



HAL
open science

Un outil de génération automatique de tableaux de bord

Praveen Soni, Cyril de Runz, Fatma Bouali, Gilles Venturini

► To cite this version:

Praveen Soni, Cyril de Runz, Fatma Bouali, Gilles Venturini. Un outil de génération automatique de tableaux de bord. Atelier “ La place des usagères et usagers dans les outils de fouille et d’exploration de données ” (PAUL@EGC 2024), 2024. hal-04510218

HAL Id: hal-04510218

<https://hal.science/hal-04510218v1>

Submitted on 18 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Un outil de génération automatique de tableaux de bord

Praveen Soni*, Cyril de Runz*, Fatma Bouali** **, Gilles Venturini*

*LIFAT, EA 6300, Université de Tours,
nom.prenom@univ-tours.fr, <https://lifat.univ-tours.fr/>

** IUT Lille, Université de Lille,
nom.prenom@univ-lille.fr, <https://iut.univ-lille.fr/>

1 Approche proposée

Nous proposons dans ce résumé un générateur automatique de tableaux de bord et une démonstration d'une première version de ce système. Un tableau de bord (TB) est une interface visuelle multi-vues contenant plusieurs visualisations, éventuellement liées et coordonnées, ayant pour objectif de présenter des données, de surveiller des indicateurs ou encore d'explorer des informations de manière interactive.

Les TBs concernent de nombreux domaines de l'activité humaine et ont fait l'objet de multiples développements et études. Le point spécifique qui nous intéresse est l'aide "automatisée" que l'on peut apporter aux utilisateurs pour construire un TB. Cela consiste à définir des algorithmes de recommandation ou de génération automatique s'adressant en particulier à des utilisateurs novices. On trouve dans la littérature scientifique des approches pour ce problème (en excluant les méthodes qui ne suggèrent qu'une seule visualisation et pas un TB multi-vues) : certaines utilisent des heuristiques simples comme DynSpace (El Meseery et al., 2018), ou des algorithmes d'apprentissage comme par exemple les systèmes DashBot (Da Col et al., 2021) (apprentissage par renforcement, Vizdeck (Key et al., 2012) (arbre de décision) ou DeepEye (Luo et al., 2018) (apprentissage profond). D'autres approches utilisent de l'optimisation sous contraintes comme Draco (Moritz et al., 2018). Egalement, on peut citer des logiciels industriels comme Tableau (Mackinlay et al., 2007), PowerBI ou QlikSense. Toutes ces approches peuvent faire des suggestions de visualisations à l'utilisateur, mais souvent de manière incrémentale (une par une) et sans tenir compte des visualisations déjà présentes dans le TB. Peu d'entre elles prennent en compte le retour utilisateur.

Dans nos travaux, nous avons cherché un algorithme qui puisse faire une optimisation globale des visualisations composant un TB, et ainsi optimiser de manière co-adaptée les visualisations (Soni et al., 2023). Cet algorithme doit aussi tenir compte des compétences de l'utilisateur et de ses objectifs par rapport aux données, et des ses éventuels retours. Pour cela nous proposons une approche utilisant trois modèles (données, utilisateurs, visualisations), une interface et un algorithme génétique. Cet algorithme fait évoluer une population de solutions, chaque solution étant un TB avec un nombre variable de visualisations et un appariement avec les attributs de données. Des opérateurs génétiques permettent de générer des solutions, soit de manière aléatoire (population initiale), soit en modifiant légèrement une solution (mutation

Un outil de génération automatique de tableaux de bord

qui ajoute ou supprime une visualisation ou qui vient changer l'appariement avec les attributs de données), ou en combinant deux solutions (croisement qui échange des visualisations entre deux TBs). Trois critères sont utilisés pour évaluer les solutions : la qualité de l'appariement entre les visualisations et les données (il faut trouver les visualisations les plus adaptées aux données), le respect des préférences de l'utilisateur pour certaines données (il faut choisir les attributs de données qui correspondent à ce que veut analyser l'utilisateur), la complexité du TB proposé (il faut tenir compte des compétences de l'utilisateur et sa capacité à comprendre ce que fait l'outil). En ajustant ces trois critères, notre algorithme génétique peut proposer en quelques secondes des TBs à l'utilisateur.

Nous intégrons également dans l'interface graphique une approche semi-interactive dans laquelle l'utilisateur peut, en fonction des suggestions qui lui sont faites par l'algorithme génétique, les accepter en entier ou bien les refuser partiellement ou construire un nouveau TB à partir de morceaux de solution (une ou plusieurs visualisations). Un système de retour utilisateur permet d'aider l'utilisateur à améliorer son usage de l'outil.

Nous avons réalisé une première implémentation de cet outil qui doit encore être améliorée. Nous proposons d'en faire une démonstration lors de notre présentation. Nous utilisons notamment la librairie VegaLite.

Les travaux que nous envisageons ensuite consistent principalement à finir de régler l'optimisation par algorithme génétique (notamment la fonction d'évaluation), puis à réaliser une évaluation utilisateur à plus grande échelle. Nous souhaitons également optimiser l'arrangement 2D des visualisations à l'écran et proposer automatiquement des solutions à l'utilisateur.

Références

- Da Col, S., R. Ciucanu, M. Soare, N. Bouarour, et S. Amer-Yahia (2021). Dashbot : An ml-guided dashboard generation system. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 4696–4700.
- El Meseery, M., Y. Wu, et W. Stuerzlinger (2018). Multiple workspaces in visual analytics. In *2018 International Symposium on Big Data Visual and Immersive Analytics (BDVA)*, pp. 1–12. IEEE.
- Key, A., B. Howe, D. Perry, et C. Aragon (2012). Vizdeck : self-organizing dashboards for visual analytics. In *Proceedings of the 2012 acm sigmod international conference on management of data*, pp. 681–684.
- Luo, Y., X. Qin, N. Tang, et G. Li (2018). Deepeye : Towards automatic data visualization. In *2018 IEEE 34th international conference on data engineering (ICDE)*, pp. 101–112. IEEE.
- Mackinlay, J., P. Hanrahan, et C. Stolte (2007). Show me : Automatic presentation for visual analysis. *IEEE transactions on visualization and computer graphics* 13(6), 1137–1144.
- Moritz, D., C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, et J. Heer (2018). Formalizing visualization design knowledge as constraints : Actionable and extensible models in draco. *IEEE transactions on visualization and computer graphics* 25(1), 438–448.
- Soni, P., C. De Runz, F. Bouali, et G. Venturini (2023). A genetic algorithm for automatic dashboard generation : first results. In *2023 27th International Conference Information Visualisation (IV)*, pp. 77–82. IEEE.