Discussion

2021-10-07

Discussion

We propose a new integrated framework to help researcher collect and screen research publications characterized by high performance and versatility, joining the growing field of systematic review automatization (SRA) and helpers (SRH) tools.
The International Collaboration for the Automation of Systematic Reviews (ICASR) proposes a number of features tools for automatic research synthesis should possess, defining the so called Vienna Principles (Beller et al., 2018). We tried to adopt some of the concepts advocated by the ICAS; for example we are sharing the tool as an open source R package, and we tried to follow the paradigms of modular programming (McCracken, 2003), building separated components communicating through intermediate outputs stored in common formats (Comma Separated Values/Microsoft Excel™), which make it possible for users to extend the framework or integrate it with other tools in their pipeline.
Our framework has two main components, an integrated query-based citation search and management engine and an active machine learning record screener

Our search engine is capable of automatically collecting citation data from three well known scientific databases, that is, Pubmed, Web of Science and the database of the Institute of Electrical and Electronics Engineers, implementing also institutional access rights; in addition, it can process manually downloaded results from all mentioned databases plus records from the SCOPUS and EMBASE databases. Once the citations are downloaded, they are translated in a uniform format and deduplicated. The online search algorithm is efficient enough to manage tens of thousands of search results, using various expedients to overcome the online databases limitations in terms of traffic and download quota. Most SRH tools we are aware of, commercial or free to use, rely either on internal databases (e.g., Mendeley https://www.mendeley.com/), often focused on a particular topic (Visser, 2010) or relying on only one external data source (Poulter et al., 2008; Soto et al., 2019; Thomas & Brunton, 2007). To our knowledge, our tool is the first capable to fully automatize searching on three different well known scientific databases and to import raw data from five of them. Mixing different database is fundamental to have a more complete view of the literature on a topic (Bajpai et al., 2011; Wilkins et al., 2005; Woods & Trewheellar, 1998): in our results, 18.7% of the positive matches were unique for one of the different data sources and no positive record was present among all of them (data not shown).

Modern NLP techniques were used to transform textual data into actionable numeric inputs (Ananiadou & McNaught, 2006; Cohen & Hunter, 2008). One particular improvement was to use network analysis to identify associated terms, which we labelled as non-consecutive ngrams, which confer particular meaning when present together into a text while not necessary being close to each other. A similar network approach was used to merge redundant terms, in order to make model estimation more efficient and reduce possible noise. The use of network inspired modeling of text is not new (François Rousseau et al., 2015; Francois Rousseau, 2015; Violos et al., 2016) and is a valuable tool to extract semantic information not evident in one-word of consecutive ngram models.

The citation screening functionality of the framework implements active machine learning based document classification, adopting best practices and bringing further improvements at various levels. At the base of our classification algorithm we use BART (Chipman et al., 2010; Kapelner & Bleich, 2013), a Bayesian machine learning technique characterized by exceptional predictive performance and speed even in large

dataset. This model builds consecutive decision trees associating the presence/absence of set of terms to the probability of a positive match, each tree trying to decrease the prediction error of the precedent one (Hastie et al., 2009). The Bayesian approach explore distributions of trees which are justified by the data, while using a set of prior distributions to avoid overfitting by discouraging excessively deep trees, long sequences of trees, or extreme predicted probabilities. To further improve generalizability we ran multiple replications of the model and then averaged their record predictive distributions; this technique is called ensembling and it has been shown to improve out-of-sample predictive performance []. In a Bayesian setting, this effect comes from a reduction of the probability mass in the tails of the predictive distributions while not shifting their locations (i.e., it decreases variance without impacting bias) [bias/variance tradeoff, better if in Bayesian analysis and/or focusing to ensemble]. Instead, imposing stronger general priors against extreme predictions would decreased variance but also shifted the distribution mass towards a non-decision zone, therefore increasing bias. In our data it can be seen how using more then one model replica greatly improved sensitivity and efficiency, with a performance stabilization after 10 replicas (Suppl. Mat. S2); since the number of model replicas greatly impacted computation times we decided to use ten replicas since it was also associated with the higher sensitivity in the same performance cluster. The variability between models derives only from the randomness in the MCMC fitting phase []; an alternative was to impose additional variability by bootstrapping [] but this approach actually decreased performance (Suppl. Mat. S2), probably due to the low proportion of positive matches in the data and the loss of information with bootstrapping in unbalanced classification tasks [].

