Génération automatique de texte en langage naturel pour les systèmes de questions-réponses

Imen Akermi* — **Johannes Heinecke*** — **Frédéric Herledan***

* *Orange Innovation, 2 av. Pierre Marzin, 22307 Lannion, France*
imenakermi@yahoo.fr,
{johannes.heinecke,frederic.herledan}@orange.com

RÉSUMÉ. *Cet article traite de la génération du langage naturel dans le contexte des systèmes de questions-réponses. Les différents travaux portant sur ces systèmes se sont focalisés sur la génération d'une réponse courte ou d'un paragraphe contenant la réponse, à partir de données structurées ou de pages Web. La longueur de ces réponses n'est généralement pas appropriée du fait que les réponses peuvent être perçues comme trop brèves ou trop longues pour être lues à haute voix par un assistant intelligent. Dans ce travail, nous présentons une approche non supervisée de génération de réponses concises qui ne nécessite pas de données annotées. Testée sur des corpus de données en anglais et en français, l'approche proposée montre des résultats très prometteurs.*

MOTS-CLÉS : *systèmes de questions-réponses, génération du langage naturel, analyse en dépendances.*

TITLE. *Compact answer generation with a Transformer based approach.*

ABSTRACT. *This paper presents an unsupervised approach for natural language generation within the framework of question-answering systems. This approach addresses the issue of generating answers that are usually too short or too long without having to resort to annotated data. This approach shows promising results for English and for French.*

KEYWORDS: *question answering systems, natural language generation, dependency analysis.*

1. Introduction

Les systèmes de questions-réponses (SQR) analysent et traitent les questions des utilisateurs afin de leur fournir des réponses pertinentes (Hirschman et Gaizauskas, 2001). L'intérêt pour les SQR s'est accru avec la popularité récente des assistants intelligents. Ces derniers permettent aux utilisateurs de poser des questions en langage naturel, en utilisant leur propre terminologie, et d'avoir directement des réponses sans avoir à parcourir une longue liste de documents pour trouver les réponses appropriées. Les SQR sont ainsi devenus un élément central des échanges « humain-machine ».

La plupart des travaux de recherche existants se focalisent sur le traitement et l'interprétation de la question. Ils accordent souvent peu d'importance à la représentation de la réponse donnée en sortie. Généralement, la réponse est soit représentée par un ensemble de termes courts répondant exactement à la question, soit par un passage dans un document qui contient la réponse exacte mais qui peut aussi intégrer d'autres informations inutiles ne relevant pas du contexte de la question posée.

Prenons l'exemple de la question *Qui vivait au Costa Rica avant les Espagnols ?*, avec les SQR actuels, deux structures de réponses sont généralement renvoyées :

Amérindiens

À l'époque précolombienne, les Amérindiens de l'actuel Costa Rica faisaient partie d'un complexe culturel connu sous le nom de « zone intermédiaire », entre les régions culturelles mésoaméricaine et andine.

La première réponse pourra être perçue par les utilisateurs comme trop brève et ne rappelant pas le contexte de la question. La deuxième pourra être perçue comme trop longue et nécessitera à l'utilisateur une lecture attentive pour deviner la réponse à sa question au milieu d'informations non pertinentes.

Pour répondre aux attentes des utilisateurs, il conviendrait de produire une réponse synthétique qui soit concise, c'est-à-dire ne contenant pas d'autres informations que la stricte réponse à la question et qui soit aussi complète, c'est-à-dire rappelant le contexte de la question posée. Pour cette tâche, une approche simple pourrait consister à utiliser des règles prédéfinies pour générer les structures possibles de réponses (Reiter et Dale, 1997). Cependant, de telles approches ne sont pas généralisables et échouent à capturer les spécificités de la réponse. D'autre part, les approches par apprentissage supervisé qui se basent sur des architectures neuronales nécessitent de larges corpus de données qui feront correspondre une question à une réponse complète et concise. Elles sont très performantes pour générer des réponses pour des données du domaine sur lequel elles ont été entraînées mais elles ont souvent des limites pour généraliser sur d'autres domaines.

L'approche de génération de réponses que nous proposons est non supervisée et ne nécessite donc pas de corpus d'apprentissage. Elle peut facilement être adaptée à n'importe quelle langue. La performance de cette approche est attestée par des expérimentations et une évaluation humaine sur des données de test en anglais et en

français. Les données ont été acquises par écrit. Nous avons pu vérifier que les derniers systèmes de transcription de la parole sont capables de sortir du texte prenant en compte certaines spécificités de l'écrit comme les majuscules pour les entités. Nous avons donc estimé que ces données étaient aussi pertinentes pour l'usage oral que nous envisagions. En effet, bien que l'expression orale diffère de celle de l'écrit, le prototype (Rojas Barahona *et al.*, 2019) nous a montré que cette différence est moindre pour un usage de questions-réponses. Ces données portaient exclusivement sur la connaissance générale. Hormis pour des constructions syntaxiques très spécifiques à certains métiers, nous ne voyons pas d'obstacle à appliquer notre méthode aux données de domaines de spécialité. Elle ouvre également des pistes très prometteuses, d'une part pour générer des réponses synthétiques dans un contexte de dialogue et, d'autre part, pour créer automatiquement un corpus d'entraînement ou de test afin de creuser ultérieurement des approches supervisées. Les principales contributions de cet article peuvent être résumées comme suit :

– une approche de génération de réponses synthétiques en langage naturel ;

– une approche de construction automatique de corpus de questions-réponses pour creuser ultérieurement la capacité des systèmes supervisés à générer des réponses synthétiques

L'article est organisé comme suit. Nous présentons dans la section 2 une revue de la littérature sur les approches de génération automatique de texte de l'analyse en dépendances dans le contexte des systèmes de questions-réponses. La section 3 détaille l'approche de génération proposée. Nous décrivons dans la section 4 les expériences menées et nous introduisons dans la section 5 une approche de construction automatique de corpus de questions-réponses. Nous concluons dans la section 6 avec un résumé des approches proposées dans cet article ainsi que des réflexions pour les travaux futurs.

2. État de l'art

2.1. *Génération automatique de texte*

La génération automatique de texte (GAT) est considérée comme un sous-domaine de l'intelligence artificielle et de la linguistique computationnelle. Elle s'intéresse à la construction de systèmes capables de produire des textes en langage naturel qui soient compréhensibles, et ceci à partir d'informations extraites de textes, de données structurées ou de données visuelles telles que les images ou les vidéos (Reiter et Dale, 1997). Elle trouve une application particulière dans les SQR.

De nos jours, la grande quantité d'informations disponibles rend la recherche d'information complexe et chronophage. En renvoyant directement la réponse exacte à une question posée en langage naturel, les SQR évitent à l'utilisateur de devoir filtrer lui-même les informations renvoyées. Les SQR couvrent principalement trois tâches : l'analyse de la question, la recherche d'information et l'extraction de la ré-

ponse (Lopez *et al.*, 2011). Dans les travaux existants, ces tâches sont abordées de différentes manières, en fonction des bases de connaissances utilisées, des types de questions traitées (Iida *et al.*, 2019 ; Zayaraz *et al.*, 2015) et de la façon avec laquelle la réponse est présentée. Dans la littérature, nous distinguons généralement deux formes de représentation. La réponse peut prendre la forme d'un paragraphe sélectionné à partir d'un ensemble de passages textuels extraits du Web ou à partir de bases de connaissances (Asai *et al.*, 2018 ; Du et Cardie, 2018), comme elle peut également être uniquement une réponse courte, par exemple un groupe nominal (Wu *et al.*, 2003 ; Bhaskar *et al.*, 2013 ; Le *et al.*, 2016). Dans les systèmes qui extraient les réponses à partir de bases de connaissances, la réponse prend généralement une forme très concise se limitant à une information brièvement représentée et qui, certes, permet de répondre à la question, mais qui manque considérablement de contexte. Ces formes de réponses, trop brèves ou trop longues pourraient considérablement entraver le dialogue « homme machine » en le rendant moins naturel.

Malgré l'abondance des travaux dans le domaine des SQR, la problématique de formulation des réponses a reçu très peu d'attention. Une première approche traitant indirectement cette tâche a été proposée dans Brill *et al.* (2001) et Brill *et al.* (2002). En effet, les auteurs avaient pour but de diversifier les motifs possibles de réponses en permutant les termes de la question en vue de maximiser le nombre de documents extraits susceptibles de contenir la réponse. Une autre approche de représentation de réponse basée sur des règles de reformulation a été également proposée dans Agichtein et Gravano (2000) et Lawrence et Giles (1998) dans le contexte de l'expansion de requêtes pour la recherche de documents et non pour l'extraction de la réponse exacte.

