Une technologie NoSQL au service de moteur de recherche en santé

Romain Lelong[*,****] , Lina Soualmia[*,**] Saoussen Sakji[***]
Badisse Dahamna[****], Stéfan Darmoni[*,**,****]

[*]TIBS - LITIS EA 4108, Université de Rouen, France
romain.lelong@gmail.com
[**]INSERM, LIMICS UMR-1142, Paris, France
[***]Université de Jendouba, Tunisie
ssakji@yahoo.fr
[****]CHU de Rouen, 1 rue de Germont, 76000 Rouen, France
stefan.Darmoni@chu-rouen.fr,

Résumé. Afin de répondre au besoin d'accès à des volumes de données en perpétuelle croissance, de nouvelles technologies telles que les bases Not Only SQL (NoSQL) ont vu le jour. Ces bases offrent un haut degré d'accès aux données mais souffrent néanmoins d'un manque de standards qui complexifie les tâches de Recherche d'Information (RI). Cette étude présente la mise œuvre du moteur de recherche sémantique sous-jacent à LiSSa (Littérature Scientifique en Santé) à l'aide d'une solution NoSQL. Initialement basée sur un Système de Gestion de Base de Données Relationnelles, l'exploitation de cette solution permet une nette amélioration des performances mais impose également le développement d'outils annexes pour réaliser les tâches de RI classiques. Cependant, une évaluation du Système de Recherche d'Information en contexte devra en outre être réalisée.

1 Introduction

Depuis le début des années 2000, Internet est devenu la source majeure d'information pour les professionnels de santé comme pour les patients. La croissance du volume de données de santé présentes sur Internet et la diversité de ces dernières soulèvent aujourd'hui différents défis aussi bien purement techniques qu'en matière de Recherche d'Information (RI). La littérature scientifique, a priori de bonne qualité, à également fait l'objet de cette croissance notamment au sein des bases de données bibliographiques comme MEDLINE et de nombreux de moteurs de recherche destinés à la RI au sein de ces bases ont vu le jour (e.g. PudMed, GoogleScholar). Ce volume de données impose également des efforts techniques quant à la modélisation et l'architecture des bases de données bibliographiques et des moteurs de recherche associés afin d'assurer une recherche précise et performante en terme de temps de traitement. Avec la montée en puissance du volume de données (« Big Data » ou données massives), de nouvelles technologies d'accès aux données telles que les bases Not Only SQL (NoSQL) ont vu le jour

afin de pallier à l'insuffisance des Système de Gestion de Base de Données Relationnelles (SGBDR) classiques. Un grand nombre d'applications exploitent ces technologies aujourd'hui. Ces solutions technologiques souffrent cependant de leur « jeunesse » et de manque de standard rendant la RI plus complexe à mettre en œuvre. Cette étude a pour but de présenter la mise œuvre d'une technologie NoSQL comme support d'accès aux données du moteur de recherche Semantic Search Engine NoSQL (SSE_{NoSQL}) en arrière plan des applications Doc'CISMeF Darmoni et al. (2001) et LiSSa Griffon et al. (2017) respectivement dédiées à la recherche d'information documentaire et bibliographique dans la domaine de la santé en langue française.

2 Matériel

2.1 Le projet CISMeF

Le projet Catalogue et Index des Sites Médicaux de langue Française (CISMeF) [1] (initié en février 1995 au sein du Centre Hospitalier Universitaire de Rouen (CHUR) Darmoni et al. (2000)) recense, filtre et évalue manuellement selon des critères de qualité du Net Scoring Darmoni et al. (1999) les diverses sources d'informations de santé de qualité disponibles sur Internet (guides de bonnes pratiques, conférences, cours, etc.). L'organisation des ressources est basée sur deux standards : (a) les éléments et metadonnées du Dublin Core (b) des terminologies/ontologies médicales dans le processus d'indexation manuelle avec comme terminologie pivot le thesaurus Medical Subject Headings [2] (MeSH) de la National Library of Medicine (NLM).

