

## Une structure pour les questions enchainées

Kévin Séjourné

Université de Paris Sud XI, Limsi-CNRS, BP 133, 91403 Orsay Cedex

Bâtiment 508

kevin.sejourne@limsi.fr

**Résumé.** Nous présentons des travaux réalisés dans le domaine des systèmes de questions réponses (SQR) utilisant des questions enchainées. La recherche des documents dans un SQR est perturbée par l'absence d'informations sur la valeur à accorder aux éléments de texte éventuellement utiles à la recherche d'informations qui figurent dans les questions liées. Les récentes campagnes d'évaluation montrent que ce problème est sous-estimé, et n'a pas fait l'oeuvre de technique dédiée. Afin d'améliorer la recherche des documents dans un SQR nous étudions une nouvelle méthode pour organiser les informations liées aux interactions entre questions. Celle-ci se base sur l'exploitation d'une structure de données adaptée à la transmission des informations des questions liées jusqu'au moteur d'interrogation.

**Abstract.** We present works realized in the field of questions answering systems(SQR) using chained questions. The search for documents in a SQR is disrupted by the absence of information on the value to be granted to the possibly useful elements of text in search of information which appear in bound questions. The recent campaigns of evaluation show that this problem is under estimated, and did not make the work of dedicated technique. To improve the search for documents in a SQR we study a new method to organize the information bound to the interactions between questions. This one is based on the operation of a structure of data adapted to the transmission of the information of bound questions up to the search engine.

**Mots-clés :** Question réponse enchainée.

**Keywords:** chained question answering.

### 1 Introduction aux questions enchainées

Les questions enchainées sont des questions destinées à des systèmes de questions réponses classiques, mais qui présentent une difficulté supplémentaire. Chaque question doit être in-

1	Où se trouve la cathédrale Sainte-Sophie en Russie ?
2	Qui était son archiprêtre en 1995 ?
3	Quel Écossais a construit la cathédrale ?
4	Quelle impératrice russe l'a accredité pour la construire ?

TAB. 1 – Exemple d'un groupe de questions enchainées tirées du corpus utilisées pour la campagne d'évaluation CLEF 2007.

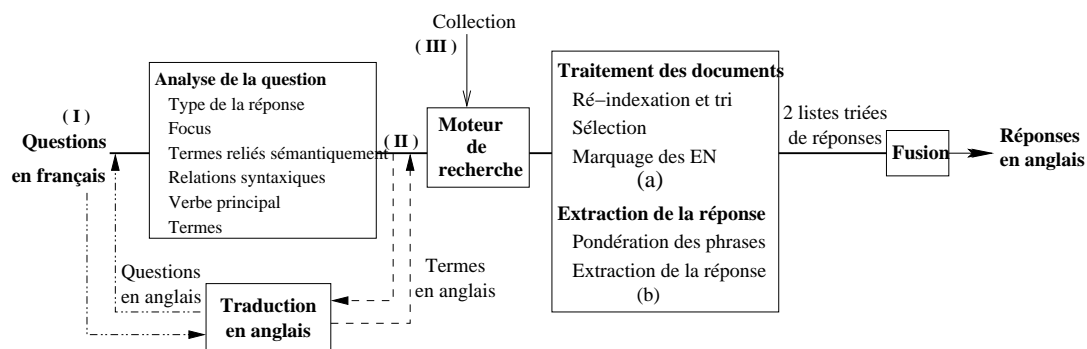


FIG. 1 – Architecture du système Musclef en mode inter-lingue

interprétée en connaissance de l'historique des questions et des réponses précédentes. Il y a eu récemment plusieurs campagnes d'évaluation de systèmes de questions réponses (SQR) où des questions enchainées étaient proposées. Selon les corpus, les questions enchainées peuvent faire référence à un contexte global (ou sujet global) préalablement introduit comme ce fut le cas dans la campagne d'évaluation TREC (Zhou *et al.*, 2006). Elles peuvent aussi faire référence aux réponses précédentes ou avoir de multiples références vers d'autres questions. Les questions enchainées peuvent présenter toutes ces difficultés sans les annoncer explicitement, comme dans la campagne d'évaluation des SQR Clef07 (Penas *et al.*, 2007) ; la première question peut même parfois avoir le rôle d'un introducteur de contexte. Le tableau 1 montre un exemple de groupe de questions enchainées. Parfois les SQR sont inter-lingue, cela signifie que la langue des documents est différente de la langue des questions, comme c'était le cas pour une des pistes de la campagne Clef07. C'est le corpus de cette campagne que nous utilisons par la suite.

