Universite de Montpellier
Rapport de projet

Claims checking

Belkassim BOUZIDI
Chakib ELHOUITI
Massili KEZZOUL
Abdelkader NEDJARI
Ramzi ZEROUAL

Encadrant :
M^r Konstantin Todorov

10 mai 2020

Remerciements

Tout d'abord nous souhaitons adresser nos remerciements au corps professoral et administratif de la faculté des sciences de Montpellier qui déploient des efforts pour assurer à leurs étudiants une formation actualisée.

En second lieu, nous tenons à remercier notre encadrant Mr Konstantin Todorov pour ses précieux conseils et son aide durant toute la période du travail.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre projet en acceptant d'examiner notre travail.

Nous remercions Mr Yahia Zeroual pour sa relecture attentive de ce rapport.

Table des matières

Chapitre 1

Organisation du projet

1.1 Méthodes d'organisation

Afin de mener à bien le développement du projet, nous avons décidé de travailler un maximum de temps ensemble et de manière très régulière. Nous nous sommes réunis trois à quatre fois par semaine, en vue de faire le point sur l'avancement du projet et de définir les objectifs restant à atteindre.

Ainsi, selon l'état de progression de la conception, nous réalisâmes les tâches en retard durant le week-end pour ne pas cumuler de retard et respecter l'intégralité du cahier des charges.

Toutes les semaines, nous nous sommes réunis avec notre encadrant , Mr Konstantin Todorov Lors de ces réunions, des mises au point relatives au projet, nous furent prodiguées, cela nous a permis de bénéficier de précieux conseils.

1.2 Decoupage du projet

Nous avons découpé la réalisation du projet en trois grandes phases.

1.2.1 Phase de modélisation

Durant cette étape, nous nous sommes réunis pour définir les fonctionnalités demandées par le projet. Notamment séparer les fonctionnalités importantes de celle moins importantes. Nous avons également choisi les outils de travail collaboratifs et les principales technologies utilisées, ainsi qu'une première modélisation du projet.

1.2.2 Phase de développement

Durant cette phase, nous avons commencé à implémenter les différentes fonctionnalités que nous avons modélisées lors de l'étape précédente, toute en améliorant la modélisation au fur et à mesure de l'avancement de notre projet. Nous avons notamment réalisé des tests pour les différents modules afin de s'assurer de leur bon fonctionnement.

1.2.3 Finalisation du projet

Cette étape a consisté en la réalisation des tests finaux afin de s'assurer que les défferents scripts fonctionnent en toute circonstance et éventuellement corriger les bogues qui peuvent apparaitre.

1.3 Outils de collaboration

Afin de s'organiser, nous avons décidé d'utiliser Git au travers du serveur GitLab hébergé par le service informatique de la faculté. En effet le logiciel libre Git a facilité grandement la collaboration entre nous. Le serveur GitLab quant à lui est fourni gratuitement par le service informatique de la faculté.

En ce qui concerne la rédaction de ce rapport, nous avons utilisé LaTeX, système de composition de documents créé par Leslie Lamport, pour faciliter la rédaction à plusieurs.

Schéma 1.1 – Logo du GitLab

Schéma 1.2 – Logo de Latex

Chapitre 2

Introduction au sujet

2.1 Fact-checking

2.1.1 Présentation du principe de fact-checking

Le fact-checking ou le vérification des faits, est une technique consistant d'une part à vérifier la véracité des faits et l'exactitude des chiffres présentés dans les médias et les différents réseaux sociaux et les blogues etc... . Cette notion est apparue aux États-Unis dans les années 1990. Mise en pratique par des journalistes d'investigation dans le cadre de leur profession, la méthode s'est démocratisée grâce à des logiciels aidant les particuliers à vérifier les faits.

L'entrée officielle du fact-checking en France date de 1995, quand est créée l'association Acrimed, qui se présente comme « l'observatoire des médias ».

En vue de la prolifération très rapide des informations, il devient de plus en plus important de s'assurer de la véracité des informations qui se trouvent partout sur Internet et autres médias, tant du point de vue de la société que de celui de la recherche. De nombreuses approches récentes dans diverses communautés scientifiques portent sur des problèmes tels que la vérification des faits, la détection de la pertinence ou de point de vue des documents par rapport à des revendications particulières.