2.2 Doc'CISMeF

Doc'CISMeF est le moteur de recherche dédié à la RI dans le CISMeF Darmoni et al. (2001). Basé auparavant sur divers SGBDR (nottament Oracle) Doc'CISMeF donne aujourd'hui accès à 118 033 ressources et repose sur la technologie NoSQL Infinispan Marchioni et Surtani (2012). Il peut être interrogé en langage naturel ou via deux langages de requête logiques différents. Doc'CISMeF est un moteur de recherche sémantique exploitant : (a) les relations sémantiques des termes et terminologies indexant les ressources (b) les différents champs et métadonnées des ressources (c) de la recherche plein texte (titre, résumé, etc.). De plus, certains éléments sémantiques tel que les méta-termes Thirion et Darmoni (1999) et les types de ressources Darmoni et Thirion (2000) ont été ajoutés au cours du temps.

2.3 LiSSa

LiSSa (Littérature Scientifique en Santé [3]) est un moteur de recherche donnant accès à plus de 1 191 977 références bibliographiques de santé en Français Griffon et al. (2017) issues soit des éditeurs tels que Elsevier-Masson ou de la base de données PubMed. Il est issu du projet Base de Données Bibliographiques en Français (BDBfr) (projet n° ANR–14–CE17–0020) de l'Agence Nationale de la Recherche (ANR) dans le programme Technologies pour la Santé

1. http ://www.chu-rouen.fr/cismef/
2. https ://www.nlm.nih.gov/mesh/
3. http ://www.lissa.fr

(TecSan) 2015 Griffon et al. (2014). Le moteur de recherche LiSSa est technologiquement identique à celui de Doc'CISMeF.

2.4 Les SGBD NoSQL

Depuis leur apparition les SGBD NoSQL ont connu un succès croissant en remplaçant les SGBDR classiques (e.g. Oracle) dans nombre d'applications y compris dans les grande entreprise (e.g. Google, Facebook, Amazon). Leur capacité en terme de scalabilité horizontale (architectures matérielles) et de répartition de charge explique le succès de ce type d'architecture qui permet d'assurer de bonnes performances d'accès avec la montée en charge. Ces technologies demeurent cependant immatures et hétérogènes et présentent certains inconvénients (notamment pour les problématiques de RI) tel que le non respect strict de contraintes ACID (Atomicité, Cohérence, Isolation, Durabilité) et le manque de standard (maintenance difficile) tel qu'un langage de requête standardisé pleinement adopté à l'image du SQL pour les SGBDR Nayak et al. (2013).

3 Méthode

3.1 Choix de la technologie NoSQL

Cette étude a pour objectif principal de présenter le moteur de recherche SSE_{NoSQL} sous-jacent aux applications Doc'CISMeF et LiSSa migrés en 2017 sur le plan opérationnel vers la solution NoSQL Infinispan. Infinipsan est un cache distribué (cluster) de la famille des In-Memory Data Grid (IMDG). Ces solutions permettent de stocker les données directement dans la Random Access Memory (RAM) des machines offrant ainsi un haut degré d'accès à ces dernières. Plus formellement, les données sont représentées sous forme d'associations clé/valeur au sein de table de hachage (Map) stockée en RAM fournissant ainsi un accès très rapide aux valeurs par simple spécification de la clé. On utilisera dans la suite la notation $NomMap : cle \longrightarrow valeur$ pour décrire une Map. C'est en raison de ce critère de rapidité Salhi et al. (2017) d'accès aux données que le choix s'est porté sur cette solution.

3.2 Modélisation des données

La modélisation des données au sein du SSE_{NoSQL} a été conçue de manière générique en suivant un paradigme Entité-Association et plus spécifiquement le modèle de graphe générique Resource Description Framework (RDF). Ce modèle formalise toutes les données comme des entités ou des attributs d'entités. Chaque entité $entity$ est identifiée par un id et un type d'entité $type$ et peut être associée soit à des attributs $attribut$ de noms $name$ (e.g. de type numérique, texte, date, etc.) soit à d'autres entités par l'intermédiaire d'une relation portant un libellé $label$. Cette relation peut être vue comme un triplet $(id_s, label, id_t)$ liant l'identifiant de l'entité source id_s à celui de l'entité cible id_t par l'intermédiaire de la relation $label$. La généricité de l'architecture de l'IMDG permet une modélisation « compacte » des données basée sur un nombre réduit de Map (voir tableau 1). Map_E est une Map générique associant à chaque id d'entité l'entité elle-même. Elle contient toutes les entités aussi différentes soient-elles les unes des autres (e.g. ressources bibliographiques (LiSSa), auteurs, ressources terminologiques, etc.).