Le système Musclef (figure 1) développé au Limsi a globalement une architecture semblable aux SQR classiques. Le problème que nous nous posons est alors de savoir comment organiser les informations des questions précédentes pour que celles-ci puissent être utilisées pour améliorer la recherche de la réponse ?

Dans la suite de notre article, nous proposons les définitions suivantes :

**Terme** : Un terme est un mot ou un groupe de mots considéré lors de l'interprétation de la question et donc jugé pertinent par rapport à des ressources dont nous disposons (documents, dictionnaires, questions précédentes ...). Ainsi les termes de la première question du tableau 1 sont «la cathédrale Sainte-Sophie» et «en Russie».

**Élément utile à la recherche d'information (RI)** : Un élément utile à la RI peut être de plusieurs natures : cela peut être un groupe de termes d'une question, de réponse ou des résultats intermédiaires de calcul d'un SQR. Par exemple, la catégorie de la réponse est un élément. Les éléments de la première question sont les termes «la cathédrale Sainte-Sophie», «en Russie», la catégorie de la question «*lieu*», et la réponse «Novgorod».

**Dépendance d'un couple question/réponse vers un autre** : Une dépendance d'un couple QR vers un autre est un lien hiérarchique lexical/ syntaxique/ sémantique/ pragmatique nécessaire à la bonne résolution d'une question. Les dépendances vont d'une paire question/réponse vers une autre paire question/réponse. La question 2 du tableau 1 dépend de la 1 : Cette dépendance a pour origine le mot «son», qui renvoie à un attribut d'un *objet* précédemment évoqué. Comme il n'y en a pas dans la question 2, il s'agit de la question 1 ; la question 3 dépend de la 1 ; la question 4 dépend de la 3 mais pas directement de la 1, un élément est manquant pour cette résolution. Par contre, les dépendances sont transitives : 4 hérite de la dépendance de 3 vers 1.

**Contexte (d'une question d'un groupe)** : Un contexte est un groupe d'éléments reliés via des dépendances. C'est l'ensemble de tous les éléments utiles à la RI pour la résolution d'une question en tenant compte des questions auxquelles elle est liée. Pour considérer un contexte possible d'une question il faut les éléments de chaque couple question/réponse obtenu en suivant les dépendances de tous les couples question/réponse qui la précèdent dans le groupe.

Notre problème peut alors se reformuler ainsi : Si nous considérons que les données issues des dépendances peuvent améliorer la recherche de la réponse, comment structurer le contexte pour que cette recherche soit plus efficace ? Chiori et Al. (Chiori *et al.*, 2003 4) rencontrent ce problème dans leur système de question réponse interactif écrit/oral, mais lui apportent une solution basée sur une interrogation de l'utilisateur.<sup>1</sup>

Dans la littérature on trouve essentiellement trois approches possibles :

1) Ajouter les éléments de la question liée au texte de la nouvelle question et mettre en entrée d'un SQR classique.

Cela revient à modifier les questions au niveau du point I de la figure 1. Il faut alors disposer d'un système capable d'assurer que la jointure sera réalisée correctement. Il faut aussi pouvoir gérer les différentes formulations et interprétations, et dans tous les cas disposer d'un module de plus pour ne pas oublier le type attendu de la réponse ou d'autres éléments pertinents. Souvent l'ensemble est mal formulé et donc mal analysé. La question «Quelle impératrice russe l'a accredité pour la construire ?» peut se ré-écrire de nombreuses manières.<sup>2</sup>

– Ajout à la fin avec séparation du contexte par un ' ; ' : *Quelle impératrice russe l'a accredité pour la construire ; [un] écossais, Vladimir [of/de] Novgorod, cathédrale Sainte-Sophie Russie ?*

– Unifications syntaxiques des éléments : *Quelle impératrice russe a accredité [[un] écossais|Vladimir [of/de] Novgorod] pour construire la cathédrale [Sainte-Sophie [en Russie]] ?*

Cette approche est limitée à ce que l'on peut re-structurer dans une phrase sans trop perturber le système d'analyse de la question. Pour les systèmes de QR inter-lingues il faudrait aussi prévoir de traduire la réponse.