2.1.2 Présentation de ClaimsKG

Le LIRMM [1] en collaboration avec 2 équipes allemandes (L3S Hannover et l'institut de sciences sociologiques GESIS à Cologne), a construit et mise à disposition la base de connaissance ClaimsKG [2] qui recueillit les informations et méta-données provenant d'un grand nombre de sites journalistiques internationaux de fact checking, tels que Politifact [3] ou Snopes [4]. ClaimsKG est un graphe de connaissances d'assertions annotées et liées qui facilite la création de requêtes structurées sur les assertions, leurs valeurs de vérité (True, Mostly False, etc...), leurs auteurs, date de publication, etc. ClaimsKG est généré par un pipeline entièrement automatisé qui collecte des assertions et des métadonnées à partir des sites de fact-checking, transforme les données en graphes de connaissances selon un modèle établis, et annote les assertions avec des entités DBpedia (Wikipedia). La base actuelle comprend plus de 32,000 assertions publiées depuis 1996 et est mis à jour régulièrement.

1. Le Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier
2. https://github.com/claimskg
3. https://www.politifact.com/
4. https://www.snopes.com/

2.1.3 Travail à réaliser

Le sujet du TER consiste en l'enrichissement de cette base de connaissances avec des nouvelles données provenant des sites web suivants :

Fatabyyano [5] Jordani, en Arabe, Fatabyyano (terme en arabe qui veut dire "Alors montrez-le" en arabe) est la première et la seule plateforme arabe certifiée par l'IFCN [6] ;

Vishvas.news [7] Un site Internet de vérification des faits multilingue (en hindi, anglais ...) qui s'engage à combattre la désinformation et les informations erronées.

Le but du TER sera d'identifier les assertions individuelles dans chaque histoire, ainsi que leur label de véracité et par la suite identifier les relations entre les assertions dans chaque histoire, ainsi que les relations entre ces histoires. Les données produites par ce projet seront intégrées à la base de connaissance ClaimsKG.

La principale difficulté qu'on s'attend à rencontrer est la gestion des différentes langues proposée par ces sites web. Notamment dans l'identification des différentes relations entre les assertions. En effet afin de reconnaitre les différents mots-clés du sujet traité par les articles, on utilise habituellement TAGME. un puissant outil de reconnaissance d'entités nommées dans un texte. Les langues utilisées par ces sites web ne sont pas prises en charge par TAGME. Il s'agit donc de trouver une alternative afin de reconnaitre ses différentes assertions. Ce problème ainsi que sa résolution sera détaillé plus tard dans ce rapport.

2.2 Technologies utilisées

Nous avons implémenter l'application en Python. Choix qui s'impose de lui même car couplé à la bibliothèque BeautifulSoup il devient très facile de faire du web scraping [8].

En ce qui concerne la traduction nous avons choisi d'utiliser Yandex Translator [9]. En effet en vu de la taille des données à traduire nous avons pas pu utiliser l'api de traduction de Google puisque cette dernière devient payante à partir d'un certain nombre de caractères.

En fin, la reconnaissance des différents entitées concerné est faite grace à l'outil TAGME [10] a travers son API python [11].

Schéma 2.1 – Logo de Python

Schéma 2.2 – Logo de BeautifulSoup

Schéma 2.3 – Logo de Yandex

5. https://fatabyyano.net/
6. International Fact-Checking Network : https://www.poynter.org/ifcn/
7. https://www.vishvasnews.com/
8. Le web scraping est une technique d'extraction du contenu de sites Web, via un script ou un programme
9. https://translate.yandex.com/
10. https://tagme.d4science.org/tagme/
11. Lien Github vers l'api Python de TAGME : https://github.com/marcocor/tagme-python

Chapitre 3

Conception et implementation du projet

3.1 Conception et Modélisation

3.1.1 Analyse de ClaimKG

Lors de cette première phase, le plus important a été de comprendre et s'imprégner du code et de la structure déjà mis en place et mis à notre disposition (Claimskg), comprendre les outils utilisés ainsi que la structure déjà établie.

La structure

Comme le projet est d'une envergure immense bien comprendre la structure était primordiale afin de respecter un maximum les outils utilisés. Notre encadrant nous à mis a disposition un écrit (que vous trouverez dans le dossier autre/ sous le nom de ClaimsKG.pdf) qui explique la structure global du projet dont voilà ci-dessous un résumer.

Schéma 3.1 – ClaimKG

Comme vous pouvez le voir sur le schéma, ClaimsKG est découpé en plusieurs phases. La première consiste à extraire les données brutes à partir de site web de fact-checking. Ensuite les données sont structurées afin de produire un fichier CSV. En deuxième phase le fichier CSV est passé à un générateur de Graphes de connaissances [1]. Dans notre cas, on s'occupe de la première phase. C'est-à-dire, extraire les données brutes du site web afin de produire un fichier CSV.

Le fichier CSV en question contient l'ensemble des faits traiter par un site web. Chaque fait est passer à un script de scraping afin d'extraite chaque information et de la structurer. La structure d'un fait est faite selon un modèl précis dont voici une partie :

rating value La valeur de véracité du fait ;

creativeWork author name Auteur de la claim ;

1. Knowledge Graph Generator

creativeWork datePublished Date de publication de la claim ;
claimReview author Auteur de l'article/analyse de la claim ;
extra entities claimReview Les différentes entitées presente dans la review.