The strong imbalance between relevant and non-relevant records (Sampson et al., 2011) can affect sensitivity []. To overcome the problem, we oversampled the positive records ten times before model fitting. Our hyperparameter analysis showed that together with model ensembling, the oversampling rate was the parameter with the highest impact on performance. An alternative approach would be record weighting to change error costs during estimation [] but the BART implementation we used did not allow observation weighting and using simple oversampling allows to use different modeling engines. Also, the simple query ordering of the record during the creation of the initial training set allowed to have a far higher prevalence of relevant records (17.2%) compared to the overall one (0.11%) and this boosts the overall sensitivity of the model. A known risk with oversampling of the target class is the missclassification of negative records as positive [], but since in our approach all predicted positive are manually reviewed we are ensured to achieve always 100% specificity/positive predictive value.

One of the innovation in our approach is the overcoming of simple stopping rules in SRA classification tasks []. Static supervised learning models requires to manually label a certain amount of records (i.e., the training set) which are then used to train the algorithm and classify the rest (the test set) [], but this method requires to decide the size of the training set in advance. Active learning tries to alleviate the problem by splitting the classification task in an iterative process in which prediction and review phases alternate, but even in this case, one has to define a stopping rule regarding how many records to screen manually in every iteration []. By exploiting the Bayesian nature of BART we produce a distribution of probabilities and not a single value for each record; by joining these distribution in the positive and negative record groups it's possible to identify a "uncertainty zone" so that all record whose predictive distribution crosses this zone needs manual verification. Therefore the researcher will have a defined number or records to check at every iteration which dynamically decreases as uncertain predictions are reviewed and the uncertainty shrinks (Suppl. Mat. S2 Fig. 1). In this model, researcher do not need to choose a priori the number of records to review but just the size of the uncertainty zone (by setting the PPD quantiles used to built it) and after how many iterations with no new positive matches to stop; both the parameters are more theoretical and general than a simple stopping rule and nevertheless the hyperparameter tuning did not indicate a big impact of these parameters. Since our approach requires researchers to review both unlabelled records with a positive predicted label and those with a PPD inside the uncertainty zone, it can be considered as a unifying syinthesis of the "certainty" and "uncertainty" paradigms of active learning (Miwa et al., 2014).

To evaluate performance we avoided the classic random out-of-sample approaches like train-test sampling, out-of-bag bootstrapping or cross-validation (Kohavi & others, 1995) because we deemed them not realistic in this setting. It is assumed that the corpus to be screened is not a random sample, but the entire population of citation for a given query; a random validation would be useful to estimate the performance of

this method on a new dataset derived by the same data-generating process, but this is a totally theoretical goal with no usefulness since all the existing data related was alread acquired. Instead a more relevant question is to estimate the long run sensitivity of our prediction method in finding all relevant records in the whole data set. Since the records are ordered, again a random approach would not work since it assumes that the prevalence of positive records is equal on average in every possible sample (Tashman, 2000), while instead it drops as more records are considered (Fig. 2). We instead used a Bayesian model to estimate the likelihood of a positive match given the prediction scores of the BART model. This approach showed to fit the data very well (high R^2) and the Bayesian predictive distribution can be exploited to draw a worse case scenario given the data.
In our study our approach reached 98.8% sensitivity during the hyperparameter optimization (deterministic, small sample) and on the whole data set we predicted a theoretical sensitivity of 100% [93.5%, 100%] in the first session and of 97.3% [73.8%, 100%] in the second, a lower value as is expected by the probabilistic accumulation of random events; both results are above the usual results in the field (O'Mara-Eves et al., 2015) and in line with the 92% average sensitivity estimated after human only screening (Edwards et al., 2002). In one interesting case, the model spotted a missclassification during initial labeling, demonstrating its robustness and its value also as a second screener, as already suggested by previous studies (Bekhuis & Demner-Fushman, 2010, 2012; Frunza et al., 2010).
Also we showed that such sensitivity was achieved screening just 1.34% of all downloaded records, a sensible reduction in workload; finally, albeit the simple query based ordering did concentrate most of the relevant matches in the first 20-25 thousands records, some relevant records would have required almost the full data set to be manually checked to be found.
The model takes ~5-20 minutes per iteration to perform predictions in session 1 (17755 documents) and 20-40 minutes in session 2 (98371 documents) on a 8 core, 2.5 GhZ, 16 GB RAM laptop from 2014 and that equate to 1-3 days per session to manually review and perform predictions, for a total of 1-2 weeks for the whole process, a huge saving of time compared to the months usually required for the screening phase of systematic reviews (Allen & Olkin, 1999; Bannach-Brown et al., 2019; Borah et al., 2017). To our knowledge, our data sets are larger than what common in most SRA studies (O'Mara-Eves et al., 2015; Olorisade et al., 2016), emphasizing the reliability of the tool in real world scenarios.