2) Ajouter les éléments de la question liée directement dans la structure des termes destinés au moteur de recherche.

C'est à dire au niveau du point II de la figure 1. Les termes composant les éléments de la question liée sont alors mêlés à ceux de la nouvelle question (Buscaldi *et al.*, 2007). La structure envoyée au moteur de recherche est une structure qui contient notamment le résultat de la résolution des anaphores. Cependant, cet ajout ne permet pas de faire de distinction entre ce qui vient de la question courante ou de la précédente. Dans le groupe du tableau 1, après la résolution d'anaphores, il n'est plus possible de savoir que l'élément de texte «cathédrale Sainte-Sophie Russie» provient de la première question. Il est possible d'avoir simultanément deux documents dans lesquels on trouve qu'un écossais a construit quelque chose et un autre dans lequel on trouve qu'une personne a construit une cathédrale en Russie. Mais il est possible que l'on ne dispose pas de documents parlant simultanément d'écossais, de cathédrale et de Russie.

Quel poids accorder à ces différents éléments quand on ne sait pas d'où ils viennent ?

<sup>1</sup>La solution examine un score d'ambiguïté structurelle d'une question pour déterminer de nouvelles questions de désambiguïsation. Puis l'utilisateur est censé y répondre correctement pour apporter des informations dont les dépendances inter question-réponse peuvent être correctement résolues.

<sup>2</sup>«Vladimir of Novgorod» est la réponse de Muscief à la question 3 du tableau 1.

Par exemple : dans le contexte de la cathédrale de Novgorod, dans l'ensemble des documents de clef 2007, il était possible de trouver le nom d'une personne ayant construit la cathédrale<sup>3</sup>. Mais rien ne confirme ou n'infirme que cette personne soit de nationalité écossaise<sup>4</sup>.

Ce n'est donc pas satisfaisant, si en plus les groupes de questions forment des séquences où les éléments de texte sont repris implicitement ou avec une variation/oubli partielle, cette approche conduit à une perte de performance proportionnelle à la longueur des séquences de questions.

3) Réaliser une première recherche sur la première question.

Cette première question définit le contexte du groupe. Puis réaliser les recherches suivantes uniquement sur l'ensemble de documents trouvés lors de cette première recherche (Zhou *et al.*, 2006; Hickl *et al.*, 2006). On réduit à chaque question la collection utilisée au niveau du point III de la figure 1. On peut remarquer qu'avec cette stratégie, une question dépendant de  $n$  autres dispose alors d'un ensemble de recherche réduit  $n$  fois. Il n'y a aucun moyen de re-ouvrir l'espace de recherche. Pour des séquences de questions avec des dépendances, cette stratégie n'est pas généralisable. Par exemple, imaginons que dans le groupe présenté en exemple nous ayons déjà analysé les dépendances entre questions et que l'on puisse restreindre l'étude à la séquence de questions 1 3 4. La résolution de la question 4 se fera sur les seuls documents qui parlent de «cathédrale Sainte-Sophie en Russie» et d'«Écossais». Si la réponse est dans un document lié au nom de l'Écossais seul sans sa nationalité, alors la réponse ne sera pas trouvée.

Dans les SQR, les documents sont généralement découpés en paragraphes de taille équivalente avant leur indexation. L'impact de la taille des paragraphes est évidemment très important. Trop gros ils ne restreignent pas suffisamment la recherche, et la seconde question est traitée presque 'hors contexte'. Trop petits ils la restreignent trop, et il n'y plus assez d'information autour de la réponse. Pour trouver une taille de paragraphe adaptée aux questions enchainées, il faudrait aussi que les questions réfèrent toujours à des informations proches dans les paragraphes des documents réponses des premières questions. Il faut aussi que cette proximité puisse être quantifiée afin de définir la taille adaptée des paragraphes.

Dans les 3 approches présentées ci-dessus, les éléments ont la même importance, ce qui pose de problème et ne permet pas une recherche efficace. Aucune de ces stratégies n'est satisfaisante. Plus généralement, la quantité d'éléments pour une recherche unique augmente linéairement avec la profondeur des dépendances entre questions. Le problème se pose pour le choix des éléments : ne prendre que ceux de la question ? ou prendre tous ceux de toutes les questions liées ? C'est un problème de contrôle du bruit par rapport au silence dans le nombre de documents retournés par le moteur de recherche.