Le peu de travaux qui se sont intéressés à cette tâche dans le cadre des SQR l'ont adressée sous l'angle de la génération de résumés de textes (Ishida *et al.*, 2018 ; Iida *et al.*, 2019 ; Rush *et al.*, 2015 ; Chopra *et al.*, 2016 ; Nallapati *et al.*, 2016 ; Miao et Blunsom, 2016 ; See *et al.*, 2017 ; Oh *et al.*, 2016 ; Sharp *et al.*, 2016 ; Tan *et al.*, 2016 ; dos Santos *et al.*, 2016). La majorité de ces travaux n'ont considéré que les questions de causalité de type « *pourquoi* » où les réponses sont des paragraphes. Pour rendre ces réponses plus concises, ils procèdent à un compactage des paragraphes extraits.

D'autres approches (Kruengkrai *et al.*, 2017 ; Girju, 2003 ; Verberne *et al.*, 2011 ; Oh *et al.*, 2013) ont exploré cette tâche comme un problème de classification où il s'agit de prédire si un passage de texte pourrait constituer une réponse à une question donnée.

Il faut noter que ces approches ont pour seul but de diversifier au maximum les formules possibles pour augmenter la probabilité d'extraire la bonne réponse et non pour générer une réponse qui soit conviviale pour l'utilisateur. Il faut également souligner que ces approches ne sont valables que pour les SQR qui génèrent les réponses sous forme d'extraits de textes et ne pourront pas être appliquées aux réponses courtes.

Les travaux présentés dans (Pal *et al.*, 2019) ont tenté d'aborder ce problème en proposant une approche supervisée dont l'apprentissage s'est fait sur un petit ensemble de données dont les paires questions-réponses ont été extraites à partir de corpus de

données axés sur la tâche de compréhension de texte et ont été également ajoutées manuellement, ce qui rend la généralisation et la capture de la variation très limitées.

Notre approche de génération de réponses concises diffère de ces travaux car elle est non supervisée, elle peut s'adapter à n'importe quel type de questions factuelles (à l'exception de celles de type *pourquoi*) et elle s'appuie sur des données facilement accessibles et non annotées. Pour cela, nous nous basons sur l'analyse en dépendances afin d'avoir une idée du rôle de la réponse dans la question posée, par exemple, pour savoir si la réponse est le sujet ou l'objet de la question.

2.2. *Analyse en dépendances*

L'analyse en dépendances est très utilisée pour obtenir la structure d'une phrase et bien d'autres tâches de TALN s'appuient sur cette analyse. C'est notamment le cas depuis l'arrivée des outils d'analyse à base de transition en apprentissage supervisé (Nivre, 2003), faciles à utiliser, comme par exemple Maltparser, Nivre *et al.* (2006)) et plus récemment des outils d'analyse à base de graphes (Kiperwasser et Goldberg, 2016 ; Dozat *et al.*, 2017). Des approches et des outils très performants et d'une haute qualité ont été présentés lors de deux campagnes d'évaluation de l'analyse en dépendances CoNLL 2017 (Zeman *et al.*, 2017) et CoNLL 2018 (Zeman *et al.*, 2018). En 2017, des équipes participantes utilisaient des parsers à base de transition ou à base de graphes (Kübler *et al.*, 2009), mais suite aux bons résultats du gagnant de CoNLL 2017 (Dozat *et al.*, 2017) (un parser à base de graphes) les outils les plus performants sont maintenant presque tous à base de graphes, comme, par exemple, les gagnants en termes de MLAS de CoNLL 2018 (Straka (2018) ; Kondratyuk et Straka (2019)).

Pour l'instant la plupart des analyseurs à base d'apprentissage supervisé utilisent les treebanks fournis par le projet Universal Dependencies (UD)[1] (Nivre *et al.*, 2016). Ce projet fournit 183 treebanks en 104 langues[2]. Certains treebanks sont néanmoins très petits, mais comme les approches de la campagne CoNLL 2018 ont pu le montrer, la plupart d'entre eux permettent d'apprendre des analyseurs pour obtenir des résultats de bonne qualité. Malgré le fait qu'une partie des treebanks a été créée antérieurement au projet UD (p. ex. Abeillé *et al.* (2003) pour le français et Marneffe et Manning (2008) pour l'anglais). Des apprentissages crosslingues sont possibles, car tous les treebanks ont été annotés en suivant le même guide d'annotation et utilisent les mêmes catégories pour désigner les parties de discours ainsi que les relations en dépendances et les traits syntactico-morphologiques.

Les treebanks du projet UD sont actuellement en train d'être enrichis par des *enhanced dependencies*, qui en plus des relations de base indiquent des relations indirectes entre mots comme le sujet d'un verbe coordonné, etc. (Nivre *et al.*, 2018 ; Oepen

1. `http://universaldependencies.org`
2. Version 2.7 du 15 novembre 2020, `http://hdl.handle.net/11234/1-3424`

et al., 2020). En revanche, pour l'instant les treebanks pour l'anglais ou le français ne sont pas encore exhaustivement annotés.

Pour évaluer l'analyse en dépendances, quatre métriques similaires sont actuellement utilisées :

– *Unlabeled Attachement Score* (UAS), qui exprime le pourcentage des mots étant attachés à la bonne tête sans prendre en compte le type de relation de dépendance ;

– *Labeled Attachement Score* (LAS), qui exprime le pourcentage des mots étant attachés à la bonne tête et ayant le bon type de relation de dépendance. Le LAS est la valeur f-mesure (Zeman *et al.*, 2018) :

$$P = \frac{\textit{\#nœudsCorrects}}{\textit{\#nœudsPrédits}} \qquad R = \frac{\textit{\#nœudsCorrects}}{\textit{\#nœudsGold}}$$

$$LAS = F_1 = \frac{2PR}{P+R}$$

Si le LAS n'est pas pondéré, P et R sont toujours identiques, donc $LAS = F_1 = P = R$. Pour le LAS pondéré *#nœudCorrects*, *#nœudsPrédits* et *#nœudGold* ne sont pas simplement les sommes des mots correctement annotés, mais pour les mots fonctionnels un facteur n (par exemple 0,1) est appliqué. Une erreur de partie de discours influence donc le LAS et précision et rappel ne sont plus forcément identiques ;

– *Content Word Labeled Attachement Score* (CLAS), une variante du LAS, qui ignore les relations de dépendance des mots fonctionnels afin de pouvoir comparer des scores de langues très différentes (Nivre et Fang, 2017). Par exemple, le finnois ayant des cas locaux au lieu de prépositions a moins de mots que le français pour une phrase comme *tu vas de Helsinki à Turku*, donc un parser ne peut pas attacher incorrectement un mot absent (par exemple le pronom et les prépositions dans la traduction finnoise : *menet Helsingistä Turkuun*) ;

– *Morphology Aware Labeled Attachement Score* (MLAS), une extension du CLAS, qui en plus prend en compte les valeurs des parties de discours et les traits morphologiques en plus de l'arbre syntaxique (Zeman *et al.*, 2018).

Pour une comparaison de qualité de deux versions de modèles pour une langue, LAS est la métrique à préférer, car il exprime la qualité globale de l'analyse. Pour la comparaison crosslingue d'un modèle CLAS et MLAS sont plus adaptés afin de ne pas « désavantager » les langues avec beaucoup des mots fonctionnels (comme le français).

3. Approche de génération d'une réponse naturelle

L'approche de génération de réponses que nous décrivons dans cet article est un composant d'un SQR qui a été développé par Rojas Barahona *et al.* (2019). Comme illustré dans la figure 1, l'architecture de ce système se compose d'un frontal de traite-

ment de la parole, d'un composant de compréhension, d'un gestionnaire de contexte, d'un composant de génération et d'un composant de synthèse.

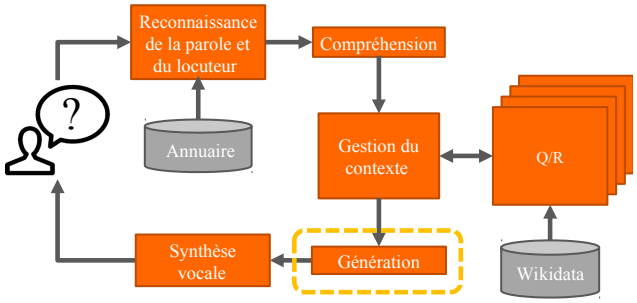

Figure 1. *L'architecture globale du système de questions-réponses conversationnel*

Il s'agit premièrement de comprendre la question posée par l'utilisateur puis de traduire cette question en langue naturelle (français ou anglais) dans une représentation formelle pour ensuite transformer cette représentation formelle en une requête Sparql[3]. Grâce à la requête Sparql nous cherchons la réponse dans une base de connaissances RDF, dans notre cas Wikidata[4]. La réponse est toujours une liste d'URI ou de valeurs. Prenons l'exemple de la question *Qui a joué le rôle de Gus Fring dans Breaking Bad ?* la requête Sparql

```
SELECT DISTINCT ?uri WHERE { wd:Q5620660 ^pq:P453/ps:P161 ?uri }
```

sera envoyée à Wikidata pour extraire la réponse. L'entité `wd:Q5620660` représente le rôle de *Gus Fring* ainsi que le prédicat `^pq:P453/ps:P161`[5] représente le chemin dans le graphe de connaissance entre `wd:Q5620660` et la réponse (`wd:Q726142`) qui réfère à l'acteur *Giancarlo Esposito* en passant par d'autres nœuds.

