	Identifiant pays
admin_code	Identifiant état/province

Description 9: Relations GEONAMES, GEONAMES_ADMIN et GEONAMES_COUNTRY

Cette architecture permet de déterminer une localisation que l'information récupérée de Twitter soit :

- des coordonnées géographiques
- une saisie manuelle d'un nom de ville, d'un état ou de pays
- un fuseau horaire.

Nous décrivons maintenant le processus mis en place. Nous cherchons tout d'abord à identifier l'emplacement géographique à partir des informations de localisation fournies puis en cas d'échec, nous regardons à partir du fuseau horaire.

Si les informations transmises sont des coordonnées géographiques, pour chaque ville nous calculons une distance entre les coordonnées du tweet et de la ville à partir de la formule développée par [VINCENTY, 1975]:

$$R \times acos(\cos(LaT) \times \cos(LaV) \times \cos(LoV - LoT) + (\sin(LaT) \times \sin(LaV)))$$

Avec :

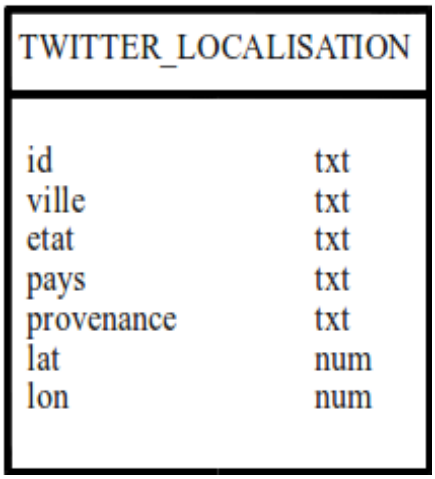
- R = 6366 (correspond au rayon de la terre en kilomètre)
- LaT = latitude tweet en radian
- LaV = latitude ville en radian
- LoT = latitude tweet en radian
- LoV = longitude ville en radian

Nous retenons la ville pour laquelle la distance entre ses coordonnées et celles du tweet est la plus faible.

Si les informations transmises ne sont pas des coordonnées géographiques, nous considérons alors le mot précédent une éventuelle virgule. Ce premier mot est suffisant pour retrouver la localisation (par exemple "Los Angeles, USA"). De plus, nous avons constaté que se limiter au premier mot permettait d'atténuer le bruit généré. Nous recherchons une correspondance entre une ville (relation *GEONAMES*) et ce mot. Le cas échéant nous affectons au tweet le trio: ville, état/province, pays. Sinon, nous cherchons une correspondance sur l'état/province (relation *GEONAMES_ADMIN*). En cas de correspondance trouvée, nous affectons au tweet les valeurs: * (caractère étoile), l'état/province, pays. Enfin nous cherchons une dernière correspondance sur le pays (*GEONAMES_COUNTRY*) et affectons au tweet les valeurs * (caractère étoile),* (caractère étoile) , pays si correspondance il y a.
Les * (caractère étoile) ont été choisies comme ville ou état de substitution.

Si aucune correspondance n'a été trouvée alors nous regardons les informations apportées par le fuseau horaire. Les informations disponibles sont soit un nom de ville ou d'état ou de pays soit un nom de fuseau horaire. La démarche est la même que pour la recherche de correspondance sur la localisation. En revanche, un fuseau horaire étant rattaché à un pays, nous affectons au tweet la valeur * (caractère étoile) pour la ville et l'état/province et ce, même si la correspondance a été trouvée sur l'un de ces éléments. En France, le fuseau horaire est celui de Paris où que l'on se situe sur le territoire. Il serait faux de rattacher tous les tweets émis avec ce fuseau horaire à la capitale. En revanche les tweets émis d'Italie seront en général rattachés au fuseau horaire de Rome (même si il n'y a pas de décalage horaire entre Rome et Paris).

Les localisations retenues sont stockées dans une relation spécifique nommée *TWITTER_LOCALISATION*.



Relation 10: TWITTER_LOCALISATION

Champs	Contenu
id	Identifiant du tweet
ville	Nom de la ville ou *
etat	Nom de l'état ou de la province ou *
pays	Nom du pays
provenance	Identification de la provenance (localisation ou fuseau horaire)
lat	Coordonnée latitude
lon	Coordonnée longitude

Description 10: Relation TWITTER_LOCALISATION

Nos analyses ont montré que ce processus permettait d'affecter une localisation dans 75% des cas. Comme le montre la répartition sur la figure 32, plus de 43% des localisations qui peuvent être effectuées le sont grâce à la ville ou aux coordonnées (et ont donc une précision importante).

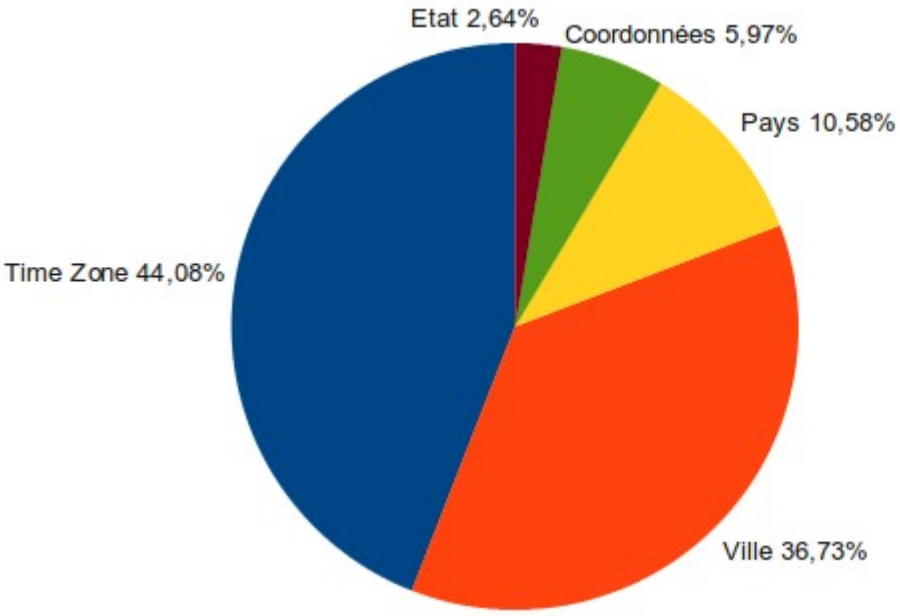


Figure 32: Localisation ou fuseau horaire, qui fournit l'information ?

La phase de normalisation de la localisation est résumée dans la figure 33.

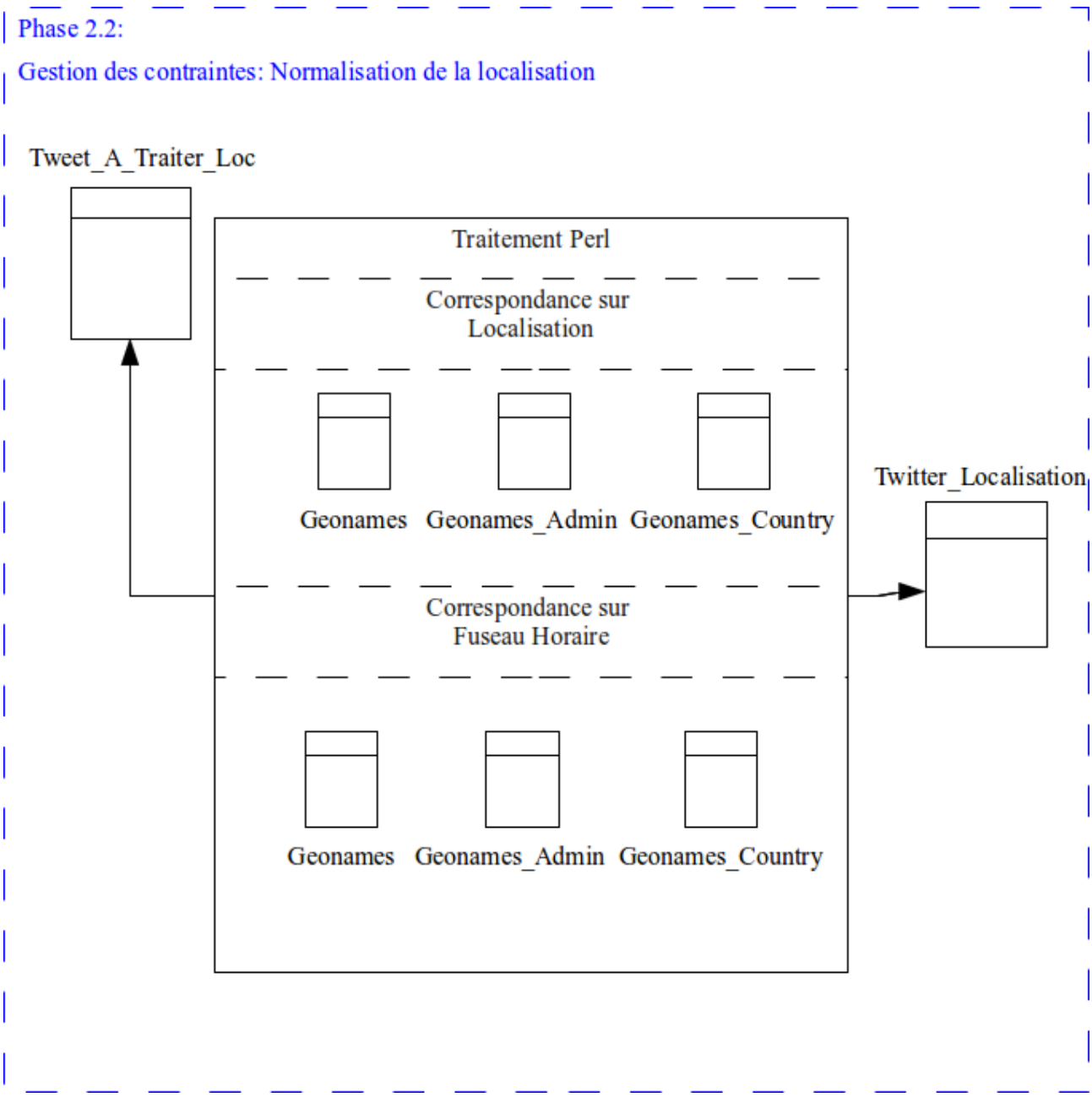


Figure 33: Phase de normalisation de la localisation

Ce traitement nous a permis d'affecter une ville, un état ou un pays à une majorité de Tweet. Cette localisation est normalisée et donc est exploitable.

Après avoir traité les ambiguïtés liées aux données du tweet, nous traitons maintenant les contraintes introduites par le thésaurus utilisé.

5.3.3 Gestion de la désambiguïsation

5.3.3.1 Motivations

Il a été montré dans la sous-section 4.3.4 que dans le thésaurus utilisé, différents termes peuvent apparaître à plusieurs niveaux de la hiérarchie. Il est donc indispensable de rattacher chacun des mots du tweet à une seule partie de la hiérarchie du MeSH (ne sont donc concernés que les mots présents dans un tweet et dans le thésaurus du MeSH). Il existe plusieurs approches pour déterminer à quelle hiérarchie un mot donné pour un tweet donné appartient.
Considérons la hiérarchie suivante:

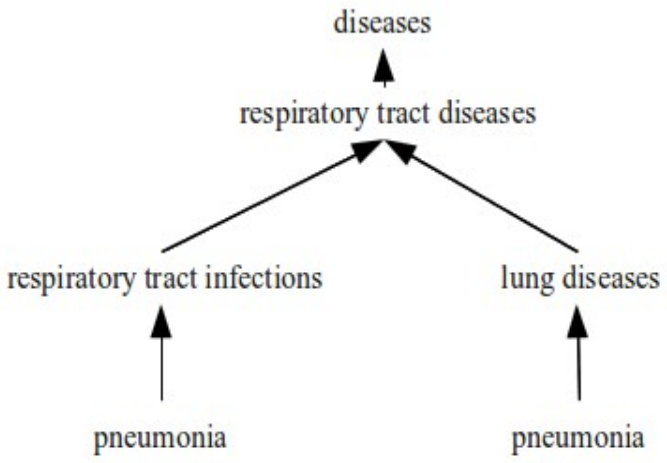


Figure 34: Hiérarchie "pneumonia"(sous ensemble du thésaurus du MeSH)

Considérons ensuite le tweet suivant :
"*pneumonia & serious nerve problems. can't stand up. possible myasthenia gravis treatable with meds.*".
Si nous recherchons dans MeSH le concept associé à *pneunomia*, nous constatons que ce mot intervient à plusieurs endroits (c.f. Figure 34):

- soit dans la hiérarchie *pneumonia >respiratory tract infections > respiratory tract diseases > diseases*
- soit dans la hiérarchie *pneumonia > lung diseases > respiratory tract diseases > diseases* .

En fonction de la position, une opération de *roll-up* sur *pneunomia* ne donnera pas le même résultat (soit "*respiratory tract diseases*" ou "*lung diseases*").

Si nous examinons le tweet, nous pouvons constater que les mots utilisés peuvent aider à déterminer le contexte. Il est ainsi évident que le sujet du tweet concerne davantage le concept "*respiratory tract infections*" que le concept "*lung diseases*".

Aussi, par la suite, nous considérons l'hypothèse suivante : plus le mot d'un tweet apparaît fréquemment avec le parent d'un concept dans le même contexte du tweet, plus le mot appartient à ce concept .

5.3.3.2 Mesure de fouille du web

De manière à désambiguïser les mots polysémiques de la hiérarchie de MeSH, nous avons adapté la méthode $\text{AcroDef}_{\text{IM3}}$ décrite dans [ROCHE et PRINCE, 2008]. Cette mesure qui est fondée sur l'Information Mutuelle au cube ([DAILLE, 1994]) calcule la dépendance de deux mots dans un contexte donné. Contrairement à l'Information Mutuelle, l'Information Mutuelle au cube privilégie les co-occurrences fréquentes. Par ailleurs, $\text{AcroDef}_{\text{IM3}}$ prend en considération le contexte dans lequel les co-occurrences sont présentes. En appliquant un contexte C (mots illustrant un contexte), l'approche $\text{AcroDef}_{\text{IM3}}$ est donnée par la formule ci-dessous.

$$AcroDef_{IM3}(m_1, m_2) = \frac{(nb(m_1 \wedge m_2 \wedge C))^3}{(nb(m_1 \wedge C) \times nb(m_2 \wedge C))}$$

Dans notre cas, nous souhaitons calculer la dépendance entre un mot *m* du tweet et son concept parent *p* de la hiérarchie. Cette dépendance est calculée par rapport au contexte du tweet. Un tel contexte est caractérisé par les mots proches de *m* que nous souhaitons désambiguïser. Ce contexte est déterminé dans une fenêtre de 5 mots dans le texte du tweet. Dans la pratique, nous prenons en compte les deux mots précédant et suivant le terme à désambiguïser. Ne sont retenus que les noms, les verbes et les adjectifs sélectionnés via un étiquetage grammatical préalable effectué par TreeTagger. Ceci revient à effectuer, au plus, quatre requêtes distinctes : '*m and p and* mot_i' où mot_i désigne les mots du contexte *C* des tweets. Ainsi, *nb(m, p)*, qui prend en compte un contexte *C* de la même manière que la formule ci-dessus, correspond à la somme des pages retournées par ces quatre requêtes.

Reconsidérons le tweet: "*pneumonia & serious nerve problems. can't stand up. possible myasthenia gravis treatable with meds.*" et la hiérarchie présentée Figure 34.

Le contexte est formé des deux mots porteurs d'informations (noms, adjectifs, verbes) suivant "*pneunomia*" : "*serious"* et "*nerve"*. Le but est alors de déterminer à quelle partie du thésaurus MeSH nous devons associer le mot "*pneunomia*" du tweet. Pour cela, le calcul des numérateurs donne les résultats ci-dessous :

- nb(*pneunomia*; "*lung diseases*") = 145 (nombre de pages retournées avec les requêtes '*pneunomia* "*lung diseases*" *serious*' et '*pneunomia* "*lung diseases*" *nerve*')
- nb(*pneunomia*; "*respiratory tract infections*") = 360

Le calcul ci-dessous donne alors la dépendance des termes :

AcroDef_{IM3}(*pneunomia*; "*lung diseases*") = $\dfrac{nb(\,pneunomia\,,'\,lung\ diseases\,')^3}{nb(\,pneunomia\,)\times nb(\,'\,lung\ diseases\,')} = 6.10^{-5}$

AcroDef_{IM3}(*pneunomia*; "*respiratory tract infections*") =

$\dfrac{nb(\,pneunomia\,,'\,respiratory\ tract\ infections\,')^3}{nb(\,pneunomia\,)\times nb(\,'\,respiratory\ tract\ infections\,')} = 48.10^{-5}$

Ainsi, dans cet exemple, pour le mot "*pneunomia*", nous allons privilégier une agrégation au niveau du concept "*respiratory tract infections*" de MeSH.

Notons que cette étape de désambiguïsation qui se révèle indispensable pour les données de MeSH permet d'obtenir des résultats corrects en terme d'affectation (entre 60% et 70% d'affectations correctes) mais moyenne en terme de taux de traitement (entre 50% et 60% des tweets seulement sont classés). De plus elle est assez coûteuse en terme de nombre de requêtes à mener (quatre par mot à désambiguïser) et en terme de temps (six mots désambiguïsés par minute).

Afin d'améliorer la problématique liée aux temps de traitement, nous avons adopté une autre approche.

5.3.3.3 Mesure de Recherche d'Information

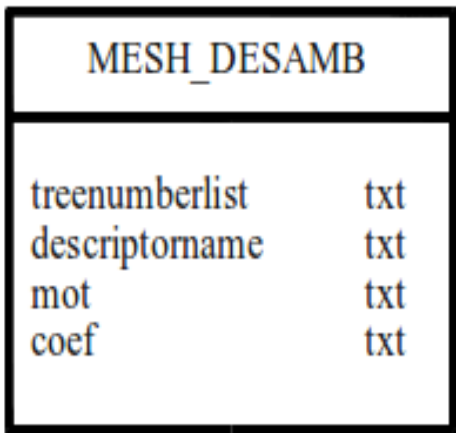
Notre problématique est d'améliorer le temps de traitement lié aux quatre requêtes nécessaires. L'idée est de rattacher à chaque terme du MeSH les mots fréquemment employés avec le concept. Nous partons de l'hypothèse qu'en affectant à un concept du MeSH un ensemble de mots fréquemment associé avec ce concept et son parent dans la hiérarchie MeSH, nous pourrons retrouver des correspondances avec les mots présents dans le tweet.