Dans un groupe de questions enchainées, les contraintes sur la recherche d'informations sont les dépendances entre questions ainsi que la structuration des ensembles de termes en éléments et contextes. Ainsi, nous voulons proposer une nouvelle structure permettant de lier les contraintes classiques de recherche d'informations (les termes), à des contraintes concernant l'ordre dans lequel ces termes peuvent être relaxés en cas de silence trop important. Nous pouvons donc obtenir des contraintes sur la manière de relaxer d'autres contraintes. L'utilisation de cette structure suppose de concevoir une stratégie de relaxation qui soit adaptée au domaine d'application du SQR. Toute relaxation de contraintes sera alors réalisée en tenant compte des performances de la recherche. Une recherche plus performante pourra être mise en place et une justification des

<sup>3</sup>Vladimir of Novgorod, cf Article wikipedia anglais «Saint\_Sophia\_Cathedral\_in\_Novgorod» et «Church\_of\_Scotland» de la cathédrale sainte-sophie de Novgorod

<sup>4</sup>Une analyse pragmatique approfondie avec un moteur d'inférence aurait peut-être trouvé la nationalité de cette personne.

## Une structure pour les questions enchainées

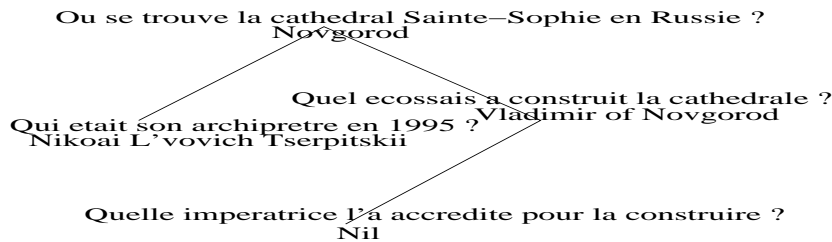


FIG. 2 – L'arbre correspondant au groupe du tableau 1

termes choisis fournie à l'utilisateur. Dans le cas idéal, nous sommes aussi convaincus que les SQR capables de résoudre des questions enchainées de cette manière sont une étape vers des systèmes de dialogue homme-machine en *domaine ouvert*.

Nous voulons donc créer une structure qui permet de représenter les dépendances d'un groupe afin d'améliorer la recherche dans les documents. Nous nous intéressons plus particulièrement à la partie structure et organisation du problème. C'est ce que nous allons présenter maintenant.

## 2 Construction de la structure du contexte

Nous avons voulu construire une structure pour représenter plus finement les dépendances. Nous allons maintenant préciser cette structure et expliquer une méthode de construction. Nous analyserons alors sa performance à trouver ces dépendances. En s'inspirant des travaux sur les structures de dialogue (Vilnat, 2005), de la nature séquentielle des groupes de questions et du partage des termes des questions déjà résolues du groupe, nous choisissons d'organiser la structure du contexte d'un groupe de questions en un arbre (figure 2).

### 2.1 L'organisation

La structure du contexte est un arbre, à sa racine on trouve le contexte commun à toutes les questions. Chaque branche est constituée d'un arbre de paires question/réponse. A chaque noeud sont indiqués la question et son contexte. Ce sont les dépendances qui encodent la structure de l'arbre. Le contexte est composé d'une liste d'éléments faisant éventuellement référence à un noeud réponse.

La structure d'arbre nous permet de représenter efficacement les groupes où les questions ne reprennent que le contexte issu de la première question. L'ajout des éléments d'informations utiles à la recherche d'information à chaque noeud permet une représentation homogène des groupes où les questions réutilisent des contextes liés les uns aux autres. Les questions ne réutilisant pas le contexte des précédentes sont rattachées au noeud **groupe**. Ce noeud **groupe** peut également recevoir des éléments afin de contraindre l'espace de recherche à la manière des évaluations de Trec 2006<sup>5</sup> (Hickl *et al.*, 2006).

<sup>5</sup>Un contexte était donné explicitement pour chaque groupe de questions.