De même, les différentes relations entre entités sont stockées dans les mêmes Maps génériques. Les relations ont été séparées en trois catégories différentes : (a) les relations hiérarchiques (Map_{Rh}) notamment pour les relations intra-terminologiques (e.g. le concept MeSH *asthme* est un fils de *hypersensibilité respiratoire*), (b) les relations d'indexation (Map_{Ri}) (e.g. *asthme* indexe la ressource X) et (c) les autres types de relation (Map_R) (e.g. l'auteur A est un auteur de la ressource Y). Les Maps Map_R, Map_{Rh} et Map_{Ri} permettent toutes les trois d'accéder aux relations à partir de l'identifiant id_s de l'entité source de cette relation. Enfin, la Map Map_M permet de modéliser la sémantique de chaque type $type$ d'entité (e.g. modélisation de la symétrie d'une relation, appartenance d'un attribut à une entité d'un type spécifique).

Map	:	clé	$\longrightarrow$	valeurs
Map_T (types d'entité)	:	id	$\longrightarrow$	$type$
Map_E (entité)	:	id	$\longrightarrow$	$entity$
Map_A (attributs)	:	id	$\longrightarrow$	$\langle(name, attribut)\rangle$
Map_R (relations)	:	id_s	$\longrightarrow$	$\langle(id_s, label, id_t)\rangle$
Map_{Rh} (relations hiérarchiques)	:	id_s	$\longrightarrow$	$\langle(id_s, label, id_t)\rangle$
Map_{Ri} (relations d'indexation)	:	id_s	$\longrightarrow$	$\langle(id_s, label, id_t)\rangle$
Map_J (jointure)	:	id_s+$label$+$type_t$	$\longrightarrow$	$\langle id_t\rangle$
Map_{Ji} (jointure d'indexation)	:	id_s+$label$+$type_t$	$\longrightarrow$	$\langle id_t\rangle$
Map_M (modéle)	:	$type$	$\longrightarrow$	$model$

TABLE 1 – *Ensemble des Maps génériques nécessaires à la modélisation des données du SSE_{NoSQL}. Les notations avec chevron du type « $\langle X\rangle$ » indiquent une liste d'éléments de type X. Le caractère symétrique éventuel d'une relation $(id_s, label, id_t)$ est modélisé par l'existence de la relation $(id_t, label, id_s)$ au sein des Maps de relations.*

3.3 Requêtage sémantique

Le moteur de recherche SSE_{SQL} sous-jacent à Doc'CISMeF est historiquement pourvu d'un langage de requête booléen orienté attributs inspiré de celui de Ovid [4] (voir exemples de requêtes dans le tableau 2) et augmenté de certaines fonctionnalités telles que l'expansion sémantique par défaut Darmoni et al. (2001). Ce langage permet de requêter des ressources à l'aide de contraintes portant sur les attributs structurés et non structurés relatifs à ces ressources. Le SSE_{NoSQL}, repose lui sur un nouveau langage de requêtes orienté entité fournissant une expressivité de requêtage plus importante notamment au niveau sémantique puisque qu'il permet de construire des requêtes faisant intervenir les entités elle-mêmes et non plus seulement les attributs d'une ressource documentaire (voir exemple de requête figure 1). La syntaxe de ce langage est construite à partir « clause entité » du type `ENTITE(CONTRAINTES)` permettant de requêter une entité à l'aide de contrainte sur cette dernière. Les clauses entité peuvent être imbriquées afin de requêter des entités sémantiquement liées (e.g. `ENTITE1(ENTITE2(...))` désigne toutes les entités `ENTITE1` liées aux `ENTITE2`). Initialement conçu dans le cadre du projet ANR Ravel, ce langage de requête Lelong et al. (2017) reposait sur un SGBDR. Quelques concessions pour son exploitation dans le cadre d'une technologie

4. https ://ovidsp.ovid.com/

NoSQL de type IMDG ont été nécessaires. Une retranscription de la nouvelle syntaxe vers l'ancienne a en outre été mise en place afin d'assurer une rétrocompatibilité des requêtes exprimées à l'aide de l'ancienne syntaxe.

