Institut National des Langues et Civilisations Orientales

Département Textes, Informatique, Multilinguisme

Étude de similarités textuelles pour un système de questions-réponses dans le domaine de la Paye

MASTER

TRAITEMENT AUTOMATIQUE DES LANGUES

Parcours :

Ingénierie Multilingue

par

Katharine JIANG

Directeur de mémoire :

Damien Nouvel

Encadrante :

Gaëlle Recourcé

Année universitaire 2022/2023

TABLE DES MATIÈRES

Liste des figures

Liste des tableaux

Remerciements

Je tiens tout d'abord à adresser mes plus sincères remerciements à l'ensemble des professeurs du master TAL, pour leur accompagnement et la qualité de leur enseignement. Je tiens tout particulièrement à remercier mon encadrant, Damien Nouvel pour son soutien et ses conseils durant la rédaction de ce mémoire.

Je remercie bien évidemment la Revue Fiduciaire pour leur confiance et leur accueil chaleureux, notamment l'équipe « Projets » dans laquelle j'ai travaillé, sous la bienveillante direction de ma maître de stage, Gaëlle Recourcé. Merci également aux collègues avec lesquels je me suis liée d'affection durant ce stage, et qui ont participé à son bon déroulement.

Enfin, je tiens profondément à remercier mes camarades de master, pour leur bonne humeur, encouragement et entraide à toute épreuve.

Introduction

Résumé

Ce mémoire s'inscrit dans un contexte de valorisation des contenus du domaine de la Paye de l'entreprise Groupe Revue Fiduciaire, par le biais d'un système de question-réponses permettant d'interroger le fonds documentaire en langage naturel. Ce travail a pour objectif d'utiliser et de comparer différentes similarités textuelles existantes, appliquées sur un domaine spécialisé, afin d'associer un fragment réponse dans un document à une question posée. Nous chercherons à appliquer des méthodes à la fois statistiques (TF-IDF [Hiemstra, 2000]), et également à base de réseaux de neurones profonds, état de l'art actuel sur les questions de similarité (SBERT « Sentence-Transformers » [Reimers and Gurevych, 2019]). Ces approches seront testées individuellement, mais également combinées de manière hybride, simultanément ou en cascade, afin d'en observer leur efficacité et d'en dégager la combinaison la plus satisfaisante. Les résultats obtenus penchent en faveur d'une approche en cascade, avec une combinaison de pré-traitement type racinisation couplé à la méthode TF-IDF, puis un basculement en modèle SBERT selon un paramètre arbitraire défini. Ces expériences soulignent également un manque de modèles adaptés pour la langue française et un besoin de fine-tuning indispensable sur le domaine traité.

Mots-clés : système de questions-réponses, fouille de texte, TF-IDF, passage retrieval, SBERT, semantic textual similarity, RRF

Plan de lecture

Le mémoire s'articule en plusieurs chapitres, exposés ci-dessous :

— Le chapitre 1, intitulé « Cadre de l'étude », présente les enjeux de l'entreprise ainsi que les problématiques soulevées dans le cadre de ce travail.

— Le chapitre 2, nommé « État de l'art », vise à présenter les différentes méthodes et travaux sur les notions de notre sujet, à savoir le système de questions-réponses et la similarité textuelle.

— Le chapitre 3 « Corpus et métriques » décrit le processus de création de deux corpus : le corpus de travail et le corpus d'évaluation. Il aborde également les différentes métriques d'évaluation.

— Le chapitre 4 « Expérimentation et discussion » présente l'ensemble des approches testées, les résultats obtenus et leur analyse. Il discute également des limites et perspectives de ce travail.

Première partie

Contexte général

CHAPITRE 1

CADRE DE L'ÉTUDE

Sommaire

1.1 Introduction

A l'ère de ChatGPT, où la génération automatique de réponse et les agents conversationnels sont devenus des produits exploitables et plus crédibles auprès des entreprises, les domaines spécialisés comme le médical ou le juridique ne sont pas en reste. Suscitant une explosion d'intérêt ces dernières années, ces outils en plein essor posent cependant la question des sources utilisées, cruciales dans le domaine du droit, et de la responsabilité engagée lors de réponses erronées.

1.2 Présentation du projet de l'entreprise

Le travail exposé dans ce mémoire se place dans le contexte d'une entreprise d'édition juridique, s'adressant à des experts de plusieurs domaines (fiscal, social, paye, vie des affaires, comptabilité et patrimoine) à travers différents produits tels que des dictionnaires spécialisés, des revues ou encore des dépêches, sous format papier et numérique.

L'entreprise cherche à valoriser ces fonds documentaires, en permettant notamment à un utilisateur de trouver l'information qu'il recherche de manière efficace. Les documents pouvant être longs, faciliter l'accès à l'information, peut à la fois prendre la forme d'un passage surligné dans un document lors de questions précises, ou bien une synthèse de plusieurs passages pertinents selon une question posée dans un assistant conversationnel. Cet accès à l'information est souvent réalisé par des outils d'analyse de l'information, essentiellement textuelle. Quel qu'en soit l'application finale, le choix d'un découpage des documents en amont, en « unité de sens » informatives et autonomes, est primordial. C'est à partir d'un découpage fiable que la sélection du ou des passages les plus pertinents selon la requête posée par l'utilisateur pourra se baser. Dans le cadre de ce travail, le choix d'un retour direct au bon passage source, de manière contextualisée et contrôlée, a été privilégiée.

1.3 Sujet et périmètre du travail

Plus concrètement, le travail consiste dans un premier temps, à découper le contenu textuel des documents le plus pertinemment possible en « unités de sens », en se basant de la structure initiale existante, afin d'établir une base de départ qualitative. Dans un second temps, le travail vise à mener des tests de comparaison de similarité textuelle à l'aide de méthodes existantes, afin de trouver la méthode la plus adéquate disponible pour ce jeu de données particulier. Plus précisément, nous cherchons à répondre à une question posée en retournant une portion de document, que l'on nommera « fragment », tout au long de ce mémoire. L'évaluation de ce jeu de données quant à elle se base sur des couples de question/réponse récupérés d'une ressource interne complémentaire, dans une section dédiée où les rédacteurs répondent aux questions provenant d'abonnés.

Autrement dit, notre problème se décompose en ces différents points :

— quelles sont les unités de sens pertinentes dans notre cas d'application et comment les délimiter ?
— quelle méthode existante est la plus adéquate afin de relier la question à son bon fragment sur notre corpus spécialisé ? Le lien sémantique entre question et réponse peut-il être capturé par les différentes approches vectorielles de similarité ?
— quelle mesure utiliser pour évaluer nos approches ?

CHAPITRE 2

ÉTAT DE L'ART

Sommaire

2.1 Introduction

Dans cette partie, nous abordons les principales notions de notre sujet, afin de mieux cerner non seulement les différentes solutions existantes, mais également la position que nous adoptons dans le cadre de ce travail. Nous présenterons tout d'abord le système de questions-réponses : sa définition, ses caractéristiques distinctives par rapport au moteur de recherche ou encore de l'agent conversationnel, son évolution au cours du temps ainsi que sa structure. Pour une meilleure lisibilité, les différentes métriques d'évaluation d'un tel système seront exposées dans le chapitre suivant. Dans une seconde partie, nous nous focaliserons particulièrement sur l'approche basée sur la notion de similarité, en passant en revue la plupart des concepts permettant sa compréhension : la notion du pré-traitement, les modèles vectoriels ainsi que les mesures de similarité.

2.2 Systèmes de questions-réponses

2.2.1 Définition et distinction

Les systèmes de questions-réponses (*question-answering system* en anglais), sont une forme avancée de la recherche d'information (*information retrieval*), visant à répondre automatiquement à des requêtes formulées en langage naturel par un utilisateur à partir d'une base de connaissances. Les systèmes de questions-réponses vont au-delà de la recherche de documents car ils visent à fournir une réponse précise

à la question posée plutôt qu'une liste de documents qui seraient susceptibles de contenir la réponse [Voorhees and Tice, 2000], comme le ferait par exemple un moteur de recherche. Quant à l'agent conversationnel (aussi appelé *chatbot*), il se distingue du système de questions-réponses notamment par la simulation de dialogue (plusieurs tours de parole) ainsi que des réponses quasiment toujours générées, permettant ainsi une illusion de conversation avec l'utilisateur.

2.2.2 Historique

Les premiers Systèmes de Questions-Réponses (SQRs) remontent aux années 60-70 : on peut citer comme exemples le système BASEBALL répondant à des questions concernant les jeux de baseball de la ligue américaine [Green Jr et al., 1961] ou encore le système LUNAR répondant à des questions sur les roches lunaires [Woods, 1973]. Ces systèmes étaient restreints à des domaines spécifiques en raison des limites à la fois technologiques et de données de l'époque.

Depuis lors, les SQRs n'ont cessé d'évoluer, notamment avec l'avènement d'internet. Les nombreuses évolutions technologiques ont en effet permis leur développement, en ouvrant un champ encore plus large de langues, d'informations, et donc de possibilités : START par exemple, a su exploiter les ressources de l'Internet (World Wide Web) pour créer son système de questions-réponses [Katz, 1997].

Puis, face à la surcharge d'informations émanant du web ainsi qu'aux besoins accrus de systèmes de questions-réponses pour y remédier, de nombreuses campagnes d'évaluation ont vu le jour, en particulier celle organisée par TREC (Text REtrieval Conference) à partir de 1999 [Soboroff, 2021] qui a grandement encouragé et contribué aux travaux dans ce domaine. D'autres campagnes ont suivi cette dernière, telles que CLEF [Magnini et al., 2004] et NTCIR [Eguchi et al., 2002] pour aller au-delà de la langue anglaise et élargir la variété des langues aux langues européennes et asiatiques respectivement ; on peut également citer la campagne d'évaluation EQUER [Ayache et al., 2006] pour la langue française.

Les domaines couverts par les SQRs aujourd'hui sont nombreux, y compris le domaine particulièrement complexe du juridique. Diverses approches ont été menées par les SQRs dans ce domaine, dont des approches symboliques, à base de règles linguistiques, d'ontologies ou encore de graphes [Martinez-Gil, 2023].

Ces premières approches, hautement interprétables par l'humain mais dont le développement reste très coûteux, peu flexibles et limités au niveau de la performance, ont laissé peu à peu place à des modèles statistiques avec l'introduction du TF-IDF [Hiemstra, 2000] ou encore des CRFs [Pinto et al., 2003] qui démontraient de meilleures performances pour un moindre coût. La compréhension et résolution de ces systèmes face aux questions complexes restaient cependant limitées.

Récemment, ces approches statistiques ont elles-mêmes été dépassées par l'arrivée des solutions à base de réseaux de neurones grâce aux progrès considérables apportés par des modèles à base de transformers comme BERT [Tenney et al., 2019]. Ces derniers surpassent la plupart des résultats obtenus par des modèles antérieurs, bien qu'ils aient pour désavantage d'être peu interprétables par l'humain et de demander une grande masse de données pour être opérationnel.

2.2.3 Structure et types de système des questions-réponses

Les SQRs regroupent des systèmes divers pouvant aller de simples systèmes binaires oui/non à des systèmes complexes pouvant offrir une réponse synthétisée à

l'utilisateur, en recoupant plusieurs données [Voorhees and Tice, 2000], se trouvant par exemple dans différents documents. En effet, bien que la structure en trois parties du système de questions-réponses soit bien définie [Mitkov, 2022], à savoir :

1. analyse de la question
2. recherche des documents
3. sélection/extraction de la réponse

les systèmes peuvent varier de l'un à l'autre non seulement par rapport aux données (structurées ou non-structurées), mais également les aspects ci-dessous [Martinez-Gil, 2023] :

Domaine

Tout d'abord, le SQR peut être à domaine dit « ouvert » ou « fermé », c'est-à-dire répondant à n'importe quel domaine ou au contraire à un domaine spécifique, avec un périmètre de vocabulaire contrôlé. Il est important de noter que chaque domaine spécifique peut comporter des données hétérogènes avec des caractéristiques propres. Dans le domaine juridique par exemple, les comptes-rendus de jugement différeront des articles de lois, de dépositions de plainte, ou dans notre cas, d'articles éditoriaux juridiques.

Type de question

Les questions soumises aux SQRs peuvent se concentrer sur certains types de questions définis, les deux grandes catégories étant des questions de type factoïdes ou non-factoïdes, c'est-à-dire, respectivement, des questions factuelles « Quel est le salaire minimum légal ? » ou plus complexes telles que « Pourquoi les impôts sont-ils proportionnels aux revenus ? ».

