

Algorithme de recherche par lieux

L'utilisateur de l'application Savoirs peut choisir d'orienter les suggestions par lieux. Un article contient des lieux à deux endroits différents :

- * dans le header de l'article qui contient les lieux les plus importants de l'article.
- * dans le corps du texte par le biais d'un taguage préalable.

Pour préserver l'importance des lieux dans le header sans oublier ceux dans le corps de texte, il faut appliquer une pondération

Pondération :

lieux du corpus	Pondération de 1 multiplié par le nombre d'occurrence de ce lieu
lieux du header	Pondération la plus élevée des lieux du corpus incrémenté de 1

Requête :

La requête doit être géographique, si l'utilisateur choisit d'orienter sur Paris, les suggestions doivent fournir :

- les textes traitant de Paris dans l'ordre décroissant des pondérations
- des textes géographiquement les plus proches de Paris

Algorithme de recherche par dates

L'utilisateur de l'application Savoirs peut choisir d'orienter les suggestions par dates. Un article contient des dates à deux endroits différents :

- * dans le header de l'article qui contient les dates les plus importantes de l'article.
- * dans le corps du texte par le biais d'un taguage préalable.

Pour préserver l'importance des dates dans le header sans oublier ceux dans le corpus de texte, il faut appliquer une pondération

Pondération :

dates du corpus	Pondération de 1 multiplié par le nombre d'occurrence de ce lieu
périodes du corpus	Pondération de 1 multiplié par le nombre d'occurrence de ce lieu puis divisé par la longueur de la période (voir exemple)
dates du header	Pondération la plus élevée des dates du corpus incrémenté de 1
périodes du header	Pondération la plus élevée des dates du corpus incrémenté de 1 puis divisé par la longueur de la période (voir exemple)

exemple de pondération sur une période :

$\begin{matrix} 1600 & 1601 & 1602 & 1698 & 1699 \\ \text{XVII}^{\text{e}} \text{ s.} = (& 1/100 & 1/100 & 1/100 \dots 1/100 & 1/100) \end{matrix}$

$\begin{matrix} & & 1939 & 1945 \\ \text{Seconde guerre mondiale} = (& 1/7 & \dots & 1/7) \end{matrix}$

Requête :

La requête est simple, la pondération à lieu à l'enregistrement. Lors de la requête, l'utilisateur renseigne une date ou une période et Soir retourne les articles ordonnés de manière décroissante par le poids renseigné.

Il n'y a pour le moment pas de recherche de proximité de prévu.

Algorithme de recherche par concepts

L'utilisateur de l'application Savoirs peut choisir d'orienter les suggestions par concept. Un article contient des concepts à deux endroits différents :

- * dans le header de l'article qui contient les concepts les plus importants de l'article.
- * dans le corps du texte par le biais d'un taguage préalable.

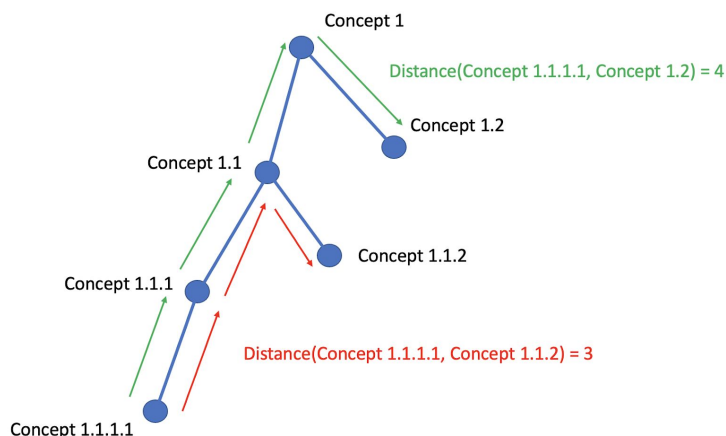
Pour préserver l'importance des concepts dans le header sans oublier ceux dans le corps de texte, il faut appliquer une pondération

Pondération :

concepts du corpus	Pondération de 1 multiplié par le nombre d'occurrence de ce concept
concepts du header	Pondération la plus élevée des concepts du corpus incrémenté de 1

Tableau des distances :

Pour pouvoir réaliser il faut connaître la distance entre les concepts. La distance entre chaque concept est calculé en fonction du parcours de l'arbre des concepts comme expliqué dans ce schéma :



Il pourra être imaginé un poids différent si on remonte l'arbre ou si on le descend, ainsi que si on change de branche.

Requête :

Pour effectuer la requête on va prendre le concept choisi par l'utilisateur, aller chercher tous les concepts à une distance de 1 avec un poids fort, puis tous les concepts avec une distance de 2 avec un poids inférieur au précédent et ainsi de suite.

La valeur des poids sera calculé en inverse du nombre d'itération réalisé pour rechercher les concepts. Exemple si on fait une requête avec une profondeur de trois concepts : le concept à une distance de 1 aura un poids de trois, le concept à une distance de 2 aura un poids de deux et le concept de à une distance de 3 aura un poids de 1.