

Un outil d'exploration pour le Défi EGC 2016

Olivier Parisot, Yoanne Didry, Thomas Tamisier

Luxembourg Institute of Science and Technology (LIST), Belvaux, Luxembourg
olivier.parisot@list.lu, <http://www.list.lu>

Résumé. Dans le cadre du défi EGC 2016, nous avons développé une application web pour explorer les données décrivant les articles publiés depuis 2004 lors des conférences EGC. L'outil permet de découvrir les thèmes importants qui ont été abordés dans ces papiers. De plus, il permet de déterminer automatiquement les articles sémantiquement similaires à des thèmes donnés.

1 Introduction

Dans le cadre du défi EGC, l'association EGC a mis à disposition les informations concernant les articles présentés lors des conférences entre 2004 et 2015¹. Ces données sont fournies via un fichier CSV dans lequel les caractéristiques de chaque article sont détaillées : *a)* L'année de publication. *b)* Le titre. *c)* Les auteurs. *d)* Le résumé. *e)* La première page de l'article.

Afin d'extraire des renseignements utiles sur les articles parus dans la conférence EGC, nous avons développé une plateforme permettant de charger ce fichier afin de l'analyser. La première partie de l'article décrit l'architecture et les technologies utilisées au sein de la plateforme, tandis que la seconde détaille le processus d'extraction des données. La dernière section donne un bref aperçu des différentes fonctionnalités du prototype.

2 Prototype

La plateforme développée est une application multi-tiers : les traitements des données et les calculs sont réalisés côté serveur ; côté client, les résultats sont présentés à l'utilisateur dans une interface web. De cette manière, les calculs lourds peuvent être réalisés sur une machine de puissance raisonnable, tandis que l'interface de l'outil est accessible via un simple navigateur.

Du point de vue technique, le développement a été réalisé avec Grails, un framework permettant de développer des applications client/serveur performantes (Ledbrook et Smith (2014)) : Java/Groovy pour la couche applicative, HTML5/Javascript pour l'interface web.

Des bibliothèques JAVA et des API externes ont été utilisées dans la plateforme (Table 1). Ces composants ont été intégrés de différentes manières : *a)* Les bibliothèques JAVA et open source ont été intégrées directement dans l'application. *b)* Les APIs externes sont invoquées lors de l'exécution en effectuant des appels distants (une connexion réseau est donc nécessaire).

1. http://editions-rnti.fr/files/RNTI_articles_export.txt.zip

TAB. 1 – Composants externes intégrés dans la plateforme.

Partie	Nom	Type	Usage
Serveur	OpenCSV	librairie JAVA	Lecture de fichier
Serveur	Apache Tika	librairie JAVA	Détection de langage
Serveur	Apache OpenNLP	librairie JAVA	Traitement du langage
Serveur	AlchemyAPI	API distante	Extraction de concepts
Serveur	DISCO	librairie JAVA	Analyse de similarité
Serveur	MALLET	librairie JAVA	<i>Topic modeling</i>
Client	jquery & plugins	librairie JAVASCRIPT	Interactivité & Nuages de points
Client	Highchart	API distante	Visualisation de données

Afin de bien utiliser ces composants, la plateforme a été développée en s'inspirant des architectures micro-services (Namiot et Sneps-Snepe (2014)). En pratique, ce type d'architecture permet (entre autres) d'intégrer intelligemment des APIs distantes, en gérant notamment les problèmes d'accès et la mise en cache des résultats.

3 Traitement des données

La plateforme est constituée de plusieurs modules permettant de réaliser le processus classique de traitement de données : *a)* le chargement des données. *b)* le nettoyage/filtrage des données. *c)* l'analyse des données. *d)* la visualisation des résultats obtenus.

Le chargement des données est réalisé simplement en utilisant la librairie OpenCSV². En complément, nous avons créé et traité des fichiers *cfp2016_fr.txt* et *cfp2016_en.txt* contenant les détails des appels à contribution de la conférence EGC pour chaque langue.