Types de réponse

Les réponses quant à elles peuvent se présenter sous diverses formes et dépendent également du type de la question posée, il peut s'agir, comme évoqué plus haut, de réponses binaires oui/non ou vrai/faux, pour répondre à des questions fermées de type factuelles.

Il peut également s'agir de réponses sous forme d'extraction de segments textuels, avec différents de niveaux de granularité possibles :

— un paragraphe parmi un ensemble de documents (aussi appelé *paragraph retrieval*)
— une phrase au sein d'un paragraphe (*answer sentence selection*)
— une entité exacte au sein d'un paragraphe (date, personne, ...)

Comme dernier type de réponse, il existe également la réponse en langage naturel totalement générée, utilisée notamment dans les agents conversationnels.

La tâche peut être encore complexifiée si l'on ajoute la possibilité de non-réponse, par exemple lorsqu'aucun document dans la base de connaissances ne contient la réponse ou que la question est erronée « Pourquoi le salaire minimum est-il fixé à 500 euros ? ».

2.3 Similarité

Parmi les méthodes sur lesquels les systèmes extractifs peuvent s'appuyer, la similarité est une des approches les plus utilisées actuellement. Nous allons faire un point sur quelques notions théoriques essentielles sur le pré-traitement d'un texte qui serviront à mieux comprendre nos expériences, avant d'aborder les techniques les plus courantes de représentation vectorielle de textes. Les modèles utilisés dans la partie expérimentation seront également détaillés.

2.3.1 Notions théoriques sur le pré-traitement des données

Tokenisation

La tokenisation est une étape fondamentale en pré-traitement des données, permettant de découper le texte en petites unités. La tokenisation dépend de la langue traitée, du tokenizer utilisé mais aussi simplement de l'unité élémentaire considérée comme « token » que l'on souhaite passer au modèle : il peut s'agir de mots, de ponctuations, de caractères, ou encore d'expressions. Les n-grammes quant à eux sont des séquences de tokens de longueur n.

Normalisation

La normalisation est un procédé permettant de réduire la forme d'un mot à sa forme simplifiée et a souvent pour conséquence la réduction du vocabulaire et donc de la dimensionnalité des vecteurs. En dehors du passage des mots en minuscule, de la suppression de ponctuations ou encore de mots vides (*stopwords* en anglais [Rajaraman and Ullman, 2011], il existe deux types d'opérations principales sur les mots : la racinisation et la lemmatisation.

— La racinisation (en anglais, *stemming*), comme son nom l'indique, cherche à conserver uniquement la forme « racine » d'un mot. Cette opération est souvent réalisée en découpant de manière rudimentaire les suffixes et préfixes des mots ainsi que leur désinence, sans prise en compte du contexte.

  Exemple : le nom « obligation » comme l'adjectif « obligée » devient « oblig » après racinisation. Cependant, étant donné que l'analyse est basée sur les affixes, l'adjectif « obligatoire » sera quant à lui ramené sous la forme racine « obligatoir ».

— La lemmatisation consiste à ramener le mot sous sa forme canonique aussi appelée « lemme », qui correspond à la forme d'entrée de dictionnaire. Dans le cas du français, il s'agit par exemple de la forme masculin singulier pour les noms, adjectifs et articles ou encore l'infinitif pour les verbes. Elle se distingue de la racinisation par son analyse morphologique et syntaxique plus fine. La lemmatisation permet notamment de distinguer les homographies.

  Exemple : le mot « livre » en tant que nom, aura comme lemme « livre ». Cependant s'il est un verbe (comme dans la phrase « il livre le produit »), son lemme sera alors « livrer ».

2.3.2 Type de similarité et méthodes

La similarité textuelle est considérée comme une tâche complexe et constituant un défi majeur dans le domaine du TAL, notamment de par sa subjectivité. Elle

trouve de nombreuses applications dans différents travaux tels que la détection de plagiat [Hariharan, 2012], le résumé du texte [Aliguliyev, 2009], ou dans notre cas, les systèmes de questions-réponses. La tâche de question-réponse peut être considérée comme une tâche de similarité avec comme particularité, une asymétrie entre la taille de la requête et celle de la réponse.

La similarité peut-être mesurée par des méthodes mathématiques s'appuyant sur des représentations des textes, souvent numériques. La vectorisation est un moyen de représentation possible : il s'agit d'une transformation permettant de passer de données textuelles à des données numériques, en les projetant dans un espace vectoriel, tout en tentant autant que possible de conserver les propriétés linguistiques du texte d'origine. Cette représentation en vecteurs permet alors d'effectuer des mesures de similarité, par application d'opérations vectorielles sur ces textes transformés en vecteurs mathématiques. Il est cependant à noter que les représentations ne capturent pas toujours bien le sens des textes, menant à des calculs sur ces représentations qui peuvent être approximatifs voire faux.

On distingue deux types de similarités, lexicale et sémantique :

— La similarité lexicale (aussi appelée syntaxique), où les textes sont considérés comme des ensembles de mots, la similarité entre deux textes est caractérisée par le degré de recouvrement entre les ensembles de mots relatifs aux textes considérés.

— La similarité sémantique, qui permet de capturer des termes ayant le même sens, en intégrant cette fois-ci la sémantique des mots notamment via la vectorisation de texte, avec des techniques comme celle des plongements de mots (*word embeddings*). Il existe également des plongements de phrases, ou encore de documents.

Similarité lexicale

TF-IDF

Le TF-IDF (*Term Frequency - Inverse Document Frequency* en anglais) [Salton, 1975] est une mesure statistique fréquemment utilisée en recherche d'information. Cette méthode est basée sur une transformation des représentations en sacs de mots (ou Bag of Words). En opposition à ce dernier, qui considère les mots comme ayant la même importance, la méthode TF-IDF permet d'ajuster le poids d'un mot dans un document, en fonction de sa fréquence dans le document et sa présence dans l'ensemble des documents, permettant ainsi d'évaluer l'importance globale du mot. Cette approche permet donc d'obtenir, pour chaque texte, une représentation vectorielle qui comporte des vecteurs de poids et non plus d'occurrences. Elle se base sur le principe que des mots récurrents dans un document, et rares dans le reste du corpus, sont plus informatifs que des mots apparaissant de façon homogène.

Soit D un ensemble de documents et T un ensemble de termes. Le TF_{ij} (Term Frequency) se calcule par le nombre d'occurrences d'un terme t_i dans un document d_j. Mathématiquement, cela s'exprime comme suit :

$$TF_{ij} = \frac{\text{Nombre d'occurrences de } t_i \text{ dans } d_j}{\text{Nombre total de termes dans } d_j}$$

Le IDF_i (Inverse Document Frequency) se calcule comme le logarithme du rapport entre le nombre total de documents N dans la collection et le nombre de documents contenant le terme t_i, qui s'exprime par la formule suivante :

$$IDF_i = \log\left(\frac{N}{\text{Nombre de documents contenant } t_i}\right)$$

Enfin, la formule complète du TF-IDF pour un terme t_i dans un document d_j est donnée par le produit du TF_{ij} et du IDF_i :

$$TFIDF_{ij} = TF_{ij} \times IDF_i$$

Le résultat $TFIDF_{ij}$ donne une mesure de l'importance du terme t_i dans le document d_j par rapport à la collection entière de documents. Plus le $TFIDF_{ij}$ est élevé, plus le terme est considéré comme important pour ce document particulier.

Cette méthode statistique présente cependant plusieurs inconvénients. Le comptage d'occurrence des mots ne prend pas en compte l'ordre des mots ni leurs relations. Se basant seulement sur la forme de surface, la sémantique des mots, notamment les cas de synonymie et plus largement la sémantique des phrases est mal capturée par le TF-IDF. Le recouvrement au niveau du vocabulaire n'est en effet pas un critère suffisant pour parler de similarité, la langue naturelle permettant d'exprimer une même idée sous différentes formes.

De plus, les représentations BoW et TF-IDF produisent des vecteurs larges et creux (*sparse vectors*), c'est-à-dire contenant une grande majorité de 0 car la taille des vecteurs dépend de celui du vocabulaire : plus ce dernier est important, plus le vecteur sera grand.

Similarité sémantique

Word2Vec, Glove, fastText

Les plongements de mots (*word embeddings*) tentent de pallier aux problèmes du TF-IDF évoqués à la section précédente. Nous pouvons citer, parmi les modèles les plus populaires, Word2Vec [Mikolov et al., 2013], Glove [Pennington et al., 2014] ou encore fastText [Bojanowski et al., 2017]. Cette approche consiste à produire, pour chaque mot, un vecteur dense et de taille fixe. Ainsi, deux mots présents dans des contextes similaires seront considérés comme ayant une signification similaire et auront des vecteurs plus proches dans l'espace vectoriel en termes de distance (cf. 2.3.3). Cette représentation du mot par son voisinage s'appuie sur l'hypothèse distributionnelle en linguistique, qui peut être résumée par la phrase de « You shall know a word by the company it keeps » (« Vous connaîtrez un mot par ses fréquentations ») [Firth, 2020]. La représentation vectorielle d'un document entier peut être calculée en prenant la moyenne des vecteurs de chaque terme présent dans le document.

Néanmoins si cette nouvelle représentation de texte permet de mieux approximer la signification d'un mot, elle ne permet pas toujours de capturer le sens d'un mot en cas d'homonymie graphique (homographie). Un mot employé dans des contextes différents n'aura qu'un vecteur représentatif de son sens.

Exemple : « avocat » le fruit ou « avocat » le métier, n'auront pas la même signification mais seront pourtant représentés par un seul et même vecteur.

Transformers

BERT

Les transformers sont les modèles de représentation textuelle les plus avancés actuellement dans le domaine du TAL. Contrairement aux précédentes, ces modèles prennent en compte le fait qu'un mot, selon le contexte, aura un sens différent et donc sera représenté par un vecteur différent. Un des modèles « état de l'art » nommé BERT [Devlin et al., 2018] est basé sur une architecture à réseaux de neurones avec un mécanisme dit d'attention, s'appuyant sur le principe des transformateurs bidirectionnels. Ce mécanisme permet de capter les relations entre les mots, même si ces derniers sont éloignés dans la séquence, capturant ainsi des relations sémantiques complexes. Les modèles les plus connus pour la langue française sont CamemBERT [Martin et al., 2019] et FlauBERT [Le et al., 2019]. Cependant, BERT n'est pas le plus performant pour certaines tâches, dont la similarité sémantique [Reimers and Gurevych, 2019].

De plus, bien qu'il existe des modèles basés sur BERT ou ses dérivés français traitant de la tâche de question-réponse (*question answering*), tel que par exemple le modèle *camembert-base-squadFR-fquad-piaf*[1], ces derniers ne correspondant pas exactement à notre cas d'application. Il s'agit de questions ciblées sur des entités exactes à retrouver dans un paragraphe, de type date, personne, lieu etc., tandis que l'on cherche à retrouver des portions de texte entiers dans un document s'approchant de la taille d'un paragraphe.

SBERT

SBERT [Reimers and Gurevych, 2019] est une méthode de représentation de phrases, basée sur le modèle BERT. Cette méthode génère des représentations vectorielles au niveau des phrases plutôt que des représentations de mots individuels et est particulièrement utile pour la recherche de similarité sémantique entre différentes phrases. Il existe dans les modèles disponibles, deux types de similarité :

— la similarité symétrique, entre deux textes de même taille (phrase-phrase ou document-document)
— la similarité asymétrique, qui compare deux textes de longueur différente comme une requête courte et un fragment plus long.

Dans notre cas, la similarité asymétrique correspond à notre tâche de question-réponse. Cependant, il est à noter que les modèles pré-entraînés asymétriques à disposition ne sont qu'en langue anglaise tandis que ceux symétriques comportent des modèles en langue anglaise et multilingues (français inclus). De l'autre côté, il est difficile d'entraîner son propre modèle asymétrique, dans la langue et le domaine souhaité, car cette démarche requiert des données propres pour l'entraînement. Pour adapter le domaine d'un modèle pré-entraîné, nous avons besoin d'un jeu de données assez volumineux sur le français juridique, avec des paires similaires notées comme plus ou moins similaires à l'aide d'une échelle, ce qui est hors de portée de ce travail.