3.4 Moteur booléen

Le processus d'exécution du moteur booléen SSE_{NoSQL} se décompose en trois étapes. La figure 1 illustre les étapes 1 et 2 de ce processus.
Étape 1 (analyse syntaxique) : Elle consiste en la construction d'un objet muni d'une structure arborescente représentative de la structure booléenne et sémantique de la requête logique. Cette analyse est réalisée à l'aide d'un analyseur lexical basé sur une grammaire implémentée à l'aide du générateur de parser JavaCC. L'arbre construit est ainsi exploitable informatiquement. Lorsque la requête est en langage naturel, l'outil d'Extraction de Concepts Multi-Terminologique (ECMT) Cabot et al. (2016) est utilisé pour réaliser une indexation automatique basée sur l'algorithme du sac de mots et ainsi construit une requête logique.
Étape 2 (transformation logique) : Elle permet de transformer l'arbre initial en un arbre logiquement équivalent et propice à son exécution. L'exécution des branches de l'arbre généré revient alors soit à une simple recherche d'entités à partir des valeurs (e.g. `NLM(year=2016)`) soit à la recherche d'une entité par simple jointure (e.g. `NLM(CPT(...))`).
Étape 3 (exécution) : La dernière étape est destinée à l'exécution de l'arbre à partir des Maps de l'IMDG. L'arbre est alors exécuté des nœuds terminaux vers le nœud racine de l'arbre. L'utilisation d'un IMDG comme source de données impose certaines limites (notamment de l'absence de langage de requête standard) et rend nécessaire la mise en place de briques de RI spécifiques. Afin de permettre une recherche des entités par l'intermédiaire de leurs attributs (e.g. recherche d'un auteur par son nom, recherche d'une ressource d'une année de publication donnée), des index basés sur la technologie Lucene [5] ont été créés. Les jointures entre entités (e.g. `...NLM(JOURNAL(...))`, `...NLM(CPT(...))`) sont réalisées grâce aux Maps Map_J et Map_{Ji}. Map_J permet d'obtenir la liste des entités d'une certain type ($type_t$) en relation avec une entité source (id_s) et via un type de relation donné ($label$). Le rôle de Map_{Ji} est similaire et permet donc de récupérer la liste des concepts indexant une ressource et avec un type d'indexation donné. L'étape 2 assurant que chaque branche unitaire de l'arbre (c.à.d. dépourvue d'opérateurs booléens (e.g. `NLM(CPT(lab="asthme"))`, `NLM(JOURNAL(title="..."))`)) se résume en une succession de recherches de valeurs ou de jointure, les branches unitaires sont exécutées en parallèle (et une seule et même fois en cas de branche similaire) dans un premier temps. La logique booléenne est exécutée dans un second temps à l'aide des opérateurs ensemblistes classiques (union, intersection, différence).

4 Résultats

Pour une première évaluation, la plus-value de l'utilisation de l'IMDG est évaluée sur 10 requêtes (données dans le tableau 2). Ces requêtes sont lancées sur le SSE_{NoSQL} et leur temps d'exécution est comparé à celui de l'ancienne version SSE_{SQL} du moteur basée sur la technologie Oracle et le SQL. Les requêtes sont données dans un langage de requête logique

5. https ://lucene.apache.org/

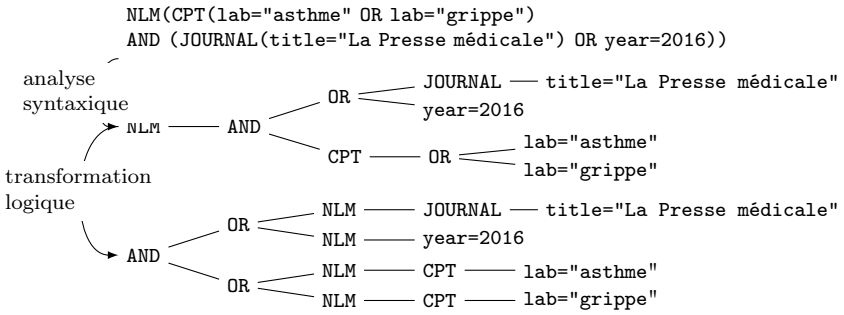

FIGURE 1 – *Exemple illustrant le processus du SSE_{NoSQL} à partir d'une requête logique*

