Résumé automatique de texte par extraction

IFT6010

Lara Haidar-Ahmad

Résumé par extraction

- **Automatic summarization** is the process of reducing a text document with a computer program in order to create a summary that retains the most important points of the original document. As the problem of information overload has grown, and as the quantity of data has increased, so has interest in automatic summarization. Technologies that can make a coherent summary take into account variables such as length, writing style and syntax. An example of the use of summarization technology is search engines such as Google. Document summarization is another.
- Generally, there are two approaches to automatic summarization: *extraction* and *abstraction*. Extractive methods work by selecting a subset of existing words, phrases, or sentences in the original text to form the summary. In contrast, abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what a human might generate. Such a summary might contain words not explicitly present in the original. Research into abstractive methods is an increasingly important and active research area, however due to complexity constraints, research to date has focused primarily on extractive methods.

https://en.wikipedia.org/wiki/Automatic_summarization

Résumé par extraction

- **Automatic summarization** is the process of reducing a text document with a computer program in order to create a summary that retains the most important points of the original document. As the problem of information overload has grown, and as the quantity of data has increased, so has interest in automatic summarization. Technologies that can make a coherent summary take into account variables such as length, writing style and syntax. An example of the use of summarization technology is search engines such as Google. Document summarization is another.
- Generally, there are two approaches to automatic summarization: *extraction* and *abstraction*. Extractive methods work by selecting a subset of existing words, phrases, or sentences in the original text to form the summary. In contrast, abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what a human might generate. Such a summary might contain words not explicitly present in the original. Research into abstractive methods is an increasingly important and active research area, however due to complexity constraints, research to date has focused primarily on extractive methods.

https://en.wikipedia.org/wiki/Automatic_summarization

Résumé par extraction

- **Automatic summarization** is the process of reducing a text document with a computer program in order to create a summary that retains the most important points of the original document.
- Generally, there are two approaches to automatic summarization: *extraction* and *abstraction*. Extractive methods work by selecting a subset of existing words, phrases, or sentences in the original text to form the summary. In contrast, abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what a human might generate.

https://en.wikipedia.org/wiki/Automatic_summarization

LexRank: Graph-based Lexical Centrality as Salience in Text Summarization

Güneş Erkan GERKAN@UMICH.EDU
Department of EECS
University of Michigan, Ann Arbor, MI 48109 USA

Dragomir R. Radev RADEV@UMICH.EDU
School of Information & Department of EECS
University of Michigan, Ann Arbor, MI 48109 USA

Résumé par extraction : choix des phrases

- Première approche : Centroide
- Deuxième approche : Centralité
- Méthode Lexrank

Prérequis : IDF

- IDF = inverse document frequency

$$\mathrm{idf}_i = \log\left(\frac{N}{n_i}\right)$$

N nombre de document,

ni le nombre de document ou le mot i apparait au moins une fois

- Fréquence d'un mot dans la langue

Centroide

- Les mots les plus rares dans la langue, qui apparaissent le plus souvent dans le texte, sont des mots spécifiques au sujet

- Pour chaque mot : tf_i x idf_i

- Pour chaque phrase :$\Sigma(tf_i \times idf_i)$

Centralité

- On se base sur les similarités entre chaque paire de phrases
- Une phrase ayant beaucoup de similarités avec d'autres phrases est dite "centrale"
- Pour chaque paire de phrase:

$$\text{idf-modified-cosine}(x, y) = \frac{\sum_{w \in x,y} \text{tf}_{w,x} \text{tf}_{w,y} (\text{idf}_w)^2}{\sqrt{\sum_{x_i \in x} (\text{tf}_{x_i,x} \text{idf}_{x_i})^2} \times \sqrt{\sum_{y_i \in y} (\text{tf}_{y_i,y} \text{idf}_{y_i})^2}}$$

- Le Degree Centrality d'une phrase est le nombre de phrase qui lui sont similaire

LexRank

- Extension de la méthode basée sur la centralité
- Certaines phrases peuvent être problématiques et augmenter le score des autres phrases
- On veut donner un poids aux votes selon leurs provenances
- Graphe dont les nœuds sont les phrases et les liens sont les similarités entre les phrases

$$p(u) = \sum_{v \in adj[u]} \frac{p(v)}{deg(v)}$$

- Sous forme matricielle : $\mathbf{p} = \mathbf{B}^{\mathrm{T}}\mathbf{p}$
- Trouver p revient à trouver la distribution stationnaire de probabilité p

Résultats

	2003 Task2		
	min	max	average
Centroid	0.3523	0.3713	0.3624
Degree (t=0.1)	0.3566	0.3640	0.3595
LexRank (t=0.1)	0.3610	0.3726	0.3666

(a)

	2004 Task2		
	min	max	average
Centroid	0.3580	0.3767	0.3670
Degree (t=0.1)	0.3590	0.3830	0.3707
LexRank (t=0.1)	0.3646	0.3808	0.3736

(b)

	2004 Task4a		
	min	max	average
Centroid	0.3768	0.3901	0.3826
Degree (t=0.1)	0.3863	0.4027	0.3928
LexRank (t=0.1)	0.3931	0.4038	0.3974

(c)

	2004 Task4b		
	min	max	average
Centroid	0.3760	0.3962	0.4034
Degree (t=0.1)	0.3801	0.4147	0.4026
LexRank (t=0.1)	0.3837	0.4167	0.4052

(d)

Table 3: ROUGE-1 scores for different MEAD policies on DUC 2003 and 2004 data.

Conclusion

- Une approche prometteuse

- En général, la méthode LexRank fonctionne mieux que les autres