1. https ://huggingface.co/etalab-ia/camembert-base-squadFR-fquad-piaf

2.3.3 Mesure de similarité

Une fois le moyen de vectorisation choisi pour représenter nos textes, différentes mesures de similarité sont envisageables. Dans les opérations vectorielles les plus courantes, on retrouve ainsi : la distance euclidienne, la similarité cosinus ou encore le produit scalaire (*dot product*). Calculer la similarité entre un document et une requête, dans le cadre du question-réponse, est une manière de voir à quel point ces éléments sont proches dans l'espace vectoriel : plus les vecteurs de phrases sont proches dans cet espace vectoriel, plus les phrases sont considérées comme sémantiquement similaires. Les vecteurs peuvent être, selon la mesure choisie, proches au niveau de la distance à parcourir entre leurs extrémités, de leur angle ou des deux à la fois. Autrement dit, plus ces vecteurs ont une distance ou un angle petit, plus la similarité entre les deux augmente.

Distance euclidienne

Soient $A = (a_1, a_2, \ldots, a_n)$ et $B = (b_1, b_2, \ldots, b_n)$ deux points dans un espace de dimension n, la distance euclidienne entre A et B est donnée par :

$$d(A, B) = \sqrt{(b_1 - a_1)^2 + (b_2 - a_2)^2 + \ldots + (b_n - a_n)^2} \tag{2.1}$$

La distance euclidienne a cependant pour défaut de prendre en compte la longueur des textes. En effet, plus un fragment sera long, plus le nombre de mots partagés sera élevé sans pour autant que ces documents soient réellement similaires.

Produit scalaire

A la différence de la distance euclidienne, le produit scalaire prend en compte à la fois la magnitude et l'angle des vecteurs. Il se calcule comme suit pour deux vecteurs A et B, de dimension n :

$$\mathbf{A} \cdot \mathbf{B} = a_1 * b_1 + a_2 * b_2 + \ldots + a_n * b_n \tag{2.2}$$

Similarité cosinus

Quant à la similarité cosinus, elle se calcule en effectuant le produit scalaire des deux vecteurs, le tout divisé par le produit des normes des deux vecteurs. Elle prend ainsi en compte l'angle entre les vecteurs et non leur taille, réduisant leur impact sur le calcul de similarité. Cette approche est la plus couramment utilisé, notamment car les valeurs sont normalisées et se situent entre [-1, 1], à la différence du produit scalaire.

Soit A et B deux vecteurs n-dimensionnels :

$$\cos\theta = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| * \|\mathbf{B}\|} \tag{2.3}$$

Avec la norme d'un vecteur V de dimension n se calculant ainsi :

$$\|\mathbf{V}\| = \sqrt{v_1 * v_1 + v_2 * v_2 + \ldots + v_n * v_n} \tag{2.4}$$

2.4 Positionnement de notre travail

Pour résumer notre positionnement, nous nous situons dans le cadre d'un système de questions-réponses dans un domaine fermé, à savoir celui de la Paye d'un point de vue juridique. Les questions sont dans notre cas de type factoïdes, et la réponse attendue s'apparente au « paragraph retrieval », i.e. le niveau intermédiaire entre la recherche de document et l'extraction de réponse exacte [Othman and Faiz, 2016]. Dans notre cas cependant, nous parlerons plutôt de « fragment » de document que de paragraphes visuels (c.f. 3.2.3).

Ce choix de réponse contextualisée se rapproche de la vision [Lin et al., 2003] qui rappelle l'importance de l'interaction humain-machine et des préférences de l'utilisateur, qui, à la date de parution, étaient des réponses avec contexte au niveau du paragraphe ou du document. Le SQR souhaité ayant une visée d'assistant juridique automatique, l'interaction humain-machine ainsi que les préférences utilisateurs ne seront pas mis de côté par l'évolution de la technologie, qui se tourne vers des systèmes avec une extraction toujours plus exacte de la réponse, et éliminant souvent tout contexte. Notre vision est donc à l'opposé du travail [Hsu et al., 2021], qui cherche au contraire à reformuler des passages corrects estimés trop longs et où les passages ne répondant pas directement à la question sont considérés comme superflus.

En d'autres termes, aucune reformulation ni troncation stricte ne seront effectuées afin de ne pas dénaturer le sens des réponses et de fournir une réponse en contexte. Ce choix offre un plus grand contrôle et est très précieuse dans notre cas d'étude et dans le domaine dans lequel nous nous situons.

2.5 Conclusion

Dans ce chapitre, nous avons pu explorer et mieux comprendre les notions de notre sujet. Tout d'abord, nous avons passé en revue les différentes approches des systèmes de questions-réponses, qui ont évolué en même temps que les nouveaux besoins utilisateurs. Les modèles les plus performants aujourd'hui se tournent vers des modèles à base de neurones profonds. Les types de systèmes étant nombreux, nous avons également détaillé les différents aspects pouvant caractériser un système de questions-réponses et le différencier d'un autre, au niveau du domaine, du type de question traité et du type de réponse attendu notamment. Nous nous sommes concentrés par la suite sur une des méthodes possibles pour le système de questions-réponses, à savoir celle la similarité. Pour traiter cette notion, nous avons parlé du pré-traitement des données avant d'évoquer la différence entre similarité lexicale et sémantique, tout en montrant des exemples de méthodes. En dernier point, trois formules traditionnelles de similarité ont été exposées, ainsi que leurs avantages et/ou limites respectives.

Deuxième partie

Expérimentations

CHAPITRE 3

CORPUS ET MÉTRIQUES

Sommaire

3.1 Introduction

Dans cette partie, nous nous intéressons aux ressources mises à disposition pour établir notre corpus de travail mais aussi notre corpus d'évaluation. Pour le corpus de travail, nous nous penchons tout d'abord sur le format de ces données ainsi que leurs spécificités, en terminant par quelques statistiques globales. Nous parlons par la suite du travail important sur le pré-traitement de ce corpus, indispensable pour avoir des données d'une qualité satisfaisante afin de mener à bien nos expérimentations. Enfin, pour permettre une meilleure compréhension des résultats attendus et de mieux interpréter les résultats obtenus dans le prochain chapitre, nous abordons la constitution du corpus d'évaluation ainsi que les mesures retenues.

3.2 Corpus de travail

3.2.1 Description des données

L'entreprise possède un grand nombre de ressources, catégorisées en six grands domaines : Fiscal, Social, Paye, Vie des affaires, Comptabilité et Patrimoine. Chaque domaine comporte différents types de contenus variables : des articles, des dictionnaires, des guides, des dépêches... Il est important de noter que tous les textes sont au format XML, mais que la structure du document sera déterminée par son type. En effet, chaque type aura une structure XML et des caractéristiques qui lui seront

propres, que cela soit au niveau de ses métadonnées ou au niveau de ses données textuelles.

Dans le cadre de notre travail nous nous concentrons sur les documents spécialisés dans le domaine de la Paye. Le domaine est certes juridique mais se distingue totalement des textes de lois (législation), de rendus de décisions (jurisprudence), ou de vulgarisation (service public). Il s'agit de textes rédigés par des rédacteurs juridiques sur le domaine du droit et ses notions, à destination de professionnels, à savoir ici les gestionnaires de paye.

Les données stockées par l'entreprise dans le domaine de la Paye contiennent principalement des articles, des dépêches et deux dictionnaires. Après discussion, le périmètre de notre corpus s'est limité au dictionnaire de la Paye (regard générique) et au dictionnaire Paye DSN (regard avec un focus sur la DSN), car ces derniers ont un volume satisfaisant tout en reprenant tous les concepts utiles au domaine. Ce type de données est le plus complet comparé aux autres ressources, qui pouvaient reprendre des éléments déjà évoqués par les dictionnaires. Ce dernier a également le grand avantage d'être le mieux structuré des types car il se base sur de vrais dictionnaires physiques, avec des entrées, des sections, des titres etc. Concernant la fenêtre temporelle du corpus, nous prenons la version la plus récente possible, à savoir les dictionnaires du millésime 2023, correspondant au dernier renouvellement en date.

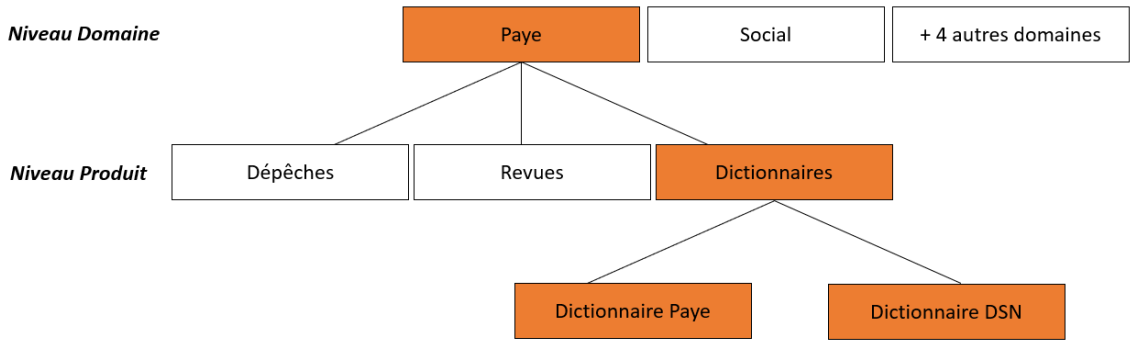


FIGURE 3.1 – Schéma représentant la sélection des données pour notre corpus de travail

3.2.2 Problématique

Dans le périmètre établi du dictionnaire Paye, chaque document XML représente une entrée de dictionnaire. Cette entrée est souvent longue, comportant au minimum une dizaine de paragraphes, en passant par des définitions, des précisions légales, des cas particuliers, des exemples, des tableaux etc. Cependant, notre objectif n'est pas de renvoyer le document entier contenant la réponse, mais bien de retourner directement un passage de texte considéré comme la zone pertinente qui répond à la question posée. Pour y parvenir, il est donc indispensable d'effectuer un découpage intelligent des documents afin de passer du niveau document à un niveau plus fin. Ce découpage ne peut être le fruit du hasard et doit, autant que possible, constituer des « unités de sens » cohérents et indépendants afin de créer des fragments dits « pertinents ». Pour cela, la structuration initiale des documents nous aide déjà grandement à délimiter ce que l'on nomme « fragment ». Ce dernier peut être défini succinctement comme une unité autonome sémantiquement, pouvant comporter plusieurs paragraphes et

abordant une sous-notion complète. Il est important de garder à l'esprit que ce choix de découpage reste subjectif et s'appuie sur la structure initiale choisie par le rédacteur.

3.2.3 Pré-traitement

Pour constituer notre corpus en fragments, deux grandes étapes ont été réalisées : la première étape de parsing à l'aide d'une feuille de style XSL et la deuxième étape de fragmentation avec un script python.

Nettoyage et parsing via XSL

Afin d'extraire le contenu textuel des documents XML, une transformation XSL spécifique sur les données structurées du type dictionnaire a tout d'abord été réalisé, afin de récupérer le contenu sans les liens externes, listes de références internes ou autres éléments bruyants ne faisant pas partie du texte affiché. En plus de cette extraction, un tag de la forme « @TAG » a été rajouté devant chaque élément textuel nettoyé. Les exemples de tags peuvent être, entres autres : « @TITRE_1 », « @TEXTE » ou encore « @TABLEAU ».

Fragmentation via script python

Les tags créés dans la première étape servent au script python à repérer les couples de « tag-texte » et permettent ainsi de regrouper ou séparer les éléments textuels nettoyés, selon des règles plus ou moins triviales.

Par exemple, nous regroupons des suites d'éléments textuels « @TEXTE » correspondant à des paragraphes distincts d'une même sous-partie. Ces derniers ont moins de sens isolés car il s'agit souvent de cas de co-référence, où le contexte est repris par pronom (c.f. deuxième fragment de la figure 3.3). En outre, nous avons ajouté par concaténation, à la tête de chaque fragment, le ou les titres auxquels il appartient. Ce supplément d'informations peut s'avérer utile dans nos expérimentations de calcul de similarité, ajoutant du contexte au fragment, qui, seul, peut ne pas être assez explicite. Cette concaténation de titres a une profondeur maximale de trois niveaux.

Durant la fragmentation, nous avons également décidé de ne pas garder certains éléments qui étaient en dehors de notre cadre de travail à savoir : les modèles à remplir et les tableaux car ils ne constituent pas un fragment réponse possible pour une question posée dans notre cas d'application et donc un fragment réponse que l'on souhaite garder. Par ailleurs, le format extrêmement hétérogène des tableaux aurait fait l'objet d'un travail à part entière afin de les découper en fragments pertinents et de les transformer en phrases compréhensibles.