Bien que nous arrivons à trouver la réponse exacte à une question, sa représentation n'est pas conviviale pour l'utilisateur. De ce fait, nous proposons une approche non supervisée qui intègre l'utilisation des modèles transformers tels que BERT (Devlin *et al.*, 2019) et GPT (Radford *et al.*, 2018). Le choix d'une approche non supervisée émane du fait qu'il n'existe pas un corpus d'apprentissage associant une question à une réponse compacte, exhaustive et qui permettrait d'appliquer en mode supervisé une architecture neuronale *end-to-end* apprenant à générer une phrase répondant à une question. Cette approche part du fait que nous avons déjà extrait la réponse exacte à une question posée. Nous supposons qu'une réponse bien formulée n'est que la reformulation de la question même associée à la réponse exacte. Cette approche comprend deux étapes fondamentales. La première étape consiste à effectuer une analyse en dé-

3. `https://www.w3.org/TR/sparql11-overview/`
4. `https://www.wikidata.org/`
5. `P453` « rôle de », `P161` « acteur ».

pendances de la question en entrée et nous procédons dans une deuxième étape à la génération de la réponse.

3.1. *Analyse en dépendances*

Pour cette première étape d'analyse en dépendances, nous utilisons une version améliorée de UdpipeFuture (Straka, 2018) qui était le système gagnant en termes de la métrique MLAS de la tâche de l'analyse en dépendances (Zeman *et al.*, 2018). UdpipeFuture est un analyseur, qui fait l'étiquetage en parties de discours et la lemmatisation avec un LSTM. L'analyse en dépendances est faite avec un parser à base de graphes, inspiré de Dozat *et al.* (2017).

Notre modification consiste à intégrer les plongements contextuels à UdpipeFuture lors de l'apprentissage. Pour cela, nous nous sommes orientés vers BERT multilingue (Devlin *et al.*, 2019), XLM-R (Conneau *et al.*, 2019) (pour l'anglais et le français), RoBERTA (Liu *et al.*, 2019) (pour l'anglais), FlauBERT (Le *et al.*, 2020) et CamemBERT (Martin *et al.*, 2019) (pour le français[6]) lors de l'apprentissage des treebanks French-GSD et English-EWT[7], issus du projet Universal Dependencies (UD). L'ajout des plongements contextuels a significativement augmenté les résultats pour les trois métriques, LAS, CLAS et MLAS (cf. tableau 1 pour le français et l'anglais, pour d'autres langues cf. Heinecke (2020))

français (UD-French-GSD)

	Straka (2018)	FlauBERT	BERT	CamemBERT	**XLM-R**
MLAS	77,29	79,53	81,64	82,17	**82,62**
CLAS	82,49	84,16	86,21	86,45	**86,94**
LAS	85,74	87,98	89,68	89,67	**89,82**

anglais (UD-English-EWT)

	Straka (2018)	BERT	RoBERTA	**XLM-R**
MLAS	74,71	81,16	82,38	**82,91**
CLAS	79,14	85,89	86,89	**87,24**
LAS	82,51	88,63	89,40	**89,54**

Tableau 1. *Analyse en dépendances du français et de l'anglais (UD 2.2), les meilleurs résultats en gras*

Néanmoins, pour l'analyse en dépendances des questions simples de type quiz, les deux treebanks UD (French-GSD et English-EWT) ne sont pas adaptés, car leurs

6. Nos expérimentations s'appuient sur `flaubert_base_cased` et `camembert-base`.
7. Comme la campagne d'évaluation CoNLL 2018, nous utilisons la version 2.2 des treebanks UD pour la comparaison.

corpus d'apprentissage ne contiennent pas ou très peu de questions[8]. Les mauvais résultats de l'analyse en dépendances des questions avec des modèles appris avec ces treebanks sont résumés en tableau 2.

français (Fr-GSD)

	BERT	**CamemBERT**	FlauBERT	XLM-R
MLAS	60,52	**61,32**	58,09	59,23
CLAS	73,04	**75,26**	70,96	73,52
LAS	79,27	**80,49**	78,40	79,27

anglais (En-EWT)

	BERT	RoBERTa	XLM-R
MLAS	**80,45**	80,68	80,68
CLAS	**88,02**	89,17	89,42
LAS	**90,58**	91,49	91,88

Tableau 2. *Analyse des questions avec des modèles appris sur UD sans modifications*

Afin d'améliorer l'analyse, nous avons enrichi les treebanks d'apprentissage French-GSD et English-EWT en annotant 309 questions anglaises des challenges QALD7 (Usbeck *et al.*, 2017) et QALD8[9] (ainsi 91 questions pour le test) en supprimant les doublons. Pour le français, nous avons traduit des questions issues de QALD7, et formulé des questions nous-mêmes (66 pour le test, 267 pour l'apprentissage). Les annotations ont été effectuées par deux linguistes avec le guide d'annotation du projet Universal Dependencies[10] et la documentation des treebanks French-GSD (pour les questions françaises) et English-EWT (pour l'anglais). Nous avons procédé à une pré-annotation automatique des deux corpus de questions avec des modèles appris sur French-GSD et English-EWT. Puis nous avons effectué deux passes de validation et corrections. Ensuite nous avons fait une évaluation 4-fold. Comme le tableau 3 le montre, la qualité de l'analyse augmente considérablement. Les *embeddings* CamemBERT (pour le français) et BERT (anglais) ont à nouveau le meilleur impact.

Nous nous appuyons sur la version Udpipe-Future[11] que nous avons améliorée avec BERT et CamemBERT et qui donne les meilleurs résultats en termes d'analyse en dépendances pour procéder au découpage de la question en fragments textuels (appelés également *chunks*) : $Q = \{c_1, c_2, \ldots, c_n\}$.

8. Il existe également un treebank de questions, French-FQB (Seddah et Candito, 2016), mais la plupart des questions de ce treebank sont plutôt des questions longues, et peu similaires aux questions « typiques » du quiz.
9. `https://github.com/ag-sc/QALD`
10. `https://universaldependencies.org/guidelines.html`
11. `https://github.com/Orange-Opensource/udparse`

français (French-GSD)

	BERT	**CamemBERT**	FlauBERT	XLM-R
MLAS	91,20	**92,12**	90,53	91,23
CLAS	96,10	**97,37**	94,74	96,14
LAS	97,55	**98,26**	96,86	97,56

anglais (English-EWT)

	BERT	RoBERTa	XLM-R
MLAS	**84,85**	83,08	83,08
CLAS	**91,92**	91,67	90,66
LAS	**94,24**	93,85	93,59

Tableau 3. *Analyse des questions avec des corpus d'apprentissage UD de base enrichis de questions*

Si on reprend l'exemple précédent de la question *Qui a joué le rôle de Gus Fring dans Breaking Bad ?* l'ensemble des fragments textuels serait *Q* = {*Qui, a joué, le rôle de Gus Fring, dans Breaking Bad*}.

3.2. *Processus de génération de réponses*

Dans cette deuxième étape, nous procédons d'abord à un premier test de l'ensemble Q pour vérifier si le fragment textuel qui contient un marqueur de question (*quel*, *quand*, *qui*, etc.) représente le sujet *nsubj* ou l'objet *obj* dans l'arbre en dépendances de la question analysée. Si c'est le cas, nous remplaçons tout simplement ce fragment textuel par la réponse que nous avons identifiée précédemment. Reprenons l'exemple précédent de la question *Qui a joué le rôle de Gus Fring dans Breaking Bad ?*. Le système détecte automatiquement que le fragment textuel contenant le marqueur de question *Qui* représente bien le sujet. Ce sujet sera donc remplacé directement par la réponse exacte *Giancarlo Esposito*. Par la suite, la réponse générée sera *Giancarlo Esposito a joué le rôle de Gus Fring dans Breaking Bad*. Autrement, nous procédons à la suppression du fragment textuel contenant le marqueur de question que nous avons détecté et nous rajoutons la réponse *R* à l'ensemble *Q* :

$$Q = \{c_1, c_2, \ldots, c_{n-1}, R\}$$

À partir de l'ensemble de fragments textuels *Q*, nous générons par permutation toutes les structures de réponses possibles qui peuvent former la phrase répondant à la question traitée :

$$S = \{s_1(R, c_1, c_2, \ldots, c_{n-1}), s_2(c_1, R, c_2, \ldots, c_{n-1}), \ldots, s_m(c_1, c_2, \ldots, c_{n-1}, R)\}$$

Nous nous référons à l'utilisation d'un modèle de langue (ML) qui permet d'assigner une probabilité d'occurrences pour les séquences de mots générées. Dans notre approche, l'ensemble des structures S sera évalué par un modèle de langue basé sur des modèles transformer qui permettra d'extraire la séquence de fragments textuels la plus probable qui servira de réponse :

$$structure^* = s \in S; p(s) = \underset{s_i \in S}{\operatorname{argmax}}\, p(s_i)$$