Un nettoyage/filtrage a ensuite été réalisé : les articles pour lesquels certains champs sont incomplets ou manquants ont été ignorés. Ainsi, 742 articles ont été considérés.

L'analyse de chacun de ces articles a ensuite été réalisée en plusieurs étapes :

1) La langue dans laquelle est écrite l'article est automatiquement détectée par Apache Tika, via une heuristique de classification basée sur des corpus gérés par la librairie (Mattmann et Zitting (2011)). En effet, les actes de la conférence EGC contiennent des articles rédigés en différentes langues : en pratique, l'outil en a retenu 687 rédigés en français et 55 en anglais.

2) Les champs textuels (titres et résumés) sont traités de manière à filtrer les mots usuels en se basant sur une extension personnalisée de la liste utilisée par *Snowball*³.

3) Les mots mal orthographiés ou avec les mots collés suite à un espace manquant sont ignorés (ex : '*LeWeb*' ou '*quipermet*'). Ceci est réalisé grâce à DISCO, une librairie d'analyse de données textuelles basée sur des corpus extraits de *wikipedia* puis pré-traités (Kolb (2008)).

4) Pour chaque article, les principaux concepts sont extraits à partir du titre et du résumé via AlchemyAPI, un service en ligne de traitement automatique du langage⁴.

5) Pour l'ensemble des articles, les thèmes importants sont identifiés via MALLET, une librairie de *topic modeling* (McCallum (2002)).

2. <http://opencsv.sourceforge.net>

3. <http://snowball.tartarus.org/algorithms/french/stop.txt>

4. <http://www.alchemyapi.com>

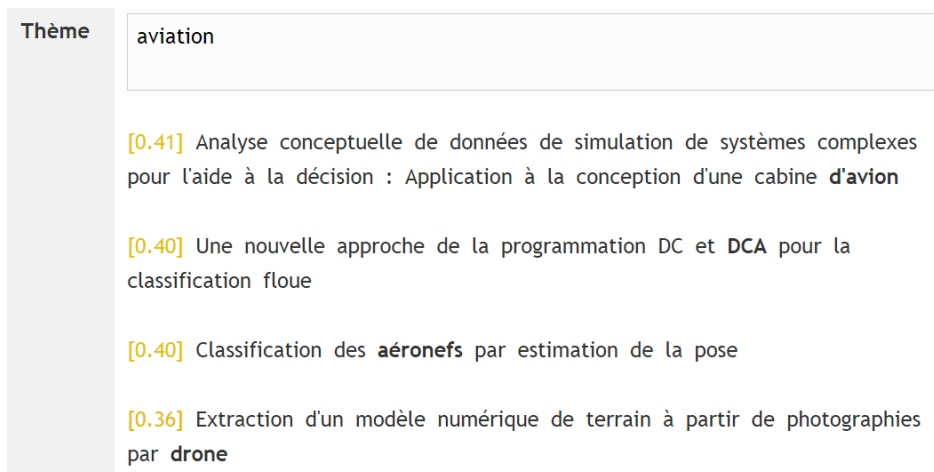
Dans le nuage de mots créé à partir de l'appel à contribution, nous constatons sans surprise que les termes *donnée*, *connaissance*, *extraction*, *fouille* et *gestion* sont les plus mentionnés. Dans le nuage de mots créé à partir des résumés des articles écrits en français (Figure 1), nous retrouvons également sans surprise les mêmes thèmes importants.

En combinant ces deux nuages de mots, et en utilisant un dégradé de couleur pertinent (Harrower et Brewer (2003)), nous pouvons observer les termes : *a*) souvent présents dans les résumés mais peu dans l'appel à contribution. *b*) présents dans l'appel à contribution et souvent dans les résumés. *c*) présents dans l'appel à contribution mais absents dans les résumés. Cela nous aide à voir que des techniques relatives aux *graphes*, *règles*, *ontologie* sont souvent utilisées dans les articles. De plus, nous voyons que la conférence vise explicitement des papiers en rapport avec des domaines d'application concrets. D'après notre outil, certains sont souvent abordés (*santé*) alors que d'autres sont peu ou pas traités (*défense*, *économie*, *éducation*).