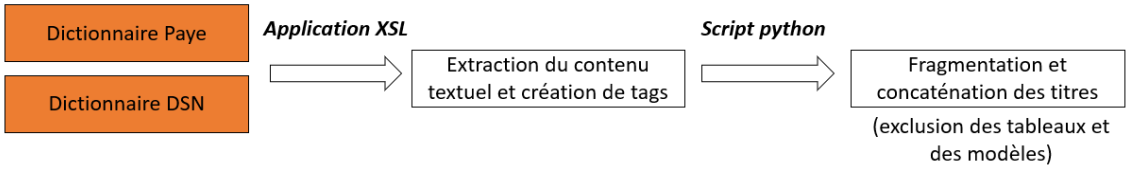


FIGURE 3.2 – Schéma récapitulatif de la chaîne de traitement de fragmentation des données

3.2.4 Résultats et statistiques

Le corpus composé du dictionnaire paye et du dictionnaire DSN, comporte les caractéristiques suivantes.

	Dictionnaire Paye	Dictionnaire DSN	Total
Nombre de documents	620	101	701
Nombre de fragments	8344	1286	9630
Nombre de tokens	1 501 448	183 220	1 684 668

TABLE 3.1 – Statistiques du corpus en nombre de documents, fragments et tokens

Nous présentons également ci-dessous un exemple de fragmentation réalisé. Une partie du document XML peut être trouvé en annexe (A.1), afin de mieux en comprendre sa structure.

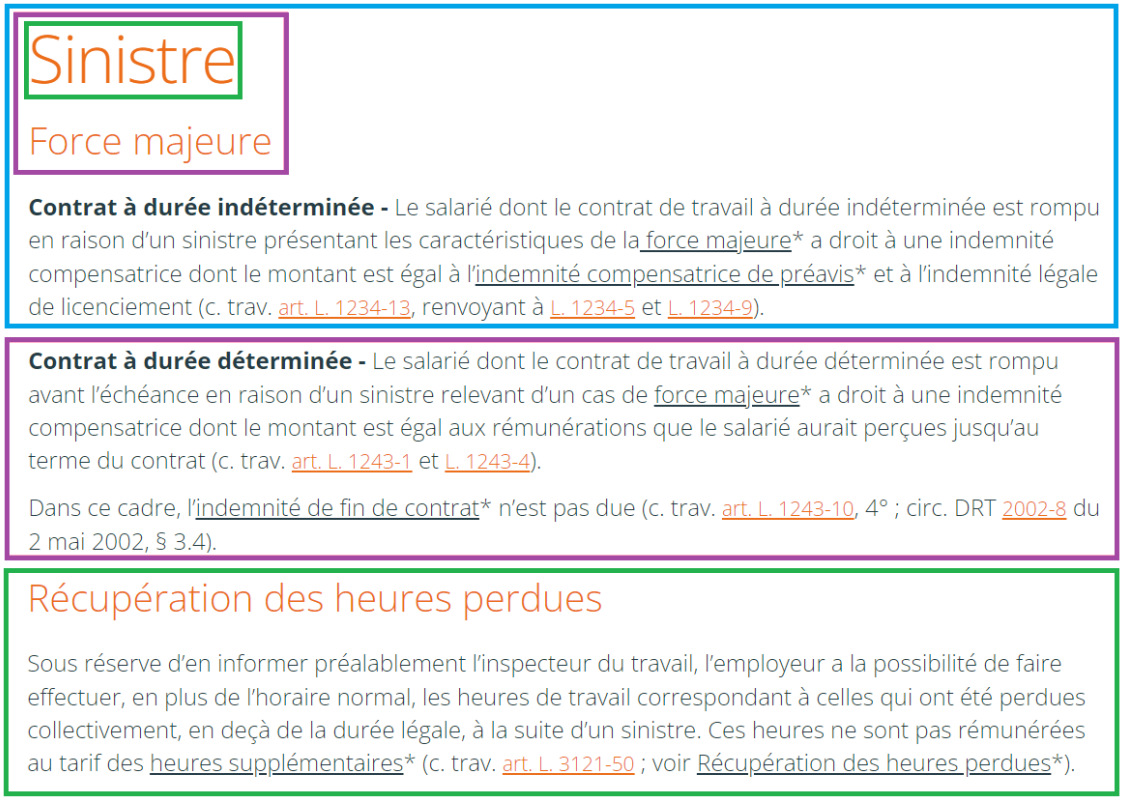


FIGURE 3.3 – Exemple de découpage d'un document comportant trois fragments, avec concaténation de leur titre et sous-titre

3.2.5 Limites

Tout d'abord, et comme évoqué dans la partie problématique, le découpage profite de la structure initiale existante, mais reste un choix subjectif pris par les rédacteurs, à la frontière entre choix sémantique et éditorial (voire pédagogique).

De plus, bien que les documents soient structurés, certaines coquilles dûes au formatage peuvent subsister. Par exemple, il existe des sections <COMPLEMENT> qui n'ont pas à être intégrées aux fragments réponses candidats mais qui sont un cas

particulier qui n'ont été repérées qu'après-coup. Il en est de même pour les chapô, qui ont été conservés à tort dans les fragments réponses candidats, car ces derniers sont des courts résumés introductifs du document et ne peuvent constituer une réponse à une question posée. Cela signifie que ces deux types de fragments sont éligibles en tant que fragment réponse dans le corpus d'évaluation et lors des expériences menées alors que ces éléments auraient dû être supprimés.

Pour finir, comme dit plus haut dans la partie fragmentation, les tableaux ont été écartés car ils ne correspondent pas à des fragments réponses candidats dans notre cas d'application. Dans un cas d'usage plus large, ces derniers devraient être cependant traités et fragmentés.

3.3 Constitution du corpus d'évaluation de référence

3.3.1 Données

Un des points cruciaux de notre travail a été de déterminer quelle méthode d'évaluation serait pertinente pour ce corpus si spécifique, que nous ne pouvions annoter nous-mêmes par manque d'expertise. Pour ce faire, nous avons exploité des couples question/réponse dont nous disposions dans une source de données complémentaire. Il s'agit de sections courtes dans lesquels les rédacteurs répondent à des questions des lecteurs. Ces questions sont particulièrement soignées et peuvent avoir été reformulées lorsqu'elles étaient imprécises ou incorrectes.

Dans ces couples question/réponse, nous avons d'une part des questions, et d'autre part, les réponses à ces questions, rédigées par les rédacteurs. Cependant, ces réponses rédigées ne se superposent pas toujours avec des passages trouvables tels quels dans le dictionnaire. Néanmoins, après observation, nous nous sommes rendus compte qu'un certain nombre de réponses rédigées correspondaient à des passages de dictionnaire, à quelques tournures près.

3.3.2 Construction automatique du corpus de référence

Principe

De la constatation précédente est née l'idée de créer un corpus d'évaluation à partir de ces réponses rédigées. Plus concrètement, nous cherchons à mettre en correspondance la réponse rédigée du rédacteur avec un fragment du dictionnaire afin de passer d'un corpus « question/réponse rédigée » à un corpus d'évaluation « question/fragment réponse du dictionnaire ».

Pour ce faire, nous avons opté pour la méthode TF-IDF (c.f. 2.3.2) à l'aide d'un vectoriser sklearn. De toutes les méthodes statistiques, cette méthode est de loin la plus appropriée pour repérer des textes similaires syntaxiquement (c.f 2.3.2) , c'est-à-dire avec des termes en commun « graphiquement », tout en prenant en compte le poids de chaque mot du document et de la longueur de ce dernier. D'autres méthodes de comparaison de chaînes similaires comme l'indice de Jaccard [Jaccard, 1901] (nombre d'éléments en commun entre deux ensembles) serait déséquilibré par les documents trop longs, tandis que la distance de Levenshtein [Levenshtein et al., 1966] (nombre minimal d'opérations pour passer d'une chaîne à l'autre) serait très coûteuse pour un résultat non forcément meilleur.

En outre, afin de limiter les inconvénients du sac de mots, nous utilisons en plus des tokens en tant que vocabulaire du vecteur, tous les bigrammes du fragment afin de

capturer des concepts composés plus larges. Les séquences de deux tokens permettent ainsi de prendre légèrement en compte l'ordre des mots. Cet ajout de bigrammes dans le vecteur est facilement implémentable avec sklearn, en modidifiant l'attribut ngram_rang à (1,2) au lieu de (1,1).

Notre corpus d'évaluation se constitue donc au final, d'une question d'un lecteur associé à un fragment réponse trouvée dans le dictionnaire, qui est le plus proche syntaxiquement de la réponse rédigée par le rédacteur grâce à un calcul TF-IDF. Ces éléments constituent ainsi notre corpus de question/réponse de référence.

Résultats et statistiques

Afin de mieux illustrer les propos précédents, deux exemples de résultat de notre corpus de référence sont présentés ci-dessous :

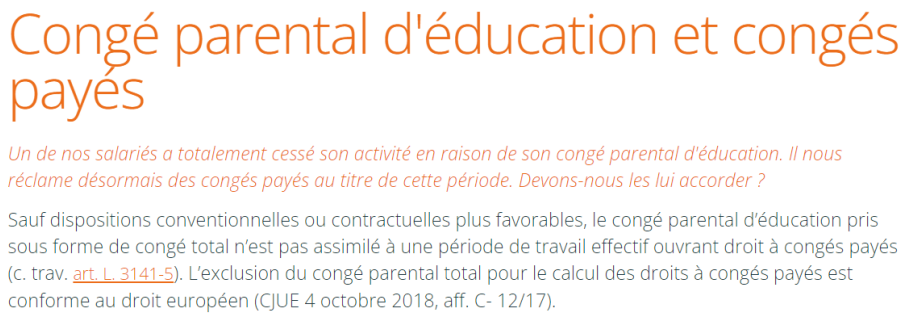


FIGURE 3.4 – Exemple 1 de couple question abonné / réponse rédacteur

FIGURE 3.5 – Passage du dictionnaire associé à la réponse de l'exemple 1, avec score de similarité de 83,36%

Questions : Les questions de notre corpus d'évaluation sont particulières : elles sont à la fois représentatives de questions d'utilisateurs car ce sont des véritables situations et questions que se posent par des experts, mais restent des questions plus ou moins revues par les rédacteurs, corrigées et reformulées si celles-ci comportaient des erreurs. Elles s'apparentent donc plus à des questions de Foire Aux Questions et comportent notamment les concepts abordés dans le dictionnaire.

Finalement, certains termes utilisés dans les questions sont à prendre au pied de la lettre car ils font référence à un concept précis, tandis que d'autres requièrent

FIGURE 3.6 – Exemple 2 de couple question abonné / réponse rédacteur

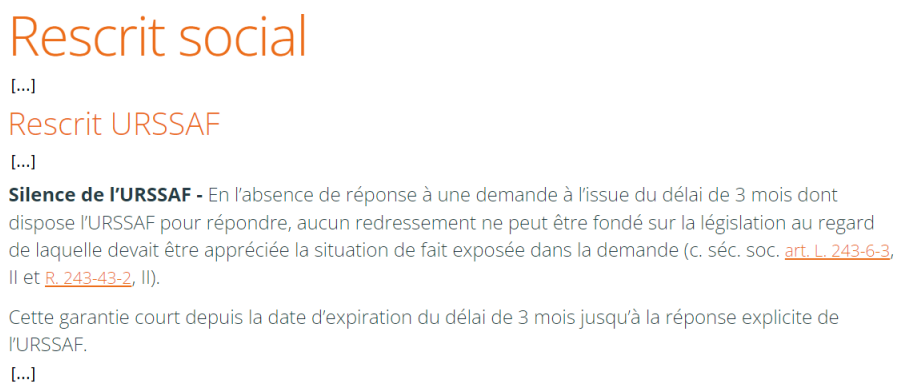


FIGURE 3.7 – Passage du dictionnaire associé à la réponse de l'exemple 2, avec un score de similarité de 53,90%

une plus marge sémantique, car ils sont moins précis et plus sujet à variation. Par exemple : lorsqu'on pose une question sur un contrat CDD, on parle de ce type de contrat et pas d'un autre, même s'il se rapproche sémantiquement du terme CDI. De l'autre côté, si l'on évoque le « calcul d'un salaire » pour une situation donnée, nous aimerions retrouver sémantiquement dans la réponse, des termes se rapprochant du concept de « rémunération », car nous ne retrouverons sûrement pas la même formulation dans le fragment réponse.

De manière générale, les questions sont longues et comportent de 1 à 5 phrases, exposant souvent une situation en une ou plusieurs phrases suivie d'une question. Le nombre moyen de tokens d'une question entière est d'environ 29 tokens (ponctuation incluse). 35% des questions sont des questions directes contre 65% de questions situationnelles suivie d'une question.