Une fois que nous avons identifié la structure qui représentera la réponse à la question traitée, nous passons à la génération des termes manquants. En effet, nous supposons qu'il pourrait y avoir un ou plusieurs termes qui ne figurent pas nécessairement dans la question ou dans la réponse mais qui sont en revanche nécessaires à la génération d'une bonne structure grammaticale de la réponse. Ce processus nécessite que nous définissions deux paramètres, le nombre de termes manquants possible et leurs positions dans la structure sélectionnée. Dans cet article, pour fixer ces deux paramètres, nous faisons l'hypothèse qu'un seul terme pourrait être manquant et qu'il est situé avant la réponse courte dans la structure identifiée, comme cela pourrait être le cas pour un article défini manquant (*la, les*, etc.) ou encore une préposition (*dans, à*, etc.) par exemple. Par conséquent, pour prédire ce terme manquant, nous utilisons des modèles de génération (MG) basés sur le modèle transformer BERT pour sa capacité à capturer de manière bidirectionnelle le contexte d'un mot donné dans une phrase. Dans le cas où le modèle de génération renvoie une séquence de caractères non alphabétiques, nous supposons que la structure optimale, telle que prédite par le ML, n'a pas besoin d'être complétée par un terme supplémentaire. Dans ce qui suit, nous illustrons le déroulement des différentes étapes de l'approche proposée avec un exemple en anglais :

Question : *how far is Ponte Vedra beach from Jacksonville FL ?*

1) Analyse de la question et extraction de la réponse moyennant notre SQR (Rojas Barahona *et al.*, 2019) :
   Réponse_courte = {eighteen miles southeast}
2) Découpage de la question en fragments textuels à partir de l'analyse en dépendances que nous avons définie :
   Q = {*How far*, *is*, *Ponte Vedra beach*, *from Jacksonville FL*}
3) Suppression des marqueurs de question (*How far*) :
   Q = {*is*, *Ponte Vedra beach*, *from Jacksonville FL*}
4) Ajout de la réponse courte extraite :
   Q = {*is*, *Ponte Vedra beach*, *from Jacksonville FL*, *eighteen miles southeast*}
5) Génération des structures de réponses possibles *S* :
   S = {*Ponte Vedra beach*, *is*, *from Jacksonville FL*, *eighteen miles southeast* ;
   from Jacksonville FL, *Ponte Vedra beach*, *is*, *eighteen miles southeast* ; ...}
6) Évaluation des structures par un modèle de langue :
   p(structure)* $= \operatorname{argmax}_{s_i \in S} p(s_i)$:
   *structure** = *Ponte Vedra beach*, *is*, *eighteen miles southeast*, *from Jacksonville FL*

7) Génération des termes possiblement manquants à *structure** par un modèle de génération (mot manquant = *about*) :
Ponte Vedra beach is [mot manquant] *eighteen miles southeast from Jacksonville FL*

Réponse : *Ponte Vedra beach is about eighteen miles southeast from Jacksonville FL.*

Comme nous pouvons le remarquer, la réponse finale générée avec l'ajout du terme manquant s'apparente considérablement à une réponse naturelle qui pourrait être émise par un humain.

4. Expérimentation et évaluation

Les corpus de tests existants pour l'évaluation des SQR sont soit adaptés aux systèmes qui génèrent la réponse exacte à la question et donc une réponse courte, soit plus axés vers la tâche de *Machine Reading Comprehension* où la réponse est un passage de texte contenant la réponse exacte. Par la suite, nous avons créé un jeu de données qui consiste à associer des questions extraites du corpus QALD-7 challenge (Usbeck *et al.*, 2017) avec des réponses en langage naturel qui ont été définies manuellement par un linguiste et que nous avons revues individuellement. Ce corpus appelé Quereo consiste en 150 questions avec leurs réponses exactes. On note en moyenne trois réponses possibles en langage naturel pour chaque question. Ce corpus existe en versions française et anglaise.

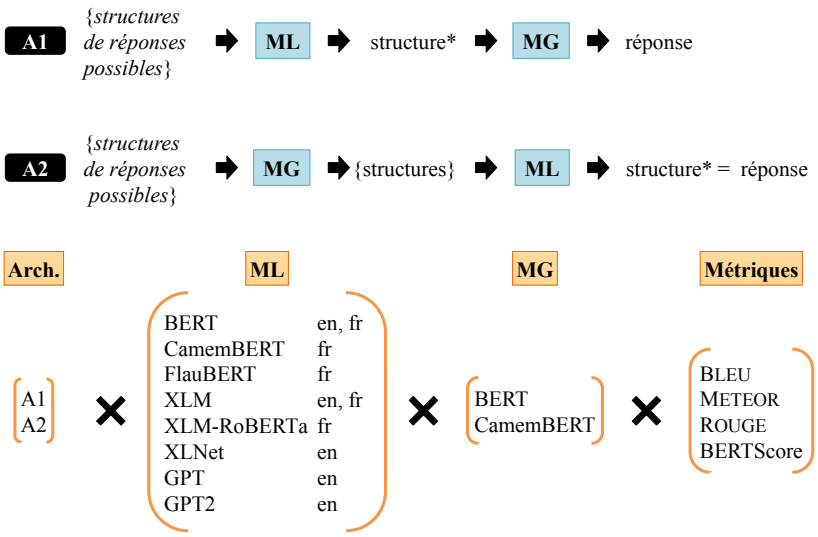

Figure 2. *Cadre d'expérimentation pour identifier la meilleure configuration*

Comme illustré dans la figure 2, nous avons évalué deux architectures possibles de notre approche pour la génération de réponses. La première architecture *A1* consiste

d'abord à lister toutes les structures de réponses possibles, puis à les faire évaluer par un ML qui permet la sélection de la structure optimale, enfin de générer avec le MG le terme manquant dans la structure sélectionnée. La deuxième architecture *A2* consiste d'abord à lister toutes les structures de réponses possibles, puis à générer avec le MG les termes manquants dans chaque structure, enfin à faire évaluer l'ensemble de ces structures par le ML pour sélectionner la structure optimale. Pour cet article, nous supposons qu'il y a seulement un terme manquant par structure.

Pour évaluer l'approche proposée, nous avons opté pour trois métriques *n*-gram (BLEU, METEOR et ROUGE) utilisées dans la littérature pour évaluer ce type de tâche et la métrique BERTScore qui exploite les plongements lexicaux pré-entraînés de BERT pour calculer la similarité entre la réponse générée et la réponse de référence. Pour pouvoir comparer les différentes variantes de l'approche, nous nous sommes référés au test de Friedman (Milton, 1939) qui permet de détecter les écarts de performances entre plusieurs modèles évalués par plusieurs métriques en se basant sur les rangs moyens.

Nous avons également mené une évaluation humaine pour les versions française et anglaise du corpus de données, dans laquelle nous avons demandé à 20 locuteurs natifs des deux langues d'évaluer la pertinence d'une réponse générée (*correcte* ou *pas correcte*) pour une question donnée en indiquant le ou les types d'erreurs détectées (*accord grammatical*, *préposition incorrecte*, *ordre des mots*, etc.). La figure 3 présente le cadre d'évaluation que nous avons mis en œuvre et fourni aux participants. Les résultats de chaque participant sont enregistrés dans un fichier *json*. Le taux d'interaccords entre les participants qui a été mesuré par le coefficient Kappa de Fleiss (Fleiss, 1971) a atteint 70 %, ce qui indique un accord substantiel d'après le tableau d'interprétation de Landis et Koch (1977). À travers l'étude d'évaluation humaine, nous voulions explorer dans quelle mesure les métriques standard sont fiables pour évaluer les approches GAT dans le contexte des systèmes de questions-réponses.

Le tableau 4 (corpus français) et le tableau 6 (corpus anglais) ne présentent que les résultats obtenus pour les trois meilleurs modèles selon le classement du test de Friedman. Les modèles de langue utilisés sont adaptés selon la langue du corpus mis en test. Vanté par ses mérites en tant que modèle génératif très puissant entraîné sur un très large corpus de données constitué de 8 millions de pages Web associant entre autres des termes en anglais et en français, le modèle GPT a été également testé avec le corpus français pour voir s'il arrivait à détecter la meilleure structure à choisir pour une question. En effet, notre corpus d'évaluation peut, dans certains, cas inclure des questions qui associent des termes en anglais et en français, tels que le nom d'un film. Les valeurs mises entre crochets représentent le rang d'un modèle selon la métrique utilisée.