## 2.2 La construction

Nous appelons dépendance unitaire un chemin de longueur 1 dans l'arbre de dépendance. Pour détecter des dépendances, nous combinons un ensemble de critères linguistiques concernant une dépendance unitaire. La combinaison de critères donne un score de fiabilité d'existence (cf 2.2.2) de cette dépendance. Il est possible d'ajouter autant de critères que l'on veut afin d'améliorer la détection. La formalisation en dépendances unitaires nous permet donc d'obtenir une résolution multi-critère qui n'est pas dépendante d'un concept linguistique sous-jacent. Chaque critère retourne un unique score pour un couple de questions. C'est la valeur de ce score qui permet de déterminer l'existence d'une dépendance entre 2 questions.

### 2.2.1 Critères utilisés pour construire les vecteurs de scores

L'algorithme générique de création d'arbre de dépendances tient compte d'un nombre quelconque de critères linguistiques. Nous avons utilisé les 4 critères suivants pour illustrer sa mise en oeuvre et réaliser des tests.

1) Les catégories<sup>6</sup> des questions détectées par Musclef. Si deux questions ont des catégories (Ligozat, 2006) identiques il a beaucoup de chance qu'il n'y ait pas de dépendances de l'une vers l'autre :

*Citer le nom d'un aliment contenu dans le régime alimentaire de base d'Asie du sud-est.*

*Citer le nom d'un aliment contenu dans le régime alimentaire de base d'Europe.*

Les catégories sont identiques, et n'ont donc pas de lien.

2) La présence d'anaphores inter-questions. Un système de résolutions développé au laboratoire<sup>7</sup>, permet d'associer des éléments d'une question à l'autre. Celui-ci utilise essentiellement un dictionnaire pour déterminer le genre et le nombre des éléments et référents pour les comparer. Les référents sont calculés via des critères morpho-syntaxiques inclus dans une mini-grammaire. Une anaphore inter-questions indique une dépendance entre 2 questions.

3) La longueur relative d'une question par rapport à une autre. Une question avec moins de **caractères** réintroduit probablement moins d'éléments et vraisemblablement réutilise plus les éléments déjà introduits. Comme c'est une mesure de l'entropie relative entre 2 questions, seules les différences importantes sont significatives, comme dans l'exemple suivant :

*Quel était le nom de la barge qui a coulé à Porto Rico le 7 janvier 1994 ?*

*Qu'a heurté la barge ?* Les mots, à cause de leur plus faible nombre relativement aux caractères et à cause de leur capacité d'agrégation de sens ne forment pas un bon candidat pour construire un critère de score de dépendance.

4) Les entités nommées identiques d'une question sur l'autre. Ce critère a tendance à montrer que les 2 questions ne sont pas liées, au contraire la répétition partielle d'une EN<sup>8</sup> dans une question indique plutôt une dépendance vers la question dans laquelle elle est complète.

*De quel instrument Swann jouait-il dans le duo 'Flanders et Swann' ?*

*Dans quel pays Swann est-il né ?*

*Dans quelle ville Swann est-il mort ?*

Nous allons maintenant voir comment ces critères sont utilisés.

<sup>6</sup>Dans Musclef chaque question a une catégorie. Celle-ci permet de connaître le type de la réponse attendue.

<sup>7</sup>inspiré des travaux de (Hernandez, 2004)

<sup>8</sup>entité nommée

Num quest	1	2	3	1	2	3
2	[1,1,0,0]	X	X	[7,8,0,0]-> <b>15</b>	X	X
3	[1,1,0,0]	[1,0,0,0]	X	[7,8,0,0]-> <b>15</b>	[7,0,0,0]-> <b>7</b>	X
4	[1,1,0,0]	[1,0,0,0]	[0,2,0,0]	[7,8,0,0]-> <b>15</b>	[7,0,0,0]-> <b>7</b>	[0,16,0,0]-> <b>16</b>

TAB. 2 – Les critères sont [catégorie, lien anaphorique, taille,entité nommée]. Première partie : Les vecteurs de scores pour le groupe en exemple. Seconde partie : Les vecteurs de scores après pondération et projection.