Réponses : En ce qui concerne les fragments réponses, nous avons observé qu'en-dessous d'un certain seuil de similarité, la réponse de référence et le fragment le plus proche retourné par le TF-IDF étaient trop différents l'un de l'autre pour que le fragment trouvé soit considéré comme fragment valide. Ce seuil a été fixé arbitrairement fixé à 0.3. Il s'agissait de réponses qui ne s'apparentent pas suffisamment à un fragment du dictionnaire, soit car la réponse rédigée recoupe plusieurs notions de plusieurs fragments, soit car elle est en dehors du périmètre du dictionnaire. Ces couples ont donc été écartés pour cette raison.

Couples Q/R totaux récupérés	187
Couples Q/R retenus (seuil score de similarité 0.3)	131

TABLE 3.2 – Statistiques globales sur les couples Q/R

Les deux graphiques suivants (figures 3.8 et3.9) montrent des statistiques sur les fragments réponses qui sont à garder à l'esprit lors de nos expérimentations. Par exemple, pour le premier graphique (figure 3.8) nous pouvons observer que deux documents ne comportent qu'un seul fragment. Sachant que l'on donne le bon document à la recherche sémantique, deux résultats seront d'office corrects car un seul fragment ne pourra être choisi. Ce cas reste cependant rare : la majeure partie des documents se situe aux alentours de 5 à 25 fragments.

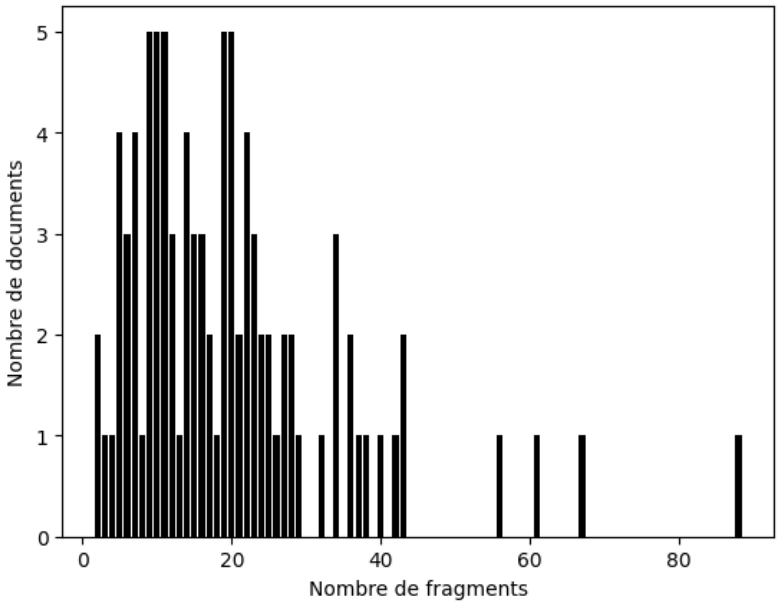


FIGURE 3.8 – Nombre de fragments sur les 131 documents réponses

Le deuxième graphique (figure 3.9) représente la répartition du nombre de tokens dans nos fragments réponses. Environ 44% des fragments réponses comportent 128 tokens ou moins, 77% des fragments comportent 256 tokens ou moins, et environ 95% se situent en dessous de 512 tokens. Ces données sont importantes car la partie expérimentation (c.f. chapitre 4) utilise certains modèles ne pouvant supporter plus de 128, 256 ou 512 tokens.

Limites

Ce corpus d'évaluation créé via les ressources dont nous disposions comporte bien entendu des limites. Ce dernier a été constitué automatiquement sans relecture manuelle, et reste donc un « silver standard » plus qu'un « gold ».

En effet, bien que la méthode TF-IDF est la plus adaptée pour la constitution du corpus de référence, elle peut tout de même retourner un fragment ne répondant pas à la question, notamment car elle ne se trouve pas dans les données. Malgré le seuil défini en-dessous duquel la réponse de dictionnaire est jugée trop différente, il reste possible de trouver quelques cas où le fragment réponse du dictionnaire peut être jugé non pertinent.

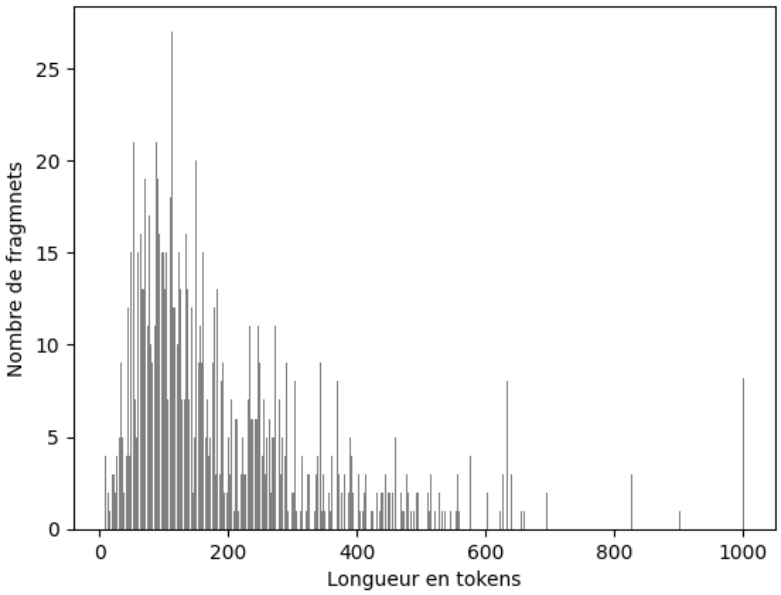


FIGURE 3.9 – Nombre de tokens sur les 131 fragments réponses

De plus, en dehors du caractère imparfait de certains fragments réponses retournés, le corpus n'est pas volumineux. Les questions sont variées en terme de situations, mais peu variées en terme de forme. Ces dernières sont très normées, correctement formulées et peuvent ne pas être représentatives de vraies questions posées par les utilisateurs mais plutôt des « questions types » à la manière d'une Foire Aux Questions (F.A.Q).

3.4 Mesures d'évaluation

En ce qui concerne les métriques d'évaluation des systèmes de questions-réponses, ces dernières sont nombreuses, car « l'évaluation de tout système permettant d'extraire de l'information est difficile » [Perret, 2005]. Les mesures d'évaluation pour les systèmes de question/réponses dépendent surtout du type de réponse attendu.

Sur l'axe macroscopique, il existe des métriques traditionnelles de la recherche d'information de ranking multiples tels que le Mean Average Precision (MAP), le Mean Reciprocal Rank (MRR), ou encore le normalised Discounted Cumulative Gain (nDCG), pour un classement de plusieurs réponses triées par pertinence. Ces mesures sont utiles lorsque plusieurs réponses sont jugées comme valides.

Au niveau de la granularité, si l'on attend en réponse une entité exacte et non un passage ou un document, l'exact matching sera une mesure intéressante dans ce cas de recherche d'une fenêtre précise.

Nous nous situons cependant dans le cas d'un seul fragment réponse attendu. Le fragment réponse est soit correct selon notre corpus de référence (VP : vrai positif, le fragment trouvé est celui attendu), soit incorrect (FP : faux positif, le fragment trouvé n'était pas celui attendu). Ces deux éléments permettent de calculer le taux de précision (formule 3.1). A noter que dans notre cas de notation binaire correct/incorrect, il n'existe pas de vrais négatifs ni de faux négatifs afin de calculer un quelconque rappel ou une f-mesure.

$$\text{Précision} = \frac{\text{vrais positifs}}{\text{vrais positifs} + \text{faux positifs}} \tag{3.1}$$

Cette mesure permet de voir en globalité si nos différentes approches, menées dans la section suivante fonctionnent. En réalité, pour une approche plus qualitative, il faut se pencher sur chaque cas pour analyser si les fragments retournés sont réellement incorrects, ou incorrects à cause d'une erreur dans le corpus de référence créé ou encore une réponse à moitié valable.

3.5 Conclusion

Nous avons vu dans ce chapitre la constitution de deux corpus : le corpus de travail et le corpus d'évaluation. Pour le corpus de travail, nous avons limité le périmètre de nos données à deux dictionnaires, comportant toutes les notions fondamentales du domaine. Par la suite, les documents retenus ont été fragmentés, non pas par paragraphes, mais par « unités de sens », déterminés par la structure-même des documents. Au-delà du découpage effectué, nous avons concaténé tous les titres et sous-titres se rapportant au fragment, afin de mieux les contextualiser. Pour le corpus d'évaluation, ce dernier s'est basé sur des couples de question/réponse récupérés d'une source interne annexe. Ces couples ne pouvant être exploités tels quels, les réponses rédigées ont été reliées à des fragments réponses appartenant au corpus de travail. Cette mise en correspondance a été réalisée de manière automatique, à l'aide de la méthode TF-IDF contenant unigrammes et bigrammes. Pour finir, nous avons listé les différentes métriques d'évaluation utilisés pour mesurer la qualité des systèmes de questions-réponses et retenue celle de la précision.

CHAPITRE 4

EXPÉRIMENTATION ET DISCUSSION

Sommaire

4.1 Introduction

Dans ce chapitre, nous décrivons les différentes approches qui ont été testées. Nous commençons avec la méthode statistique TF-IDF [Hiemstra, 2000], en effectuant différents pré-traitements pour tenter d'améliorer les résultats. Nous enchaînons avec des modèles état de l'art au niveau de la capture de sémantique des phrases, à base de réseaux de neurones : les modèles SBERT [Reimers and Gurevych, 2019]. Nous testons les modèles à disposition, à savoir à la fois des modèles symétriques (anglais et multilingue) et asymétriques (anglais) avec leur calcul de similarité recommandé (cosinus ou produit scalaire). Puis nous terminons par une combinaison des deux méthodes dans une partie d'hybridation, par fusion des résultats puis par application des modèles en cascade. Par ailleurs, nous quantifions, exemplifions et analysons les résultats obtenus par les diverses approches.

4.2 Expérimentations individuelles

Nous cherchons à voir si les questions posées contiennent assez d'information pour déterminer le bon fragment réponse dans un document et s'il est possible de déterminer un seuil de confiance. Nous formons comme hypothèse que la méthode TF-IDF sera puissante dans notre cadre d'application, les questions étant très bien

dirigées. De l'autre côté, la capture du sens des phrases des modèles SBERT peut nous être utile, bien que non entraîné sur notre domaine.

4.2.1 Méthode statistique TF-IDF

Première approche

Dans un premier temps, nous tentons l'approche statistique TF-IDF comme évoquée pour la création du corpus d'évaluation. Cette fois-ci le TF-IDF sera utilisé en vectorisant la question et les fragments propositions du document comportant la réponse et non entre la réponse rédigée et les fragments totaux du dictionnaire. De plus, nous prenons en compte cette fois-ci seulement les unigrammes de la question et des fragments, et non plus les bigrammes.

Nous estimons que le vocabulaire utilisé dans les questions y est tellement précis, que ce modèle, malgré ses limites, peut constituer un bon candidat. Et en effet, la précision obtenue de cette baseline est plutôt élevée compte tenue de nos données, à hauteur de 41,22% de précision.

Amélioration par pré-traitements

Afin d'améliorer les résultats, nous testons deux méthodes préliminaires avant d'appliquer le TF-IDF : une lemmatisation avec la bibliothèque spacy (fr_core_news_sm [1]) et une racinisation avec la bibliothèque nltk [2] (c.f. 2.3.1) des questions et des fragments candidats, en plus de la normalisation en lettres minuscules. Nous observons une amélioration des résultats dans les deux cas, mais sensiblement plus grande à l'aide de la racinisation que de la lemmatisation.

En effet, les racines plus que les lemmes, nous permettent de réduire, par exemple, sous la forme « rémuner », le verbe « rémunérer » dans une question et le nom commun « rémunération » dans un fragment réponse, tandis que la lemmatisation nous donnerait deux formes distinctes, éloignant le fragment réponse de la question. Ce gain permis par la racinisation est possible grâce au manque de bruit qu'elle ramène car la recherche se situe dans le bon document. Plus de risques d'erreur dûes à la racinisation seraient observables par recherche sur l'ensemble du corpus. La racinisation pourra par exemple rapprocher le verbe « falloir » à la 3ème personne du singulier dans une question « Faut-il ... » qui aura comme racine « faut » comme le nom « faute » dans « faute grave » ou « faute lourde ». Ce problème est cependant plus réduit si l'on se situe déjà dans le bon document réponse.