Nous notons que le score de précision le plus élevé pour le français d'environ 85 % a été obtenu avec la première architecture avec BERT comme modèle de génération (MG) et CamemBERT comme modèle de langage (ML). On remarque également que l'architecture *A1*, qui considère l'évaluation de la structure par un ML avant de générer les termes manquants, fonctionne mieux. Étonnamment, en tant que modèle génératif,

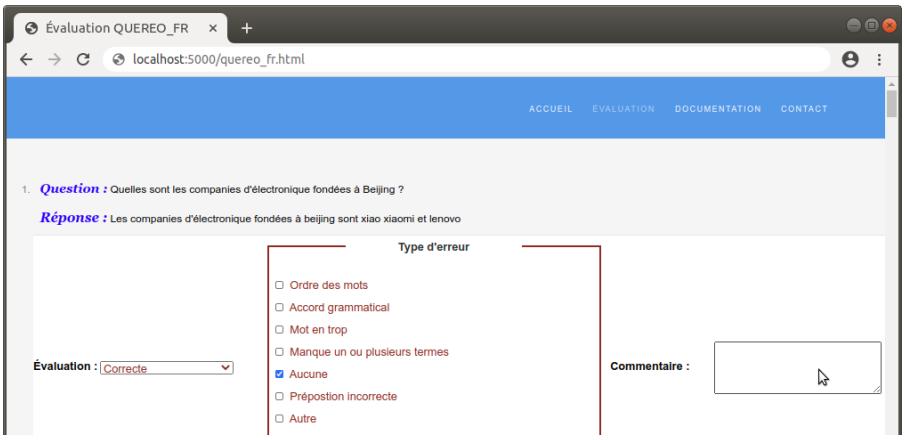

Figure 3. *La plate-forme d'évaluation*

le modèle multilingue BERT prédit mieux les termes manquants que CamemBERT pour les réponses du corpus français. Ces résultats sont également confirmés par le test de Friedman où nous pouvons clairement voir que la première configuration classée correspond à la meilleure configuration classée selon l'évaluation humaine, avec une légère différence par rapport aux autres configurations. Voyons si cela signifie que les quatre métriques sont corrélées à la précision humaine.

D'après le tableau 5 qui présente la corrélation Pearson (Benesty *et al.*, 2009) entre la précision humaine avec les quatre métriques et la figure 4 qui illustre le classement donné par chaque métrique d'évaluation avec le jugement humain pour chaque configuration (c.-à-d. *configuration* = *MG* × *architecture* × *ML*) testée, nous pouvons clairement voir que les résultats de l'évaluation humaine sont positivement et fortement corrélés aux scores BLEU, METEOR et BERT. Ces métriques correspondent pratiquement au classement humain et sont donc évidemment capables d'identifier quelle configuration donne les meilleurs résultats. En revanche, la métrique ROUGE, utilisée pour l'évaluation des questions-réponses en français, est modérément corrélée à l'évaluation humaine, ce qui signifie que cette métrique ne doit pas être considérée comme la seule métrique d'évaluation pour évaluer une telle tâche. En revanche, lorsque la métrique ROUGE est considérée avec les autres métriques, elle permet de se rapprocher du jugement humain.

Le tableau 6 présente les résultats pour le corpus de données anglais et indique que la meilleure précision est d'environ 72 % avec *A1*, BERT comme modèle de génération et le transformer GPT (*Generative Pretrained Model*) comme modèle de langue. Selon les trois premières configurations, c'est l'architecture *A2* qui se démarque et le transformer GPT qui prend le dessus sur les autres modèles de langue.

rang hum.	rang Friedman	Arch.	MG	ML	*accuracy* humaine
1	**1**	**A1**	**BERT**	**CamemBERT**	**84,85**
2	2	A2	BERT	FlauBERT-small-cased	84,09
2	3	A1	BERT	XLM-RoBERTa-base	84,09
2	9	A1	BERT	BERT-base-multilingual-cased	84,09
5	4	A1	BERT	FlauBERT-base-uncased	83,33
5	5	A1	BERT	mlm-1024	83,33
5	6	A2	BERT	GPT2	83,33
5	10	A2	BERT	XLM-clm-enfr-1024	83,33
5	11	A1	BERT	XLM-clm-enfr-1024	83,33
5	12	A2	BERT	FlauBERT-large-cased	83,33
5	13	A2	BERT	mlm-1024	83,33
5	14	A2	BERT	XLM-RoBERTa-base	83,33

rang hum.	rang Friedman	BLEU score	BLEU rang	METEOR score	METEOR rang	ROUGE score	ROUGE rang	BERTS score	BERTS rang
1	**1**	**86,28**	**[1]**	**96,76**	**[1]**	**93,69**	**[6]**	**97,89**	**[2]**
2	2	85,87	[7]	96,75	[2]	94,22	[1]	97,96	[1]
2	3	85,93	[4]	96,63	[6]	93,79	[5]	97,88	[3]
2	9	85,01	[19]	96,52	[22]	93,81	[4]	97,79	[7]
5	4	86,17	[2]	96,72	[3]	93,56	[14]	97,81	[6]
5	5	85,39	[10]	96,60	[8]	93,61	[10]	97,83	[4]
5	6	85,46	[9]	96,67	[4]	93,48	[17]	97,76	[10]
5	10	85,89	[6]	96,55	[18]	93,57	[13]	97,71	[19]
5	11	84,99	[20]	96,52	[23]	93,87	[3]	97,76	[12]
5	12	86,15	[3]	96,57	[13]	93,14	[37]	97,79	[8]
5	13	85,90	[5]	96,54	[20]	93,30	[27]	97,76	[11]
5	14	85,32	[13]	96,46	[28]	93,63	[9]	97,71	[17]

Tableau 4. *Classement des modèles selon l'évaluation humaine (meilleur en gras) et le test Friedman (meilleur en jaune), corpus français*

	BLEU	METEOR	ROUGE	BERT-score
Quereo_fr	98 %	99 %	46 %	97%
Quereo_en	85 %	80 %	83 %	88 %

Tableau 5. *Coefficient de la corrélation de Pearson entre les quatre métriques et l'évaluation humaine*

Ces résultats sont confirmés par le test de Friedman avec une légère différence de classement et également appuyés par les scores de corrélation entre l'évaluation humaine et chacune des quatre métriques comme le montre le tableau 5.

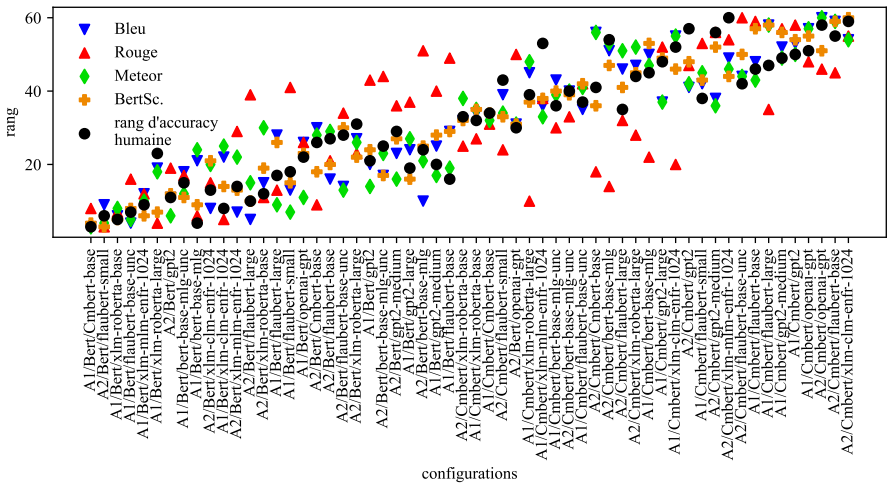

Figure 4. *Corrélation entre les évaluations humaines et les métriques* BLEU, METEOR, ROUGE *et* BERT *score pour le Q/R en français (« CmBERT » veut dire* CamemBERT*)*

L'objectif de ces résultats est d'identifier parmi les métriques standard celles qui sont fiables pour évaluer la tâche de génération de réponses dans le contexte des SQR.

Nous avons également procédé à l'analyse des erreurs signalées par les participants. Comme nous pouvons le constater à partir de la figure 5, l'erreur la plus courante signalée pour les deux corpus de données anglais et français est l'ordre des mots. Ceci sous-entend un problème lié à la phase d'évaluation des structures de réponses possibles par le modèle de langue. La deuxième erreur la plus signalée est l'indication d'un ou plusieurs termes manquants dans la réponse (corpus français) ou la présence de termes intrus (anglais). Ceci concerne le processus de génération (MG).