Pour constituer les ensembles de mots de références, nous avons récupéré 75 résumés fournis par un moteur de recherche pour une requête portant sur le mot du MeSH et son père dans la hiérarchie.

Un résumé correspond à une description de la page sur 250 caractères qui reprend les mots clefs importants. Il est soit écrit dans la page par les développeurs, soit généré par le moteur de recherche. Il est affiché dans les résultats d'un moteur de recherche et son rôle est d'inciter l'internaute à cliquer sur le lien. Nous avons préféré Yahoo à Google parce qu'il n'est pas possible de récupérer plus de 8 résumés en parallèle avec l'API mise à disposition par Google. Pour chaque résumé, nous utilisons TreeTagger pour déterminer la fonction grammaticale de chaque mot (seuls les verbes, noms, adjectifs sont retenus) et nous conservons la forme lemmatisée des termes qui ne sont pas des mots "vides".

Nous affectons un coefficient à chaque terme qui correspond au nombre de fois où le mot a été associé dans le résultat de la requête web.

Nous avons constitué, pour chaque concept du MeSH à désambiguïser, un ensemble de couples mot/coefficient stocké dans la relation *MESH_DESAMB*.



Relation 11: MESH_DESAMB

Champs	Contenu
treenumberlist	Identifiant du terme dans la hiérarchie du MeSH
descriptorname	Terme dans la hiérarchie du MeSH
mot	Mot trouvé dans les résultats de la reqête
coef	nombre de fois ou le mot est associé avec le terme du MeSH dans le résultat des requêtes

Description 11: Relation MESH_DESAMB

Le choix de 75 résumés a été validé après avoir étudié les résultats obtenus avec des ensembles de mots constitués à partir de 10, 20, 50, 75 et 100 résumés sur un corpus de 901 termes (c.f. Tableau 11).

Nombre de résumé observé	% d'affectation erronée	% d'affectation correcte
10	77,36%	22,64%
20	53,16%	46,84%
50	40,29%	59,71%
75	37,18%	62,82%
100	40,07%	59,93%

Tableau 11: Désambiguïsation Analyse du nombre de résumés nécessaires

Cette étape est longue (6 mots traités à la minutes) mais ne doit être réalisée qu'une seule fois pour chaque terme du MeSH apparaissant plusieurs fois dans la hiérarchie.

Une fois les ensembles de mots constitués, nous avons appliqué la méthode de la mesure cosinus entre les mots du tweets et les mots fréquemment associés au concept à désambiguïser. La mesure cosinus (ou similarité cosinus) permet de calculer la similarité entre deux vecteurs à *n* dimensions en déterminant l'angle entre eux. Cette métrique est fréquemment utilisée pour mesurer la ressemblance entre deux documents.

Soit deux vecteurs *A* et *B*, l'angle θ s'obtient par le produit scalaire et la norme des vecteurs :

$$\theta = \arccos\left(\frac{A.B}{\|A\|.\|B\|}\right)$$

Comme l'angle θ est compris dans l'intervalle [0,π], la valeur π indiquera des vecteurs résolument opposés, π / 2 des vecteurs indépendants (orthogonaux) et 0 des vecteurs colinéaires. Les valeurs intermédiaires permettent d'évaluer le degré de similarité.

Cette formule se généralise dans un espace de dimension 2 à L avec 2 vecteurs V_1 et V_2 :

$V_1 = \{a_1 , \dots , a_j , \dots , a_L\}$

$V_2 = \{b_1 , \dots , b_j , \dots , b_L\}$

$$\cos(V_1, V_2) = \frac{\sum_{j=1}^{L} a_j b_j}{\sqrt{\sum_{j=1}^{L} a_j^2}\sqrt{\sum_{j=1}^{L} b_j^2}}$$

En comparant les angles des vecteurs de mots formés par le tweet (contenu dans la relation *TWITTER_TREETAGGER*) et les différents vecteurs représentant les hiérarchies auxquelles il pourrait appartenir, on détermine celle dont il est le plus proche.

Cependant, le vecteur servant de base au calcul de la mesure cosinus a dû être adapté au contexte du tweet afin de prendre en compte la limite de 140 caractères. Un mot fortement représentatif d'une hiérarchie n'apparaît guère plus d'une fois dans le tweet.
Or nous souhaitons affecter un poids plus important aux mots les plus discriminants d'un contexte qu'aux mots qui apparaissent mois souvent. Nous décidons de considérer autant de fois un mot commun entre le vecteur rattaché à la hiérarchie et le vecteur tweet que le mot est rattaché au vecteur hiérarchie.

Considérons le tweet :
Honour killings, domestic violence still a problem in Turkey: Women face an uphill battle in the court system, along... http://eq6pf.tk

Le mot *turkey* apparaît à deux niveaux dans la hiérarchie du MeSH, selon que l'on considère le pays ou la volaille.

- *organism >> eukaryota >> animal >> chordata >> vertebrate >> bird >> poultry >> turkey*
- *geographicals >> geographic location >> Asia >> Asia , western >> middle east >> turkey*

Nous souhaitons déterminer à quelle hiérarchie du mot *turkey*, le tweet fait référence.
Les mots "utiles" du tweet sont :

- *honour*
- *kill*
- *domestic*
- *violence*
- *problem*
- *women*
- *batlle*
- *court*
- *systeme*

Supposons que les mots fréquemment rattachés au concept *Turkey* dans la hiérarchie volaille soient

(avec le nombre de fois où le mot est associé dans le résultat de la requête web entre parenthèse) :

- *kill*(1)
- *battle* (1)
- *honour* (1)

Supposons ensuite que les mots fréquemment rattachés au concept *Turkey* dans la hiérarchie pays soient : *women* (5)

Dans notre exemple, nous normalisons les vecteurs comme le montre le tableau 12.

Mot	contexte tweet	contexte hiérarchie volaille
honour	1	1
kill	1	1
Domestic	1	0
violence	1	0
problem	1	0
women	1	0
batlle	1	1
court	1	0
systeme	1	0

Tableau 12: Vecteur Turkey dans le contexte Volaille

Puis nous calculons l'angle formé par le vecteur tweet (1-1-1-1-1-1-1-1-1) et le vecteur hiérarchie du contexte volaille (1-1-0-0-0-0-1-0-0).

$$\cos(V_{tweet}, V_{volaille}) = \frac{(1*1)+(1*1)+(1*0)+(1*0)+(1*0)+(1*0)+(1*1)+(1*0)+(1*0)}{\sqrt{1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2}*\sqrt{1^2+1^2+0^2+0^2+0^2+0^2+1^2+0^2+0^2}}$$

$$\cos(V_{tweet}, V_{volaille}) = \frac{(1*1)+(1*1)+(1*1)}{\sqrt{1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2}*\sqrt{1^2+1^2+1^2}}$$

$$\cos(V_{tweet}, V_{volaille}) = \frac{3}{\sqrt{9}\sqrt{3}} = 0{,}577350269$$

Puis nous calculons l'angle formé par les vecteurs normalisés entre le tweet et le contexte pays en prenant en compte le poids du mot *women* dans le contexte de la hiérarchie comme le montre le tableau 13.

Mot	contexte tweet	contexte hiérarchie pays
honour	1	0
kill	1	0
Domestic	1	0
violence	1	0
problem	1	0
women	1	1
women	1	1
women	1	1
women	1	1
women	1	1
batlle	1	0
court	1	0
systeme	1	0

Tableau 13: Vecteur Turkey dans le contexte Pays

$$\cos(V_{tweet}, V_{pays}) = \frac{(1*1)+(1*1)+(1*1)+(1*1)+(1*1)}{\sqrt{1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2} * \sqrt{1^2+1^2+1^2+1^2+1^2}}$$

$$\cos(V_{tweet}, V_{pays}) = \frac{5}{\sqrt{13}\sqrt{5}} = 0{,}620173673$$

Après comparaison de l'angle formé par les vecteurs (plus le cosinus est grand, plus les angles sont petits), nous pouvons conclure que les vecteurs V_{tweet} et V_{pays} sont plus proches que les vecteurs V_{tweet} et $V_{volaille}$.

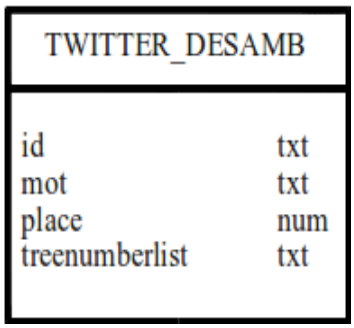
La hiérarchie faisant référence au mot *turkey* dans le contexte géographique sera retenue.

Afin de valider définitivement le choix de la mesure cosinus comme méthode de désambiguïsation, nous avons évalué les mesures de précision et de rappel bien connues en fouilles de données. La précision correspond au nombre de tweets pertinents retrouvés par rapport au nombre de tweets sélectionnés. Une précision de 100% signifie que tous les tweets retrouvés sont pertinents. Le rappel correspond au nombre de tweets pertinents retrouvés par rapport au nombre de tweets. Un rappel de 100% signifie que tous les tweets pertinents ont été retrouvés.
Ce calcul a été effectué sur la base de 1566 tweets contenant le mot "turkey" évalués manuellement afin de déterminer si nous sommes dans un contexte géographique (806 tweets) ou de volaille (760 tweets). Les résultats sont présentés dans le tableau 14.

	Précision	Rappel
Contexte Volaille	68,57%	94,74%
Contexte Géographique	92,25%	59,06%

Tableau 14: Désambiguïsation Précision et Rappel

Nous validons le choix de la mesure cosinus comme algorithme pour notre de désambiguïsation. A chaque mot présent dans le tweet et le thésaurus, nous affectons l'identifiant unique (*treenumberlist*) du MeSH. Nous stockons ces valeurs dans la relation *TWITTER_DESAMB*.



Relation 12: TWITTER_DESAMB

Champs	Contenu
id	nombre de fois où le mot est associé avec le terme du MeSH dans le résultat de la requête
mot	Terme dans la hiérarchie du MeSH
place	Place du mot dans le tweet
treenumberlist	Identifiant du terme dans la hiérarchie du MeSH

Description 12: Relation MESH_DESAMB

La phase de désambiguïsation est résumée dans la figure 34.

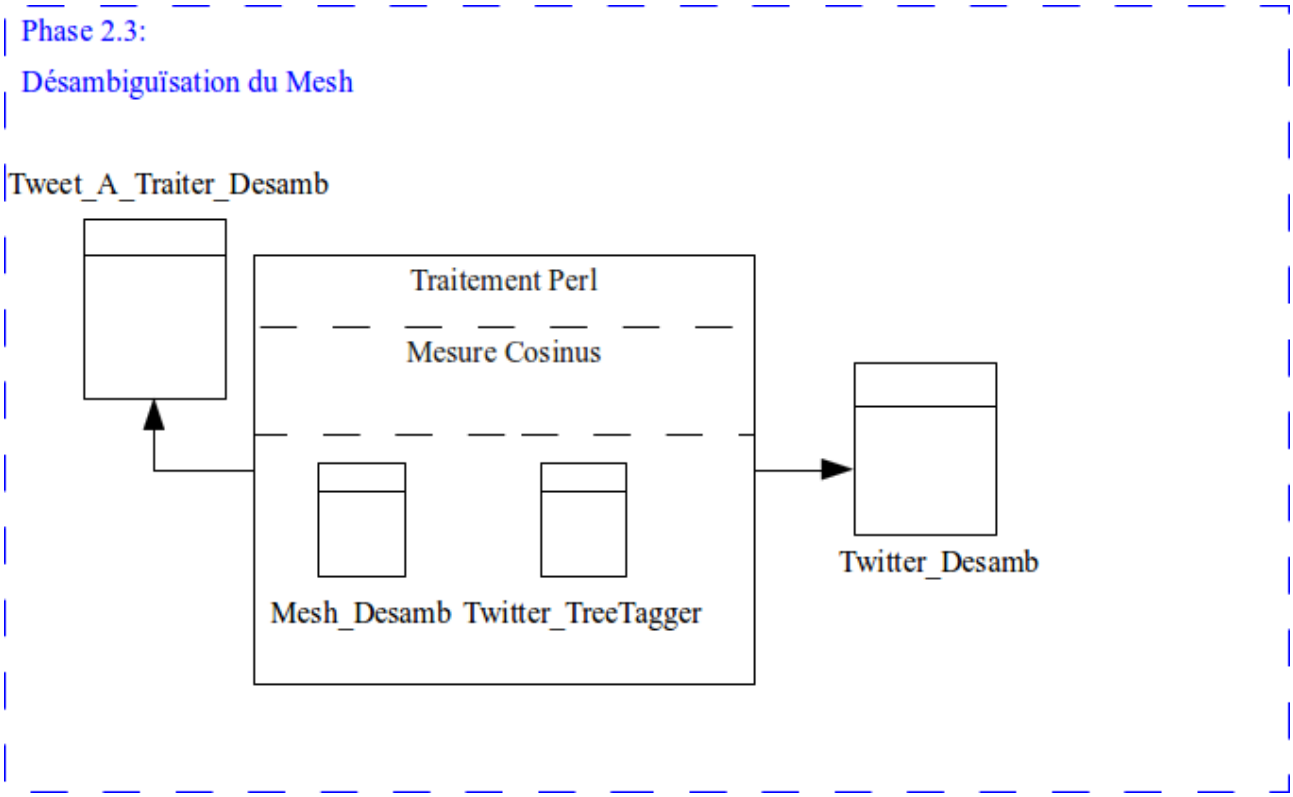


Figure 35: Phase de désambiguïsation du MeSH

Ces traitements représentent le cœur du processus de transformation des données. Ils doivent être rapides afin d'intégrer l'ensemble du flux reçu. La capacité d'absorption pour chaque phase est de :

- 70 tweets par minute pour la normalisation du tweet, l'analyse morpho-syntaxique occupant à elle seul 50 % du temps nécessaire.
- 180 tweets par minute pour la gestion de la localisation.
- 210 tweets par minute pour la désambiguïsation de nos hiérarchies (40 fois plus rapide que

l'approche utilisant *AcroDef*).

A l'issue de ces traitements, les tweets sont nettoyés, localisés et les ambiguïtés ont été levées. Pour conclure cette section et avant de passer à la phase d'alimentation de données, nous récapitulons nos relations dans le schéma relationnel de la figure 36. Le dictionnaire de données associé est disponible en annexe 1.

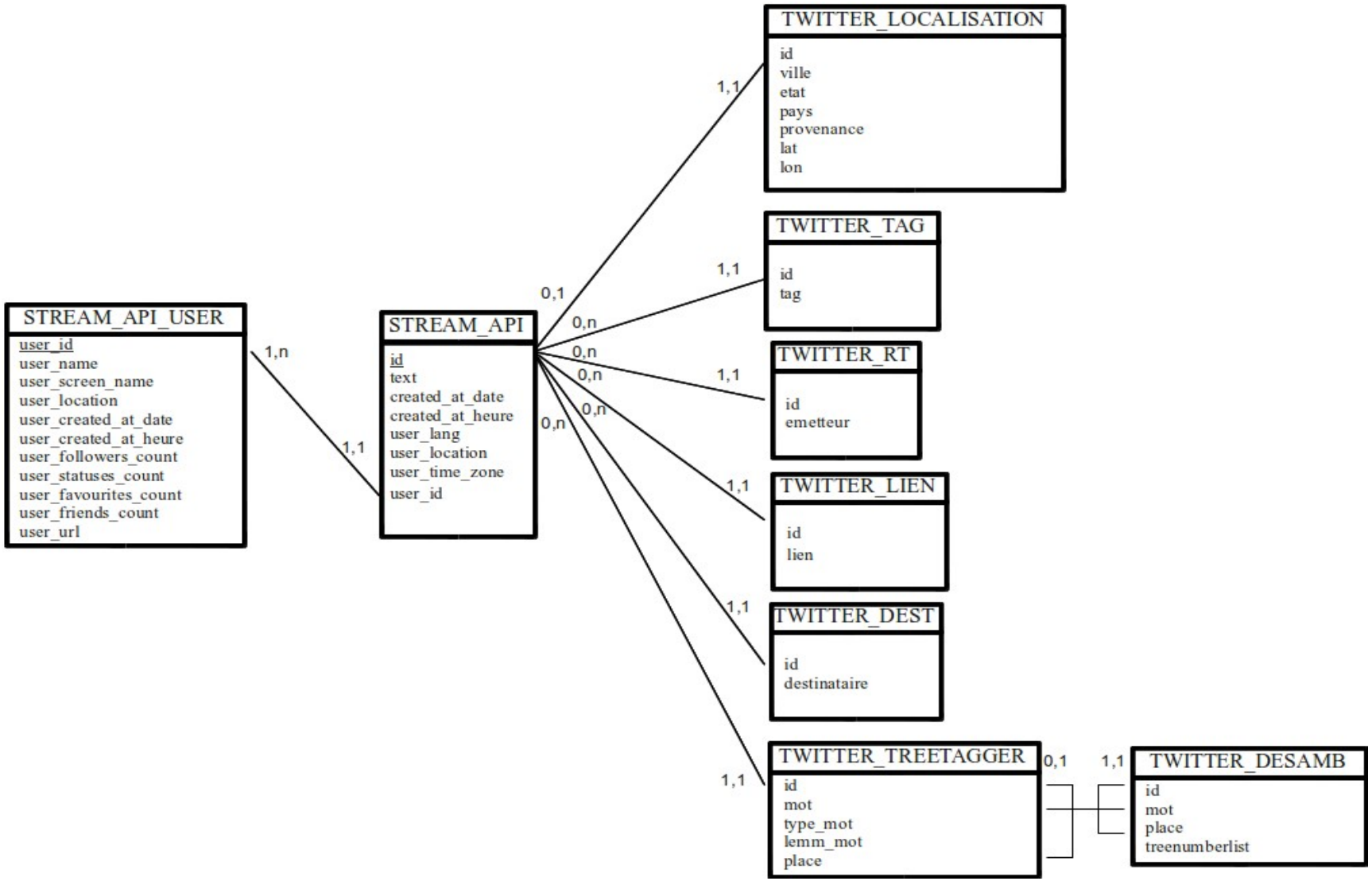
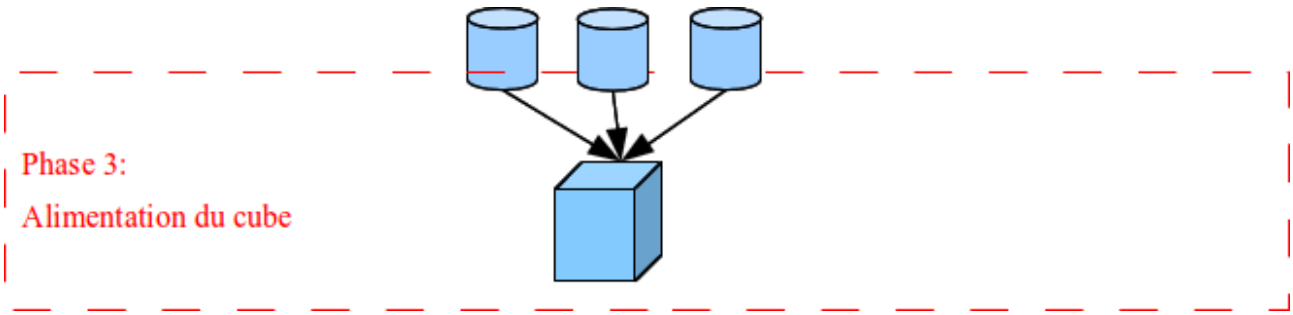


Figure 36: Schéma relationnel – partie traitement des données.