Pour finir, l'utilisation de AlchemyAPI et de MALLET permet de compléter ces résultats par une approche non supervisée en détectant automatiquement les thèmes importants comme *classification*, *visualisation*, *graphes* ou *ontologie*.

5 Recherche des articles par thème

L'interface permet à l'utilisateur de sélectionner un thème et/ou une phrase, et l'outil fournit une liste des articles en rapport avec ce qui a été saisi. Par exemple, la saisie du mot *aviation* permet d'obtenir une liste d'articles en rapport avec ce thème, notamment les articles parlant d'*avion*, d'*aéronef* et de *drone* (Figure 2).



The screenshot shows a search interface with a 'Thème' label and a text input field containing 'aviation'. Below the input field, there is a list of search results, each starting with a score in brackets and followed by a description of an article. The words 'avion', 'aéronefs', and 'drone' are bolded in the descriptions.

Thème	aviation
[0.41]	Analyse conceptuelle de données de simulation de systèmes complexes pour l'aide à la décision : Application à la conception d'une cabine d'avion
[0.40]	Une nouvelle approche de la programmation DC et DCA pour la classification floue
[0.40]	Classification des aéronefs par estimation de la pose
[0.36]	Extraction d'un modèle numérique de terrain à partir de photographies par drone

FIG. 2 – Recherche d'articles par thème, en utilisant le calcul de similarité entre phrases : exemple des résultats obtenus en cherchant les articles écrits en français et relatifs à l'aviation. Les mots en gras sont ceux qui ont le plus d'importance dans le calcul de la similarité.

La liste des articles est obtenue en calculant avec DISCO la similarité entre le thème sélectionné par l'utilisateur et les titres et/ou résumés des articles (le thème peut être constitué de

plusieurs mots). Les articles sont ensuite triés en fonction de cette similarité, et les meilleurs résultats sont proposés à l'utilisateur. Afin d'améliorer le processus, nous avons également filtré les mots importants dans l'appel à contribution de la conférence EGC, de manière à ignorer les termes très fréquents (*données, fouille, etc.*).

Pour explorer visuellement les résultats du calcul de similarité sémantique, une projection MDS peut être utile (Kruskal et Wish (1978)), et l'idée de projeter des mots/textes/documents a déjà été traitée dans le passé (notamment dans Blanco et Martín-Merino (2007)). Nous avons donc créé un module dans le prototype qui génère une telle projection, notamment en transformant la similarité en distance ($distance = 1 - similarité$). De plus, en colorant chaque point en fonction de la similarité de l'article considéré par rapport à un thème donné, nous pouvons mettre en évidence les thèmes et/ou techniques souvent abordées durant les conférences.

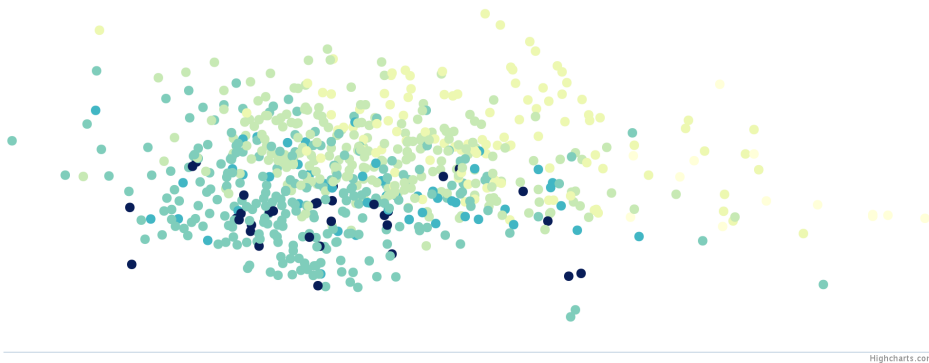


FIG. 3 – Projection MDS réalisée avec les articles rédigés en français. La distance entre les points représente la distance sémantique entre les titres, calculée en adaptant la similarité obtenue avec DISCO. Dans cette projection, la couleur de chaque point représente la similarité de l'article avec le thème 'visualisation' (dégradé YlGnBu – Harrower et Brewer (2003)).