Figure 5. *Distribution d'erreurs de la génération*

rang hum.	rang Fr.	arch.	MG	ML	*accuracy* humaine
1	**1**	**A2**	**BERT–base-multiling-cased**	**GPT**	**72,36**
1	2	A1	BERT-base-multiling-cased	GPT	72,36
3	2	A1	BERT-large-cased	GPT	71,55
3	4	A2	BERT-large-cased	GPT2	71,55
3	5	A2	BERT-base-multiling-cased	GPT2	71,55
3	6	A2	BERT-base-multiling-cased	GPT2-large	71,55
3	7	A2	BERT-base-multiling-cased	GPT2-medium	71,55
3	7	A2	BERT-large-cased	GPT2-medium	71,55
3	10	A2	BERT-large-cased	GPT	71,55
10	7	A2	BERT-large-cased	GPT2-large	70,73
10	30	A1	BERT-base-multiling-cased	BERT-base-unc.	70,73

rang	rang	BLEU		METEOR		ROUGE		BERTS	
hum.	Friedman	score	rang	score	rang	score	rang	score	rang
1	**1**	**78.25**	**[2]**	**94,63**	**[1]**	**92,83**	**[2]**	**97,21**	**[3]**
1	2	78,25	[1]	94,51	[2]	92,53	[10]	97,23	[2]
3	2	77,12	[6]	94,45	[3]	92,80	[5]	97,32	[1]
3	4	76,98	[7]	94,40	[7]	92,86	[1]	97,17	[5]
3	5	77,53	[4]	94,39	[8]	92,82	[4]	97,14	[6]
3	6	77,85	[3]	94,41	[5]	92,65	[7]	97,07	[10]
3	7	77,42	[5]	94,40	[6]	92,58	[9]	97,10	[9]
3	7	75,96	[13]	94,41	[4]	92,60	[8]	97,18	[4]
3	10	76,28	[11]	94,30	[9]	92,76	[6]	97,14	[7]
10	7	76,74	[8]	94,26	[10]	92,83	[3]	97,14	[8]
10	30	74,85	[30]	93,94	[31]	90,86	[26]	96,61	[28]

Tableau 6. *Classement des modèles selon l'évaluation humaine (meilleur en gras) et le test Friedman (meilleur en jaune), corpus anglais*

5. Une approche de génération automatique de corpus de questions-réponses

Les résultats prometteurs que nous avons obtenus suite aux deux évaluations que nous avons conduites, nous ont amenés à envisager d'utiliser cette même approche pour construire des corpus de type questions-réponses qui associent une réponse en langage naturel à une question en langage naturel. Notre idée est de générer ainsi des corpus d'apprentissage de grande taille pour entraîner des approches neuronales de type *end-to-end*. Notre approche de génération de réponses concises conduit évidemment à des réponses synthétiques un peu stéréotypées. Pour introduire un peu de variété dans l'expression des réponses, il était nécessaire de rajouter dans ce corpus d'apprentissage des réponses plus naturelles.

Nous avons eu l'idée d'exploiter les corpus MRQA (*Machine Reading for Question Answering*) existants pour alimenter nos corpus d'apprentissage. Dans un corpus MRQA, chaque entrée comporte une question en langage naturel, une réponse courte et une réponse longue, représentée sous forme d'un paragraphe de plusieurs phrases dont au moins une contient la réponse courte. La figure 6 montre un exemple de ce type de représentation de réponse[12].

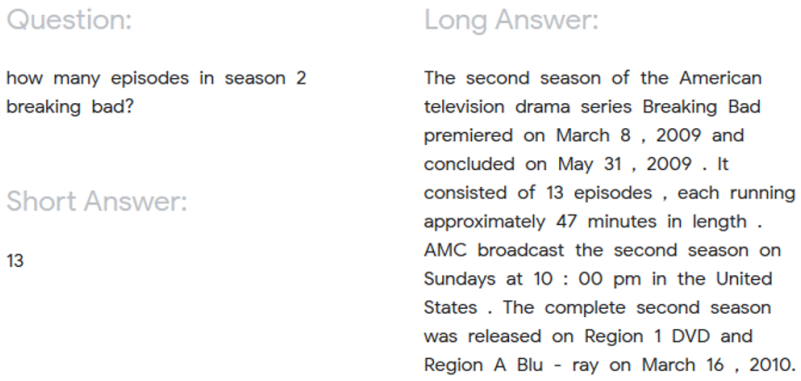

Figure 6. *Un exemple de questions-réponses extrait du corpus GNQ*

L'approche que nous proposons consiste à explorer l'utilisation des corpus MRQA pour d'une part, générer une réponse synthétique à partir de la question et de la réponse courte (avec l'approche de génération décrite précédemment) et d'autre part, extraire une réponse concise et naturelle à partir de la réponse longue. Ce procédé d'augmentation des données (Shorten et Khoshgoftaar, 2019) permet de régulariser et de réduire le surajustement lors de l'apprentissage.

Pour extraire une réponse concise et naturelle à partir de chaque réponse longue, nous avons d'abord supprimé de ces réponses les références externes vers d'autres pages ou les notes de bas de page, choses que l'on retrouve souvent dans les pages Wikipédia ou dans des articles. Puis, nous avons découpé chaque réponse longue en phrases, pour ne retenir que celles contenant la réponse courte. Enfin, nous avons calculé la similarité sémantique entre la question et chaque phrase candidate pour ne retenir comme réponse naturelle que la phrase ayant le score de similarité le plus élevé.

On dénote :

$$C = \{(Q_1, Sa_1, La_1), (Q_2, Sa_2, La_2), \ldots, (Q_i, Sa_i, La_i)\}; i \in [1, m]$$

un corpus MRQA qui contient un ensemble de triplets *(question, réponse courte, réponse longue)*, m étant le nombre de triplets dans le corpus. La_i =

12. `https://ai.google.com/research/NaturalQuestions`

$\{S_1, S_2, \ldots, S_j\}; j \in [1, n]$ est la réponse longue pour la question Q_i segmentée en un ensemble de phrases S_j, *n* étant le nombre de phrases que peut contenir une réponse longue. Nous avons procédé à un prétraitement de ces phrases afin de supprimer les références externes vers d'autres pages ou les notes de bas de pages que l'on retrouve souvent dans les pages Wikipédia ou dans des articles.

Notre approche consiste à extraire de l'ensemble La_i un sous-ensemble de phrases S_j candidates pour la réponse naturelle à la question posée Q_i :

$$S_{candidates} = \{S_1, S_2, \ldots, S_k\} \sqsubseteq La_i; k \in [1, n] \forall S_k \in S_{candidates} : Sa_i \in S_k$$

Ce sous-ensemble ne rassemble que les phrases qui contiennent la réponse courte Sa_i identifiée pour la question Q_i. Sachant que le nombre de phrases candidates varie dans un intervalle $[1, n]$, deux cas se présentent. Le premier cas survient quand nous avons plus qu'une phrase candidate $k > 1$, nous procédons alors au calcul de ce que l'on appelle *un score de confiance* ($confidence_{\text{score}} \in [0, 1]$) pour chaque phrase candidate. En fait, ce score de confiance représente la similarité sémantique entre une phrase candidate et la question. Ceci permet de ne sélectionner que la phrase qui relève du contexte de la question.

Prenons l'exemple de la question *Qui est le maire de paris ?* illustrée dans la figure 7. On remarque qu'il y a plusieurs phrases candidates comportant la réponse courte (Anne Hidalgo) à la question. Pourtant, toutes ces phrases candidates ne sont pas des réponses acceptables. Nous choisissons donc la phrase dont le sens est le plus proche de celui de la question. Pour cela, nous calculons la similarité sémantique de la question et celle de chaque phrase candidate puis, sélectionnons celle ayant la représentation la plus proche de celle de la question.

Question	Réponse longue
Qui est le maire de Paris ?	Ana María Hidalgo Aleu, dite Anne Hidalgo, née le 19 juin 1959 à San Fernando (Espagne), est une femme politique française possédant également la nationalité espagnole. Anne Hidalgo est titulaire d'une maîtrise de sciences sociales du travail, obtenue à l'université Jean-Moulin-Lyon-III et d'un DEA de droit social et syndical. Membre du Parti socialiste, elle est première adjointe au maire de Paris de 2001 à 2014 et conseillère régionale d'Île-de-France de 2004 à 2014. À l'issue des élections municipales de 2014, Anne Hidalgo devient la première femme maire de Paris et est réélue à la suite des élections municipales de 2020.
Réponse courte	
Anne Hidalgo	

Figure 7. *Un exemple de questions-réponses*

Pour la mesure de similarité sémantique, nous utilisons l'approche Simbow (Charlet et Damnati, 2017) qui a prouvé sa performance dans la tâche de similarité entre les questions dans le challenge SemEval-2017. Cette métrique s'apparente à la métrique *soft-cosinus* mais considère en plus les relations entre mots qui peuvent être d'ordre lexical ou sémantique en faisant introduire dans la formule une matrice de relations :

$$Simbow_{\cos}(X, Y) = \frac{X^t \cdot M \cdot Y}{\sqrt{X^t \cdot M \cdot X}\sqrt{Y^t \cdot M \cdot Y}}$$

$$X^t \cdot M \cdot Y = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i m_{i,j} y_j$$

où M est une matrice d'élément $m_{i,j}$ qui représente la relation entre les termes i et j et qui est calculée par la métrique *cosinus* entre les vecteurs de plongement `word2vec` représentatifs des termes i et j. Ceci introduit la notion de similarité sémantique qui permet d'éviter de se retrouver avec une similarité nulle dans le cas où deux textes n'ont aucun terme en commun.

$$S^* = S \in S_{candidates}$$

$$confidence_{\text{score}}(S^*) = \underset{S_i \in S_{candidates}}{\text{argmax}} \; Simbow_{\cos}(S_i, Q)$$

Dans le cas où il n'existe qu'une seule phrase candidate $S_{candidates} = \{S^*\}$, cette dernière sera considérée comme la meilleure réponse à la question.