The last component of our framework is a data-driven query generation algorithm. The creation of an efficient and efficacious search query is a complex task (Hammerstrøm et al., 2010; Lefebvre et al., 2011) since it requires building a combination of positive and negative terms to maximize the number of relevant search results while minimizing the total number of results. We propose a solution based on concurrent decision trees (Blanco-Justicia & Domingo-Ferrer, 2019; Moore et al., 2018) built on the posterior predicted probabilities; the technique tries to extract high sensitivity subqueries from the labeled data, enrich them with negative terms to increase specificity (Abdelmgeid Amin, 2008) and then let researcher evaluate them and pick the most meaningful ones. The framework will then join them and remove redundancies. The aim is to generate a second query that complement the first human made one and help find possible missing records not found during the first session.
The generated query allowed to retrieve few more positive matches not found in session 1 at the cost of a large increase in the amount of documents. One interesting aspect of this functionality is that it provides a human-readable overview of the classification rules internalized by the classification model, showing which combination of terms was particularly relevant and even spotting authors and locations associated to the topic of study. The generated query therefore acted as a tool for machine learning explainability (Bhatt et al., 2020; Burkart & Huber, 2021), a feature useful to spot bias in black box classification algorithms (Malhi et al., 2020) and that is increasingly legally required for high-stack machine learning applications (Bibal et al., 2020, 2021).^ It is important to note that this process is entirely data-driven. The algorithm is only aware of the "world" defined by the data set, itself generated by a specific search query focused on a particular topic. Therefore, the new query may not be specific once applied to an unbounded search domain, returning an unmanageable amount of unrelated results. The solution we found was to add another component specifying the general topic (antimicrobial resistance and healthcare associated infections) of our research.

As reported, our framwork builds on modularity. We designed it in order to easily implement complete independency of the main modules in future iterations; especially, we plan to allow users create their custom made components in addition or in place of the original ones, further expanding its functionalities. In our view it would be possible to easily add custom citation search and parsing functions for other scientific databases, alternative text features building algorithms and alternative machine learning modules.
We deem such interoperability extremely relevant, because the main strength of our tool is the composition of many solutions and the general approach related to Bayesian active machine learning, but each of its components could benefit greatly from the recent improvements in text mining.
For example, our text feature generation approach is based on the boolean bag-of-words paradigm, and surely it could be improved by more nuanced text representations: it could be evaluated if feature transformations like Tf-Idf (Ananiadou & McNaught, 2006; Baeza-Yates et al., 1999), would be advantageous, even if we hypothesize that tree based classification algorithms like BART are robust enough to not need such operation. Word embedding could be considered, which transform terms in semantic vectors based on the surrounding text Minaee et al. (2021), and could be used to eliminate semantically redundant terms or differentiate identical terms with different meanings given the context; another option would be to use unsupervised learning models like Latent Dirichlet Analysis, Latent Semantic Analysis, etc., (Q. Chen et al., 2016; Landauer et al., 1998; Pavlinek & Podgorelec, 2017) to extract topics to enrich the feature space. Our classification algorithm can be implemented with any Bayesian supervised machine learning method which produces full PPDs; therefore alternative models could be evaluated, like Gaussian Processes which are known for their flexibility (S.-H. Chen et al., 2015; Jayashree & Srijith, 2020). Even more interesting would be to test advanced learning algorithms that surpass the bag-of-words approach, taking into consideration higher level features in the text like term context and sequences, long distance term relationships, semantic structures, etc., (Cheng et al., 2019; Farkas, 1995; Lai et al., 2015; Li et al., 2020; Minaee et al., 2021; Yang et al., 2020), given that a Bayesian implementation of such algorithms is available (for example C. Chen et al. (2018)).

Abdelmgeid Amin, A. (2008). *Using a query expansion technique to improve document retrieval*.

Allen, I. E., & Olkin, I. (1999). Estimating time to conduct a meta-analysis from number of citations retrieved. *Jama*, *282*(7), 634–635.

Allen, I. E., & Olkin, I. (1999). Estimating time to conduct a meta-analysis from number of citations retrieved. *Jama*, *282*(7), 634–635.