5.4 Phase 3: Alimentation du cube



Suite aux précédentes étapes, les données sont intégrées dans le cube (c.f. Figure 37). La table de faits est alimentée quotidiennement avec les données de la veille. Celles-ci sont agrégées au moment de l'intégration selon les trois dimensions retenues (MeSH, Localisation, Temps).

Elle est alimentée à partir des relations *TWITTER_DESAMB, TWITTER_LOCALISATION* et *STREAM_API*. Les mesures retenues dans l'implémentation courante sont le nombre de tweets ainsi que le TF-IDF et le TF-IDF Adaptatif.

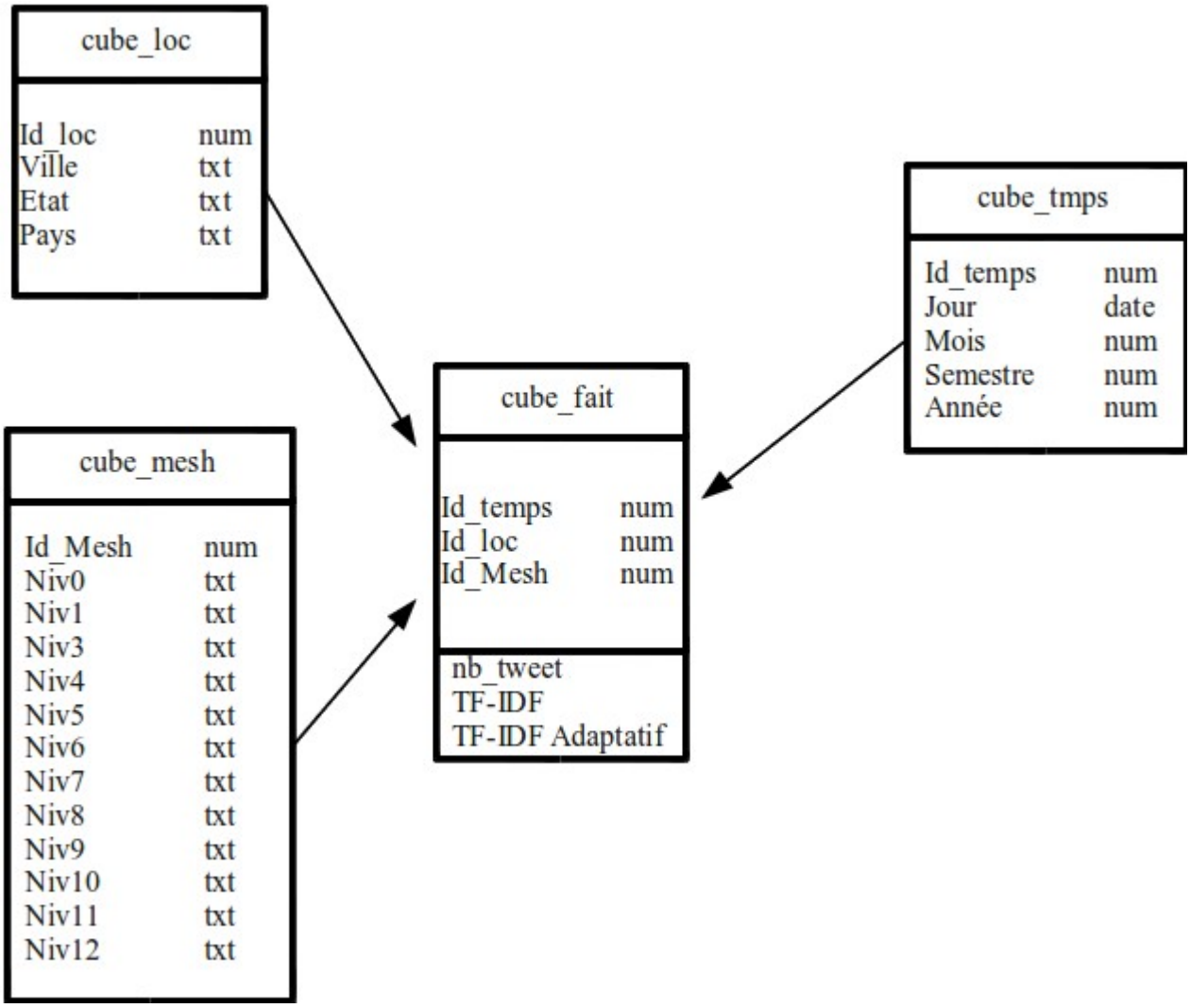


Figure 37: Modèle Logique de Données retenu

La relation CUBE_FAIT, relation centrale pour notre analyse de données est décrite dans le Description 13.

Champs	Contenu
id_temps	identifiant de la relation CUBE_TEMPS
id_loc	identifiant de la relation CUBE_LOC
id_mesh	identifiant de la relation CUBE_MESH
nb_tweet	nombre de tweets contenant le mot
Tf-idf	Tf-idf selon la formule classique
Tf-idf adaptatif	Tf-idf selon la formule adaptative

Description 13: Relation CUBE_FAIT

Elle référence les relations qui supportent nos dimensions.

La relation CUBE_MESH décrite dans la Description 14 représente nos hiérarchies de mots du domaine médical. Elles est issue du MeSH.

Champs	Contenu
id_mesh	identifiant de la relation CUBE_MESH
Niv0	Niveau 0 de la hiérarchie du MeSH
Niv1	Niveau 1 de la hiérarchie du MeSH
Niv2	Niveau 2 de la hiérarchie du MeSH
Niv3	Niveau 3 de la hiérarchie du MeSH
Niv4	Niveau 4 de la hiérarchie du MeSH
Niv5	Niveau 5 de la hiérarchie du MeSH
Niv6	Niveau 6 de la hiérarchie du MeSH
Niv7	Niveau 7 de la hiérarchie du MeSH
Niv8	Niveau 8 de la hiérarchie du MeSH
Niv9	Niveau 9 de la hiérarchie du MeSH
Niv10	Niveau 10 de la hiérarchie du MeSH
Niv11	Niveau 11 de la hiérarchie du MeSH
Niv12	Niveau 12 de la hiérarchie du MeSH

Description 14: Relation CUBE_MESH

La relation CUBE_LOC décrite dans la Description 15 représente nos hiérarchies géographique.

Champs	Contenu
id_loc	identifiant de la relation CUBE_LOC
ville	nom de la ville ou caractère *
etat	nom de l'état ou caractère *
pays	Nom du pays

Description 15: Relation CUBE_LOC

Enfin la relation CUBE_TMPS décrite dans la Description 16 représente une hiérarchie temporelle classique.

Champs	Contenu
id_temps	identifiant de la relation CUBE_TEMPS
jour	Date du jour
mois	numéro du mois (1 à 12)
semestre	numéro du semestre (1 ou 2)
année	année sur 4 caractères numériques

Description 16: Relation CUBE_TMPS

Ces trois dimensions sont alimentées en amont du projet et n'ont pas vocation à être mises à jour de façon fréquente. Le dictionnaire de données associé à l'ensemble des relations est disponible en annexe 1.

Les différentes fonctions utilisées dans notre approche sont soit des développements internes au projet, soit des outils tiers mis à la disposition des développeurs. Nous les résumons dans le tableau 15.

Phase	Outils utilisés	Fonction
Acquisition des données	Api Twitter	Accéder au données
	Développement interne	Intégrer les données dans l'entrepôt
Normalisation du texte du tweet	Développement interne	Nettoyer le tweet
	TextCat	Déterminer la langue
	TreeTagger	Analyse morphosyntaxique
Normalisation de la localisation	Développement interne	Déterminer la localisation
Gestion de la désambiguïsation	Api Yahoo	Récupérer les mots en relation avec les concepts du Mesh
	TreeTagger	Constituer les ensembles de mot utiles fréquemment associés au concept.
	Développement interne	Désambiguïser les mots
Alimentation du cube	Développement interne	Intégrer les données dans le cubes

Tableau 15: Outils utilisés

Nous possédons à l'issue de ce chapitre un ensemble de relations alimentées depuis Twitter ainsi qu'un cube de données prêt à être analysé. Nous proposons dans le chapitre 6 un ensemble de statistiques et des exemples d'utilisations en conditions réelles réalisés pour valider notre approche.

6 Chapitre 6 : Restitutions et visualisation des analyses

Notre solution d'alimentation est opérationnelle depuis le mois de janvier 2011. Les données ont par la suite été intégrées pour simuler une activité réelle depuis janvier. Nous présentons maintenant une synthèse des résultats obtenus après quatre mois de fonctionnement Dans ce chapitre nous commençons par préciser les conditions d'utilisation sur lesquelles les analyses reposent (6.1). Nous présentons ensuite un ensemble d'analyses statistiques (6.2) puis quelques exemples d'analyses multidimensionnelles (6.4) et pour conclure ce chapitre nous proposons un exemple de navigation (6.4) au sein du cube tel qu'un décideur pourrait être amené à la pratiquer.

6.1 Condition d'utilisation et définition du périmètre d'analyse

Le processus de récupération des données via l'interface de programmation Twitter est opérationnel depuis le 21 janvier 2011. Notre solution est installée sur un Pentium Dual Core 3GHz avec 2Go de mémoire vive sous un Linux 2.6.20, la base de données est une base Postgresql version 8.4.7 et le moteur Olap est Mondrian de Pentaho version 3.2.0.

L'utilisation de l'API STREAM impose une limitation de 200 mots-clés. Nous avons fait le choix de retenir l'ensemble des termes correspondant à la hiérarchie MeSH relative aux maladies virales (identifiant MeSH : C.C02). Cette sous-hiérarchie se compose de 363 termes dont 198 termes uniques parmi lesquels nous retrouvons des termes génériques comme "*virus*" ou "*influenza*" et des termes plus spécialisés comme "*hiv-associated lipodystrophy syndrome*" ou "*feline acquired immunodeficiency syndrome*". Bien que nous supposons peu probable la présence du terme "*feline acquired immunodeficiency syndrome*" dans un tweet, nous avons décidé de rechercher les 198 mots uniques composants cette sous hiérarchie afin de ne pas introduire de biais. La liste exhaustive est disponible en annexe 3.
La sous-hiérarchie *Disease >> Virus Disease* est décomposée comme suit:

Hiérarchie (niveau)	Nombre de termes
Niveau 1 (C.C02)	1
Niveau 2	19
Niveau 3	85
Niveau 4	131
Niveau 5	83
Niveau 6	43
Niveau 7	1

Tableau 16: Décomposition hiérarchie Disease >> Virus Disease

Depuis le 21 janvier 2011, 1 269 998 tweets ont été intégrés (chiffre arrêté au 20 mai 2011). L'évolution journalière est représentée dans la Figure 38.

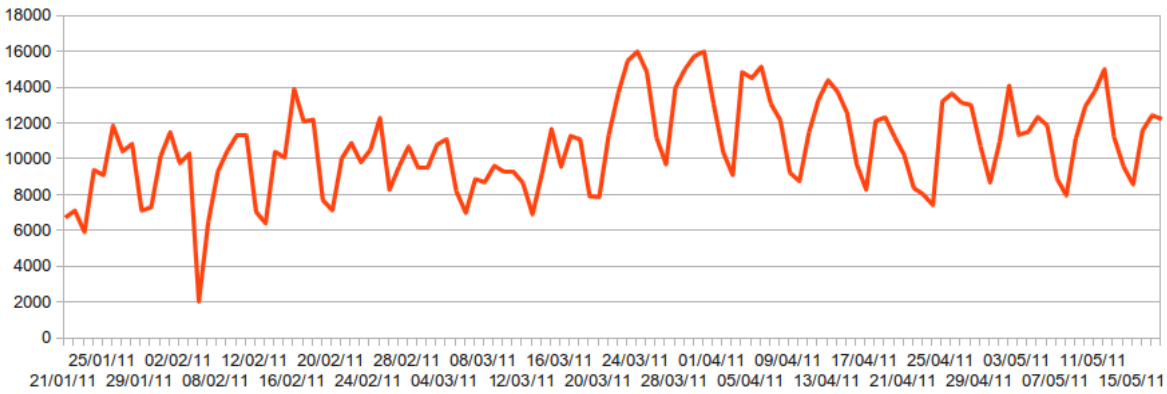


Figure 38: Évolution du nombre de tweets intégrés par jour

Nous observons une variation journalière régulière qui correspond à un cycle hebdomadaire (moins de tweets émis le week-end, le maximum sur le milieu de semaine).

6.2 Comparaison des tweets en lien avec notre domaine d'application

Nous proposons dans cette section un ensemble de statistiques qui correspondent aux tweets relatifs à notre domaine d'application tel que défini au chapitre 6.1 que nous comparons aux statistiques relevées tous domaines confondus.

6.2.1 Liens

Une étude des liens présents dans les tweets relatifs à notre périmètre d'application montre que 40% des tweets font référence à une page web.

Nombre de liens	512239
Nombre de tweets possédant au moins un lien	503592
Pourcentage de tweets possédant au moins un lien	40%

Ce pourcentage peut indiquer que notre domaine d'application est trop spécialisé pour être traité en 140 caractères. Le taux de lien était en 2010 tout domaine d'application confondu de 16%[26]. Il peut s'agir d'une spécificité de notre domaine d'application.

6.2.2 Retweet

Les tweets liés au monde médical font-ils le *buzz* ? Autrement dit un tweet en lien avec notre périmètre d'application fait-il l'objet d'un retweet plus fréquent ?

Nombre de retweet	309370
Pourcentage de retweet	24%
Nombre de tweet retweetés (un tweet peut être retweeté plusieurs fois)	100349
Pourcentage de tweet retweetés	8%

Avec 8% de tweets qui sont des retweet, nous nous trouvons face à un taux similaire au taux constaté tout domaine d'application confondu (6%)[27].

6.2.3 Destinataires @

Nous observons aussi une similarité entre le domaine médical et les statistiques tout domaine confondu (27% pour 23%).[28]

26 http://frenchweb.fr/twitter-chiffres-2010-et-usages-a-la-loupe/
27 http://leblog.wcie.fr/tag/retweet/
28 http://leblog.wcie.fr/tag/retweet/

Nombre de destinataires	393564
Nombre de destinataires uniques	219199
Nombre de tweets indiquant au moins un destinataire	338940
Pourcentage de tweets indiquant au moins un destinataire	27%

Nous n'observons pas de destinataire privilégié (c.f. Tableau 17).

Destinataire	Répartition
@mannix1000	1,15%
@blogdokter	0,68%
@televan	0,60%
@combatadengue	0,56%
@minsaude	0,49%
@ttscontradengue	0,46%
@cslewisdaily	0,42%
@emiliosukita	0,36%
@detikcom	0,34%
@abolishcancer	0,33%

Tableau 17: Top 10 des destinataires

6.2.4 Tags

Nous analysons ensuite les sujets des tweets indiqué par le tag #.

Nous constatons que moins de 20 % des tweets utilisent ce type de marqueur.

Nombre de sujet	332060
Nombre de sujet uniques	57776
Nombre de tweets contenant au moins un tag	224622
Pourcentage de tweets contenant au moins un tag	18%

Les principaux sujets en lien avec notre périmètre d'application sont #dengue, #hepatitis et #influenza.

Tag	Répartition
#dengue	6%
#hepatitis	2%
#influenza	1,6%
#health	1,50%
#meningitis.	1,36%
#news	0,95%
#leukemia	0,94%
#meningitis	0,81%
#dengue:	0,69%
#fb	0,66%
#measles	0,60%
#pneumonia	0,60%
#dengue.	0,59%
#livingproof	0,59%
#fatigue	0,55%
#hiv	0,50%
#h1n1	0,49%
#syndrome	0,44%
#dengue,	0,42%
#chickenpox	0,37%

Tableau 18: Top 20 des tag présents au sein des tweets

L'étude des sujets ne permet pas de mettre un *buzz* en évidence.

6.2.5 Composante géographique d'un tweet

Une analyse des tweets concernant les aspects géographiques de notre périmètre d'application confirme la prédominance du continent nord-américain. La répartition par continent (Figure 39) montre que la moitié des tweets proviennent du continents nord-américain.