### 2.2.2 Un algorithme à 6 étapes

A chaque combinaison de deux questions d’un groupe est associé un vecteur de scores. Dans l’étape 1 nos vecteurs sont ceux du tableau 2(première partie) pour le groupe en exemple du tableau 1. Le vecteur de la case (1,2)=[1,1,0,0] indique que seules une différence de catégorie et une anaphore ont été trouvées, entre les questions 1 et 2.

La seconde étape consiste à projeter chaque vecteur sur une valeur unique après harmonisation de leurs composantes. La projection se fait selon une somme pondérée par l’efficacité *a priori* de chaque critère linguistique. La normalisation corrige les résultats pour tenir compte des taux d’erreurs différents. La projection est alors une somme pondérée de chaque composante. L’harmonisation utilisée dans notre implémentation est [0.5,0.4,1,1] ; elle est calculée par rapport à la valeur maximum d’un critère. Par exemple, la valeur maximum que donne le critère des catégories<sup>9</sup> est 2. L’harmonisation pour ce critère est donc de «1/2». La pondération utilisée dans notre implémentation est [14,20,5,-5], elle est choisie par rapport à la confiance *a priori* en chaque critère. On obtient alors les données du tableau 2(seconde partie). Le vecteur de la case (1,2) devient donc [0.5x14x1, 0.4x20x1, 1x5x0, 1x-5x0] soit une somme de 15.

A la troisième étape chaque valeur est convertie en une probabilité d’existence d’une dépendance unitaire. Le programme utilise la valeur du score maximum possible du système pour la totalité de ces critères afin de calculer la confiance dans les associations. On en déduit la probabilité d’existence d’une dépendance unitaire. Les valeurs harmonisées-pondérées-extremums sont données par le vecteur [7,24,5,-5]. La valeur maximum pour les calculs de probabilité d’existence est donc  $7x1 + 24x1 + 5x1 + -5x0 = 36$ .<sup>10</sup>

A la quatrième étape on utilise une valeur limite choisie à l’avance à l’aide d’un apprentissage supervisé. Cette valeur permet d’éliminer les combinaisons à scores trop faibles (dans notre exemple il s’agit des scores qui lui sont inférieurs). Cette valeur limite est calculée à partir des résultats de l’étape 3 et des dépendances encodées sous la forme ‘1 : existe ’ et ‘0 : existe pas’ préalablement annotée. Les combinaisons ‘2-3’ et ‘2-4’ sont oubliées dans l’exemple. Dans notre implémentation, les 122 questions situées en rang 2 et plus d’un groupe ont reçu un score qui a été comparé aux 92 dépendances annotées (cf 2.2.3). Nous avons alors créé une table qui indique pour chaque score l’existence ou non d’une dépendance. Le système weka (Frank *et al.*, 2005) nous a permis de déployer facilement un outil de classification statistique nous permettant alors d’évaluer une valeur limite.

La cinquième étape consiste à comparer les combinaisons restantes. Si deux combinaisons ont la même question pour origine, celle de plus faible score est éliminée. Ceci garantit qu’une question ne réutilise le contexte que d’une seule autre question qui éventuellement aura construit

<sup>9</sup>Les catégories sont comparées sur deux niveaux de similarités : «définition-personne» «définition-autre»

<sup>10</sup>Les projections dont la valeur est négative sont supprimées, il n’y a donc pas de *probabilité négative*.

son contexte sur celui d'une précédente. Cette étape est réalisée en utilisant le maximum de chaque ligne. On obtient donc '1-2', '1-3' et '3-4'.

La sixième étape construit l'arbre après un tri topologique des dépendances unitaires. Le tri topologique donne 2 ordres valides (et donc deux arbres).



Le premier est toujours choisi, car il respecte l'ordre d'introduction des questions. Pour un groupe de questions donné, la structure arborée n'est pas forcément unique. Si une question '1' introduit les éléments A et B. Si une question '2' réutilise uniquement l'élément A et que la question '3' réutilise les éléments A et B. Alors, il est possible de rattacher la question '3' soit à la question '1' (Stratégie de la première introduction d'un élément), soit à la question '2' (Stratégie de la définition la plus récente d'un élément). Si on ne dispose pas de critères sémantiques/pragmatiques, l'élément peut être décrit par les mêmes mots, mais correspondre à des sens différents. Étant donné le type des questions posées il peut être intéressant de privilégier la stratégie de la première introduction, alors que sur un système de dialogue on préférera peut-être la dernière afin d'avoir une représentation plus concise ne demandant pas de mémorisation des événements anciens.