Vous trouverez ci-dessous la structuration complète.

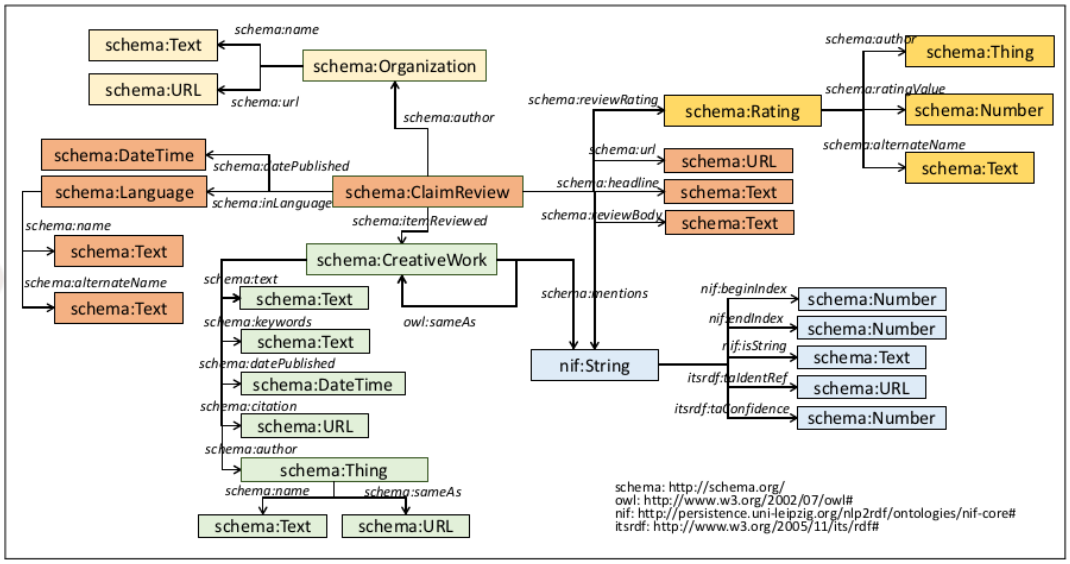

Schéma 3.2 – Structuration complète d'un fait

L'implémentation

Après la partie théorique de l'analyse nous avons commencer à regarder les implémentations déjà faite [2]. Nous avons analysé les classes principales comme celle de « Claim.py », puis comprendre comment l'exécution de l'extraction ce fessait avec « __init__.py ». nous avons ensuite analysé les différents scripts d'extraction de sites de fact-cheking tel que « africacheck.py ».

Schéma 3.3 – Ici une image UML pour expliquer
la structuration de leur implémentation

3.1.2 Structure des scripts

Parler des class et un peu d'UML

De ce que nous avons compris du projet il fallait écrire une classe qui représente l'extracteur, puis écrire des méthodes d'extraction (extraction du titre, extraction de la claim ... etc.) sur une page que nous appliquerons sur tout le site via un script python, chaque méthode représente une

2. Vous trouverez cette implémentation à cette addresse : https://github.com/claimskg/claimskg-extractor

colonne du fichier csv final où tous les éléments extraits sont stockés, tous les résultats de chaque méthode sont regroupés dans une méthode qui les regroupent dans la classe principale ‘Claim.py‘ et c’est sur celle-ci que le script ‘ init.py ‘ est appliqué.

parler de l’extraction des liens

puis l’extraction d’une claim

Puis parler du format des données que notre prog doit rendre

L’étape suivante était de trouver un site web respectant les critères de fact cheking ainsi que les critères (une véracité, une claim écrite, un auteur, des tags ... etc) du projet initial (Claimskg), pour apporter un maximum de diversité ainsi que notre contribution personnelle spécifique nous avons décidé avec l’accord de notre encadrant d’extraire les données d’un site d’une langue différente pas encore présente dans le projet, comme tous les étudiants présents dans ce projet ont une connaissance de la langue arabe il a été logique de commencer par cette langue, le seul site officiel de fact cheking en arabe est https ://fatabyyano.net/ nous avons donc analysé la structure du site web puis commencé son l’implementation, l’étape d’après fut de trouver comment faire une ”named entity recognition” sur les mêmes principes que celle déjà présenté (TagMe qui renvoi vers des liens wikipedia).

Le second site web sur lequel nous avons travaillé est https ://www.vishvasnews.com/, 11 langues différentes sont présentent sur ce site : L’anglais, Hindi, Pendjabi, Ourdou, Bengali, Tamoul, Malayalam, Goudjerati, Télougou, Marathi et L’odia , nous avons analysé la structure du site puis extrait ses données.