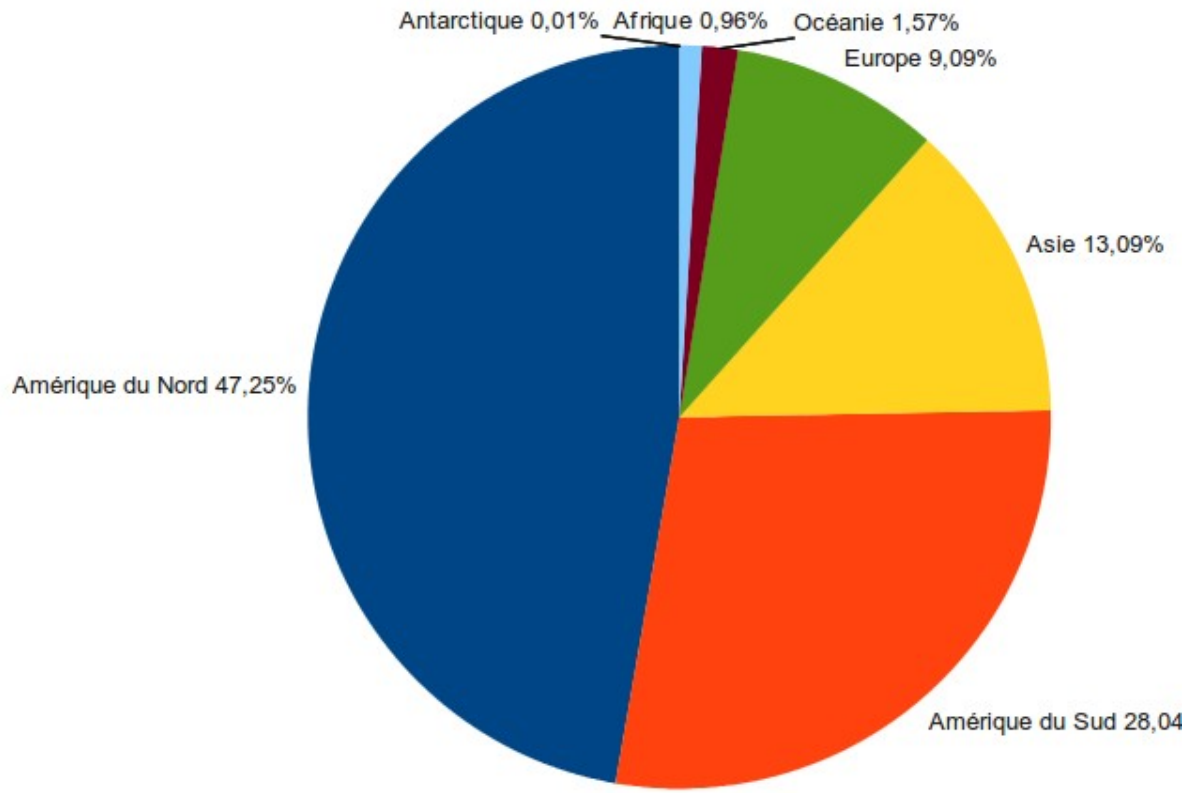


Figure 39: Répartition par continent

La carte (Figure 40) met en évidence la répartition par pays et confirme que les États-Unis occupe de loin la première place en tant que pays émetteur dans notre contexte d'application.

Figure 40: Mappemonde des pays émetteurs

Le bon score du Groenland, qui arrive en sixième position des pays émetteur (Figure 41), s'explique plus par le paramétrage par défaut du fuseau horaire (Twitter inscrit «Groenland» par défaut s'il ne parvient pas à le déterminer) que par l'attrait des groenlandais pour le service de micro-blogging.

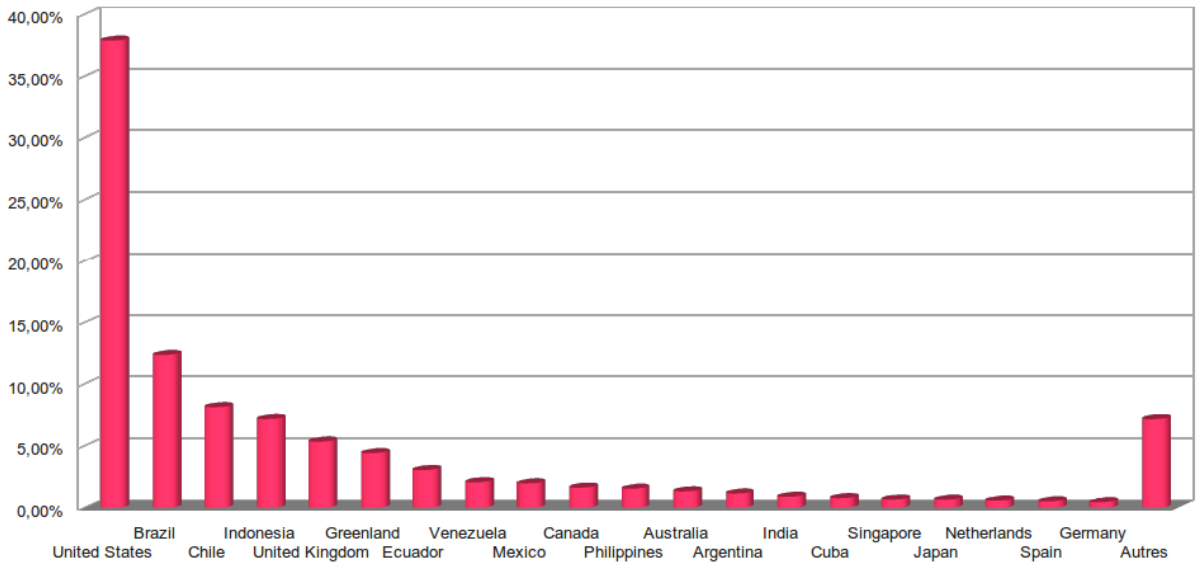


Figure 41: Top 20 des pays émetteurs

Cette répartition des pays se ressent lors de l'analyse des langues employées.

6.2.6 Répartition de la langue

Sans surprise l'anglais domine le classement (c.f. Figure 42). Avec un dictionnaire de recherche en anglais, il est vrai que cela était plus facile puisque le mot *hepatitis* n'a pas lieu d'apparaître dans un tweet écrit en français (qui utilisera le mot hépatite).

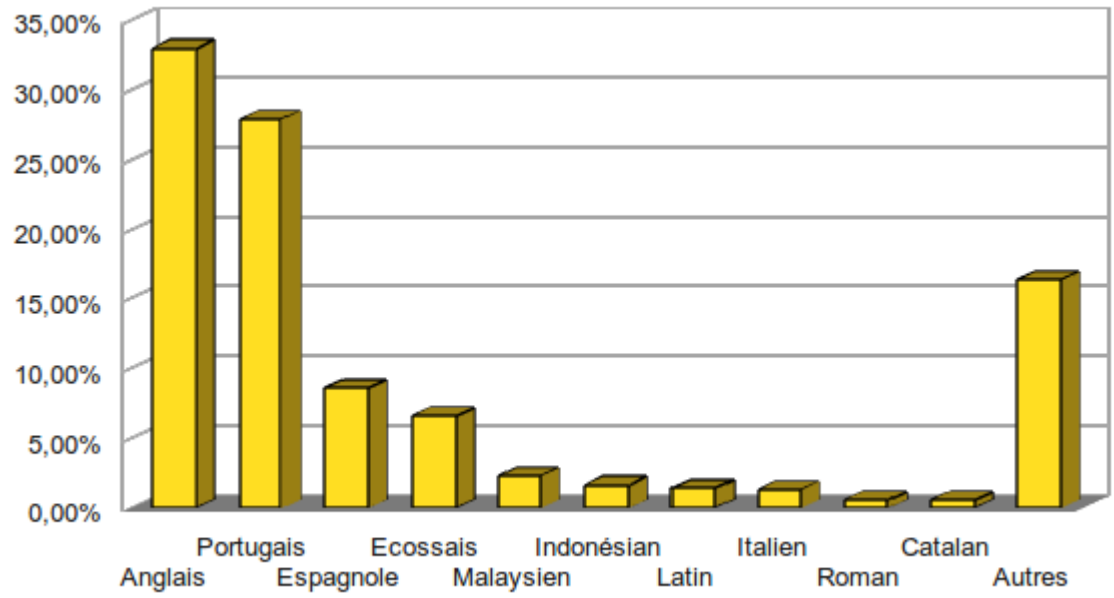


Figure 42: Top 10 des langues utilisées dans les tweets

La Figure 42 met pourtant en évidence que les langues ibero-romanes (portugais, espagnole) représentent plus du tiers de nos tweets. Nous trouvons deux explications à cela, d'une part le continent sud américain est un gros pourvoyeur de tweets (le Brésil notamment), d'autre part un certain nombre de mots de notre périmètre d'application sont similaires en anglais et en portugais/espagnole (dengue, influenza, hepatitis …).

6.2.7 Analyse grammaticale

Nous avons ensuite pratiqué une analyse grammaticale sur les tweets de langue anglaise que nous livrons dans le tableau 19. Il en ressort deux points :

- un mot sur trois dans un tweet est un nom, un sur dix est un verbe.
- les noms sont beaucoup plus diversifiés (et donc plus discriminants) que les autres entités et

représentent 75% des mots.

	Nombre de mots	Nombre de mots uniques	%	% Unique
Nom	3635745	109372	33,72%	74,41%
Verbe	1439299	9158	13,35%	6,23%
Ponctuation	1244228	8	11,03%	0,01%
Préposition ou conjonction de coordination	899530	110	8,34%	0,07%
Adjectif	766713	25258	7,11%	17,18%
Déterminant	570942	22	5,29%	0,01%
Adverbe	455656	2831	4,23%	1,93%
Nombre	424428	40	3,94%	0,03%
Pronom Personnel	313538	24	2,91%	0,02%
Conjonction de coordination	265043	19	2,46%	0,01%
Autres	768409	136	7,64%	0,09%
Total	10783531	146978		

Tableau 19: Analyse grammaticale des tweets de notre périmètre d'application

Pour conclure sur les mots, les dix mots les plus utilisés sont donnés dans le tableau 20.

• pneumonia	• meningitis
• wart	• influenza
• leukemia	• common cold
• hepatitis	• fever
• rabies	• dengue

Tableau 20: Top 10 des mots les plus employés

Nous nous apercevons que toutes ces statistiques, même si elles permettent de mieux appréhender la réalité des tweets dans notre périmètre d'application, ne permettent pas une analyse assez fine et

ciblée pour un décideur.

6.3 Analyse multidimensionnelle

Même si il n'est pas possible de ressortir ici l'ensemble des possibilités d'analyse, notre solution permet de générer un certains nombre de représentations graphiques. Par exemple nous pouvons étudier la répartition géographique d'un concept. La figure 43 présente la répartition des tweets contenant le mot *"leukomia"*. Les Etats-Unis et le Canada ont été retirés car, de par leur poids, ils ne permettaient plus de mettre en évidence une différence entre les autres pays.

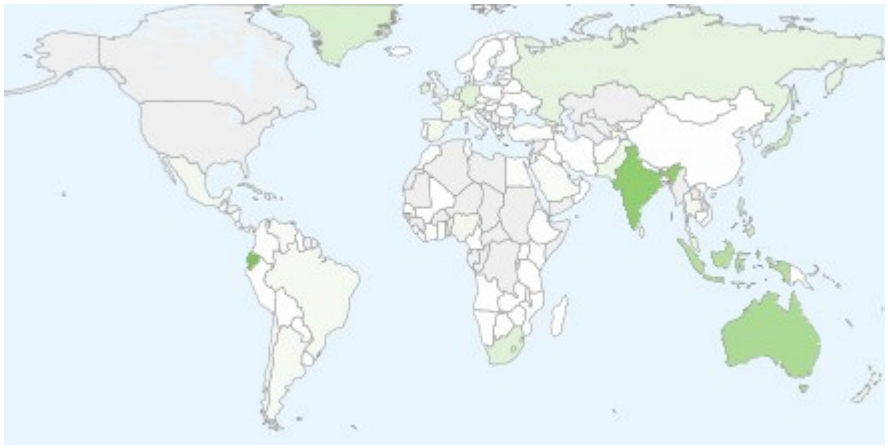


Figure 43: répartition du mot leukomia hors Etats-Unis et Canada

Un autre exemple, nous pouvons visualiser l'évolution d'un concept dans le temps sur une ville ou un pays donnés. Ainsi la figure 44 représente l'évolution du nombre de tweets contenant le terme "*pneumonia"* au Royaume-Uni. Il met clairement un évidence un pic à la fin du mois de janvier qui pourrait correspondre à une épidémie soit à l'impact d'une campagne de sensibilisation autour de la pneumonie.

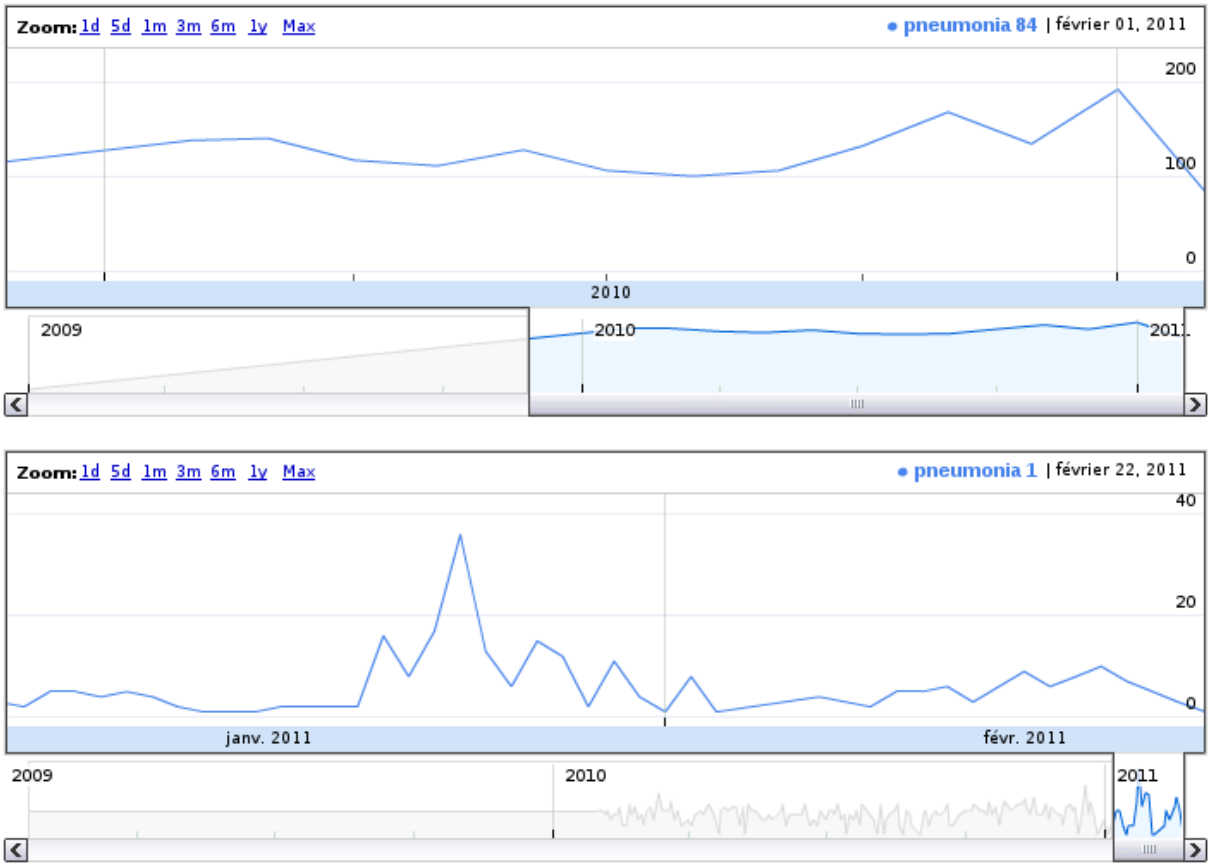


Figure 44: Evolution du mot pnuemonia au Royaume-Uni aux mois de Janvier-Février

Nous pourrions fournir bien d'autres tableaux à la demande mais nous souhaitons fournir avant tout à l'utilisateur un outil d'aide à la décision. Nous souhaitons que les utilisateurs naviguent de manière autonome au travers des données et trouvent seul les réponses aux interrogations qu'il ne manquera pas de soulever au cours de son exploration.

6.4 Exemple de possibilité d'analyse en ligne des résultats

Un utilisateur peut à loisir croiser des données (par exemple le nombre de tweet par mois et par pays) ou suivre l'évolution de tendance (par exemple le nombre de tweet par jour sur un sujet donné).

Mais l'intérêt principal réside dans la navigation au sein des données. Un utilisateur pourra naviguer selon une ou plusieurs des trois hiérarchies définies au moment de la conception du cube de données (hiérarchie géographique, hiérarchie temporelle, hiérarchie de mots en lien avec le domaine médical).

Pour comprendre, imaginons un docteur en médecine s'intéressant au concept de "maladies virales du système nerveux central".

Il souhaite observer dans un premier temps si le concept de "maladies virales du système nerveux central" est un concept discuté au travers des tweets. Il s'aperçoit qu'un nombre non négligeable de tweets sont en lien avec ce sujet et cela l'interpelle. Il décide donc d'approfondir ces recherches en ajoutant d'abord une dimension géographique à son analyse. En se positionnant au niveau le plus haut de la hiérarchie (Pays), il constate que l'ensemble des tweets concernés viennent des États-Unis. Et lorsqu'il descend son analyse d'un niveau hiérarchique, il se situe au niveau de l'état et il s'aperçoit que seul l'état de la Californie est concerné par ces échanges de tweets. Le niveau suivant de la hiérarchie n'apporte aucune information (distribution équilibrée du nombre de tweet entre les villes de l'état de Californie) et notre docteur décide donc de rester au niveau de l'état.

Il s'interroge sur une éventuelle saisonnalité des échanges et de la même manière qu'il a procédé avec la dimension localisation, il affine sa recherche pour arriver à la conclusion que les échanges ont principalement eut lieu au mois de Mars de l'année 2011, plutôt sur le début de mois.

Il effectue des recherches en parallèle pour comprendre mais ne trouve rien qui pourrait expliquer l'intérêt soudain des californiens pour les "maladies virales du système nerveux central". Il revient donc à son analyse et ayant déterminé le lieux (la Californie) et la période (mars 2011), il interroge la hiérarchie lié au domaine médical afin de déterminer avec plus de précision le ou les concepts concernés. la hiérarchie du concept "maladies virales du système nerveux central" est précisé dans la figure 45.