Amélioration par remplacement de concept

Nous tentons d'effectuer un autre pré-traitement en utilisant une base croisée entre un thésaurus et une terminologie interne à l'entreprise, qui a été constitué semi-manuellement et qui permet d'avoir, pour une forme appartenant au thésaurus, ses variations synonymiques, bien que la liste ne soit pas exhaustive. Exemple : le concept « contrat de travail à durée indéterminée » aura comme formes alternatives : « CDI », « contrat en CDI », « contrat à durée indéterminée », mais aussi leurs formes plurielles, « contrats de travail à durée indéterminée », « contrats à durée indéterminée ». Ce couplage avec cette base de données nous semble prometteuse dans la mesure où

1. https ://spacy.io/models/fr
2. https ://www.nltk.org/howto/stem.html

certains cas de synonymies non pris en compte par la méthode TF-IDF pourraient être partiellement traités grâce à un remplacement de termes synonymes sous une seule et même forme. Le concept étant de longueur variable en termes de token et afin de ne pas créer un déséquilibre entre les concepts courts et les concepts longs, chaque concept a été réduit en un seul mot en supprimant tous ses espaces.

Malheureusement, malgré certains résultats qui se sont améliorés dans les cas où le manque de synonymie ou de remplacement par concept faisait défaut, d'autres résultats se sont dégradés, nous donnant une performance globale à peine améliorée par rapport au TF-IDF de base.

Dans le premier exemple ci-dessous, le fragment a été correctement repéré par le TF-IDF associé à la base de connaissance par rapport au TF-IDF simple pour la question suivante : « Un salarié est en **temps partiel thérapeutique**. Est-il concerné par le **maintien de salaire** pour **maladie** ? ». Nous observons que la détection du concept « maintien de salaire », rare dans le document entier, a su faire remonter le bon fragment réponse.

NB : Les concepts trouvés du thésaurus sont mis en gras tandis que ceux en couleur, correspondent à un concept également présent dans la question.

Fragment trouvé par TF-IDF sans base de connaissance

Temps partiel thérapeutique
Cadre du **temps partiel thérapeutique**
En lien avec une **maladie** non professionnelle

En cas de **maladie** non professionnelle, il n'est pas exigé que le **temps partiel thérapeutique** soit immédiatement précédé d'un **arrêt de travail** à **temps complet** indemnisé (c. **séc. soc.** art. L. 323-3, 1°). Pour tous les assurés, le **temps partiel thérapeutique** peut être décidé dans le **cadre** d'un maintien au travail ou d'une reprise du **travail à temps partiel**.

Fragment trouvé par TF-IDF et base de connaissance

Temps partiel thérapeutique
Statut du salarié et rémunération
Rémunération d'activité

L'employeur rémunère le salarié en fonction du travail accompli. En revanche, il n'a pas à le faire bénéficier du **maintien de salaire*** prévu par la loi ou les **conventions collectives** en cas d'**arrêt de travail** pour **maladie** ou **accident du travail**, puisque le **contrat de travail** du salarié n'est plus suspendu (**cass. soc**. 21 mars 2007, n° 06-40891 D). Bien entendu, l'employeur est tenu par les éventuelles dispositions conventionnelles plus favorables au salarié.

Dans ce deuxième exemple, au contraire, nous avons un bon fragment trouvé par le TF-IDF simple pour la question « Nous préparons les **documents de fin de contrat** d'un **CDD** rompu par commun accord avec le salarié. L'attestation **chômage** fait-elle partie des documents à remettre ? ». Nous avons en effet une perte de bon fragment en utilisant la base de connaissance, dû notamment à la non-correspondance entre « documents de fin de contrat » existant dans la base de connaissances et « documents à remettre au salarié en fin de contrat » qui n'a pas été inscrit dans la base comme une variante possible du concept. Un autre fragment contenant un mot de la base de connaissance à savoir « CDD » sera lui, remonté à tort vers les meilleurs résultats réponses.

En d'autres termes, la base de connaissances n'étant pas exhaustive, certains manquements ne permettent pas la récupération des bons fragments. L'apparition, parfois répétée, d'un concept appartenant au thésaurus pèse sur le calcul de similarité car il peut être considéré comme rare. Il y aura donc une tendance à choisir un fragment qui possède les termes du thésaurus par rapport aux autres termes de la question, s'ils n'ont pas été bien capturés.

Fragment trouvé par TF-IDF sans base de connaissance

Démission
Obligations de l'employeur
Documents à remettre au salarié en fin de contrat

L'employeur doit remettre au salarié démissionnaire un **certificat de travail***, un **reçu pour solde de tout compte*** et une attestation d'**assurance chômage*** (**cass. soc.** 15 mars 2017 n° 15-21232, **BC V** n° 50).

Fragment trouvé par TF-IDF et base de connaissance

Démission
Droit de démissionner (en bref)
Pas de démission pour les **CDD**

Un salarié en **contrat à durée déterminée*** (**CDD**) ne peut pas démissionner. Dans certains cas, il peut **rompre son CDD de façon anticipée**.

Pour clôturer cette partie sur notre approche statistique, les résultats obtenus avec les différents pré-traitements sont listés dans le tableau suivant (tableau 4.1)

Méthode	Mesure	Précision (%)
TF-IDF (baseline)	cos	41,22
Racinisation + TF-IDF	cos	**47,32**
Lemmatisation + TF-IDF	cos	45,80
Remplacement synonymes par leur concept + TF-IDF	cos	41,98

TABLE 4.1 – Tableau récapitulatif des résultats du TF-IDF couplé ou non avec différents pré-traitements

4.2.2 Méthode à base de réseaux de neurones profonds SBERT

Dans un second temps, nous tentons une approche « état de l'art », à savoir des modèles existants pré-entraînés basés sur les réseaux de neurones profonds Sentence Transformers [Reimers and Gurevych, 2019]. Bien que ces derniers ne soient pas fine-tunés pour notre type de corpus, nous utilisons ces modèles pré-entrainés pour observer de premiers résultats sur nos données. Comme évoqué dans dans la partie 2.3.2 il existe deux types de modèles SBERT : symétrique et asymétrique. Nous choisissons de tester tous les modèles, quelque soit leur symétrie, avec leur mesure de similarité de préférence (cos ou dot), sans aucun pré-traitement particulier. Attention, une longueur maximale de séquence, dépendant des modèles, est imposée : la limite peut être de 128 ou 512 tokens. En d'autres termes, une troncation automatique sera effectuée si la longueur du fragment est supérieure à la limite définie par le modèle. Deux modèles SBERT, sur la langue française uniquement *sentence-camembert-large* [3] et *french_semantic* [4] n'ont pas pu être testées car ces derniers n'effectuent pas une troncation et retournent directement une erreur dans les cas où la limite est dépassée.

Sur les 27 modèles, les trois modèles ayant le mieux performé sont les suivants :

Modèle	Symétrie	Multilingue	Mesure	Longueur	Précision (%)
distiluse	oui	oui	cos	128	**39,69**
multi-qa	oui	oui	cos	512	35,11
paraphrase-multi	oui	oui	cos	128	34,35

TABLE 4.2 – Tableau des trois meilleurs résultats obtenus avec modèles SBERT sur nos données

Le meilleur résultat obtenu est celui de *distiluse-base-multilingual-cased-v1* (écourté en distiluse) avec calcul de similarité cosinus, qui devance les deux autres modèles *multi-qa-mpnet-base-dot-v1* (multi-qa) et *paraphrase-multilingual-mpnet-base-v2* (paraphrase-multi) avec au minimum 4% de précision supplémentaire. Sans surprise, les modèles non-multilingues, entraînés sur la langue anglaise, obtiennent une moins bonne performance et n'apparaissent donc pas dans les meilleurs résultats du tableau. C'est le cas des modèles asymétriques, tous entraînés sur la langue anglaise, et bien que la configuration soit plus adaptée pour notre tâche. Il est à noter cependant que le meilleur modèle SBERT pour notre cas d'application, *distiluse-base-multilingual-cased-v1*, n'admet pas plus de 128 tokens, ce qui signifie qu'il se s'est basé que sur le début de nos fragments.

4.3 Hybridation

Une approche peut être choisie au détriment d'une autre si elle montre de meilleures performances globales. C'est le cas ici de l'approche TF-IDF avec un pré-traitement de racinisation qui donne de meilleurs résultats totaux que les différents modèles à base de réseaux de neurones disponibles. Cependant, si l'on regarde en détail les résultats, on peut noter que les bonnes ou mauvaises performances de chacun des modèles ne se situent pas sur les mêmes questions. Si l'on superpose la meilleure performance TF-IDF obtenue et la meilleure performance de modèle SBERT *distiluse-base-multilingual-cased-v1*, on observe que seulement 60% des réponses trouvées par

3. https ://huggingface.co/dangvantuan/sentence-camembert-large
4. https ://huggingface.co/Sahajtomar/french_semantic

le modèle SBERT se recoupent avec celles du TF-IDF, le taux est de 50% dans le sens inverse. Ce qui signifie que l'approche statistique a réussi à trouver de bons fragment réponses que le modèle transformer n'a pas su retrouver et vice versa. Nous tentons donc dans cette partie de combiner les forces de chacune des approches. Tout au long de cette partie, nous parlons du modèle *distiluse-base-multilingual-cased-v1* lorsque nous évoquons le modèle SBERT, qui a été retenu de par ses meilleurs résultats.

4.3.1 Approche hybride simultanée par fusion des résultats

Nous tentons comme première approche à l'aide d'une méthode s'appuyant sur le reranking RRF (Reciprocal Range Fusion) [Cormack et al., 2009] qui consiste à effectuer un reclassement des données selon les classements retournés de plusieurs méthodes. Dans notre cas, nous utilisons ce reclassement seulement pour relever le meilleur fragment réponse, en fusionnant les résultats trouvés par les modèles TF-IDF et SBERT. La formule est la suivante :

$$RRFscore(d \in D) = \sum_{r \in R} \frac{1}{k + r(d)} \tag{4.1}$$

Ici, k est une constante (fixé à $k = 60$ dans [Cormack et al., 2009] et dans notre expérience bien qu'elle ait en réalité peu d'incidence sur le calcul final), et r représente une permutation de classement sur l'ensemble des documents. $r(d)$ représente ainsi le classement du document d dans la permutation r. La formule somme l'inverse de la position de classement de chaque document d de l'ensemble des documents D dans les classements R, ajusté par la constante k. Le score RRF pour un document est calculé en additionnant ces valeurs inverses sur toutes les permutations de l'ensemble R.

Cette méthode à l'avantage de ne pas prendre en compte l'échelle de similarité qui a déterminé le ranking mais le ranking en lui-même. Les valeurs de similarité cosinus par exemple se situent dans l'intervalle [-1 ;1] tandis qu'un produit scalaire sera un nombre positif, sans limite théorique. La différence d'échelle n'aura pas d'incidence avec cette méthode de reranking. Cependant, le résultat du calcul peu interprétable est n'est qu'un produit dérivé.

En fusionnant les résultats du meilleur résultat obtenu avec le TF-IDF et le meilleur modèle SBERT grâce à ce calcul, nous dépassons la barre des 50% de bons fragments retournés, en atteignant précisément un score de 50,38%.

4.3.2 Approche hybride en cascade

Nous expérimentons une dernière approche par combinaison des méthodes en cascade. Tout d'abord, nous tentons de voir, pour la méthode TF-IDF comme la meilleure méthode SBERT, s'il n'existe pas un seuil à partir duquel il y aurait un optimum de vrais positifs. De cette manière, nous pourrions récupérer des vrais positifs considérés comme sûrs selon les seuils TF-IDF et SBERT définis.

A priori, il semble qu'aucun seuil intéressant ne peut être exploité, pour le modèle TF-IDF comme le modèle SBERT (figures 4.1 et 4.2). Les résultats de calcul de similarité ne peuvent en effet donner à coup sûr les bons fragments réponse. Même lorsque l'on se situe dans les scores les plus élevés, des fragments incorrects se mêlent aux vrais positifs.