3.2 Implémentation

Chapitre 4

Analyse des résultats

4.1 Résultats

Comme premier résultat du site fatabayyano nous avons obtenu un fichier csv contenant l'url de la claim le rating et sa traduction la claim, la date , le titre, tous les liens présents sur la page et les tags tout cela en langue arabe.

Url	rating	rating_transl	claim	date	tags
https://fataby	صحيح	TRUE	في ولاية راخين	2019-07-25	"الشرطة,بورما
https://fataby	صحيح	TRUE	في رجل مسلم	2019-07-25	"اكات,بورميين
https://fataby	صحيح	TRUE	اكان في بورما	2019-07-25	"بورما,تعذيب
https://fataby	صحيح	TRUE	تين غرب بورما	2019-07-25	"تدمير,تكشف
https://fataby	صحيح	TRUE	لهاداً في العالم	2019-07-25	"هادا,الروهينجيا
https://fataby	صحيح	TRUE	يوك عن العمل	2019-07-03	"وم,علوم,طبي
https://fataby	صحيح	TRUE	نيقة أم إشاعة؟	2019-05-15	"وم,علوم,طبي
https://fataby	صحيح	TRUE	يقة أم مفبرك؟	2019-05-06	"وم,علوم,طبي
https://fataby	صحيح	TRUE	لثانية في 2019	2019-04-14	"وم,علوم,طبي
https://fataby	صحيح	TRUE	، أبوظبي و دبي	2019-04-03	"أخبار_حقيقية,
https://fataby	صحيح	TRUE	وسط كوبنهاغن	2019-03-26	"دنماركيون#,ة
https://fataby	صحيح	TRUE	خدمي فيسبوك	2019-03-23	"وم,علوم,طبي
https://fataby	صحيح	TRUE	خلال بث مباشر	2019-03-16	"Fraser_Anni
https://fataby	صحيح	TRUE	، تقرير فتبينوا ..	2019-03-15	"newzealand,
https://fataby	صحيح	TRUE	مل حول العالم	2019-03-13	"وم,علوم,طبي
https://fataby	صحيح	TRUE	لادش المجاورة	2019-02-27	"غلادش,حقيقة
https://fataby	صحيح	TRUE	قلب ولا نبض ؟	2019-02-23	"وم,علوم,طبي
https://fataby	صحيح	TRUE	دين الإسلامي ؟	2019-02-15	"Amazon,مات
https://fataby	صحيح	TRUE	يقة ام إشاعة ؟	2019-02-06	"Islam,هولندا,
https://fataby	صحيح	TRUE	ر ضد الإمارات؟	2019-01-11	"حطيرة_حيوان
https://fataby	صحيح	TRUE	ي سماء كوينز؟	2018-12-29	"ت,ضوء#,أزرق
https://fataby	صحيح	TRUE	، بعيون حمراء؟	2018-09-01	"وم,علوم,طبي
https://fataby	صحيح	TRUE	طاولة الجزار؟	2018-08-31	"وم,علوم,طبي
https://fataby	صحيح	TRUE	معدني منصهر؟	2018-08-19	"وم,علوم,طبي
https://fataby	صحيح	TRUE	ي الحرم المكي	2018-06-09	"وم,علوم,طبي
https://fataby	صحيح	TRUE	يداخل شجرة؟	2018-05-11	"وم,علوم,طبي
https://fataby	صحيح	TRUE	نيقة أم خرافة؟	2018-03-01	"وم,علوم,طبي

Schéma 4.1 – premier fichier csv obtenu

Puis après avoir appliqué le processus de traduction sur la claim et la review et après obtention des entités présentes dedans, deux autres colonnes sont ajoutées contenant les résultats du tagme du projet initial, ces résultats sont les noms attribués par la base de données wikipedia de cettte entité.

Les résultats du second site web "vishvasnews" sont aussi regroupés dans un fichier csv, qui lui est celui généré par le programme du projet initial compte tenu du fait qu'une des langues extraite est l'anglais, les colonnes sont les ratings values, l'auteur de la claim, l'auteur de la review de claim, l'url de la claim, la review, la date, la claim entière, les liens présents, le titre et les tags. Tout cela est rangé par langue l'anglais, Hindi, Pendjabi, Ourdou, Bengali, Tamoul, Malayalam, Goudjerati,

Schéma 4.2 – premier fichier csv obtenu

Télougou, Marathi puis L'odia puis par catégorie.

Schéma 4.3 – premier fichier csv obtenu

4.2 Problèmes rencotrés

Chapitre 5

Bilan et conclusions

5.1 Ce qu'on a fait et pas fait

5.2 Perspective

Annexe A

Annexe