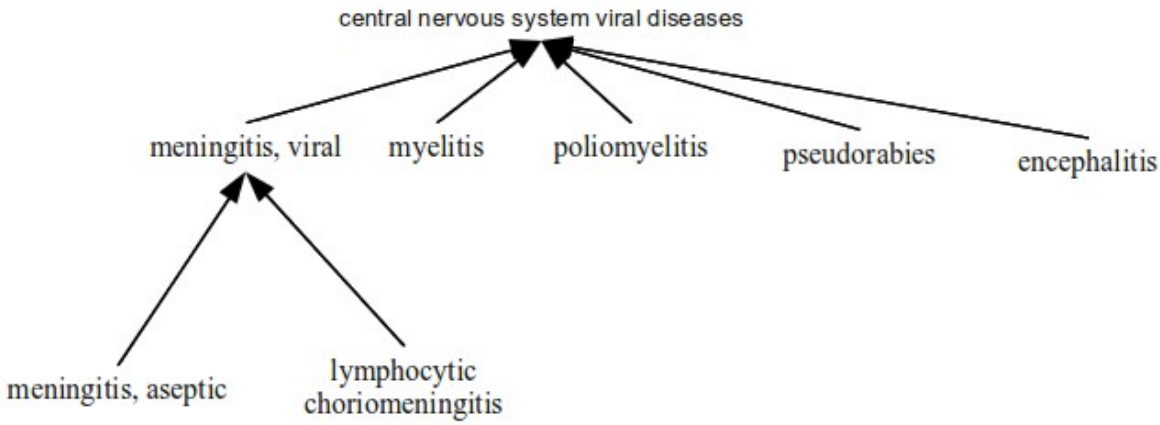


Figure 45: Hiérarchie du concept "maladies virales du système nerveux central"

En navigant au sein de cette hiérarchie, il constatera d'abord que le concept de "méningite virale" est concerné puis conclura que le *buzz* est à rattacher au concept de "chorioméningite

lymphocytaire". A partir des éléments, "Californie", "Mars 2011" et "chorioméningite lymphocytaire", il pourra trouver peut être qu'un ex-acteur et ex-gouverneur a été traité avec succès pour cette maladie au début du mois de mars 2011.
Cet exemple nous permet de présenter une des analyses possibles en navigant au travers nos hiérarchies et une mesure simple, le nombre de tweet.

Nous avons aussi souhaité mettre en place de nouvelles fonctions d'agrégation non natives de Mondrian pour permettre le calcul du TF-IDF et du TF-IDF adaptatif. Nous avons implémenté deux requêtes MDX permettant le calcul du TF-IDF en ligne selon le TF-IDF Classique ou TF-IDF Adaptatif (c.f. annexe 4).

Nous cherchons à identifier les mots les plus représentatifs de nos tweets selon ces deux mesures.
Pour l'exemple, nous focalisons notre analyse sur la ville de Londres.
La première approche permet de déterminer les **mots déterminants des tweets** par rapport aux autres tweets (et de répondre à la question: *quels sont les mots spécifiques aux tweets de la ville?*), la seconde les **mots déterminants de la ville** de Londres par rapport aux autres villes (*quels sont les mots spécifiques à la ville ?*).

Classique	Adaptatif
wart	*wart*
pneumonia	*meningitis*
meningitis	*pneumonia*
die	*die*
pls	*trust*
help	*footballer*
trust	*china*
vaccine	*wall*
common cold	*pls*
zinc	*help*

La même étude sur l'ensemble des tweets depuis le 21 janvier 2011 fait apparaître de nouvelles maladies discriminant la ville de Londres (la rougeole (*measles*), la rage (*rabies*) ou encore l'hépatite).

Adaptatif	Classique
meningitis	*meningitis*
wart	*wart*
pneumonia	*hepatitis*
die	*die*
hepatitis	*pneumonia*
rabies	*measles*
measles	*rabies*
leukemia	*pls*
pls	*leukemia*
month	*month*

Ces deux mesures offrent une lecture différente à un décideur.

Nous vous avons présenté dans cette partie des exemples de navigation et de représentations multidimensionnelle. Notre solution peut intéresser toutes personnes qui souhaiteraient analyser le contenu des tweets d'un domaine spécifique.

7 Chapitre 7 : Conclusions et perspectives

Nous avons mis en place un système permettant l'analyse multidimensionnelle de tweets. A partir de données publiques, nous pouvons déterminer les tendances et affiner en ligne notre analyse en navigant au travers des dimensions temporelle et géographique. Nous proposons aussi une analyse des tweets en lien avec un domaine particulier.

Dans notre mise en œuvre nous avons choisi le milieu médical comme domaine d'application, il suffit d'utiliser ou de construire un thésaurus différent pour analyser un autre domaine. Par exemple, nous pourrions utiliser le thésaurus WebLaw[29] du *Legal Information Access Centre (LIAC)* de la bibliothèque nationale de *New South Wales* (province australienne) si nous focalisions notre attention sur le domaine juridique.

Cette solution repose sur des logiciels et des outils libres de droits afin d'en faciliter la ré-utilisabilité.

La gestion des verrous liés aux données issues de Twitter (normalisation du message et de la localisation) ainsi que ceux liés à notre domaine d'application (désambiguïsation liée au MeSH) sont intégrés à notre solution.

Néanmoins ce système peut être amélioré. Nous proposons dans ce chapitre un état des lieux des limites auxquelles nous avons été confrontées puis dans un second temps aux perspectives mises en place pour améliorer notre solution.

7.1 Bilan critique

Certaines limites doivent être considérées comme des contraintes techniques, d'autres sont liées à nos choix de conception pour résoudre les problèmes de verrous.

7.1.1 Critiques globales du système

Tout d'abord, la mise en place d'une nouvelle fonction d'agrégation comme le calcul du TF-IDF (ou du TF-IDF Adaptatif) est consommatrice en terme de ressources et nous avons rencontré des problèmes de déconnexion sur nos serveurs web au delà d'un certain volume de données.

Ensuite nous absorbons au travers de l'interface de programmation Twitter un flux continu de

29 http://www.weblaw.edu.au

données. Afin de ne pas saturer le système, la capacité journalière de nos traitements de préparation des données doit être supérieure ou égale au nombre de tweets récupérés quotidiennement. Il s'agit là d'une problématique récurrente aux flux de données en temps réel.
Enfin nous ne traitons aujourd'hui que les tweets écrits en langue anglaise (au sens large) puisqu'ils représentent près de la moitié des tweets émis. Il existe des outils permettant de traiter le français ou l'espagnol mais nous avons préféré nous focaliser sur l'anglais. Gérer plusieurs langues aurait aussi imposé de posséder des traductions du MeSH. Il existe par exemple une version bilingue français/anglais du MeSH fournie par l'Inserm[30] mais nous avons préféré adopter une approche mono-langue.

D'autres limites sont liées aux verrous que nous avons rencontrées dans Twitter.

7.1.2 Limites constatées liées aux verrous

Nous le rappelons, un tweet est avant tout un message écrit par un être humain pour un être humain et qui comporte un certain nombre d'éléments de langage (@,#) propre au service Twitter. Si nous prenons bien en compte les spécificités propres au service comme décrit dans la sous-section 4.3.1, nous ne proposons pas, en revanche, dans notre approche la prise en compte des expressions de type SMS. Des travaux sur ce sujet sont réalisés en parallèle par un autre membre de l'équipe actuellement en stage de Recherche au sein du LIRMM et la concomitance de nos études ne nous a pas permis d'intégrer ces spécificités dans notre approche. Un bref aperçu de ces travaux sera présenté dans la section suivante.

En ce qui concerne la localisation, nous avons effectué un certain nombre de choix qui méritent d'être débattus. Nous avons décidé de ne pas traiter les homonymies au niveau des villes (par exemple Paris → France et Paris → Illinois → États-Unis).
Une première solution pour la gestion des homonymies aurait pu consister à analyser la langue utilisée. En effet, nous pouvons supposer qu'un tweet écrit en langue française qui serait posté depuis Paris le serait probablement depuis Paris, capitale de la France. Cependant la faiblesse de cette solution vient du fait que:

- Une étude menée en 2010 par la société Semiocast[31] a montré que 44% des tweets postés depuis la France étaient écrits en français et 34% en anglais[32]. Ce constat est aussi valable de

30 http://mesh.inserm.fr/mesh/
31 http://semiocast.com/
32 http://semiocast.com/downloads/Semiocast_500_000_tweets_par_jour_sont_emis_en_France_20100331.pdf

l'autre côté des Alpes, 42% de messages postés en Italie sont en italien (à l'inverse 95% des tweets du Japon sont en japonais).

- Deux villes homonymes peuvent appartenir au même pays ou à des pays de langues communes (par exemple London → Minnesota → Etats-Unis et London → Royaume-uni)

Une analyse des mots du tweets, sur le même principe que la désambiguïsation de la hiérarchie du MeSH, pourrait permettre de détecter la ville la plus probable mais les contraintes techniques (performance et volume de données) sont trop importantes par rapport au gain espéré.
De même il existe des homonymies entre certaines villes et certains pays ou certaines abréviations de pays (la ville de Usa en Tanzanie, la ville de Uk en Russie par exemple). Nous avons décidé dans ce cas de supprimer la référence à la ville de façon à affecter les tweets au pays.

Enfin la limite de 140 caractères est aussi un facteur limitant pour la qualité de nos analyses. Nous l'avons vu dans la section 4.1, nous ne possédons en moyenne que 8 mots pour désambiguïser le contexte du tweet. Il est évident qu'un plus grand nombre de mot devait nous permettre d'améliorer ces résultats. Afin de valider cette hypothèse, nous avons mené une analyse sur 15253 messages issus du réseau social Facebook qui présente des similarités avec Twitter au niveau du langage employé sans les inconvénients liés à la taille du message.
Les messages récupérés sur Facebook sont composés de 90 mots utiles en moyenne (selon la répartition présentée dans le tableau 21).

Nombre de mots utiles	Pourcentage
moins de 30 mots	4,69%
entre 31 et 50 mots	9,67%
entre 51 et 70 mots	23,52%
entre 71 et 90 mots	22,32%
entre 91 et 110 mots	27,07%
entre 111 et 120 mots	3,41%
entre 121 et 140 mots	2,53%
plus de 140 mots	6,79%

Tableau 21: Répartition du nombre de mots utiles par message Facebook

Sur ces 15253 messages, les deux-tiers (10519 exactement) devaient faire l'objet d'une désambiguïsation par rapport à notre hiérarchie du MeSH. Cette analyse nous a permis de confirmer notre hypothèse. En appliquant la méthode du cosinus retenu dans notre solution, nous obtenons un

taux d'affectation correct de 94,9%. Nous constatons donc une augmentation de 25 points (de 70% à 95 %) si nous augmentons notre nombre de mots de références (de 8 mots à 90 mots en moyenne). Nous suggérons par la suite une approche pour améliorer le nombre de mots disponibles dans un contexte de tweet.

7.2 Perspectives d'amélioration

Notre système supporte aujourd'hui un certains nombre de contraintes et nous détaillons dans cette section des pistes pour améliorer notre système par rapport à ces contraintes. Des travaux ce sont penchés sur des problématiques qui peuvent apporter des éléments d'amélioration pour notre solution.

7.2.1 A court terme et à moyen terme

7.2.1.1 Gestion de l'historisation

Le volume de tweets échangé ne cessant de croître, le volume de données intégré dans notre application deviendra de plus en plus conséquent. Afin de limiter la taille de notre entrepôt de données, nous nous intéressons aux travaux de [GIANNELLA et al., 2002]. Ils proposent un modèle de fenêtres temporelles (ou *tilted-time windows*) qui est utilisé pour modéliser et compresser la dimension temporelle.

Ce modèle s'inspire fortement du mécanisme d'oubli de la mémoire humaine et permet de stocker avec une précision maximale les données les plus récentes. En effet, plus les données vieillissent, plus le niveau de précision diminue.

Un tel modèle est donc tout à fait adapté à un contexte d'aide à la décision dans la mesure où le décideur est généralement intéressé par l'analyse des faits récents avec une grande précision mais veut malgré tout conserver une trace des données historiques pour, par exemple, évaluer les tendances. Nous pourrions imaginer l'adopter dans notre système pour limiter le volume de notre cube et améliorer les performances lors de la navigation.

Théoriquement, nous pouvons intégrer 4 956 134 373 couples (localisation, concept du MeSH) par jour (52561 mots de MeSH x 94295 localisations (88586 villes, 5461 états, 248 pays)). L'idée est d'agréger les données selon la dimension géographique et la dimension liée au MeSH pour les mois passés. Par exemple nous proposons d'agréger les données sur l'état (5461 état + 248 pays = 5709 localisations) de la dimension géographique et le niveau hiérarchique 5 de la dimension MeSH (34668 concepts) pour le mois M-1, puis sur le pays (248 localisation) et le niveau hiérarchique 2

(1797 concepts) pour les mois antérieurs.

Nous aurions ainsi sur 3 mois (base 30 jours) un volume maximal théorique de:

- Mois M : 52 561 concepts x 94 295 localisations x 30 jours = 148 684 031 190
- Mois M-1 : 34 668 concepts x 5 709 localisations x 30 jours = 5 937 588 360
- Mois M-2 : 1 797 concepts x 248 localisations x 30 jours = 13 369 680

Soit 154 634 989 230 couples (localisation, concept du MeSH) au lieu de 446 052 093 570 couples (Diminution de 65% du volume maximal théorique).

Nous pouvons observer le résultat de l'agrégation ainsi réalisé sur la dimension géographique en observant les figures 46, 47 et 48 qui affichent le niveau hiérarchique le plus fin disponible. Si nous analysons le mois M-1, nous perdons l'information sur la ville et au mois M-2, nous ne conservons plus que le pays.

Figure 46: Représentation géographique Fenêtre Temporelle Mois M

Figure 47: Représentation géographique Fenêtre Temporelle Mois M-1

Figure 48: Représentation géographique Fenêtre Temporelle Mois M-2

Au regard des volumétries réelles constatées dans notre domaine d'application, une telle approche ne paraît pas indispensable aujourd'hui mais pourrait l'être dans un autre contexte.

7.2.1.2 La grammaire du tweet

Comme annoncé précédemment, d'autres travaux en lien avec l'analyse textuelle des tweets sont menés au sein du LIRMM. L'objectif est de construire une grammaire des messages et les dictionnaires associés qui soient utilisables pour les catégoriser automatiquement.
Les travaux partent du constat que l'étiquetage des tweets est une tâche qui peut s'avérer difficile (il s'agit de messages courts, les utilisateurs ont tendance à écrire avec des abréviations qui ne sont pas typiques des dictionnaires classiques (ex : LOL)). D'autre part, les conventions d'écriture ne sont souvent que peu respectées (ex : pas d'espace après une ponctuation). De plus, certains mots sont souvent mal orthographiés (ex : tout ! tous). Certains mots sont également écrits en majuscule et dans un style télégraphique.
Les travaux de recherches cherchent à étiqueter chacun des mots d'un tweet, c'est-à-dire l'associer avec une entrée d'un des dictionnaires spécifiques à disposition. Pour cela, les travaux utilisent un ensemble de tweets et de dictionnaires adaptés aux tweets (sigles, transcription phonétique, morphologique, sentiment).
Après avoir constitué les dictionnaires spécifiques (SMS, Smileys, Sentiments) pour l'analyse des status dans les réseaux sociaux, les travaux se focalisent actuellement sur la correction des fautes de frappe et sur l'analyse des descripteurs graphiques propres aux humeurs (ex :j'adddoooorrreee).

Nous avons présenté dans cette section diverses solutions que nous pourrions imaginer intégrer à moyen terme dans notre application. D'autres pistes nous semblent intéressantes.

7.2.2 A long terme

7.2.2.1 Amélioration de la désambiguïsation sur les textes courts

Par exemple, nous avons abordé la problématique de la désambiguïsation des textes courts. Le faible nombre de caractères autorisés entraîne un faible nombre de mots pouvant être utilisés pour désambiguïser le contexte du tweet. Or nos analyses réalisées sur Facebook ont mis en évidence qu'en augmentant le nombre de mots du contexte à désambiguïser, il était possible d'obtenir un gain significatif. Nous avons cherché une solution pour améliorer les mots du contexte.
Une de nos hypothèses de base est que si un même utilisateur utilise un même mot que dans une conversation précédente alors cet utilisateur parle probablement de la même chose que précédemment.
Nous avons donc décidé d'ajouter les mots présents dans l'historique pour améliorer le système.

Par exemple si l'on souhaite désambiguïser le terme "*pneumonia*" pour l'utilisateur X, nous commençons par rechercher si le mot "*pneumonia*" a déjà été utilisé par l'utilisateur X. Le cas échéant, nous ajoutons le mots des anciens tweets au tweet à désambiguïser.
Nos premiers tests montrent que nous obtenons un gain de 10 points de bonne affectation (de 76% à 86%) lorsque nous utilisons l'historique de l'utilisateur.
Mais ces tests ont été réalisés avec un volume de tweets historiques conséquents. Malheureusement, les conditions que nous observons ne nous permettent pas de garantir de tels résultats.
Nous constatons en condition réelle que sur 651474 utilisateurs présents actuellement dans notre système, 173815 ont déposé au moins deux tweets mais seulement 22851 plus de 5 tweets (c.f. Tableau 22).

Nombre de tweets par utilisateur	Nombre d'utilisateurs
1	477659
2	95355
3	32359
4	15004
5	8246
>5	22851

Tableau 22: Répartition du nombre de tweet par utilisateur

Nous ne possédons pas l'historique nous permettant de mesurer le gain par rapport à la dégradation des performances et des temps de traitement induite par une recherche sur l'historique.

7.2.2.2 La navigation des sentiments

Si il est important de savoir quel concept est utilisé à quel moment et à quel endroit, il est aussi très intéressant de pouvoir évaluer la manière dont ce concept est perçu. En effet les décisions prises seront différentes si un concept est identifié de manière plutôt positive ou plutôt négative.
Par exemple nous pourrions souhaiter connaître le ressentie général pour une maladie dans une ville donnée et souhaiter les comparer au niveau de la ville voisine ou du pays (en tant que décideur il peut être intéressant de le savoir pour gérer les priorités ou la communication). Il serait intéressant d'intégrer dans notre approche un mesure quantitative du sentiment général véhiculé au travers des tweets autour d'un concept. Pour cela nous pouvons analyser le contenu du tweet du point de vu des sentiments comme par exemple dans [BOIY et MOENS, 2009].

Une approche consiste à utiliser un thésaurus spécialisé comme SentiWordNet[33].

SentiWordNet [ESULI et SEBASTIANI, 2006] est une ressource lexicale permettant le sondage d'opinion. Il est fondé sur WordNet[34] [MILLER, 1995] qui est une base de données lexicales développée depuis 1985 par des linguistes du laboratoire des sciences cognitives de l'Université de Princeton. Il s'agit d'un réseau sémantique de la langue anglaise, qui se fonde sur une théorie psychologique du langage. La première version diffusée remonte à juin 1991. Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue anglaise.