Le score de similarité du premier fragment retourné ne pouvant servir de critère absolu, nous observons de plus près nos données afin de voir s'il n'existe pas un autre paramètre permettant d'utiliser le TF-IDF en premier lieu, et de passer à un modèle

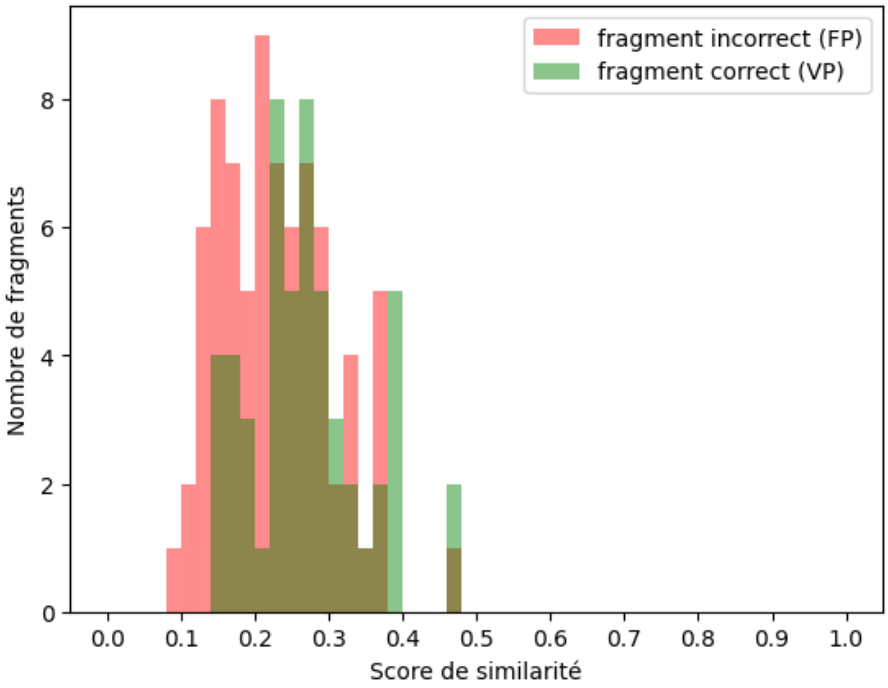


FIGURE 4.1 – Répartition des vrais positifs et faux positifs selon leur score de similarité pour le TF-IDF

transformers si le résultat trouvé par ce dernier n'est pas assez certain. La méthode TF-IDF semble en effet particulièrement fiable lorsque la différence entre le score de similarité du premier fragment retourné est bien supérieur au deuxième retrouvé. A l'inverse, dès que nous avons une différence faible selon un seuil, fixé arbitrairement à 0.03 ou moins, entre le premier segment trouvé et le deuxième fragment trouvé, la méthode SBERT, dans 80% des cas trouve une réponse correcte par rapport au TF-IDF. Si l'on combine la meilleure méthode trouvée, à savoir la méthode « TF-IDF + racinisation » avec le modèle SBERT, nous passons de 47,32% de précision à 53,43%.

4.4 Limites

Plusieurs limites à ces expérimentations peuvent être relevées. La première se situe au niveau du faible volume de couples questions/réponses. Un gain d'une dizaine de bonnes réponses fera grimper la précision de manière significative, mais il n'est pas certain que ce gain puisse s'observer et se généraliser sur un ensemble de données plus conséquent.

De plus, notre travail se limite sur la recherche du bon fragment dans le document contenant le fragment réponse. Les résultats obtenus sont donc à relativiser et à prendre en considération avec un module en amont qui pourrait relever le bon document réponse en premier lieu avant d'aller chercher le bon fragment dans ce document.

Pour finir, la meilleure combinaison trouvée dans ce travail est peu généralisable à d'autres domaines ou jeu de données, bien qu'elle donne des pistes d'exploration et de réflexion.

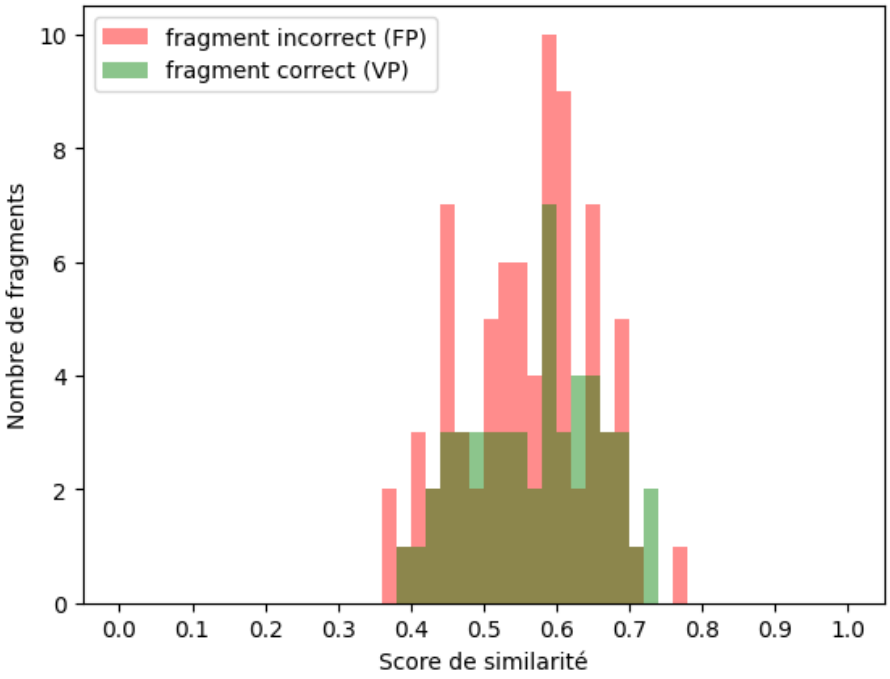


FIGURE 4.2 – Répartition des vrais positifs et faux positifs selon leur score de similarité pour le modèle SBERT distiluse-base-multilingual-cased-v1

4.5 Amélioration et perspectives

4.5.1 Données

Par manque de temps, un corpus de quiz moins bien structuré contenant des couples de phrases déclaratives et réponses binaire vrai/faux avec commentaires rédigés, n'a pas été exploité dans le cadre de cette étude. Ces déclaratives auraient pu être exploitées après transformation en question type « Est-il vrai que ... ? ». Ces dernières auraient permis une comparaison intéressante, notamment lors de d'affirmations fausses ou avec des termes inexactes, qui peuvent orienter la sémantique dans une mauvaise direction et ayant pour conséquence d'éloigner la réponse attendue.

4.5.2 Pré-traitements

Dans les pré-traitements réalisés, nous avons d'emblée remplacé les termes se rapportant à un concept sous une forme unique. Cependant, d'autres choix auraient pu être considérés, comme l'ajout de concept lors de synonymie qui aurait été peut-être plus judicieux qu'un remplacement.

Pour le modèle SBERT, les pré-traitements de lemmatisation ou de racinisation n'auraient pas été utiles à la capture du sens des phrases. Cependant, afin d'aider le modèle à mieux cerner notre domaine, le remplacement de concept grâce à notre base de thésaurus-terminologie aurait pu améliorer les résultats, mais n'a malheureusement pas été testé par faute de temps.

Par ailleurs, peu d'indices linguistiques ont été exploités pour la compréhension de la question dans le cadre de ce travail. Un nettoyage en amont de la question de base par exemple, en éliminant des formulations non porteur de sens, est une partie

qui n'a pas été implémentée et qui aurait pu être un autre pré-traitement à combiner avec nos approches.

4.5.3 Modèles

Pour effectuer une adaptation de domaine avec l'outil SBERT, il existe deux solutions d'apprentissage. La première, supervisée, consiste à fournir un corpus de couples de phrases similaires du domaine, notées sur une échelle de similitude par des experts, selon par exemple un guide d'annotation. Cependant, le coût de la labellisation est un problème majeur dans la pratique, en particulier pour les domaines spécialisés [Wang et al., 2023]. Il existe en revanche une deuxième méthode d'apprentissage, non supervisée [5] permettant de répondre à ce problème. Les méthodes non supervisées s'appuient en effet sur de grandes quantités de données non annotées pour apprendre des représentations de phrases sans orientation explicite. Si cette approche permet en effet de se baser sur un corpus non labellisé à la main, elle requiert cependant des processeurs puissants que nous ne possédions pas lors de la constitution de ce travail.

En approches complémentaires, les méthodes de plongements de mots Word2Vec, Glove ou Fastext évoquées dans la partie 2.3.2 n'ont pas été exploités dans le cadre de ce travail, mais auraient pu représenter des candidats intermédiaires intéressants à comparer ou à combiner avec nos approches. Il en est de même pour l'algorithme de pondération BM25 [Robertson et al., 2009], qui aurait constitué une autre variante du sac « de mots » à comparer avec le TF-IDF.

4.6 Conclusion

Nous pouvons observer que le modèle simple TF-IDF couplé avec la racinisation surpasse des modèles d'embeddings plus sophistiqués mais non fine-tunés sur nos données (environ 8% de différence entre les deux méthodes). Ce gain de performance est possible grâce aux questions posées précises ainsi qu'une recherche intra-document qui limite les erreurs potentielles dûes à la racinisation (c.f. 2.3.1). Nous avons également tenté de réduire les limites du TF-IDF en faisant un remplacement des synonymes d'un concept sous une même forme. Si cette méthode a permis de gagner dans certains cas des bons fragments réponses, elle s'est avérée contre-productive sur les résultats globaux.

Dans une seconde partie, nous avons testé deux types d'hybridation qui ont été concluants et ont apporté une amélioration par rapport aux approches individuelles. La première, consistait à fusionner les résultats obtenus, en ne favorisant aucune des deux approches. Cette méthode a su montrer un premier gain de performance de 3% par rapport à la meilleure méthode individuelle. La deuxième approche, privilégiant la méthode TF-IDF par rapport à la méthode SBERT, permet d'obtenir un gain de précision considérable, doublé par rapport à la première approche d'hybridation (6%).

Enfin, nos deux dernières parties traitent des limites de notre travail, notamment le caractère peu généralisable et le manque de volume de nos données. Nous avons également évoqué les améliorations possibles à différents niveaux, en particulier la méthode de labellisation non supervisée pour fine-tuner un modèle SBERT afin d'effectuer une adaptation de domaine.

5. https ://www.sbert.net/examples/domain_adaptation/README.html

Conclusion générale

Dans ce mémoire, nous avons traité de l'approche de similarité textuelle pour un système de questions-réponses dans le domaine spécialisé de la Paye. Dans ce cas applicatif, en tant que non-expert et sans données de similarité labellisées, nous avons utilisé plusieurs outils génériques disponibles afin de répondre à notre problème. Notre système de questions-réponses retournant un passage du document, la première phase a consisté à délimiter des fragments réponses pertinents et autonomes de part leur sens, pour en faire un corpus de travail. Cette fragmentation, combinée à une ressource interne de couples de question/réponse, a également permis à la création d'un deuxième jeu de données, à savoir celui du corpus d'évaluation.

Lors de nos expérimentations, nous avons montré l'efficacité de l'approche statistique TF-IDF par rapport à des modèles plus sophistiqués mais non fine-tunés à notre corpus comme les réseaux de neurones profonds SBERT. Ce résultat est possible de part les questions précises posées par la cible, à savoir des experts du domaine. Après observation cependant, nous nous apercevons que le modèle SBERT, bien que moins performant que le TF-IDF dans les résultats globaux, parvient à retrouver de bons fragments non capturés par le TF-IDF. En testant deux méthodes d'hybridation afin de combiner les forces des deux modèles, la plus performante s'est avérée être celle en cascade, qui s'appuie sur le TF-IDF en premier lieu, avant de basculer selon un paramètre défini vers le modèle SBERT lorsque la méthode TF-IDF n'est pas assez fiable. Nous avons ainsi pu augmenter les scores de précision de notre baseline, c'est-à-dire un passage de 41,22% (TF-IDF) et 39,69% (SBERT) à une précision atteignant 53,43%.

Les derniers articles concernant SBERT sont encore très récents et l'étude de la tâche de similarité, en pleine croissance. Les modèles spécifiques pour la langue française sont encore assez limités, mais ce n'est qu'une question de temps avant que l'on puisse tester de nouveaux modèles performants, en plus des perspectives que nous avons évoqué dans ce mémoire.

BIBLIOGRAPHIE

[Aliguliyev, 2009] Aliguliyev, R. M. (2009). A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications*, 36(4):7764–7772. – Cité page 19.

[Ayache et al., 2006] Ayache, C., Grau, B., and Vilnat, A. (2006). Equer: the french evaluation campaign of question-answering systems. In *LREC*, pages 1157–1160. – Cité page 16.

[Bojanowski et al., 2017] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146. – Cité page 20.

[Cormack et al., 2009] Cormack, G. V., Clarke, C. L., and Buettcher, S. (2009). Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759. – Cité page 42.

[Devlin et al., 2018] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. – Cité page 21.