On peut alors décorer l'arbre avec les éléments, contextes, réponses et vecteur de scores. Les éléments de texte utiles sont repérés via un mécanisme basé sur la morpho-syntaxe. Ils sont constitués des noms, adjectifs, nombres et déterminants les entourant. L'analyse des questions de Musclef permet de s'assurer que les constituants formant le *focus* (Ferret *et al.*, 2001)<sup>11</sup> de la question sont présents dans les éléments reconstruits. De même, les entités nommées sont arbitrairement ajoutées dans la liste des éléments. Chaque élément produit est un sous ensemble contigu de la question et tous les sous éléments contigus sont fusionnés. Une fois l'arbre de dépendances construit, chaque élément de texte d'une question est traité exactement de la même manière que dans un SQR normal. On utilise juste une numérotation des termes des éléments afin de les suivre dans les étapes de sélection de mots, traduction... Les termes des éléments sont récupérés juste avant leur utilisation dans le moteur de recherche. Le vecteur de scores est ajouté afin de caractériser le type de dépendance entre les questions. Les réponses aux questions sont ajoutées ainsi que leur type (type de la réponse attendue de la question précédente). La figure 3 montre cette structure pour le groupe de 4 questions de l'exemple en tableau 1.

À l'aide de cette organisation, on peut concevoir une nouvelle interrogation des documents tenant compte de l'origine des éléments.

### 2.2.3 Résultats

Le corpus de questions de la campagne Clef2007 a été annoté à la main. Nous avons annoté un total de 92 dépendances unitaires<sup>12</sup>. Nous avons évalué notre approche sur ce corpus en comparant les dépendances unitaires annotées à celles trouvées automatiquement.

<sup>11</sup>Le focus d'une question : une phrase nominale qui a des chances d'être présente dans la réponse : Qui est le ministre ? «Le ministre» est ...

<sup>12</sup>Nous pouvons remarquer ici une différence avec l'étude réalisée par Dominique Laurent et Al. (Laurent *et al.*, 2007) qui compte 76 questions disposant d'un lien anaphorique. Comme mentionné plus haut nous ne nous arrêtons pas aux anaphores coréférentes.



```

| Groupe 97-100 (Corpus clef 07 FR-EN)
| contexte :
|
|--. Numéro : 97
|   | Question : Où se trouve la cathédrale Sainte-Sophie en Russie ?
|   | Réponse : Novgorod
|   | #éléments réutilisables : 0
|   | contexte :
|   |
|   |--. Numéro : 98
|   | | Question : Qui était son archiprêtre en 1995 ?
|   | | Réponse : Nikoai L'vovich Tserpitskii
|   | | #éléments réutilisables : 1
|   | | contexte : «la cathédrale Sainte-Sophie en Russie»
|   | | scores : [7,8,0,0]
|   |
|   |--. Numéro : 99
|   | | Question : Quel écossais a construit la cathédrale ?
|   | | Réponse : Vladimir of Novgorod
|   | | #éléments réutilisables : 1
|   | | contexte : «sainte-Sophie en Russie»
|   | | scores : [7,8,0,0]
|   |
|   |--. Numéro : 100
|   | | Question : Quelle impératrice l'a accredité pour la construire ?
|   | | Réponse : nil
|   | | #éléments réutilisables : 2
|   | | contexte : «écossais» «Réponse(99)»
|   | |                 «la cathédrale sainte-Sophie en Russie»
|   | | scores : [0,16,0,0]

```

FIG. 3 – La structure d’arbre pour un groupe de questions enchainées.

Résultat/Critère	catégories	référents	tailles	entités	réf..+tailles	cat..+tailles	cat..+réf	Les 4 ensemble
Rappel	0.63	0.66	0.73	0.15	0.65	0.631	0.614	0.65
Précision	0.68	0.70	0.15	0.04	0.728	0.706	0.814	0.80
F-Mesure	0.65	0.684	0.25	0.06	0.687	0.67	0.70	0.72

TAB. 3 – Les performances de découverte automatique de dépendances unitaires.

