Machine Learning Assisted Citation Screening for Systematic Reviews

Anjani DHRANGADHARIYA [a,1], Roger HILFIKER [b], Roger SCHAER [a] and Henning MÜLLER [a,c]

[a] *University of Applied Sciences Western Switzerland (HES-SO), Sierre, Switzerland*
[b] *School of Health Sciences, HES-SO Valais-Wallis, Leukerbad, Switzerland*
[c] *University of Geneva (UNIGE), Geneva, Switzerland*

Abstract. Evidence-based practice is highly dependent upon up-to-date systematic reviews (SR) for decision making. However, conducting and updating systematic reviews, especially the citation screening for identification of relevant studies, requires much human work and is therefore expensive. Automating citation screening using machine learning (ML) based approaches can reduce cost and labor. Machine learning has been applied to automate citation screening but not for the SRs with very narrow research questions. This paper reports the results and observations for an ongoing research that aims to automate citation screening for SRs with narrow research questions using machine learning. The research also sheds light on the problem of class imbalance and class overlap on the performance of ML classifiers when applied to SRs with narrow research questions.

Keywords. Systematic reviews, Automation, Natural language processing, Machine learning

1. Introduction

Summarizing evidence in a specific domain through the process of conducting systematic reviews (SR) becomes increasingly difficult with an exponential increase in the number of primary research publications and an increasing variety of publishers. With this increased publication-rate, citation screening for conducting the SRs has become time-consuming and labour intensive. The core of citation screening involves manually categorizing the studies found in literature search into relevant or non-relevant depending upon predefined criteria for the research question at hand [1]. This screening is usually performed by two independent reviewers, who often can not keep up with the manual process of screening the studies and constantly updating outdated SRs [2].

In the area of physiotherapy and rehabilitation such an exponential increase in the number of publications is also observed[2]. For an ongoing review on exercise and non-exercise interventions in reducing cancer-related fatigue retrieved over 30,000 references, about 2,000 of which were published in 2017 alone. It took more than 200 hours

[1]Corresponding Author: Anjani Dhrangadhariya, University of Applied Sciences Western Switzerland (HES-SO), Technopole 3, 3960 Sierre, Switzerland; E-mail: anjani.dhrangadhariya@hevs.ch.

[2]https://www.ncbi.nlm.nih.gov/pubmed?term=(physiotherapy)%20OR%20rehabilitation

each for the two independent reviewers to manually assess the titles and abstracts for relevance to the research question before the studies were taken for further meta-analysis [3].

Supervised machine learning based classification approaches are successfully applied for automation of citation screening but either only for broad and shallow SRs [4] or SRs that retrieved fewer than 6,000 studies [5]. However, there are SRs that address very specific research questions leading to narrow, predefined criteria for selection of relevant studies to be included for meta-analysis. For such narrow SRs, inclusion prevalence becomes as low as 10%, which means that out of all the studies retrieved during the search phase, 90% are excluded as non-relevant. A narrow research question combined with a low inclusion prevalence leads to class imbalance and class overlap problems for classification tasks that generally reduce classifier performance [6]. Class overlap cannot be artificially controlled but class imbalance can be tackled using oversampling or undersampling [7]. In this work, we aim to explore machine learning and natural language processing to assist citation screening in SRs with a narrow research question and low inclusion prevalence using word embeddings and random oversampling.

2. Methods

2.1. Datasets

Data sets from the open access literature were used in the work described here.

Word embedding Titles and abstracts (TA) of 2.09 million studies included in the PubMed Central Open-Access subset (PMC-OA) were used to generate semantic word embedding using the two most common architectures: word2vec and fastText [8,9].

Data for the screening process The data set used to test automation approaches included the studies identified for citation screening in an update to the systematic review by Hilfiker *et al.* [3]. The data set included TA from 31,279 studies identified during the search phase of this SR. These studies were already manually assessed for relevance and labelled by two reviewers into two mutually exclusive labels. 4,066 studies assessed as relevant were labelled "include" and the rest were labelled "exclude". The inclusion prevalence for this case is only about 13% leading to class imbalance.

2.2. Screening Automation

To automate the screening and explore the effect of class imbalance, six supervised binary classifiers were trained to classify documents into "include" (relevant) vs. "exclude" (non-relevant) using the data set of Section 2.1 represented using corpus-specific static word embedding. This approach follows steps enumerated below.

1. Word embedding generation To generate embedding, the PMC-OA data were lowercased and all punctuation except the hyphens were removed. Phrase generation was then performed using the word2phrase tool[3] to identify frequently occurring bi-grams. The output of phrase generation along with the unigrams was fed to gensim's word2vec[4] and

[3]https://github.com/travisbrady/word2phrase

[4]https://radimrehurek.com/gensim/models/word2vec.html

to fastText[5] using the hyperparamters in Table 1 to obtain two dense, semantic, real-valued word embeddings [10].