[Eguchi et al., 2002] Eguchi, K., Oyama, K., Ishida, E., Kando, N., and Kuriyama, K. (2002). Overview of the web retrieval task at the third ntcir workshop. In *NTCIR*. Citeseer. – Cité page 16.

[Firth, 2020] Firth, J. R. (2020). Papers in linguistics, 1934-1951. – Cité page 20.

[Green Jr et al., 1961] Green Jr, B. F., Wolf, A. K., Chomsky, C., and Laughery, K. (1961). Baseball: an automatic question-answerer. In *Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference*, pages 219–224. – Cité page 16.

[Hariharan, 2012] Hariharan, S. (2012). Automatic plagiarism detection using similarity analysis. *Int. Arab J. Inf. Technol.*, 9(4):322–326. – Cité page 19.

[Hiemstra, 2000] Hiemstra, D. (2000). A probabilistic justification for using tf$\times$ idf term weighting in information retrieval. *International Journal on Digital Libraries*, 3:131–139. – Cité pages 9, 16 et 37.

[Hsu et al., 2021] Hsu, C.-C., Lind, E., Soldaini, L., and Moschitti, A. (2021). Answer generation for retrieval-based question answering systems. *arXiv preprint arXiv:2106.00955*. – Cité page 23.

[Jaccard, 1901] Jaccard, P. (1901). Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines. *Bull Soc Vaudoise Sci Nat*, 37:241–272. – Cité page 31.

[Katz, 1997] Katz, B. (1997). From sentence processing to information access on the world wide web. In *AAAI Spring Symposium on Natural Language Processing for the World Wide Web*, volume 1, page 997. Stanford University Stanford, CA, USA. – Cité page 16.

[Le et al., 2019] Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., Allauzen, A., Crabbé, B., Besacier, L., and Schwab, D. (2019). Flaubert: Unsupervised language model pre-training for french. *arXiv preprint arXiv:1912.05372*. – Cité page 21.

[Levenshtein et al., 1966] Levenshtein, V. I. et al. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union. – Cité page 31.

[Lin et al., 2003] Lin, J., Quan, D., Sinha, V., Bakshi, K., Huynh, D., Katz, B., and Karger, D. R. (2003). What makes a good answer? the role of context in question answering. In *INTERACT*. – Cité page 23.

[Magnini et al., 2004] Magnini, B., Romagnoli, S., Vallin, A., Herrera, J., Penas, A., Peinado, V., Verdejo, F., and de Rijke, M. (2004). The multiple language question answering track at clef 2003. In *Comparative Evaluation of Multilingual Information Access Systems: 4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003, Trondheim, Norway, August 21-22, 2003, Revised Selected Papers 4*, pages 471–486. Springer. – Cité page 16.

[Martin et al., 2019] Martin, L., Muller, B., Suárez, P. J. O., Dupont, Y., Romary, L., de La Clergerie, É. V., Seddah, D., and Sagot, B. (2019). Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*. – Cité page 21.

[Martinez-Gil, 2023] Martinez-Gil, J. (2023). A survey on legal question–answering systems. *Computer Science Review*, 48:100552. – Cité pages 16 et 17.

[Mikolov et al., 2013] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26. – Cité page 20.

[Mitkov, 2022] Mitkov, R. (2022). *The Oxford handbook of computational linguistics*. Oxford University Press. – Cité page 17.

[Othman and Faiz, 2016] Othman, N. and Faiz, R. (2016). Question answering passage retrieval and re-ranking using n-grams and svm. *Computación y Sistemas*, 20(3):483–494. – Cité page 23.

[Pennington et al., 2014] Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543. – Cité page 20.

[Perret, 2005] Perret, L. (2005). *Extraction automatique d'information: génération de résumé et question-réponse*. PhD thesis, Université de Neuchâtel. – Cité page 35.

[Pinto et al., 2003] Pinto, D., McCallum, A., Wei, X., and Croft, W. B. (2003). Table extraction using conditional random fields. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 235–242. – Cité page 16.

[Rajaraman and Ullman, 2011] Rajaraman, A. and Ullman, J. D. (2011). *Mining of massive datasets*. Cambridge University Press. – Cité page 18.

[Reimers and Gurevych, 2019] Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*. – Cité pages 9, 21, 37 et 41.

[Robertson et al., 2009] Robertson, S., Zaragoza, H., et al. (2009). The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389. – Cité page 45.

[Salton, 1975] Salton, G. (1975). *A theory of indexing*, volume 18. SIAM. – Cité page 19.

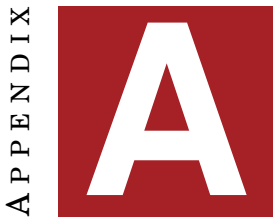
[Soboroff, 2021] Soboroff, I. (2021). Overview of trec 2021. In *30th Text REtrieval Conference. Gaithersburg, Maryland*. – Cité page 16.

[Tenney et al., 2019] Tenney, I., Das, D., and Pavlick, E. (2019). Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*. – Cité page 16.

[Voorhees and Tice, 2000] Voorhees, E. M. and Tice, D. M. (2000). Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207. – Cité pages 16 et 17.

[Wang et al., 2023] Wang, L., Chou, J., Rouck, D., Tien, A., and Baumgartner, D. M. (2023). Adapting sentence transformers for the aviation domain. *arXiv preprint arXiv:2305.09556*. – Cité page 45.

[Woods, 1973] Woods, W. A. (1973). Progress in natural language understanding: an application to lunar geology. In *Proceedings of the June 4-8, 1973, national computer conference and exposition*, pages 441–450. – Cité page 16.



ANNEXES

A.1 Exemple de données XML

```
<DOC_INTRO>
  <RUBRIQUE/>
  <SUR_TITRE/>
  <TITRE>Sinistre</TITRE>
  <CHAPO/>
</DOC_INTRO>
<DOC_DEV>
  <SECTION Id="1" Niv="1">
    <TITRE>Force majeure</TITRE>
    <TEXTE>
      <B>Contrat à durée indéterminée -</B> Le salarié dont le contrat de travail à durée
      indéterminée est rompu en raison d'un sinistre présentant les caractéristiques de la
      <REF_INTERNES><RENVOI Cache="false" Id="20113" Ref_id_doc="20120403142210352" Type="interne">
      force majeure</RENVOI></REF_INTERNES>* a droit à une indemnité compensatrice dont le montant est
      égal à l'<REF_INTERNES><RENVOI Cache="false" Id="20114" Ref_id_doc="20120403144736620" Type="interne">
      indemnité compensatrice de préavis</RENVOI></REF_INTERNES>* et à l'indemnité légale de licenciement
      (c. trav. <LIEN_EXT_BATCH><LIBELLE>art. L. 1234-13</LIBELLE><SENS>Standard</SENS><STATUT>Consultation
      </STATUT><SOURCE><FOURNISSEUR>SPAD</FOURNISSEUR><BASE>LEGI</BASE><CODE>LEGITEXT000006072050</CODE><DETAIL>
      <ARTICLE>L1234-13</ARTICLE><IDSPAD>LEGIARTI000006901129</IDSPAD></DETAIL></SOURCE></LIEN_EXT_BATCH>,
      renvoyant à <LIEN_EXT_BATCH><LIBELLE>L. 1234-5</LIBELLE><SENS>Standard</SENS><STATUT>Consultation</STATUT>
      <SOURCE><FOURNISSEUR>SPAD</FOURNISSEUR><BASE>LEGI</BASE><CODE>LEGITEXT000006072050</CODE><DETAIL><ARTICLE>
      L1234-5</ARTICLE><IDSPAD>LEGIARTI000006901118</IDSPAD></DETAIL></SOURCE></LIEN_EXT_BATCH> et <LIEN_EXT_BATCH>
      <LIBELLE>L. 1234-9</LIBELLE><SENS>Standard</SENS><STATUT>Consultation</STATUT><SOURCE><FOURNISSEUR>SPAD</FOURNISSEUR>
      <BASE>LEGI</BASE><CODE>LEGITEXT000006072050</CODE><DETAIL><ARTICLE>L1234-9</ARTICLE><IDSPAD>LEGIARTI000035644154</IDSPAD>
      </DETAIL></SOURCE></LIEN_EXT_BATCH>).</TEXTE>
    <TEXTE>
      <B>Contrat à durée déterminée - </B>Le salarié dont le contrat de travail à durée déterminée est rompu avant
      l'échéance en raison d'un sinistre relevant d'un cas de <REF_INTERNES>
      <RENVOI Cache="false" Id="20115" Ref_id_doc="20120403142210352" Type="interne">force majeure</RENVOI></REF_INTERNES>*
      a droit à une indemnité compensatrice dont le montant est égal aux rémunérations que le salarié aurait perçues
      jusqu'au terme du contrat (c. trav. <LIEN_EXT_BATCH><LIBELLE>art. L. 1243-1</LIBELLE><SENS>Standard</SENS><STATUT>
      Consultation</STATUT><SOURCE><FOURNISSEUR>SPAD</FOURNISSEUR><BASE>LEGI</BASE><CODE>LEGITEXT000006072050</CODE><DETAIL>
      <ARTICLE>L1243-1</ARTICLE><IDSPAD>LEGIARTI000029946319</IDSPAD></DETAIL></SOURCE></LIEN_EXT_BATCH> et <LIEN_EXT_BATCH>
      <LIBELLE>L. 1243-4</LIBELLE><SENS>Standard</SENS><STATUT>Consultation</STATUT><SOURCE><FOURNISSEUR>SPAD</FOURNISSEUR>
      <BASE>LEGI</BASE><CODE>LEGITEXT000006072050</CODE><DETAIL><ARTICLE>L1243-4</ARTICLE><IDSPAD>LEGIARTI000024026880</IDSPAD>
      </DETAIL></SOURCE></LIEN_EXT_BATCH>).</TEXTE>
    <TEXTE>Dans ce cadre, l'<REF_INTERNES><RENVOI Cache="false" Id="20116" Ref_id_doc="20120403144800718" Type="interne">
    indemnité de fin de contrat</RENVOI></REF_INTERNES>* n'est pas due (c. trav. <LIEN_EXT_BATCH><LIBELLE>art. L. 1243-10
    </LIBELLE><SENS>Standard</SENS><STATUT>Consultation</STATUT><SOURCE><FOURNISSEUR>SPAD</FOURNISSEUR><BASE>LEGI</BASE><CODE>
    LEGITEXT000006072050</CODE><DETAIL><ARTICLE>L1243-10</ARTICLE><IDSPAD>LEGIARTI000006901221</IDSPAD></DETAIL></SOURCE>
    </LIEN_EXT_BATCH>, 4° ; circ. DRT <LIEN_EXT_DBIT><LIBELLE>2002-8</LIBELLE><ID>81</ID><TYPE>CIRC</TYPE><SOUS-TYPE>DRT
    </SOUS-TYPE><DATE-DOC>2002-05-02</DATE-DOC></LIEN_EXT_DBIT> du 2 mai 2002, § 3.4).</TEXTE>
  </SECTION>
  <SECTION Id="2" Niv="1">
    <TITRE>Récupération des heures perdues</TITRE>
    <TEXTE>Sous réserve d'en informer préalablement l'inspecteur du travail, l'employeur a la possibilité de faire effectuer,
    en plus de l'horaire normal, les heures de travail correspondant à celles qui ont été perdues collectivement, en deçà
    de la durée légale, à la suite d'un sinistre. Ces heures ne sont pas rémunérées au tarif des <REF_INTERNES>
    <RENVOI Cache="false" Id="20117" Ref_id_doc="20120403142536426" Type="interne">heures supplémentaires</RENVOI></REF_INTERNES>
    * (c. trav. <LIEN_EXT_BATCH><LIBELLE>art. L. 3121-50</LIBELLE><SENS>Standard</SENS><STATUT>Consultation</STATUT><SOURCE>
    <FOURNISSEUR>SPAD</FOURNISSEUR><BASE>LEGI</BASE><CODE>LEGITEXT000006072050</CODE><DETAIL><ARTICLE>L3121-50</ARTICLE>
    <IDSPAD>LEGIARTI000033020250</IDSPAD></DETAIL></SOURCE></LIEN_EXT_BATCH>; voir <REF_INTERNES>
    <RENVOI Cache="false" Id="20118" Ref_id_doc="20120403150758940" Type="interne">Récupération des heures perdues</RENVOI>
    </REF_INTERNES>*).</TEXTE>
  </SECTION>
</DOC_DEV>
```

Figure A.1 – Extrait XML montrant la structure d'un document du dictionnaire