Les 4 critères utilisés simultanément permettent une détection des dépendances unitaires dont les performances sont récapitulées dans le tableau 3, pour la détection des 92 dépendances du corpus. Des études individuelles de chaque critère montrent qu’ils améliorent tous les résultats. Les deux plus efficaces sont la détection des anaphores et les différences de catégories. Cependant, les 2 autres critères peuvent être affinés et d’autres pourraient être envisagés, incluant des éléments syntaxiques et/ou sémantiques.

En première approche pour l’utilisation des informations calculées précédemment, nous avons ajouté les termes du contexte à la liste des termes des questions ayant des dépendances, sans contrôler un seul des problèmes qui motive la construction de la structure. Nous avons pu observer que dans 48 cas nous obtenons des phrases candidates là où sans la méthode de sélection nous n’en obtenons aucune. Nous pouvons aussi observer que le déploiement de la méthode proposée est robuste à la traduction des termes dans les systèmes inter-lingue. Ceci nous incite à penser que nos résultats de sélection des dépendances sont très corrects.

### 3 Conclusion

Notre travail permet d'établir des dépendances entre questions. Dans le cas général, ces dépendances sont problématiques. Nous obtenons donc ainsi des résultats encourageants qui montrent des progrès dans l'organisation des dépendances liant les éléments pour l'interrogation. La prochaine étape de ce travail consiste à modifier l'interrogation des documents afin d'utiliser au mieux cette structure d'arbre, dans le but d'améliorer les résultats du SQR sur des questions enchainées.

Des évolutions sont possibles : on pourrait prolonger cette structure jusque dans le système de sélection des réponses du SQR. Il serait alors intéressant d'étudier comment la sélection des réponses (et l'analyse des questions) pourrait s'appuyer sur cette structure, et comment fournir des éléments visant à faciliter la recherche de la réponse de la question suivante grâce à cela.

### Références

- BUSCALDI D., ANND PAOLO ROSSO Y. B. & SANCHIS E. (2007). The upv at qa@clef 2007. *Universidad Politcnica de Valencia*.
- CHIORI H., TAKAAKI H., HIDEAKI I., EISAKU M. & ANDFURUI SADAOKI K. S. (2003-4). Study on spoken interactive open domain question answering. *Spontaneous Speech Processing and Recognition (SSPR)*, p. 111–114.
- FERRET O., GRAU B., HURAUULT-PLANTET M., MONCEAUX L., ROBBA I. & VILNAT A. (2001). Finding an answer based on the recognition of the question focus. *Text retrieval conference, TREC 10*.
- FRANK E., HALL M. A., HOLMES G., KIRKBY R., PFAHRINGER B., WITTEN I. H. & TRIGG L. (2005). Weka - a machine learning workbench for data mining. In O. MAIMON & L. ROKACH, Eds., *The Data Mining and Knowledge Discovery Handbook*, p. 1305–1314. Springer.
- HERNANDEZ N. (2004). *Description et détection automatique de structures de texte*. PhD thesis, University de Paris-Sud XI LIMSI/CNRS. <http://www.limsi.fr/Individu/hernandez/research/Hernandez-these.tar.gz>.
- HICKL A., WILLIAMS J., BENSLEY J., ROBERTS K., SHI Y. & RINK B. (2006). Question answering with lcc's chaucer at trec 2006. *15th Text REtrieval Conference, Gaithersburg*, p. 1.
- LAURENT D., SÉGUÉLA P. & NÈGRE S. (2007). Cross lingual question answering using qristal for clef 2007. *Synapse Développement*.
- LIGOZAT A.-L. (2006). *Exploitation et fusion de connaissances locales pour la recherche d'informations précises*. PhD thesis, Université de Paris-Sud XI LIMSI/CNRS. .
- PENAS A., FORNER P. & GIAMPICCOLO D. (2007). Guidelines for participants in qa at clef 2007. *CELCT, Trento(IT) and UNED, Madrid*, p. 1.
- VILNAT A. (2005). *Habilitation à diriger les recherches : Dialogue et analyse de phrases*. PhD thesis, University de Paris-Sud XI LIMSI/CNRS. <http://www.limsi.fr/Individu/anne/HDR/MemoireHDR.pdf>.
- ZHOU Y., YUAN X., CAO J., HUANG X. & WU L. (2006). Fduqa on trec2006 qa track. *15th Text REtrieval Conference, Gaithersburg*, p. 1026–1033.