SentiWordNet assigne à chaque terme de WordNet 2.0 trois valeurs : Positivité, Négativité, Objectivité (respectant l'égalité : Positivité + Négativité + Objectivité = 1).

Prenons les quatre adjectifs suivants pour illustrer notre propos (les valeurs sont données dans l'ordre positivité, négativité, objectivité) :

- "*able*" (capable) est représenté par les valeurs 0.125-0-0.875
- *"unable"* (incapable) est représenté par les valeurs 0-0.75-0.25
- "*good*" (bon) est représenté par les valeurs 1-0-0
- *"abject"* (abject) est représenté par les valeurs 0-1-0

Cette ressource a été créée d'une façon semi-automatisée, en mixant des techniques linguistiques et statistiques . Elle propose aujourd'hui plus de 117000 termes.

Une première piste de recherche est de sommer les valeurs des mots composants le tweet mais cette approche a plusieurs limites parmi lesquelles :

- un même terme peut avoir plusieurs sens et donc véhiculer des sentiments différents.
- les modificateurs (un mot ou une expression qui modifie le sens du mot qu'il accompagne) doivent être pris en compte. Le terme "bon" véhicule un sentiment différent selon qu'il soit "très bon" ,"moins bon","pas bon"...
- un sentiment est souvent lié à un domaine et l'attribution d'un score général n'est pas toujours adapté (par exemple le mot "commercial" est globalement neutre mais négatif pour caractériser un film).

La détection automatique de sentiments demande un travail linguistique et informatique important, encore plus quand ce travail est réalisé sur des messages courts de type SMS.

33 http://sentiwordnet.isti.cnr.it/
34 http://wordnet.princeton.edu

7.3 Bilan personnel

Nous avons présenté dans ce mémoire une solution permettant la mise en place d'une analyse multidimensionnelle de tweets en lien avec un domaine d'application spécifique.
Ces travaux ont servi de support aux présentations qui seront faites lors de la 7ème conférence francophone sur les entrepôts de données et l'analyse en ligne (EDA 2011[35], [BRINGAY et al., 2011a]) et lors de la 22ème conférence internationale sur les bases de données et applications des systèmes experts (DEXA 2011[36], [BRINGAY et al., 2011b]). Cette dernière publication est disponible en annexe.

D'un point de vue plus personnel, ce stage de fin de cursus d'ingénieur m'a fourni l'opportunité de travailler dans un contexte différent de ce que j'ai pu connaître au cours de mes huit années d'expériences professionnelles. Il m'a permis de découvrir le fonctionnement d'un laboratoire de recherche. Il est toujours intéressant de participer à l'activité au sein d'autres organisations, de partager d'autres visions du monde professionnel et d'autres contraintes. La réalisation de ce projet m'a permis de constater que le fonctionnement d'un projet dans un contexte de laboratoire public n'est finalement pas si différent des projets que j'ai pu mener dans le secteur privé.

Ce projet a été l'occasion pour moi d'acquérir des connaissances nouvelles autour de la fouille de données et du traitement automatique du langage. J'ai pu compléter mes compétences techniques en lien avec la *BI.* Je connaissais *Business Objects, Oracle et* le *shell Unix* pour le chargement des données, j'ai découvert *Mondrian*, *Postgresql et Perl.*

C'est aussi le premier projet au cours duquel j'ai été tour à tour le chef de projet, l'installateur logiciel, l'analyste, le concepteur, le développeur, l'administrateur de base de données, le testeur et l'utilisateur. Intervenir dans toutes les phases d'un projet laisse une liberté d'action et de décision tout à fait appréciable.

35 http://eda2011.cemagref.fr/
36 http://www.dexa.org/

8 Bibliographie

BENHARDUS J., 2010. Streaming trend detection in Twitter.

BLOOMFIELD L., 1970. Le langage. Payot, Paris, trad.

BOIY E., MOENS M-F., 2009. A machine learning approach to sentiment analysis in multilingual web texts. Information Retrieval, 12(5): 526-558

BOUSSAID O., BENTAYEB F., DARMONT J., RABASÉDA S., 2003. Vers l'entreposage des données complexes. structuration, intégration et analyse. Ingénierie des Systèmes d'Information, 8(5-6) :79–107.

BRINGAY S, BÉCHET N, BOUILLOT F, PONCELET P, ROCHE M, TEISSEIRE M., 2011. Analyse de gazouillis en ligne, conférence *EDA 2011*.

BRINGAY S, BÉCHET N, BOUILLOT F, PONCELET P, ROCHE M, TEISSEIRE M., 2011. Towards an On-Line Analysis of Tweets Processing, conférence *DEXA 2011* .

CHAUDHURI S., DAYAL U., 1997. An Overview of Data Warehousing and OLAP Technology. SIGMOD Rec., 26(1) :65–74.

CODD E. F., 1993. Providing OLAP (On-Line Analytical Processing) to User-Analysts : an IT mandate. Technical report, E.F. Codd and Associates.

DAILLE B., 1994. Approche mixte pour l'extraction automatique de terminologie : statistique lexicale et filltres linguistiques. Technical report, Phd Thesis, University Paris VII, France.

ESULI A., SEBASTIANI F., 2006. SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. Actes de LREC 2006, fifth international conference on Language Resources and Evaluation, pp. 417-422.

GIANNELLA C., HAN J., PEI J., YAN X., YU P., 2002. Mining Frequent Patterns in Data Streams at Multiple Time Granularities a.

GINSBERG J., MOHEBBI M.H., PATEL R.S., BRAMMER L., SMOLINSKI M.S., BRILLIANT L., 2009. Detecting influenza epidemics using search engine query data, Nature, p1012-1014.

GOLFARELLI M., MAIO D., RIZZI S.,1998. Conceptual Design of Data Warehouses from E/R Schemes. In XXXIst Annual Hawaii International Conference on System Sciences (HICSS 98), Big Island, Hawaii, USA,volume 7, pages 334–343.

HARRIS Z-S., 1954. Distributional structure, Word, Chicago Press, pp. 146 –162. (réédité sous le titre Structural Linguistics).

INMON W., 1996. Building the Data Warehouse. John Wiley & Sons.

Jaccard P., 1901. Étude comparative de la distribution florale dans une portion des Alpes et des Jura, Bulletin de la Société Vaudoise des Sciences Naturelles 37: 547–579.

KIM H-J., LEE T-H., LEE S-G., CHUN J., 2003. Automated Data Warehousing for Rule-Based CRM Systems. In XIVth Australasian Database Conference on Database Technologies, pages 67–73. Australian Computer Society.

KIMBALL R., REEVES L., ROSS M., THORNTHWAITE W., 2000.Concevoir et déployer un data warehouse. Eyrolles. Au coeur de l'architecture décisionnelle

KIMBALL R., ROSS M., 2002. L a Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling.

MATHIOUDAKIS M., KOUDAS N., 2010. TWITTERMONITOR: trend detection over the Twitter Stream, Proceedings of SIGMOD Conference, p.1155-1158

MILLER G., 1995. Wordnet: A lexical database. Actes de ACM 38, pp. 39-41.

MOODY D., KORTINK M., 2000. From Enterprise Models to Dimensional Models : a Methodology for Data Warehouse and Data Mart Design. In Design and Management of Data Warehouses, page 5.

PENDSE N., CREETH R., 1997. Synopsis of the OLAP Report. Business Intelligence, Inc., Norwalk, CT,

PHIPPS C., DAVIS K-C., 2002. Automating Data Warehouse Conceptual Schema Design and Evaluation. In IVth International Workshop on Design and Management of Data Warehouses (DMDW 02), Toronto, Canada, volume 58 of CEUR Workshop Proceedings, pages 23–32. CEURWS.org.

ROCHE M., PRINCE V., 2008. Managing the acronym/expansion identification process for text-mining applications. International Journal of Software and Informatics, Special issue on Data Mining 2(2), 163–179.

SAKAKI T., OKAZAKI M., MATSUON Y., 2010. Earthquake shakes Twitter users: real-time event detection by social sensors, Proc. of WWW, p.851–860

SOUSSI A., FEKI J., GARGOURI F., 2005. Approche semi-automatisée de conception de schémas multidimensionnels valides. In Ière journée sur les Entrepôts de Données et l'Analyse en ligne (EDA 05), Lyon, volume B-1 of Revue des Nouvelles Technologies de l'Information, pages 71–90. Cépaduès Editions.

TESTE O., 2000. Modélisation et manipulation d'entrepôts de données complexes et historisées. Thèse de doctorat, Institut de Recherche en Informatique de Toulouse - Université Toulouse 3.

TRUJILLO J.,PALOMAR M., GOMEZ J., SONG I-Y, 2001. Designing Data Warehouses with OO Conceptual Models. Computer, 34(12) :66–75.

VINCENTY T., 1975. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations.

VAN RIJSBERGEN C-J. , ROBERTSON S-E., PORTER M-F., 1980. New models in probabilistic information retrieval. London: British Library. (British Library Research and Development Report, no. 5587).

WIEDERHOLD G.,1992. Mediators in the architecture of future information systems, in Readings in Agents,M. N. Huhns and M. P. Singh, eds., Morgan Kaufmann, San Francisco, CA, USA, pp. 185–196.

Index des Figures

Figure 1: Cycle de développement en spirale....12
Figure 2: Diagramme de Gantt prévisionnel....14
Figure 3: Illustration du MOLAP, ROLAP, HOLAP....19
Figure 4: Une architecture décisionnelle....20
Figure 5: Extraction, Transformations et Chargement....21
Figure 6: Exemple de modèle en étoile....23
Figure 7: Exemple de modèle en flocon....24
Figure 8: Exemple de modèle en constellation....24
Figure 9: Représentation graphique d'un cube de données....26
Figure 10: OLAP - Opération de rotation....27
Figure 11: OLAP - Opération de découpage en tranche....28
Figure 12: OLAP - Opération de définition d'un axe d'analyse....28
Figure 13: OLAP - Opérations Roll-Up et Drill-Down....29
Figure 14: Approche virtuelle....30
Figure 15: Approche matérialisée....31
Figure 16: Le premier tweet....32
Figure 17: le tweet au delà des étoiles....34
Figure 18: le tweet au delà des nuages....35
Figure 19: TweetSentiment, l'évolution du sentiment général....39
Figure 20: A world of tweets, d'où tweete t'on ?....40
Figure 21: Trends Map, la représentation géographique du Buzz....40
Figure 22: Twitter StreamGraph, l'association d'idées au travers des tweets....41
Figure 23: TweetSentiments: l'analyse de la personnalité au travers de ses tweets....42
Figure 24: Twittergrader ou la valeur du profil....42
Figure 25: TweetPsych, le "profiler" au service de la twittosphère....43
Figure 26: Modèle conceptuel générique....55
Figure 27: Modèle conceptuel de données retenu....56
Figure 28: Schéma global de la solution proposée....59
Figure 29: L'architecture Mondrian en image (source: http://www.osbi.fr/)....61
Figure 30: Phase d'acquisition des tweets....65
Figure 31: Phase de normalisation du tweet....74
Figure 32: Localisation ou fuseau horaire, qui fournit l'information ?....79
Figure 33: Phase de normalisation de la localisation....80
Figure 34: Hiérarchie "pneumonia"(sous ensemble du thésaurus du MeSH)....81
Figure 35: Phase de désambiguïsation du MeSH....90
Figure 36: Schéma relationnel – partie traitement des données....92
Figure 37: Modèle Logique de Données retenu....94
Figure 38: Évolution du nombre de tweets intégrés par jour....99
Figure 39: Répartition par continent....103
Figure 40: Mappemonde des pays émetteurs....104
Figure 41: Top 20 des pays émetteurs....104
Figure 42: Top 10 des langues utilisées dans les tweets....105
Figure 43: répartition du mot leukomia hors Etats-Unis et Canada....107
Figure 44: Evolution du mot pnuemonia au Royaume-Uni aux mois de Janvier-Février....108
Figure 45: Hiérarchie du concept "maladies virales du système nerveux central"....109
Figure 46: Représentation géographique Fenêtre Temporelle Mois M....116
Figure 47: Représentation géographique Fenêtre Temporelle Mois M-1....117
Figure 48: Représentation géographique Fenêtre Temporelle Mois M-2....117

Index des Tableaux

Tableau 1: Planning prévisionnel du projet....12
Tableau 2: Comparatif OLTP/OLAP....16
Tableau 3: Approche matérialisée ou virtuelle ?....30
Tableau 4: Répartition du nombre de mots utiles....48
Tableau 5: Répartition du nombre de noms utiles....48
Tableau 6: Répartition du nombre de verbes utiles....48
Tableau 7: Répartition du nombre d'adjectifs utiles....49
Tableau 8: Nombre de concepts par niveau hiérarchique du MeSH....50
Tableau 9: Illustration TF-IDF adaptatif selon la dimension géographique....57
Tableau 10: Analyse morpho-syntaxique d'un tweet avec TreeTagger 71
Tableau 11: Désambiguïsation Analyse du nombre de résumés nécessaires....84
Tableau 12: Vecteur Turkey dans le contexte Volaille....86
Tableau 13: Vecteur Turkey dans le contexte Pays....87
Tableau 14: Désambiguïsation Précision et Rappel....88
Tableau 15: Outils utilisés....96
Tableau 16: Décomposition hiérarchie Disease >> Virus Disease....98
Tableau 17: Top 10 des destinataires....100
Tableau 18: Top 20 des tag présents au sein des tweets....101
Tableau 19: Analyse grammaticale des tweets de notre périmètre d'application....105
Tableau 20: Top 10 des mots les plus employés....105
Tableau 21: Répartition du nombre de mots utiles par message Facebook....113
Tableau 22: Répartition du nombre de tweet par utilisateur....118

Index des Relations

Relation 1: DIC....61
Relation 2: STREAM_API....62
Relation 3: STREAM_API_USER....63
Relation 4: TWEET_A_TRAITER_ANN et TWEET_A_TRAITER_LOC....65
Relation 5: TWITTER_TAG, TWITTER_DEST, TWITTER_RT et TWITTER_LIEN....67
Relation 6: TWITTER_LANGUE....69
Relation 7: TWITTER_TREETAGGER....71
Relation 8: TWEET_A_TRAITER_DESAMB....72
Relation 9: GEONAMES, GEONAMES_ADMIN et GEONAMES_COUNTRY....75
Relation 10: TWITTER_LOCALISATION....77
Relation 11: MESH_DESAMB....83
Relation 12: TWITTER_DESAMB....88

Index des Descriptions

Description 1: Relation DIC....61
Description 2: Relation STREAM_API....62
Description 3: Relation STREAM_API_USER....63
Description 4: Relations TWEET_A_TRAITER_ANN et TWEET_A_TRAITER_LOC....65
Description 5: Relations TWITTER_TAG, TWITTER_DEST, TWITTER_RT et TWITTER_LIEN
....67
Description 6: Relation TWITTER_LANGUE....69
Description 7: Relation TWITTER_TREETAGGER....72
Description 8: Relation TWEET_A_TRAITER_DESAMB....72
Description 9: Relations GEONAMES, GEONAMES_ADMIN et GEONAMES_COUNTRY....75
Description 10: Relation TWITTER_LOCALISATION....77
Description 11: Relation MESH_DESAMB....83
Description 12: Relation MESH_DESAMB....89
Description 13: Relation CUBE_FAIT....93
Description 14: Relation CUBE_MESH....94
Description 15: Relation CUBE_LOC....94
Description 16: Relation CUBE_TMPS....95

Annexe 1 : Dictionnaire de données.

Relation	Champs	Description	Type
CUBE_FAIT	*id_loc*	identifiant de la relation CUBE_LOC	numeric
CUBE_FAIT	*id_mesh*	identifiant de la relation CUBE_MESH	numeric
CUBE_FAIT	*id_temps*	identifiant de la relation CUBE_TEMPS	numeric
CUBE_FAIT	*nb_tweet*	nombre de tweets contenant le mot	numeric
CUBE_FAIT	*Tf-idf*	Tf-idf selon la formule classique	numeric
CUBE_FAIT	*Tf-idf adaptatif*	Tf-idf selon la formule adaptative	numeric
CUBE_LOC	*etat*	nom de l'état ou caractère *	texte
CUBE_LOC	*id_loc*	identifiant de la relation CUBE_LOC	numeric
CUBE_LOC	*pays*	Nom du pays	texte
CUBE_LOC	*ville*	nom de la ville ou caractère *	texte
CUBE_MESH	*id_mesh*	identifiant de la relation CUBE_MESH	numeric
CUBE_MESH	*Niv0*	Niveau 0 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv1*	Niveau 1 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv10*	Niveau 10 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv11*	Niveau 11 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv12*	Niveau 12 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv2*	Niveau 2 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv3*	Niveau 3 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv4*	Niveau 4 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv5*	Niveau 5 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv6*	Niveau 6 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv7*	Niveau 7 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv8*	Niveau 8 de la hiérarchie du MeSH	texte
CUBE_MESH	*Niv9*	Niveau 9 de la hiérarchie du MeSH	texte
CUBE_TMPS	*année*	année sur 4 caractères numérique	numeric
CUBE_TMPS	*id_temps*	identifiant de la relation CUBE_TEMPS	numeric
CUBE_TMPS	*jour*	Date du jour	date
CUBE_TMPS	*mois*	numéro du mois (1 à 12)	numeric
CUBE_TMPS	*semestre*	numéro du semestre (1 ou 2)	numeric
DIC	*Mot-clef*	Mot cherché dans les tweets	texte
GEONAMES	*admin_code*	Identifiant Etat/Province	texte
GEONAMES	*country_code*	Identifiant Pays	texte
GEONAMES	*geonameid*	Identifiant numérique de la ville fourni par Geonames	numeric
GEONAMES	*latitude*	Coordonnée latitude	numeric
GEONAMES	*longitude*	Coordonnée longitude	numeric
GEONAMES	*name*	Ville ou Etat/province ou Pays	texte
GEONAMES	*name_fr*	Ville ou Etat/province ou Pays traduits ou abrégés	texte
GEONAMES_ADMIN	*admin_code*	Identifiant Etat/Province	texte
GEONAMES_ADMIN	*country_code*	Identifiant Pays	texte
GEONAMES_ADMIN	*latitude*	Coordonnée latitude	numeric
GEONAMES_ADMIN	*longitude*	Coordonnée longitude	numeric
GEONAMES_ADMIN	*name*	Ville ou Etat/province ou Pays	texte
GEONAMES_ADMIN	*name_fr*	Ville ou Etat/province ou Pays traduits ou abrégés	texte
GEONAMES_COUNTRY	*country_code*	Identifiant Pays	texte