Par exemple, une projection MDS permet de voir que de nombreux papiers présentés ont un lien plus ou moins direct avec la *visualisation* (Figure 3). Dans le cas de notre exemple précédent (les articles en rapport avec l'*aviation*), une vue MDS permet d'observer que ces derniers ne se trouvent pas proches dans la projection car ils traitent d'aspects techniques qui les différencient (*ontologie, règles, etc.*).

Pour conclure, l'outil permet donc à l'utilisateur de trouver les articles en rapport avec les sujets qui l'intéressent (selon un type de techniques et/ou un domaine d'applications).

6 Conclusion

Le prototype développé dans le cadre du défi EGC 2016 permet d'avoir une vue d'ensemble des thématiques abordées dans le cadre de la conférence EGC, entre 2004 et 2015. L'utilisation des nuages de points et des projections comme MDS permettent d'explorer les données, tandis que l'utilisation de DISCO permet d'obtenir une liste d'articles similaires à un thème prédéfini.

7 Remerciements

Le travail a été réalisé en partenariat avec la société infinAI Solutions S.A. (⁵), et nous remercions Gero Vierke et Helmut Rieder pour leur aide. Dans ce cadre, une partie de la plateforme présentée dans cet article a été utilisée pour améliorer le processus de description des produits distribués sur des sites de vente en ligne comme Amazon (Parisot et al. (2014)).

Par ailleurs, le projet a été financé par le Ministère de l'Economie et du Commerce Extérieur du Luxembourg (Loi RDI).

Références

- Blanco, Á. et M. Martín-Merino (2007). A partially supervised metric multidimensional scaling algorithm for textual data visualization. In *IDA 2007*, pp. 252–262. Springer.
- Harrower, M. et C. A. Brewer (2003). Colorbrewer.org : an online tool for selecting colour schemes for maps. *The Cartographic Journal* 40(1), 27–37.
- Kolb, P. (2008). DISCO : A multilingual database of distributionally similar words. *Proceedings of KONVENS-2008, Berlin*.
- Kruskal, J. B. et M. Wish (1978). *Multidimensional scaling*, Volume 11. Sage.
- Kuan, J. (2015). *Learning Highcharts 4*. Packt Publishing Ltd.
- Ledbrook, P. et G. Smith (2014). *Grails in Action*. Manning Publications Co.
- Lin, D. (1998a). Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL 98*, pp. 768–774.
- Lin, D. (1998b). Extracting collocations from text corpora. In *First workshop on computational terminology*, pp. 57–63. Citeseer.
- Mattmann, C. et J. Zitting (2011). *Tika in action*. Manning Publications Co.
- McCallum, A. K. (2002). MALLETT. <http://mallet.cs.umass.edu>.
- Namiot, D. et M. Sneps-Sneppe (2014). On micro-services architecture. *International Journal of Open Information Technologies* 2(9), 24–27.
- Parisot, O., G. Vierke, T. Tamisier, Y. Didry, et H. Rieder (2014). Visual analytics for supporting manufacturers and distributors in online sales. In *EMISA 2014*.
- Sarrion, E. (2012). *jQuery UI*. "O'Reilly Media, Inc."
- Sinclair, J. et M. Cardew-Hall (2008). The folksonomy tag cloud : when is it useful ? *Journal of Information Science* 34(1), 15–29.

Summary

For the EGC 2016 challenge, we have developed a web application to explore the data describing the papers that were presented from 2004 during the EGC conferences. The tool allows to discover the main topics that were discussed in the articles. Moreover, the tool aims at find the papers that are semantically similar to user-defined topics.

5. <http://www.infinait.eu>