Ananiadou, S., & McNaught, J. (2006). *Text mining for biology and biomedicine*. Citeseer.

Baeza-Yates, R., Ribeiro-Neto, B., & others. (1999). *Modern information retrieval* (Vol. 463). ACM press New York.

Bajpai, A., Davuluri, S., Haridas, H., Kasliwal, G., Deepti, H., Sreelakshmi, K., Chandrashekar, D., Bora, P., Farouk, M., Chitturi, N., & others. (2011). In search of the right literature search engine (s). *Nature Precedings*, 1–1.

Bannach-Brown, A., Przybyła, P., Thomas, J., Rice, A. S., Ananiadou, S., Liao, J., & Macleod, M. R. (2019). Machine learning algorithms for systematic review: Reducing workload in a preclinical review of animal studies and reducing human screening error. *Systematic Reviews*, *8*(1), 1–12.

Bekhuis, T., & Demner-Fushman, D. (2010). Towards automating the initial screening phase of a systematic review. *MEDINFO 2010*, 146–150.

Bekhuis, T., & Demner-Fushman, D. (2012). Screening nonrandomized studies for medical systematic reviews: A comparative study of classifiers. *Artificial Intelligence in Medicine*, *55*(3), 197–207.

Beller, E., Clark, J., Tsafnat, G., Adams, C., Diehl, H., Lund, H., Ouzzani, M., Thayer, K., Thomas, J., Turner, T., & others. (2018). Making progress with the automation of systematic reviews: Principles of the international collaboration for the automation of systematic reviews (ICASR). *Systematic Reviews*, *7*(1), 1–7.

Bhatt, U., Andrus, M., Weller, A., & Xiang, A. (2020). Machine learning explainability for external stakeholders. *arXiv Preprint arXiv:2007.05408*.

Bibal, A., Lognoul, M., De Streel, A., & Frénay, B. (2021). Legal requirements on explainability in machine learning. *Artificial Intelligence and Law*, *29*(2), 149–169.

Bibal, A., Lognoul, M., Streel, A. de, & Frénay, B. (2020). Impact of legal requirements on explainability in machine learning. *arXiv Preprint arXiv:2007.05479*.

Blanco-Justicia, A., & Domingo-Ferrer, J. (2019). Machine learning explainability through comprehensible decision trees. *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, 15–26.

Bollegala, D., Maehara, T., & Kawarabayashi, K. (2015). Embedding semantic relations into word representations. *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Borah, R., Brown, A. W., Capers, P. L., & Kaiser, K. A. (2017). Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the PROSPERO registry. *BMJ Open*, *7*(2), e012545.

Burkart, N., & Huber, M. F. (2021). A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, *70*, 245–317.

Chen, C., Lin, X., & Terejanu, G. (2018). An approximate bayesian long short-term memory algorithm for outlier detection. *2018 24th International Conference on Pattern Recognition (ICPR)*, 201–206.

Chen, Q., Yao, L., & Yang, J. (2016). Short text classification based on LDA topic model. *2016 International Conference on Audio, Language and Image Processing (ICALIP)*, 749–753.

Chen, S.-H., Lee, Y.-S., Tai, T.-C., & Wang, J.-C. (2015). Gaussian process based text categorization for healthy information. *2015 International Conference on Orange Technologies (ICOT)*, 30–33. https://doi.org/10.1109/ICOT.2015.7498487

Cheng, Y., Ye, Z., Wang, M., & Zhang, Q. (2019). Document classification based on convolutional neural network and hierarchical attention network. *Neural Network World*, *29*(2), 83–98.

Chipman, H. A., George, E. I., McCulloch, R. E., & others. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, *4*(1), 266–298.

Cohen, K. B., & Hunter, L. (2008). Getting started in text mining. *PLoS Computational Biology*, *4*(1), e20.

Edwards, P., Clarke, M., DiGuiseppi, C., Pratap, S., Roberts, I., & Wentz, R. (2002). Identification of randomized controlled trials in systematic reviews: Accuracy and reliability of screening records. *Statistics in Medicine*, *21*(11), 1635–1640.

Farkas, J. (1995). Document classification and recurrent neural networks. *Proceedings of the 1995 Conference of the Centre for Advanced Studies on Collaborative Research*, 21.

Frunza, O., Inkpen, D., & Matwin, S. (2010). Building systematic reviews using automatic text classification techniques. *Coling 2010: Posters*, 303–311.