Relation	Champs	Description	Type
GEONAMES_COUNTRY	*latitude*	Coordonnée latitude	numeric
GEONAMES_COUNTRY	*longitude*	Coordonnée longitude	numeric
GEONAMES_COUNTRY	*name*	Ville ou Etat/province ou Pays	texte
GEONAMES_COUNTRY	*name_fr*	Ville ou Etat/province ou Pays traduits ou abrégés	texte
MESH_DESAMB	*coef*	nombre de fois ou le mot est associé avec le terme du MeSH dans le résultat de la requête	texte
MESH_DESAMB	*descriptorname*	Terme dans la hiérarchie du MeSH	texte
MESH_DESAMB	*mot*	Mot trouvé dans les résultats de la reqête	texte
MESH_DESAMB	*treenumberlist*	Identifiant du terme dans la hiérarchie du MeSH	texte
STREAM_API	*created_at_date*	Date de création du tweet	texte
STREAM_API	*created_at_heure*	Heure de création du tweet	texte
STREAM_API	*id*	Identifiant du tweet	texte
STREAM_API	*text*	Tweet lui même (message)	texte
STREAM_API	*user_id*	Identifiant de l'utilisateur	texte
STREAM_API	*user_lang*	Langue définie par l'utilisateur	texte
STREAM_API	*user_location*	Localisation de l'utilisateur au moment où le tweet est écrit	texte
STREAM_API	*user_time_zone*	Fuseau horaire défini par l'utilisateur	texte
STREAM_API_USER	*user_created_at_date*	Date de création de l'utilisateur	texte
STREAM_API_USER	*user_created_at_heure*	Heure de création de l'utilisateur	texte
STREAM_API_USER	*user_favourites_count*	Statistiques: Nombre de favoris	numeric
STREAM_API_USER	*user_followers_count*	Statistiques: Nombre de followers	numeric
STREAM_API_USER	*user_friends_count*	Statistiques: Nombre d'amis	numeric
STREAM_API_USER	*user_id*	Identifiant de l'utilisateur	texte
STREAM_API_USER	*user_location*	Dernière localisation de l'utilisateur	texte
STREAM_API_USER	*user_name*	Nom de l'utilisateur	texte
STREAM_API_USER	*user_screen_name*	Pseudonyme de l'utilisateur	texte
STREAM_API_USER	*user_statuses_count*	Statistiques: Nombre de tweets	numeric
STREAM_API_USER	*user_url*	Lien Web vers le compte Twitter	texte
TWEET_A_TRAITER_ANN	*id*	Identifiant du tweet	texte
TWEET_A_TRAITER_ANN	*text*	Message contenu dans le tweet	texte
TWEET_A_TRAITER_DESAMB	*id*	Identifiant du tweet	texte
TWEET_A_TRAITER_DESAMB	*mot*	Forme lemmatisée du mot présent dans le tweet	texte
TWEET_A_TRAITER_DESAMB	*place*	Position du mot dans la phrase	numeric
TWEET_A_TRAITER_LOC	*id*	Identifiant du tweet	texte
TWEET_A_TRAITER_LOC	*user_location*	Contenu du champ user_location retourné par l'API Twitter	texte
TWEET_A_TRAITER_LOC	*user_time_zone*	Contenu du champ user_time_zone retourné par l'API Twitter	texte
TWITTER_DESAMB	id	nombre de fois ou le mot est associé avec le terme du MeSH dans le résultat de la requête	texte
TWITTER_DESAMB	mot	Terme dans la hiérarchie du MeSH	texte
TWITTER_DESAMB	place	Place du mot dans le tweet	numeric
TWITTER_DESAMB	treenumberlist	Identifiant du terme dans la hiérarchie du MeSH	texte
TWITTER_DEST	*destinataire*	Valeur du destinataire (@destinataire)	texte
TWITTER_DEST	*id*	Identifiant du tweet	texte
TWITTER_LANGUE	*id*	Identifiant du tweet	texte
TWITTER_LANGUE	*langue*	Langue évaluée par TextCat	texte
TWITTER_LIEN	*id*	Identifiant du tweet	texte

Relation	Champs	Description	Type
TWITTER_LIEN	*lien*	Valeur du lien (http://)	texte
TWITTER_LOCALISATION	*etat*	Nom de l'état ou de la province ou *	texte
TWITTER_LOCALISATION	*id*	Identifiant du tweet	texte
TWITTER_LOCALISATION	*lat*	Coordonnée latitude	numeric
TWITTER_LOCALISATION	*lon*	Coordonnée longitude	numeric
TWITTER_LOCALISATION	*pays*	Nom du pays	texte
TWITTER_LOCALISATION	*provenance*	Identification de la provenance (localisation ou fuseau horaire)	texte
TWITTER_LOCALISATION	*ville*	Nom de la ville ou *	texte
TWITTER_RT	*emetteur*	Valeur de l'émetteur (RT emetteur)	texte
TWITTER_RT	*id*	Identifiant du tweet	texte
TWITTER_TAG	*id*	Identifiant du tweet	texte
TWITTER_TAG	*tag*	Valeur du tag (#mot)	texte
TWITTER_TREETAGGER	*id*	Identifiant du tweet	texte
TWITTER_TREETAGGER	*lemm_mot*	Forme lemmatisée du mot	texte
TWITTER_TREETAGGER	*mot*	Mot présent dans le tweet	texte
TWITTER_TREETAGGER	*place*	Position du mot dans la phrase	numeric
TWITTER_TREETAGGER	*type_mot*	Genre du mot (verbes, nom, adjectif, pronom, ...)	texte

Annexe 2 : Liste des mots vides du projet.

a	anyhow	being	down	followed	hereby	itd	make	neither	or	provides	see	soon	there	trying	we	words
able	anymore	believe	downwards	following	herein	its	makes	never	ord	put	seeing	sorry	thereafter	ts	wed	world
about	anyone	below	due	follows	heres	itself	many	nevertheless	other	q	seem	specifically	thereby	twice	welcome	would
above	anything	beside	during	for	hereupon	j	may	new	others	que	seemed	specified	thered	two	went	www
abst	anyway	besides	e	former	hers	just	maybe	next	otherwise	quickly	seeming	specify	therefore	u	were	x
accordance	anyways	between	each	formerly	herself	k	me	nine	ought	quite	seems	specifying	therein	un	what	y
according	anywhere	beyond	ed	forth	hes	keep	mean	ninety	our	qv	seen	state	thereof	under	whatever	yes
accordingly	apparently	biol	edu	found	hi	keeps	means	no	ours	r	self	states	therere	unfortunately	whats	yet
across	approximately	both	effect	four	hid	kept	meantime	nobody	ourselves	ran	sent	still	theres	unless	when	you
act	are	brief	eg	from	him	keys	meanwhile	non	out	rather	seven	stop	thereto	unlike	whence	youd
actually	aren	briefly	eight	further	himself	kg	merely	none	outside	rd	several	strongly	thereupon	unlikely	whenever	your
added	arent	but	eighty	furthermore	his	km	mg	nonetheless	over	re	shall	sub	these	until	where	youre
adj	arise	by	either	g	hither	know	might	noone	overall	readily	she	substantially	they	unto	whereafter	yours
adopted	around	c	else	gave	home	known	million	nor	owing	really	shed	successfully	theyd	up	whereas	yourself
affected	as	ca	elsewhere	get	how	knows	miss	normally	own	recent	shes	such	theyre	upon	whereby	yourselves
affecting	aside	came	end	gets	howbeit	l	ml	nos	p	recently	should	sufficiently	think	ups	wherein	z
affects	ask	can	ending	getting	however	largely	more	not	page	ref	show	suggest	this	us	wheres	zero
after	asking	cannot	enough	give	hundred	last	moreover	noted	pages	refs	showed	sup	those	use	whereupon	
afterwards	at	cause	especially	given	i	lately	most	nothing	part	regarding	shown	sure	thou	used	wherever	
again	auth	causes	et	gives	id	later	mostly	now	particular	regardless	showns	t	though	useful	whether	
against	available	certain	et-al	giving	ie	latter	mr	nowhere	particularly	regards	shows	take	thoughh	usefully	which	
ah	away	certainly	etc	go	if	latterly	mrs	o	past	related	significant	taken	thousand	usefulness	while	
all	awfully	co	even	goes	im	least	much	obtain	per	relatively	significantl	taking	throug	uses	whim	

											y					
almost	b	com	ever	gone	selves	less	mug	obtained	perhaps	research	similar	tell	through	using	whither	
alone	back	come	every	got	immediate	lest	must	obviously	placed	respectivel y	similarly	tends	throughout	usually	who	
along	be	comes	everybody	gotten	immediatel y	let	my	of	please	resulted	since	th	thru	v	whod	
already	became	contain	everyone	h	importance	lets	myself	off	plus	resulting	six	than	thus	value	whoever	
also	because	containing	everything	had	important	like	n	often	poorly	results	slightly	thank	til	various	whole	
although	become	contains	everywhere	happens	in	liked	na	oh	possible	right	so	thanks	tip	very	whom	
always	becomes	could	ex	hardly	inc	likely	name	ok	possibly	run	some	thanx	to	via	whomever	
am	becoming	couldnt	except	has	indeed	line	namely	okay	potentially	s	somebody	that	together	viz	whos	
among	been	d	f	have	index	little	nay	old	pp	said	somehow	thats	too	vol	whose	
amongst	before	date	far	having	informatio n	look	nd	omitted	predomina ntly	same	someone	the	took	vols	why	
an	beforehand	did	few	he	instead	looking	near	on	present	saw	somethan	their	toward	vs	widely	
and	begin	different	ff	hed	into	looks	nearly	once	previously	say	something	theirs	towards	w	willing	
announce	beginning	do	fifth	hence	invention	ltd	necessarily	one	primarily	saying	sometime	them	tried	want	wish	
another	beginnings	does	first	her	inward	m	necessary	ones	probably	says	sometimes	themselves	tries	wants	with	
any	begins	doing	five	here	is	made	need	only	promptly	sec	somewhat	then	truly	was	within	
anybody	behind	done	fix	hereafter	it	mainly	needs	onto	proud	section	somewhere	thence	try	way	without	

Annexe 3 : Les 198 mots retenus dans notre périmètre d'application.

acquired immunodeficiency syndrome	coxsackievirus infections	hemorrhagic syndrome	marburg virus disease	respirovirus infections
adenoviridae infections	cytomegalovirus infections	henipavirus infections	marek disease	retroviridae infections
adenovirus infections	cytomegalovirus retinitis	hepadnaviridae infections	measles	rhabdoviridae infections
african horse sickness	deltaretrovirus infections	hepatitis	meningitis	rift valley fever
african swine fever	dengue	hepatitis a	mink viral enteritis	rinderpest
aids arteritis	dengue hemorrhagic fever	hepatitis b	molluscum contagiosum	rna virus infections
aids-associated nephropathy	diseases	hepatitis c	monkeypox	roseolovirus infections
aids dementia complex	distemper	hepatitis d	mononegavirales infections	rotavirus infections
aids-related complex	dna virus infections	hepatitis e	morbillivirus infections	rubella
aids-related opportunistic infections	echovirus infections	herpangina	mumps	rubella syndrome
aleutian mink disease	ecthyma	herpes genitalis	murine acquired immunodeficiency syndrome	rubivirus infections
alphavirus infections	ectromelia	herpes labialis	myelitis	rubulavirus infections
arbovirus infections	encephalitis	herpes simplex	myxomatosis	sarcoma
arenaviridae infections	encephalomyelitis	herpesviridae infections	nairobi sheep disease	severe acute respiratory syndrome
arterivirus infections	enteritis	herpes zoster	newcastle disease	sexually transmitted diseases
astroviridae infections	enterovirus infections	herpes zoster ophthalmicus	nidovirales infections	simian acquired immunodeficiency syndrome
avian leukosis	enzootic bovine leukosis	herpes zoster oticus	opportunistic infections	skin diseases
avulavirus infections	ephemeral fever	hiv-associated lipodystrophy syndrome	orthomyxoviridae infections	slow virus diseases
bell palsy	epidermodysplasia verruciformis	hiv enteropathy	papillomavirus infections	smallpox
birnaviridae infections	epstein-barr virus infections	hiv infections	paramyxoviridae infections	stomatitis
bluetongue	equine infectious anemia	hiv seropositivity	paraparesis	subacute sclerosing panencephalitis
border disease	erythema infectiosum	hiv wasting syndrome	parvoviridae infections	superinfection
borna disease	exanthema subitum	htlv-ii infections	pasteurellosis	swine vesicular disease
bovine virus diarrhea-mucosal disease	eye infections	htlv-i infections	peste-des-petits-ruminants	tick-borne diseases
bronchiolitis	fatigue syndrome	infectious bovine rhinotracheitis	pestivirus infections	togaviridae infections
bunyaviridae infections	feline acquired immunodeficiency syndrome	infectious mononucleosis	phlebotomus fever	torovirus infections
burkitt lymphoma	feline infectious peritonitis	influenza	picornaviridae infections	tumor virus infections
caliciviridae infections	feline panleukopenia	influenza in birds	pleurodynia	vaccinia
cardiovirus infections	filoviridae infections	kaposi varicelliform	pneumonia	vesicular exanthema of

		eruption		swine
central nervous system viral diseases	flaviviridae infections	keratitis	pneumovirus infections	vesicular stomatitis
chickenpox	flavivirus infections	kyasanur forest disease	poliomyelitis	viremia
circoviridae infections	foot-and-mouth disease	lassa fever	polyomavirus infections	virus diseases
classical swine fever	fowlpox	lentivirus infections	porcine reproductive and respiratory syndrome	visna
colorado tick fever	gastroenteritis	leukemia	postpoliomyelitis syndrome	warts
common cold	hantavirus infections	leukoencephalopathy	poxviridae infections	west nile fever
condylomata acuminata	hantavirus pulmonary syndrome	leukoplakia	pseudorabies	yellow fever
conjunctivitis	hemorrhagic fever	louping ill	pulmonary adenomatosis	zoonoses
coronaviridae infections	hemorrhagic fevers	lumpy skin disease	rabies	zoster sine herpete
coronavirus infections	hemorrhagic fever with renal syndrome	lymphocytic choriomeningitis	reoviridae infections	
cowpox	hemorrhagic septicemia	malignant catarrh	respiratory syncytial virus infections	

Annexe 4 : Requêtes Mdx

Requête 1:

```
---------------------------------------------------------------------------------------------------------------------
with member [Measures].[Nb occur mot] as '[Measures].[Nb mot]', solve_order = 1.0
  member [Measures].[Nb occur ts mot] as '([Mot].Parent, [Measures].[Nb mot])', solve_order = 1.0
  member [Measures].[Nb tot doc ds corpus] as '([Mot].Parent, [Measures].[Nb loc distinct])', solve_order = 1.0
  member [Measures].[Nb doc contenant] as '[Measures].[Nb loc distinct]', solve_order = 1.0
  member [Measures].[tf] as '([Measures].[Nb occur mot] / [Measures].[Nb occur ts mot])'
  member [Measures].[idf_ss_log] as '([Measures].[Nb tot doc ds corpus] / [Measures].[Nb doc contenant])'
  member [Measures].[idf] as 'Log([Measures].[idf_ss_log])'
  member [Measures].[tf_idf] as '([Measures].[tf] * [Measures].[idf])'
select {[Measures].[Nb occur mot], [Measures].[Nb occur ts mot], [Measures].[Nb tot doc ds corpus], [Measures].[Nb
doc contenant], [Measures].[tf], [Measures].[idf_ss_log], [Measures].[idf], [Measures].[tf_idf]} ON COLUMNS,
  {([Loc].[All Locs], [Mot].[All Mots])} ON ROWS
from [Cube_mmp]
```

Requête 2:

```
---------------------------------------------------------------------------------------------------------------------
with member [Measures].[Nb occur mot] as '[Measures].[Nb mot]', solve_order = 1.0
  member [Measures].[Nb occur ts mot] as '([MotMed].Parent, [Measures].[Nb mot])', solve_order = 1.0
  member [Measures].[Nb tot doc ds corpus] as '([MotMed].Parent, [Measures].[Nb tweet distinct])', solve_order = 1.0
  member [Measures].[Nb doc contenant] as '[Measures].[Nb tweet distinct]', solve_order = 1.0
  member [Measures].[tf] as '([Measures].[Nb occur mot] / [Measures].[Nb occur ts mot])'
  member [Measures].[idf_ss_log] as '([Measures].[Nb tot doc ds corpus] / [Measures].[Nb doc contenant])'
  member [Measures].[idf] as 'Log([Measures].[idf_ss_log])'
  member [Measures].[tf_idf] as '([Measures].[tf] * [Measures].[idf])'
select {[Measures].[Nb occur mot], [Measures].[Nb occur ts mot], [Measures].[Nb tot doc ds corpus], [Measures].[Nb
doc contenant], [Measures].[tf], [Measures].[idf_ss_log], [Measures].[idf], [Measures].[tf_idf]} ON COLUMNS,
  {([Loc].[All Locs], [MotMed].[All MotMeds])} ON ROWS
from [Cube_mmp]
```

Towards an On-Line Analysis of Tweets Processing

Sandra Bringay[1,2], Nicolas Béchet[3], Flavien Bouillot[1],
Pascal Poncelet[1], Mathieu Roche[1], and Maguelonne Teisseire[1,4]

[1] LIRMM – CNRS, Univ. Montpellier 2, France
{bringay,bouillot,poncelet,mroche}@lirmm.fr
[2] Dept MIAp, Univ. Montpellier 3, France
[3] INRIA Rocquencourt - Domaine de Voluceau, France – nicolas.bechet@inria.fr
[4] CEMAGREF – UMR TETIS, France – maguelonne.teisseire@cemagref.fr

Abstract. Tweets exchanged over the Internet represent an important source of information, even if their characteristics make them difficult to analyze (a maximum of 140 characters, etc.). In this paper, we define a data warehouse model to analyze large volumes of tweets by proposing measures relevant in the context of knowledge discovery. The use of data warehouses as a tool for the storage and analysis of textual documents is not new but current measures are not well-suited to the specificities of the manipulated data. We also propose a new way for extracting the context of a concept in a hierarchy. Experiments carried out on real data underline the relevance of our proposal.