Parameter	Value	Parameter	value	Parameter	value
Size	300	Alpha	0.05	bucket*	2000000
Window	5	min_alpha	0.0001	minn*	3
min_count	5	sample	0.0001	maxn*	6
sg	1	iter/ epoch	5	seed	1
hs	1	negative	5		

Table 1. Hyperparameter values used to generate word embeddings using gensim's word2vec and the fastText functionality. (*) means that these parameters were available only for the fastText embeddings.

2. Data set for text pre-processing Post deduplication and removal of non-English language studies led to 25,540 studies remaining. For these remaining studies, the text was lower-cased and tokenized into words using NLTK[6] (Natural Langauge ToolKit). Irrelevant tokens were removed using a predefined set of stop-words provided by NLTK and PubMed[7]. Additional corpus-specific stop-words identified during the experiments with the training set were removed accordingly. The text normalization process converted British English terms into American English. After token lemmatization, a corpus vocabulary was constructed from all the unique unigram and bigram tokens. To scale this vocabulary down, uninformative short tokens with fewer than five characters and the tokens with vocabulary count lower than five were removed assuming they were not representative of the classes.

3. Random oversampling Class imbalance often deteriorates the classifier performance, so in the present data set it was addressed using naive random oversampling. This method randomly duplicates data points from the minority class and brings the total number of instances in the minority class equal to the majority class [6].

4. Feature extraction Feature extraction was performed to generate real-valued, dense feature vectors from text. These features were used as input to train the supervised ML classifiers. For non-neural ML classifiers, from each of the 25,540 studies word vectors were extracted for each token using both generated word embeddings. All vectors for each token in the individual study were averaged over the entire study normalized by study length to produce a single 300-dimensional vector representing a particular study. For the Convolutional Neural Network (CNN), feature extraction was part of the model, where a static, non-trainable weight matrix generated from the word embedding was provided with the embedding layer to extract token word vectors during training.

5. Classifier training and evaluation Six classifiers including Logistic Regression (LR), Support Vector Machines (SVM), k-nearest neighbour (KNN), Decision Trees-CART (DT), Random Forest (RF), and CNNs were trained and evaluated over the generated text feature to perform binary classification. Hyperparameter tuning was performed using GridSearch. A CNN was trained using hyperparamters loosely based on suggestions

[5] https://radimrehurek.com/gensim/models/fasttext.html
[6] https://www.nltk.org/api/nltk.tokenize.html
[7] https://www.ncbi.nlm.nih.gov/books/NBK3827/table/pubmedhelp.T.stopwords/

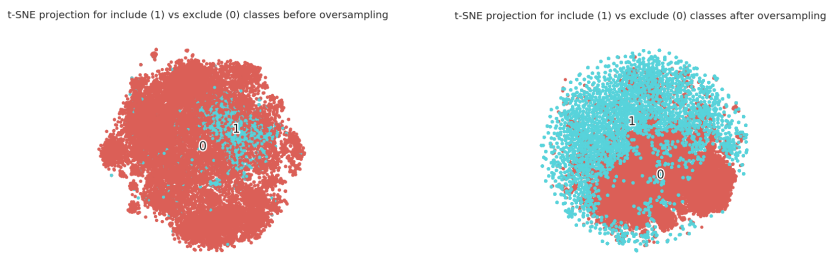

Figure 1. A t-SNE projection for 25,540 studies labelled "include" vs. "exclude" before and after oversampling.

from Zhang *et al.* [11]. Classifier performance before and after oversampling was evaluated on an unseen validation data set and metrics pertinent to imbalanced classification like precision, recall, F1 and precision-recall AUC (PR-AUC) score were tracked [7].

3. Results and Discussion

The data set used for training and evaluating the classifiers comprised 2,259 relevant studies labelled "include" and 23,281 labelled "exclude". Cosine similarity between class centroid vectors for the "include" and "exclude" classes before and after oversampling is 0.985814 and 0.985824 respectively. A high cosine similarity is indicative of high class overlap even after tackling the class imbalance using oversampling (see Figure 1).

The results obtained by applying classifiers on the data set before and after random oversampling are summarized in Table 2 and 3. Before oversampling, the classifiers focused on improving the performance for the majority class but in reality they are simply predicting the majority class as noticeable from the relatively high F1 score for the exclude class. Upon oversampling, the overall classifier performance drastically improves for the minority class especially the precision (see Table 3), while the precision for the majority class is reduced with a small improvement in recall.