Hammerstrøm, K., Wade, A., Jørgensen, A.-M. K., & Hammerstrøm, K. (2010). Searching for studies. *Education*, *54*(11.3).

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Boosting and additive trees. In *The elements of statistical learning* (pp. 337–387). Springer.

Jayashree, P., & Srijith, P. (2020). Evaluation of deep gaussian processes for text classification. *Proceedings of the 12th Language Resources and Evaluation Conference*, 1485–1491.

Kapelner, A., & Bleich, J. (2013). bartMachine: Machine learning with bayesian additive regression trees. *arXiv Preprint arXiv:1312.2171*.

Kohavi, R., & others. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Ijcai*, *14*, 1137–1145.

Lai, S., Xu, L., Liu, K., & Zhao, J. (2015). Recurrent convolutional neural networks for text classification. *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, *25*(2-3), 259–284.

Lefebvre, C., Manheimer, E., Glanville, J., Higgins, J., & Green, S. (2011). Searching for studies (chapter 6). *Cochrane Handbook for Systematic Reviews of Interventions Version*, *510*.

Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P. S., & He, L. (2020). A survey on text classification: From shallow to deep learning. *arXiv Preprint arXiv:2008.00364*.

Malhi, A., Knapic, S., & Främling, K. (2020). Explainable agents for less bias in human-agent decision making. *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, 129–146.

McCracken, D. D. (2003). Modular programming. In *Encyclopedia of computer science* (pp. 1183–1184).

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning–based text classification: A comprehensive review. *ACM Computing Surveys (CSUR)*, *54*(3), 1–40.

Miwa, M., Thomas, J., O'Mara-Eves, A., & Ananiadou, S. (2014). Reducing systematic review workload through certainty-based screening. *Journal of Biomedical Informatics*, *51*, 242–253.

Moore, A., Murdock, V., Cai, Y., & Jones, K. (2018). Transparent tree ensembles. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 1241–1244.

O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., & Ananiadou, S. (2015). Using text mining for study identification in systematic reviews: A systematic review of current approaches. *Systematic Reviews*, *4*(1), 1–22.

Olorisade, B. K., Quincey, E. de, Brereton, P., & Andras, P. (2016). A critical analysis of studies that address the use of text mining for citation screening in systematic reviews. *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*, 1–11.

Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and LDA topic models. *Expert Systems with Applications*, *80*, 83–93.

Poulter, G. L., Rubin, D. L., Altman, R. B., & Seoighe, C. (2008). MScanner: A classifier for retrieving medline citations. *BMC Bioinformatics*, *9*(1), 1–12.

Rousseau, Francois. (2015). *Graph-of-words: Mining and retrieving text with networks of features* [PhD thesis]. Ph. D. dissertation.

Rousseau, François, Kiagias, E., & Vazirgiannis, M. (2015). Text categorization as a graph classification problem. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1702–1712.

Sampson, M., Tetzlaff, J., & Urquhart, C. (2011). Precision of healthcare systematic review searches in a cross-sectional sample. *Research Synthesis Methods*, *2*(2), 119–125.

Soto, A. J., Przybyła, P., & Ananiadou, S. (2019). Thalia: Semantic search engine for biomedical abstracts. *Bioinformatics*, *35*(10), 1799–1801.

Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review. *International Journal of Forecasting*, *16*(4), 437–450.

Thomas, J., & Brunton, J. (2007). *EPPI-reviewer: Software for research synthesis*.

Turian, J., Ratinov, L., & Bengio, Y. (2010). Word representations: A simple and general method for semi-supervised learning. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 384–394.

Violos, J., Tserpes, K., Psomakelis, E., Psychas, K., & Varvarigou, T. (2016). Sentiment analysis using word-graphs. *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics*, 1–9.

Visser, E. (2010). Performing systematic literature reviews with researchr: Tool demonstration. *Technical Report Series TUD-SERG-2010-010*.

Wilkins, T., Gillies, R. A., & Davies, K. (2005). EMBASE versus MEDLINE for family medicine searches: Can MEDLINE searches find the forest or a tree? *Canadian Family Physician*, *51*(6), 848–849.

Woods, D., & Trewheellar, K. (1998). Medline and embase complement each other in literature searches. *BMJ: British Medical Journal*, *316*(7138), 1166.

Yang, J., Bai, L., & Guo, Y. (2020). A survey of text classification models. *Proceedings of the 2020 2nd International Conference on Robotics, Intelligent Control and Artificial Intelligence*, 327–334.