1 Introduction

In recent years, the development of social and collaborative Web 2.0 has given users a more active role in collaborative networks. Blogs to share one's diary, RSS news to track the lastest information on a specific topic, and tweets to publish one's actions, are now extremely widespread. Easy to create and manage, these tools are used by Internet users, businesses or other organizations to distribute information about themselves. This data creates unexpected applications in terms of decision-making. Indeed, decision makers can use these large volumes of information as new resources to automatically extract useful information.

Since its introduction in 2006, the Twitter website [5] has developed to such an extent that it is currently ranked as the 10^{th} most visited site in the world [6]. Twitter is a platform of microblogging. This means that it is a system for sharing information where users can either follow other users who post short messages or be followed themselves. In January 2010, the number of exchanged tweets reached 1.2 billion and more than 40 million tweets are exchanged per day [7]. When a user follows a person, the user receives all messages from this person, and

[5] http://twitter.com
[6] http://www.alexa.com/siteinfo/twitter.com
[7] http://blog.twitter.com/2010/02/measuring-tweets.html

in turn, when that user tweets, all his followers will receive the messages. Tweets are associated with meta-information that cannot be included in messages (e.g., date, location, etc.) or included in the message in the form of tags having a special meaning. Tweets can be represented in a multidimensional way by taking into account all this meta-information as well as associated temporal relations. In this paper, we focus on the datawarehouse [1] as a tool for the storage and analysis of multidimensional and historized data. It thus becomes possible to manipulate a set of indicators (measures) according to different dimensions which may be provided with one or more hierarchies. Associated operators (e.g., Roll-up, Drill-down, etc.) allow an intuitive navigation on different levels of the hierarchy.

This paper deals with different operators to identify trends, the top-k most significant words over a period of time, the most representative of a city or country, for a certain month, in a year, etc. as well as the impact of hierarchies on these operators. We propose an adapted measure, called $TF\text{-}IDF_{adaptive}$, which identifies the most significant words according to level hierarchies of the cube (e.g., on the location dimension). The use of hierarchies to manage words in the tweets enables us to offer a contextualization in order to better understand the content. We illustrate our proposal by using the MeSH [8] (Medical Subject Headings) which is used for indexing PubMed articles [9].

The rest of the paper is organized as follows. Section 2 describes a data model for cubes of tweets and details the proposed measure. In Section 3, we consider that a hierarchy on the words in tweets is available and propose a new approach to contextualize the words in this hierarchy. We present some results of conducted experiments in Section 4. Before concluding by presenting future work, we propose a state-of-the-art in Section 5.

2 What is the most appropriate measure for tweets?

2.1 Preliminary Definitions

In this section we introduce the model adapted to a cube of tweets. According to [2], a fact table F is defined on the schema $D = \{T, \ldots, T_n, M\}$ where T_i $(i = 1, .., n)$ are the dimensions and M stands for a measure. Each dimension T_i is defined over a domain D partitioned into a set of categories C_j. We thus have $D = \cup_j C_j$. D is also provided with a partial order $\sqsubseteq_D$ to compare the values of the domain D. Each category represents the values associated with a level of granularity. We note $e \in D$ to specify that e is a value of the dimension D if there is a category $C_j \subseteq D$ such that $e \in \cup_j C_j$. Note that two special categories are distinguished and are present on all dimensions: $\bot_D$ et $\top_D \in C_D$ corresponding respectively to the level of finer and higher granularity. In our approach, the partial order defined on the domains of the dimensions stands for the inclusion of keywords associated to the values of the dimensions. Thus, let $e_1, e_2 \in \cup_j C_j$ be two values, we have $e_1 \sqsubseteq_D e_2$ if e_1 is logically contained in e_2.

[8] http://www.ncbi.nlm.nih.gov/mesh
[9] http://www.ncbi.nlm.nih.gov/PubMed/

2.2 The data model

We instantiate the data model of the previous section to take into account the different dimensions of description and propose a specific dimension to the words associated to tweets.

Let us consider, for example, the analysis of tweets dedicated to the Duncan diet (e.g., "*The Dukan diet is the best diet ever, FACT!!! Its just meat for me for the next 5 day YEESSS*"). We wish, for example, to follow the comments or opinons of different people on the effectiveness of a diet. In order to extract the tweets, we query Twitter using a set of seed words: *Duncan, diet, slim, protein, etc.*. In this case, the original values of the word dimension are $dom(word) = \{$ *Duncan, diet, slim, protein, ...*$\}$.

Figure 1 illustrates the data model. We find the dimension ($location$ $\perp_{location}$ $= City \leq State \leq Country \leq \top_{location}$), and the dimension $time$ ($\perp_{time} = day \leq month \leq semester \leq year \leq \top_{time}$).

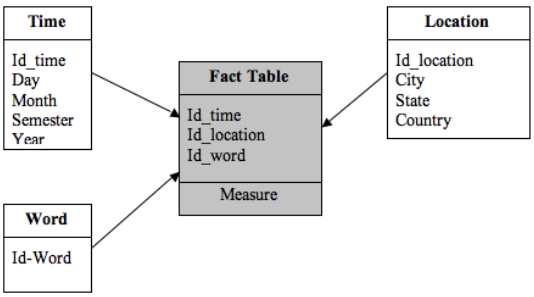


Fig. 1. The schema related to a diet application

The domain of the *word* dimension is that of the seed words with the words appearing frequently with them. In the fact table, several measures may be used. Traditionally it is the TF-IDF. This issue is addressed in the next section.

2.3 Towards an appropriate measure

Relying only on knowledge of the hierarchy in a cube does not always allow a good aggregation (i.e., corresponding to a real situation). For instance, the characteristics of the words in tweets are not necessarily the same in a State and in a City. The aggregation measure that we propose is based on approaches from Information Retrieval (IR).

In our process, the first step is to merge the number of occurrences of words specific to a level. More precisely, we list all the words located in tweets that match a given level (e.g., City, State, Country). If the user wishes to focus the search on a specific City, the words from the tweets coming from this city form a vector. We can apply this same principle to the State by using a Roll-up operator. The aim of our work is to highlight the discriminant words for each level.

Traditionally, $TF\text{-}IDF$ measure gives greater weight to the discriminant words of a document [3]. Like [4], we propose a measure called $TF\text{-}IDF_{adaptative}$ aiming to rank the words according to the level where the user is located and defined as follows:

$$TF_{i,j} - IDF_i^k = \frac{n_{i,j}}{\sum_k n_{k,j}} \times log_2 \frac{|E^k|}{|\{e_j^k : t_i \in e_j^k\}|} \quad (1)$$

where $|E^k|$ stands for the total number of elements of type k (in our example, $k = \{City, State, Country\}$) which corresponds to the level of the hierarchy that the decision maker wants to aggregate. $|\{e_j : t_i \in e_j\}|$ is relative to the number of elements of type k where the term t_i appears.

3 A hierarchy of words for tweets

In this section, we adopt a hierarchy on the words to allow the contextualization of words in tweets.

3.1 The data and the model

For the hierarchy, we use the MeSH (Medical Subject Headings)[10] National Library of Medicine's controlled vocabulary thesaurus. It consists of sets of terms naming descriptors in a twelve-level hierarchy that permits the search to be carried out at various levels of specificity. At the most general level of the hierarchical structure are very broad headings such as "Anatomy" or "Mental Disorders". More specific headings are found at more narrow levels, such as "Ankle" and "Conduct Disorder". In 2011, 26,142 descriptors are available in MeSH. There are also over 177,000 entry terms that assist in finding the most appropriate MeSH Heading, for example, "Vitamin C" is an entry term to "Ascorbic Acid".

The data model is updated to take into account this new dimension. Compared to the previous model (See Figure 1) the dimension "Word" has been replaced by MeSHWord. MeSHWord has a partial order, $\sqsubseteq_{MeSHWord}$, to compare the different values of the domain. One of the main problems with the use of this thesaurus is that different terms may occur at various levels in the hierarchy. This ambiguity raises the problem of using operators like Roll-up or Drill-down to navigate in the cube. In order to illustrate this problem let us consider the following example.

Example 1 *Let us consider the following tweet: "pneumonia & serious nerve problems. can't stand up. possible myasthenia gravis treatable with meds.". If we look in MeSH for the descriptor pneumonia, we find this term occurring in several places (See Figure 2). Depending on the position in the hierarchy, a Roll-up operation on pneunomia will not give the same result (i.e., "respiratory tract diseases" versus. "lung diseases").*

[10] http://www.nlm.nih.gov/pubs/factsheets/mesh.html

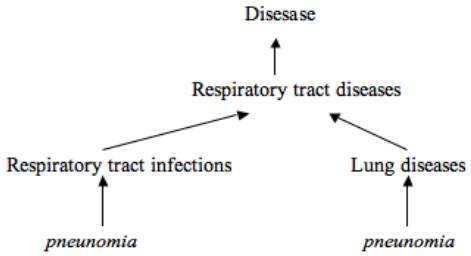


Fig. 2. An example of the MeSH thesaurus

3.2 How to identify the context of a tweet?

We have shown in Example 1, the difficulty of locating the context of a term in the hierarchy. However, a closer look at the tweet shows that the words themselves can be helpful to determine the context. In order to disambiguate polysemous words in the hierarchy of MeSH, we adapt the $AcroDef_{MI^3}$ method described in [5] where the authors show the efficiency of this method in a biomedical context. This measure is based on the Cubic Mutual Information [6] that enhances the impact of frequent co-occurrences of two words in a given context. For a context C, $AcroDef_{MI^3}$ is defined as follows:

$$AcroDef^{C}_{MI^3}(m1, m2) = \frac{(nb(m1\ and\ m2\ and\ C))^3}{nb(m1\ and\ C) \times nb(m2\ and\ C)} \quad (2)$$

In our case, we want to calculate the dependence between a word m to disambiguate and different words m_t of the tweets using the context of the hierarchy (i.e., parents p of the word m).

Example 2 *Let us consider the word 'pneunomia' to disambiguate in Example 1. Here we calculate the dependence between this word m and the other words following 'pneumonia' (nouns, verbs, and adjectives are selected with a Part-of-Speech process): 'serious' and 'nerve'. This dependence is calculated regarding the context of both possible fathers in the MeSH hierarchy. In order to predict where in the MeSH thesaurus we have to associate the word 'pneunomia', we perform the following operations:*

- $nb(pneunomia, m_t, "lung\ diseases") = 227$ *(number of returned pages with the queries 'pneunomia serious "lung diseases"' and 'pneunomia nerve "lung diseases"')*
- $nb(pneunomia, m_t, "respiratory\ tract\ infections") = 496$

The dependence of the terms is given by:

- $AcroDef^{"lung\ diseases"}_{MI^3}(pneunomia, m_t) = 0.02$
- $AcroDef^{"respiratory\ tract\ infections"}_{MI^3}(pneunomia, m_t) = 0.11$

Thus, in the tweet from Example 1, for the word pneunomia, we will preferably do the aggregation at the level of the concept 'respiratory tract infections' of the MeSH.

Note that this step of disambiguation, which is essential for data from MeSH, is quite costly in terms of the number of queries. It therefore seems more appropriate to call these functions during the ETL process rather than carrying out such processing when browsing the cube.

4 Experiments

In order to evaluate our approach, several experiments were conducted. These were performed using PostgreSQL 8.4 with the Pentaho Mondrian 3.20 environment. To extract the tweets related to the vocabulary used in MeSH, we focus on the tweets related to "Disease" and queries Twitter by using all the terms of the corresponding hierarchy. We collected 1,801,310 tweets in English from January 2011 to February 2011. In these experiments, we analyze the first words returned by the TF-IDF$_{adaptive}$ (highest scores). For example, the following table presents the first 12 words of tweets in the United States, for the State of Illinois and the City of Chicago during the month of January 2011.

United Sates	Illinois	Chicago
wart	risk	risk
pneumonia	vaccination	wart
vaccination	wart	pneumonia
risk	pneumonia	wood
lymphoma	wood	colonoscopy
common cold	colonoscopy	x-ray
disease	x-ray	death
meningitis	encephalitis	school
infection	death	vaccination
vaccine	school	eye infection
life	eye infection	patient
hepatitis	man	russia

Now we consider an example of the application of our approach. Figures *3* and *4* visualize the worldwide coverage of the words *hepatitis* and *pneunomia* excluding the United States, the United Kingdom, and Canada. This coverage is obtained by fixing the location dimension and by examining the frequency of the Word over the period.

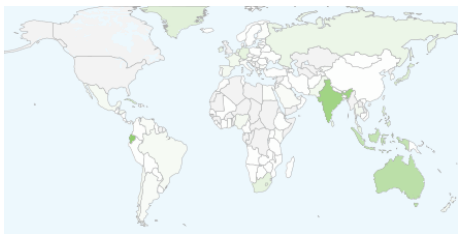


Fig. 3. Distribution of the use of the word hepatitis

Fig. 4. Distribution of the use of the word pneunomia

Finally we evaluated the prediction measure (i.e., $AcroDef_{MI^3}$) within the MeSH hierarchy (see section 3.2). We extracted more than 5,000 Facebook messages (the same kind of messages as tweets) from the *food* topic. These messages

contain at least one polysemous term (i.e. a term which can be associated to the hierarchy *food and beverages*) and one or two other hierachies of MeSH: *Eukaryota*, *lipids*, *plant structures*, and so forth. A correct prediction means that $AcroDef_{MI^3}$ associates this polysemous term with the *food and beverages* concept. In the following table, three types of elements are used in order to characterize the hierarchy (context of the $AcroDef_{MI^3}$ measure): Father (F), grand-father (GF), and father + grand-father (FGF). This table shows that (1) the use of more generic information (grand-father) is more relevant, (2) the association of all the available information improves the quality of the prediction. In our future work we plan to add other hierarchical information (e.g. son, cousins).

Elements of the hierarchy used	F	GF	FGF
Prediction	60.8%	63.6%	68.0%

5 Related work

The analysis of textual data from tweets has recently been addressed in many research studies and many proposals exist. For example, in [7], the authors propose analyzing the content of the tweets in real time to detect alarms during an earthquake. The authors of TwitterMonitor [8] present a system to automatically extract trends in the stream of tweets. A quite similar approach is proposed in [9]. However, to the best of our knowledge, most existing studies mainly focus on a specific analysis of tweets and do not provide general tools for the decision maker (i.e., for manipulating the information embedded in tweets according to their needs). Thus, few studies have been interested in the use of cubes to manage tweets. Recent work has focused on integrating textual data in data warehouses. In this context, aggregation methods suitable for textual data have been proposed. For example, in [10], the authors propose using Natural Language Processing techniques to aggregate the words with the same root or the same lemmas. They also use existing semantic tools such as Wordnet or Roget to group words together. Apart from using morpho-syntactic and semantic knowledge, other studies consider numerical approaches from the field of Information Retrieval (IR) to aggregate textual data. Thus, the authors of [11] propose aggregating documents according to keywords by using a semantic hierarchy of words found in the datawarehouses and some measures from IR. Such methods from IR are also used in the work of [2] which takes into account a "context" and "relevance" dimension to build a datawarehouse of textual data called R-cube. Other approaches add a new specific dimension. For example, in [12], the authors add a "topic" dimension and apply the PLSA approach [13] to extract the themes representing the documents in this new dimension. Finally, in [14] the authors propose aggregating parts of documents to provide the decision maker with words specific to the aggregation. In this context, the authors use a first function to select the most significant words using the classical $TF\text{-}IDF$ measure.

6 Conclusion

In this paper we proposed a new approach to analyze tweets from their multidimensional characteristics. The originality of our proposal is to define and manipulate cubes of tweets. We have shown through two different models and applications: no predefined hierarchy on tweets (i.e., diet analysis) and existing hierarchy (i.e., using the MeSH thesaurus), that the analysis of tweets requires the definition of new measures and that a contextualization step is relevant.

Future work involves several issues. First we want to extend the proposed approach to take into account opinions or feelings expressed in the tweets. Recent studies analyze the mood of people (e.g., http://twittermood.org/). We want to enhance these approaches by analyzing the content of tweets and thus be able to automatically extract knowledge such as: who are the people who followed a diet and are dissatisfied? Secondly, we wish to consider tweets as available in the form of a stream and propose new techniques for efficiently storing the data.

References

1. Codd, E., Codd, S., Salley, C.: Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate. In: White Paper. (1993)
2. Pérez-Martínez, J.M., Llavori, R.B., Cabo, M.J.A., Pedersen, T.B.: Contextualizing data warehouses with documents. Decision Support Systems **45**(1) (2008) 77–94
3. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11) (1975) 613–620
4. Grabs, T., Schek, H.J.: ETH Zurich at INEX: Flexible Information Retrieval from XML with PowerDB-XML. In: XML with PowerDB-XML. INEX Workshop, ERCIM Publications (2002) 141–148
5. Roche, M., Prince, V.: Managing the acronym/expansion identification process for text-mining applications. Int. J. of Software and Informatics **2**(2) (2008) 163–179
6. Daille, B.: Approche mixte pour l'extraction automatique de terminologie : statistiques lexicales et filtres linguistiques. PhD thesis, Université Paris 7 (1994)
7. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In: Proceedings of WWW. (2010) 851–860
8. Mathioudakis, M., Koudas, N.: Twittermonitor: trend detection over the twitter stream. In: Proceedings of SIGMOD, Demonstration. (2010) 1155–1158
9. Benhardus, J.: Streaming trend detection in twitter. In: National Science Foundation REU for Artificial Intelligence, NLP and IR. (2010)
10. Keith, S., Kaser, O., Lemire, D.: Analyzing large collections of electronic text using olap. Technical Report TR-05-001, UNBSJ CSAS (2005)
11. Lin, C.X., Ding, B., Han, J., Zhu, F., Zhao, B.: Text Cube: Computing IR Measures for Multidimensional Text Database Analysis. In: Proc. of ICDM. (2008) 905–910
12. Zhang, D., Zhai, C., Han, J.: Topic cube: Topic modeling for olap on multidimensional text databases. In: In Proc. of SIAM. (2009) 1123–1134
13. Hofmann, T.: Probabilistic latent semantic analysis. In: In Proc. of Uncertainty in Artificial Intelligence, UAI'99. (1999) 289–296
14. Pujolle, G., Ravat, F., Teste, O., Tournier, R.: Fonctions d'agrégation pour l'analyse en ligne (OLAP) de données textuelles. Fonctions TOP_KW et AVG_KW opérant sur des termes. Ingénierie des Systèmes d'Information **13**(6) (2008) 61–84