booléen (opérateurs booléens `ET`, `OU` et parenthèses) exploitant des « codes champs ». Une recherche du type `C/Q.mc` (mot clé) désigne les ressources indexées avec le concept `C` (la partie `/Q` apposée au concept est optionnelle est permet de préciser le sens d'un point de vue conceptuel). `C.mt` (méta-terme) désigne les ressources indexées avec l'ensemble des concepts que regroupe le concept générique `C`(méta-terme). `C.sr` (stratégie de recherche) est un « raccourci » permettant d'exécuter une requête booléenne complexe préalablement associée à la stratégie de recherche `C` par un expert. Enfin `L.tc` (tous champs) permet d'effectuer une recherche plein texte sur tous les champs non structurés des ressources (e.g. titre, résumé, etc.). Les requêtes sont choisies pour leurs différences d'exigence vis à vis du SSE en terme de recherche d'information. Les requêtes 1 et 2 correspondent à de simples recherches de ressources indexées avec les mots clés `asthme` et `humain` mais avec des volumétries très différentes. La requête 3 recherche les ressources traitants de l'`asthme` mais dans le cadre du `diagnostic` quand à la requête 4, elle correspond à une recherche simultanée de ressources indexées par la multitude de concepts que recouvre le méta-terme `médicament`. Les requête 5 à 8 ajoutent la gestion de la logique booléenne, 8 étant particulièrement exigeante concernant cet aspect. Enfin les requêtes 8 et 9 effectuent des recherches plein texte au sein des ressources. L'évaluation est réalisée sur le corpus bibliographique de LiSSa en raison de sa forte volumétrie 10 fois plus importante que celle de Doc'CISMeF (LiSSA 1 191 977 vs. Doc'CISMeF 118 033). On note globalement (cf. tableau 2) un gain moyen de 65,8% et des temps de traitement divisés en moyenne par 4. De manière plus détaillée, on remarque une nette amélioration pour les requêtes ayant une grosse volumétrie ($>$100 000) et ne faisant pas intervenir de recherche plein texte. Les requêtes 2, 4 et 7 font en effet apparaître un gain moyen de 84,2% et un facteur de 6,92 alors qu'en moyenne les requêtes 1, 3, 5, 6 et 8 donnent un gain de 65,5% et un facteur de 3,2. Le gain de performance est moindre pour la recherche plein texte et l'utilisation de Lucene (requêtes 9 et 10) avec un gain moyen de 39,2% et un facteur de 1,6 là où les requêtes hors recherche plein texte donnent en moyenne un gain de 72,5% et des temps d'exécution divisés par 4,6. Malgré les efforts particuliers apportés au SSE_{SQL} pour l'optimisation des performances, l'IMDG permet d'apporter une significative amélioration. Sur les 10 requêtes choisies on observe des temps d'exécution moyen de l'ordre de 3 secondes avec une variabilité plus faible (écart type 1 757,6 ms) contre 12,6 secondes pour le SSE_{SQL} et avec un écart type de 11 267,9 ms.

Partie A

nº	Requête
1	`asthme.mc`
2	`humain.mc`
3	`asthme/diagnostic.mc`
4	`médicament.mt`
5	`(asthme/traitement médicamenteux.mc OU (asthme.mc ET médicament.mt)) ET enfant.mc`
6	`médicaments.mt ET chirurgie.mt ET thérapeutique.mt`
7	`médicaments.mt OU chirurgie.mt OU thérapeutique.mt`
8	`affection iatrogénique.sr`
9	`patient.tc`
10	`(chirurgie.tc OU opération.tc) ET patient.tc`

Partie B

nº	n	t_{SQL} (ms)	t_{NoSQL} (ms)	gain (%)	facteur
1	4 879	440	139	$68,4$	$3,16$
2	445 893	29 538	3 288	$88,8$	$8,98$
3	1 013	916	315	$65,6$	$2,91$
4	141 729	14 137	3 213	$77,3$	$4,40$
5	387	4 128	2 237	$45,8$	$1,85$
6	11 042	13 962	2 755	$80,3$	$5,07$
7	315 619	36 720	4 977	$86,5$	$7,38$
8	57 555	9 305	3 046	$67,3$	$3,05$
9	259 554	9 882	5 310	$46,3$	$1,86$
10	67 155	7 863	5 349	$32,0$	$1,47$

