**Article :** Résumé automatique multi-document et indépendance de la

langue : une première évaluation en français

Cet article porte sur le résumé automatique multi-document, il présente quelques méthode pour des résumé en français, mais les méthodes ne sont pas liées à une langue particulière.

**Contexte et Problématique**

Le résumé automatique n’est pas évident, ici on propose une approche originale basée sur le rapprochement de segments morphologiquement proche. Ce sont les premiers à se pencher sur ce genre de problématique sur des corpus en français.

Le but à atteindre est assez clair : reformuler un texte dans un espace plus réduit (souvent on fixe un nombre limite de caractère à ne pas dépasser.

Pourquoi ?

Aujourd’hui le nombre de document disponible est gigantesque, et il difficile, suite à des limitations purement temporelles, de savoir quelles informations extraire de quels documents, le résumé automatique offre donc un gain de temps.

Aujourd’hui deux méthodes, les méthodes extractives et abastractives, les méthodes abstractives n’étant pas encore assez robustes, les auteurs se penchent sur une méthode extractive, on décompose les textes en segment, souvent des phrases, et on extrait les plus pertinentes.

Ce qui marche le mieux aujourd’hui est le résumé mono-document, même si cela dépend beaucoup du type de document, un article de journal sera bien plus simple à résumé qu’une oeuvre littéraire par exemple.

Pour un corpus de plusieurs documents les difficultés sont les suivantes : redondance des segments extraits, temporalité des segments, contradiction dans les segments.

Le cadre : Ici on traitera un corpus d’article de journaux en français.

**Etat de l’art**

Dès 1958 on commence à s’intéresser à des méthodes numériques pour la production d’extraits à l’aide la distribution de la fréquence des mots (Luhn, 1958), la problématique de la quantité d’informations disponible étant déjà présente. Ensuite en 1959, Edmunson propose des méthodes reposant sur la structure interne du document et sur la présence de mots clés, ce fut une énorme avancée et leur travaux sont toujours utilisés. Dans leurs approches de 1997 et 2002 Teufel & Hirao mettent en place des algorithmes supervisés, ils obtiennent de bon résultats mais leur approche est trop dépendante de la langue, ce qui est un problème.

En 2004 Mihalcea tente de résoudre ce problème en proposant une approche à base de graphe (méthode utilisé dans ce papier), son idée est de selectionner les élement.s les plus prestigieux du graphe des segment, chaque segment étant une phrase du corpus représenté par un vecteur grace au modèle des sac de mots. Ces méthodes fonctionnent mais sont très dépendante de la construction initiale du graphe.

**Méthodes**

*Toute les formules sont dans l’article*

On commence par construire le graphe après avoir fait un minimum de préprocessing sur les phrases : on enlèves les stopwords, les majuscules et la ponctuation (trois processus simples)

Ensuite on calcule la similarité à l’aide de la mesure LCS, la plus grande chaine de caractère en commun entre deux segments.

La dernière partie est variable, c’est la pondération des segments, ici ils présentent trois méthodes :

Popularité : méthode naive, une phrase est populaire si elle a beaucoup de liens dans le graphe

LexRank : basée sur le pageRank adaptée au texte, la grosse différence avec le pageRank habituel étant que le graphe n’est pas dirigé, les liens étant basés sur la similarité entre deux segments. Cette méthode utilise aussi la popularité calculée dans la première méthode.

TextRank : Variante de la méthode précédente où l’on utilise pas la popularité

Après avoir selectionnées les phrases on les assembles, on applique un seuil de similarité afin d’éviter la redondance et on positionne les phrases en fonction de la chronologie des documents.

**Résultats**

En testant toutes ces méthodes c’est les méthodes à base de graphes qui sont les plus efficaces, ils ont calculé les scores à l’aide de la méthode ROUGE (Recall-Oriented Understudy for Gisting Evaluation).

Et entre LexRank et TextRank c’est TextRank qui prime

**Pistes & Discussion**

On pourra peut être orienté la recherch vers des graphes orientés (ils ne le sont pas ici)