Afin d'évaluer cette approche de génération automatique de corpus de questions-réponses naturelles à partir des corpus MRQA, nous nous sommes basés sur le corpus Google's Natural Questions (GNQ, Kwiatkowski *et al.* (2019)), dont un exemple est présenté dans la figure 6. Ce corpus contient des questions posées par de vrais utilisateurs qui, suite à la lecture et la compréhension d'un ensemble d'articles Wikipédia, identifient les sections qui intègrent les réponses. Comme il n'existe pas des réponses de référence pour ce type de corpus, nous avons procédé avec une évaluation humaine. Pour cela, nous avons développé une plate-forme d'évaluation qui expose pour chaque question sa réponse courte, une réponse concise et naturelle, appelée *R1*, extraite de la réponse longue par le procédé que nous venons de décrire dans la présente section et la réponse concise et synthétique, appelée *R2*, générée par l'approche que nous avons exposée dans la section 3. D'un ensemble de questions-réponses qui est de l'ordre de 307 000 questions-réponses, nous avons extrait pour le test un échantillon de 1 000 questions-réponses choisies aléatoirement.

Comme affiché dans la figure, chaque participant est tenu d'indiquer, pour chaque réponse, si elle est *correcte* ou *pas correcte*. En option, il peut aussi indiquer si la réponse est également naturelle. Une réponse est considérée *correcte* si elle est grammaticalement correcte et répond bien à la question posée. Une autre option permet de signaler d'éventuelles erreurs liées, par exemple, à une syntaxe grammaticale incorrecte de la question posée ou à une réponse courte non pertinente. L'ajout de cette option s'est avéré nécessaire car nous avons relevé quelques incohérences de ce type d'erreurs dans le corpus de départ GNQ. Cette évaluation étant toujours en cours, nous exposons dans cet article les résultats que nous avons obtenus à ce jour pour un seul participant. Le tableau 9 expose les résultats préliminaires obtenus.

Ces résultats, même préliminaires, sont toutefois très encourageants et montrent un réel potentiel à l'approche hybride de génération automatique de corpus que nous proposons.

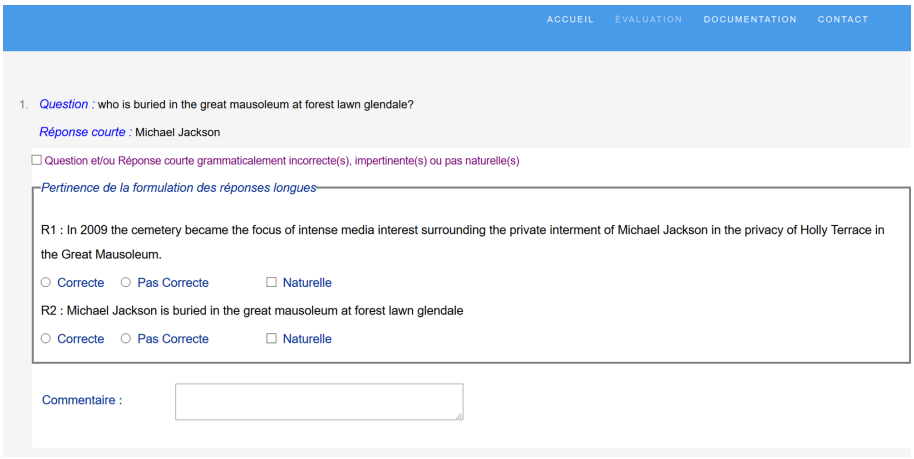

Figure 8. *Plate-forme d'évaluation pour l'approche de génération automatique de corpus de questions-réponses*

Figure 9. *Résultats préliminaires obtenus*

6. Conclusion

Dans cet article, nous avons proposé une approche pour la génération d'une réponse concise, en langage naturel, pour les systèmes de questions-réponses (SQR). Elle s'appuie sur l'analyse en dépendances de la question pour déterminer le rôle grammatical de chaque terme. Elle exploite ensuite la distribution de probabilité des séquences de mots ainsi que des modèles génératifs pour générer une réponse correcte. Les résultats obtenus avec des métriques standard sur des questions de test en français et en anglais sont très prometteurs. De plus, une expérimentation a montré une bonne corrélation entre ces métriques et le jugement humain.

Par ailleurs, nous avons aussi proposé une méthode associant notre approche à un procédé d'extraction de questions-réponses à partir de corpus MRQA dans l'objectif de construire un gros corpus synthétique qui permettrait de tester des approches supervisées sur la tâche de génération de réponses en langage naturel.

L'intégration de cette approche dans un prototype conversationnel (Rojas Barahona *et al.*, 2019) nous a permis d'observer son efficacité dans un contexte d'usage

réel. C'est ainsi que nous avons constaté que cette approche imaginée pour traiter des questions en dehors de tout contexte de dialogue était aussi efficace pour générer des réponses concises dans le cas de certaines questions en contexte. Il serait intéressant de poursuivre nos travaux d'analyse et de mesurer ce phénomène. Par ailleurs, dans le cas d'une utilisation intensive du prototype, la construction de la réponse à partir de la question peut être perçue comme trop stéréotypée. Il serait intéressant d'étudier comment l'utilisation de techniques de reformulation et/ou l'introduction de coréférences sur le sujet de la question permettent de limiter ce phénomène.

Remerciements

Nous tenons à remercier les collègues qui ont participé aux deux campagnes de l'évaluation humaine. Nous présentons également nos sincères remerciements aux relecteurs de cet article pour leurs remarques et suggestions constructives.

7. Bibliographie

Abeillé A., Clément L., Toussenel F., « Building a Treebank for French », *in* A. Abeillé (ed.), *Treebanks*, Springer, Heidelberg, 2003.

Agichtein E., Gravano L., « Snowball : Extracting relations from large plain-text collections », *Proceedings of the fifth ACM conference on Digital libraries*, p. 85-94, 2000.

Asai A., Eriguchi A., Hashimoto K., Tsuruoka Y., « Multilingual extractive reading comprehension by runtime machine translation », `https://arxiv.org/abs/1809.03275`, 2018.

Benesty J., Chen J., Huang Y., Cohen I., *Pearson Correlation Coefficient*, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 1-4, 2009.

Bhaskar P., Banerjee S., Pakray P., Banerjee S., Bandyopadhyay S., Gelbukh A., « A hybrid question answering system for Multiple Choice Question (MCQ) », *Question Answering for Machine Reading Evaluation (QA4MRE) at CLEF*, 2013.

Brill E., Dumais S., Banko M., « An analysis of the AskMSR question-answering system », *EMNLP 2002*, ACL, p. 257-264, 2002.

Brill E., Lin J., Banko M., Dumais S., Ng A., « Data-intensive question answering », *TREC 2001*, p. 393-400, 2001.

Charlet D., Damnati G., « SimBow at SemEval-2017 Task 3 : Soft-Cosine Semantic Similarity between Questions for Community Question Answering », *SemEval-2017*, ACL, Vancouver, Canada, p. 315-319, August, 2017.

Chopra S., Auli M., Rush A. M., « Abstractive sentence summarization with attentive recurrent neural networks », *NAACL*, p. 93-98, 2016.

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grace E., Ott M., Zettlemoyer L., Stoyanov V., « Unsupervised Cross-lingual Representation Learning at Scale », `https://arxiv.org/abs/1911.02116`, 2019.

Devlin J., Chang M.-W., Lee K., Toutanova K., « BERT : Pre-training of deep bidirectional transformers for language under-standing », *NAACL*, Minneapolis, p. 4171-4186, 2019.

dos Santos C., Tan M., Xiang B., Zhou B., « Attentive pooling networks », `https://arxiv.org/abs/1602.03609`, 2016.

Dozat T., Qi P., Manning C. D., « Stanford's Graph-based Neural Dependency Parser at the CoNLL 2017 Shared Task », *CoNLL 2017 Shared Task. Multilingual Parsing from Raw Text to Universal Dependencies*, ACL, Vancouver, Canada, p. 20-30, 2017.

Du X., Cardie C., « Harvesting Paragraph-level Question-Answer Pairs from Wikipedia », *ACL*, ACL, Melbourne, Australia, p. 1907-1917, July, 2018.

Fleiss J. L., « Measuring nominal scale agreement among many raters. », *Psychological bulletin*, vol. 76, nº 5, p. 378, 1971.

Girju R., « Automatic detection of causal relations for question answering », *ACL workshop on Multilingual summarization and question answering*, ACL, p. 76-83, 2003.

Heinecke J., « Hybrid Enhanced Universal Dependencies Parsing », *IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies*, ACL, Online, p. 174-180, July, 2020.

Hirschman L., Gaizauskas R., « Natural language question answering : the view from here », *natural language engineering*, vol. 7, nº 4, p. 275-300, 2001.

Iida R., Kruengkrai C., Ishida R., Torisawa K., Oh J.-H., Kloetzer J., « Exploiting Background Knowledge in Compact Answer Generation for Why-Questions », *AAI Conference on Artificial Intelligence*, vol. 33, p. 142-151, 2019.