TABLE 2 – *La partie A donne les* 10 *requêtes utilisées pour l'évaluation et la partie B, les temps d'exécution moyens* t_{SQL} *et* t_{NoSQL} *de celles-ci sur le* SSE_{SQL} *et le* SSE_{NoSQL}. n *donne le nombre de ressources renvoyées. La colonne gain donne le rapport* $\frac{t_{SQL}-t_{NoSQL}}{t_{SQL}} \times 100$ *et facteur donne le rapport* t_{SQL}/t_{NoSQL}

5 Conclusion

L'exploitation de la technologie $NoSQL$ permet une nette amélioration des performances par rapport à la technologie SQL notamment sur de grosses volumétries. L'inexistence de standard et plus particulièrement de langage de requête et l'emploi de structures de stockage simples (Maps) au sein de l'IMDG impose des efforts de modélisation et de développement dans un contexte de RI. Une prochaine évaluation réalisée en contexte pour évaluer l'outil en terme de précision et de rappel de l'outil sera menée dans les prochaines semaine.

Références

Cabot, C., L. Soualmia, B. Dahamna, et S. Darmoni (2016). Sibm at clef ehealth evaluation lab 2016 : Extracting concepts in french medical texts with ecmt and cimind. In *2016*

Conference and Labs of the Evaluation Forum, CLEF, pp. 47–60.

Darmoni, S., V. Leroux, M. Daigne, B. Thirion, P. Santamaria, C. Duvaux, et M. Gea (1999). Net scoring : critères de qualité de l'information de santé sur l'internet. *Technologie Santé 36*, 128–142.

Darmoni, S. J., J. P. Leroy, F. Baudic, M. Douyère, J. Piot, et B. Thirion (2000). Cismef : a structured health resource guide. *Methods of information in medicine 39*(1), 30–35.

Darmoni, S. J. et B. Thirion (2000). A standard metadata scheme for health resources. *J Am Med Inform Assoc 7*(1), 108–109.

Darmoni, S. J., B. Thirion, J. P. Leroy, M. Douyère, B. Lacoste, C. Godard, I. Rigolle, M. Brisou, S. Videau, E. Goupyt, J. Piott, M. Quéré, S. Ouazir, et H. Abdulrab (2001). A search tool based on 'encapsulated' mesh thesaurus to retrieve quality health resources on the internet. *Med Inform Internet Med 26*(3), 165–178.

Griffon, N., M. Schuers, G. Kerdelhué, J. Grosjean, et S. Darmoni (2017). Littérature scientifique en santé (LiSSa) : une base de données bibliographiques en français. *La Revue Du Praticien 67*(2), 134–138.

Griffon, N., M. Schuers, L. Soualmia, J. Grosjean, G. Kerdelhué, I. Kergoulay, B. Dahamna, et S. Darmoni (2014). A search engine to access pubmed monolingual subsets : Proof of concept - evaluation in french. *J Med Internet Res 16*(12), e271.

Lelong, R., L. Soualmia, B. Dahamna, N. Griffon, et S. J. Darmoni (2017). Querying ehrs with a semantic and entity-oriented query language. *Studies in health technology and informatics 235*, 121–125.

Marchioni, F. et M. Surtani (2012). *Infinispan data grid platform*. Packt Publishing Ltd.

Nayak, A., A. Poriya, et D. Poojary (2013). Type of nosql databases and its comparison with relational databases. *International Journal of Applied Information Systems 5*(4), 16–19.

Salhi, H., F. Odeh, R. Nasser, et A. Taweel (2017). Open source in-memory data grid systems : Benchmarking hazelcast and infinispan. In *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering*, pp. 163–164. ACM.

Thirion, B. et S. J. Darmoni (1999). Simplified access to mesh tree structures on cismef. *Bull Med Libr Assoc 87*(4), 480–481.

Summary

New technologies such as Not Only SQL (NoSQL) databases have emerged to meet the needs of managing the increasing volume of available online data. NoSQL databases garantee hight performances but also lack of standards which complexify Infromation Retrieval tasks. This study presents the implementation of the backend underlying search engine of LiSSa (Health Litterature in French) with a NoSQL solution. Initialy based on a RDBMS, the use of a NoSQL database enable a significant improvement of performances but also requires the implementation of tools to perform classical IR tasks. However, a contextual evaluation of this Information Retrieval System still be necessary.