		Class "include"				Class "exclude"		
Classifier	embedding	Precision	Recall	F1	PR-AUC	Precision	Recall	F1
LR	fastText	0.4044	0.8990	0.5576	0.6008	0.9891	0.8746	0.9283
SVM	fastText	0.6538	0.4640	0.5428	0.6317	0.9463	0.9746	0.9602
KNN	word2vec	0.6536	0.6066	0.6288	0.6512	0.9619	0.9685	0.9652
DT	fastText	0.2961	0.8627	0.4394	0.4287	0.9837	0.8000	0.8821
RF	fastText	0.4995	0.7892	0.6108	0.5921	0.9780	0.9209	0.9485
CNN	fastText	0.6545	0.5032	0.5690	0.6388	0.9511	0.9732	0.9620

Table 2. Classifier performance before random oversampling for the "include" and "exclude" classes

If the task is considered a classification task, a high class overlap still leads to unacceptable precision and recall values for citation screening implying the inclusion of false-positive studies and exclusion of false-negative studies. Our future work has two directions: Firstly, experimenting with systematic oversampling techniques like Synthetic Minority Over-Sampling Technique (SMOTE) and Adaptive Synthetic (ADASYN) and considering this task as data extraction rather than classification [12].

		Class “include”				Class “exclude”		
Classifier	embedding	Precision	Recall	F1	PR-AUC	Precision	Recall	F1
LR	word2vec	0.8981	0.8850	0.9116	0.9342	0.8951	0.9090	0.8816
SVM	fastText	0.8914	0.8818	0.9012	0.9378	0.8990	0.8792	0.8890
KNN	word2vec	0.8860	0.8303	0.9500	0.9321	0.9418	0.8055	0.8682
DT	fastText	0.8348	0.8201	0.8510	0.8734	0.8285	0.8463	0.8126
RF	word2vec	0.8695	0.8918	0.8488	0.9279	0.8966	0.8757	0.8560
CNN	word2vec	0.9034	0.7480	0.8183	0.9318	0.7850	0.9200	0.8471

Table 3. Classifier performance after random oversampling for the “include” and “exclude” classes.

4. Conclusion

To the best of our knowledge, this is the first attempt to explore citation screening automation for a narrow systematic review topic using domain-specific word embedding on a range of ML classifiers. The research specifically sheds light on the impact of class imbalance and class overlap on the classifier performance before and after oversampling as also discussed by Garcìa *et al.* and Prati *et al.* [6,7]. Knowledge of these challenges is useful for further development of automation approaches for citation screening.

References

[1] J. Higgins and S. Green, *Cochrane Handbook for Systematic Reviews of Interventions*. The Cochrane Collaboration, Mar 2011.

[2] S. Khangura, K. Konnyu, R. Cushman, J. Grimshaw, and D. Moher, “Evidence summaries: the evolution of a rapid review approach,” *Syst Rev* **1**, p. 10, Feb 2012.

[3] R. Hilfiker, A. Meichtry, M. Eicher, L. Nilsson Balfe, R. H. Knols, M. L. Verra, and J. Taeymans, “Exercise and other non-pharmaceutical interventions for cancer-related fatigue in patients during or after cancer treatment: a systematic review incorporating an indirect-comparisons meta-analysis,” *British Journal of Sports Medicine* **52**(10), pp. 651–658, 2018.

[4] A. Bannach-Brown, P. Przyby?a, J. Thomas, A. S. C. Rice, S. Ananiadou, J. Liao, and M. R. Macleod, “Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error,” *Syst Rev* **8**, p. 23, Jan 2019.

[5] I. Lerner, P. Créquit, P. Ravaud, and I. Atal, “Automatic screening using word embeddings achieved high sensitivity and workload reduction for updating living network meta-analyses,” *J. Clin. Epidemiol.* **108**, pp. 86–94, Apr. 2019.

[6] V. García, R. Alejo, J. S. Sánchez, J. M. Sotoca, and R. A. Mollineda, “Combined effects of class imbalance and class overlap on Instance-Based classification,” in *Intelligent Data Engineering and Automated Learning – IDEAL 2006*, pp. 371–378, Springer Berlin Heidelberg, 2006.

[7] T. Saito and M. Rehmsmeier, “The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets,” *PLoS ONE* **10**(3), p. e0118432, 2015.

[8] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality.,” *NIPS’13 Proceedings of the 26th International Conference on Neural Information Processing Systems* **2**, pp. 3111–3119, Dec 2013.

[9] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the Association for Computational Linguistics* **5**, pp. 135–146, 2017.

[10] Y. Goldberg, “A primer on neural network models for natural language processing,” Oct. 2015.

[11] Y. Zhang and B. Wallace, “A sensitivity analysis of (and practitioners’ guide to) convolutional neural networks for sentence classification,” Oct. 2015.

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and others, “SMOTE: synthetic minority over-sampling technique,” *J. Artif. Organs* , 2002.