Ishida R., Torisawa K., Oh J.-H., Iida R., Kruengkrai C., Kloetzer J., « Semi-distantly supervised neural model for generating compact answers to open-domain why questions », *AAAI*, 2018.

Kiperwasser E., Goldberg Y., « Simple and accurate dependency parsing using bidirectional LSTM feature representations », *Transactions of the Association for Computational Linguistics*, vol. 4, p. 313-327, 2016.

Kondratyuk D., Straka M., « 75 Languages, 1 Model : Parsing Universal Dependencies Universally », `http://arxiv.org/abs/1904.02099`, 2019.

Kruengkrai C., Torisawa K., Hashimoto C., Kloetzer J., Oh J.-H., Tanaka M., « Improving event causality recognition with multiple background knowledge sources using multi-column convolutional neural networks », *31st AAAI Conference on Artificial Intelligence*, 2017.

Kwiatkowski T., Palomaki J., Redfield O., Collins M., Parikh A., Alberti C., Epstein D., Polosukhin I., Kelcey M., Devlin J., Lee K., Toutanova K. N., Jones L., Chang M.-W., Dai A., Uszkoreit J., Le Q., Petrov S., « Natural Questions : a Benchmark for Question Answering Research », *Transactions of the Association of Computational Linguistics*, 2019.

Kübler S., McDonald R., Nivre J., *Dependency Parsing*, Morgan and Claypool Publishers, 2009.

Landis J. R., Koch G. G., « The measurement of observer agreement for categorical data », *Biometrics*, vol. 33, nº 1, p. 159-174, 1977.

Lawrence S., Giles C. L., « Context and page analysis for improved web search », *IEEE Internet computing*, vol. 2, nº 4, p. 38-46, 1998.

Le H., Vial L., Frej J., Segonne V., Coavoux M., Lecouteux B., Allauzen A., Crabbé B., Besacier L., Schwab D., « FlauBERT : Unsupervised Language Model Pre-training for French », *LREC*, 2020.

Le J., Zhang C., Niu Z., « Answer Extraction Based on Merging Score Strategy of Hot Terms », *Chinese Journal of Electronics*, vol. 25, nº 4, p. 614-620, 2016.

Liu Y., Ott M., Goyal N., Du Jingfei adn Joshi M., Chen D., Lewis M., Zettlemoyer L., Stoyanov V., « RoBERTa : A Robustly Optimized BERT Pretraining Approach », `https://arxiv.org/abs/1907.11692`, 2019.

Lopez V., Uren V., Sabou M., Motta E., « Is question answering fit for the semantic web ? : a survey », *Semantic Web*, vol. 2, n° 2, p. 125-155, 2011.

Marneffe M.-C. d., Manning C. D., « The Stanford typed dependencies representation », *Co-Ling, Workshop on Cross-framework and Cross-domain Parser Evaluation*, p. 1-8, 2008.

Martin L., Muller B., Suárez P. J. O., Dupont Y., Romary L., Villemonte de la Clergerie É., Seddah D., Sagot B., « CamemBERT : a Tasty French Language Model », `https://arxiv.org/abs/1911.03894`, 2019.

Miao Y., Blunsom P., « Language as a latent variable : Discrete generative models for sentence compression », *EMNLP*, 2016.

Milton F., « A correction : The use of ranks to avoid the assumption of normality implicit in the analysis of variance », *JASA*, vol. 34, n° 205, p. 109, 1939.

Nallapati R., Zhou B., dos Santos C., Gülçehre Ç., Xiang B. *et al.*, « Abstractive text summarization using sequence-to-sequence RNNs and beyond », *CoNLL*, p. 280-290, 2016.

Nivre J., « An Efficient Algorithm for Projective Dependency Parsing », *IWPT*, Dublin, p. 149-160, 2003.

Nivre J., Fang C.-T., « Universal Dependency Evaluation », *in* M.-C. d. Marneffe, J. Nivre, S. Schuster (eds), *NoDaLiDa Workshop on Universal Dependencies*, Göteborg, p. 86-95, 2017.

Nivre J., Hall J., Nilsson J., « MaltParser : A Data-Driven Parser-Generator for Dependency Parsing », *LREC*, ELRA, Genoa, Italy, May, 2006.

Nivre J., Marneffe M.-C. d., Ginter F., Goldberg Y., Goldberg Y., Hajič J., D. M. C., McDonald R., Petrov S., Pyysalo S., Silveira N., Tsarfaty R., Zeman D., « Universal Dependencies v1 : A Multilingual Treebank Collection », *10th LREC*, Portorož, Slovenia, p. 23-38, 2016.

Nivre J., Marongiu P., Ginter F., Kanerva J., Montemagni S., Schuster S., Simi M., « Enhancing Universal Dependency Treebanks : A Case Study », *Workshop on Universal Dependencies (UDW 2018)*, ACL, Brussels, Belgium, p. 102-107, November, 2018.

Oepen S., Abend O., Abzianidze L., Bos J., Hajič J., Hershcovich D., Li B., O'Gorman T., Xue N., Zeman D. (eds), *CoNLL 2020 Shared Task : Cross-Framework Meaning Representation Parsing*, ACL, Online, November, 2020.

Oh J.-H., Torisawa K., Hashimoto C., Iida R., Tanaka M., Kloetzer J., « A semi-supervised learning approach to why-question answering », *AAAI*, 2016.

Oh J.-H., Torisawa K., Hashimoto C., Sano M., De Saeger S., Ohtake K., « Why-question answering using intra-and inter-sentential causal relations », *ACL 2013*, p. 1733-1743, 2013.

Pal V., Shrivastava M., Bhat I., « Answering Naturally : Factoid to Full length Answer Generation », *2nd Workshop on New Frontiers in Summarization*, Association for Computational Linguistics, Hong Kong, China, p. 1-9, November, 2019.

Radford A., Narasimhan K., Salimans T., Sutskever I., « Improving language understanding by generative pre-training », `https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf`, 2018.

Reiter E., Dale R., « Building applied natural-language generation systems », *Nat. Lang. Eng.*, vol. 3, n° 1, p. 57–87, 1997.

Rojas Barahona L. M., Bellec P., Besset B., Dos Santos M., Heinecke J., Asadullah M., Leblouch O., Lancien J.-Y., Damnati G., Mory E., Herlédan F., « Spoken Conversational Search for General Knowledge », *SIGdial*, ACL, Stockholm, p. 110-113, 2019.

Rush A. M., Chopra S., Weston J., « A neural attention model for abstractive sentence summarization », `https://arxiv.org/abs/1509.00685`, 2015.

Seddah D., Candito M., « Hard Time Parsing Questions : Building a QuestionBank for French », *10th LREC*, ELRA, Portorož, Slovenia, 2016.

See A., Liu P. J., Manning C. D., « Get to the point : Summarization with pointer-generator networks », `https://arxiv.org/abs/1704.04368`, 2017.

Sharp R., Surdeanu M., Jansen P., Clark P., Hammond M., « Creating causal embeddings for question answering with minimal supervision », *EMNLP*, 2016.

Shorten C., Khoshgoftaar T., « A survey on Image Data Augmentation for Deep Learning », *Journal of Big Data*, vol. 6, p. 1-48, 2019.

Straka M., « UDPipe 2.0 Prototype at CoNLL 2018 UD Shared Task », *CoNLL 2018 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, ACL, Brussels, p. 197-207, 2018.

Tan M., dos Santos C., Xiang B., Zhou B., « Improved representation learning for question answer matching », *ACL 2016*, p. 464-473, 2016.

Usbeck R., Ngomo A.-C. N., Haarmann B., Krithara A., Röder M., Napolitano G., « 7th Open Challenge on Question Answering over Linked Data (QALD-7) », *in* M. Dragoni, M. Solanki, E. Blomqvist (eds), *Semantic Web Challenges*, Springer International Publishing, Cham, p. 59-69, 2017.

Verberne S., van Halteren H., Theijssen D., Raaijmakers S., Boves L., « Learning to rank for why-question answering », *Information Retrieval*, vol. 14, n° 2, p. 107-132, 2011.

Wu M., Zheng X., Duan M., Liu T., Strzalkowski T., Albany S., « Question answering by pattern matching, web-proofing, semantic form proofing », *TREC*, p. 500-255, 2003.

Zayaraz G. *et al.*, « Concept relation extraction using Naïve Bayes classifier for ontology-based question answering systems », *Journal of King Saud University-Computer and Information Sciences*, vol. 27, n° 1, p. 13-24, 2015.

Zeman D., Hajič J., Popel M., Potthast M., Straka M., Ginter F., Nivre J., Petrov S., « CoNLL 2018 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies », *in* D. Zeman, J. Hajič (eds), *CoNLL 2018 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, ACL, Brussels, p. 1-21, 2018.

Zeman D., Popel M., *et al.*, « CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies », *CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, ACL, Vancouver, p. 1-19, 2017.