Automation of Citation Screening for Systematic Literature Reviews using Neural Networks: A Replicability Study

Wojciech Kusa[1][0000−0003−4420−4147], Allan Hanbury[1,2][0000−0002−7149−5843], and Petr Knoth[3,4][0000−0003−1161−7359]

[1] TU Wien, Vienna, Austria,
[2] Complexity Science Hub, Vienna, Austria
[3] Research Studios Austria, Vienna, Austria
[4] The Open University, Milton Keynes, The United Kingdom
wojciech.kusa@tuwien.ac.at

Abstract. In the process of Systematic Literature Review, citation screening is estimated to be one of the most time-consuming steps. Multiple approaches to automate it using various machine learning techniques have been proposed. The first research papers that apply deep neural networks to this problem were published in the last two years. In this work, we conduct a replicability study of the first two deep learning papers for citation screening [16,8] and evaluate their performance on 23 publicly available datasets. While we succeeded in replicating the results of one of the papers, we were unable to replicate the results of the other. We summarise the challenges involved in the replication, including difficulties in obtaining the datasets to match the experimental setup of the original papers and problems with executing the original source code. Motivated by this experience, we subsequently present a simpler model based on averaging word embeddings that outperforms one of the models on 18 out of 23 datasets and is, on average, 72 times faster than the second replicated approach. Finally, we measure the training time and the invariance of the models when exposed to a variety of input features and random initialisations, demonstrating differences in the robustness of these approaches.

Keywords: Citation Screening, Study Selection, Systematic Literature Review (SLR), Document Retrieval, Replicability

1 Introduction

A systematic literature review is a type of secondary study that summarises all available data fitting pre-specified criteria to answer precise research questions. It uses rigorous scientific methods to minimise bias and generate clear, solid conclusions that health practitioners frequently use to make decisions [12].

Unfortunately, conducting systematic reviews is slow, labour intensive and time-consuming as this relies primarily on human effort. A recent estimate shows

that conducting a full systematic review takes, on average, 67 weeks [4], although another past study reports that the median time to publication was 2.4 years [22]. Furthermore, according to [21], 23% of published systematic reviews need updating within two years after completion.

Citation screening (also known as selection of primary studies) is a crucial part of the systematic literature review process [23]. During this stage, reviewers need to read and comprehend hundreds (or thousands) of documents and decide whether or not they should be included in the systematic review. This decision is made on the basis of comparing each article content with predefined exclusion and inclusion criteria. Traditionally it consists of two stages, the first round of screening titles and abstracts, which is supposed to narrow down the list of potentially relevant items. It is followed by a task appraising the full texts, a more detailed (but also more time-consuming) revision of all included papers from the first stage based on the full text of articles.

Multiple previous studies tried to decrease the completion time of systematic reviews by using text mining methods to semi-automate the citation screening process (see a recent systematic review on this topic: [9]). Using the machine learning paradigm, citation screening could be reduced to a binary classification problem. Then, the task is to train a model using the seed of manually labelled citations that can distinguish between documents to be included (includes) and those to be excluded (excludes). One of the challenges is a significant class imbalance (for 23 benchmark datasets, the maximum percentage of included documents is 27%, and on average, it is only 7%). Additionally, existing approaches require training a separate model for each new systematic review.

In this work, we replicate two recent papers related to automated citation screening for systematic literature reviews using neural networks [16,8]. We chose these studies since, to our knowledge, they are the first ones to address this problem using deep neural networks. Both papers represent citation screening as a binary classification task and train an independent model for each dataset. We evaluate the models on 23 publicly available benchmark datasets. We present our challenges regarding replicability in terms of datasets, models and evaluation. In the remaining sections of this article, we will use the name **Paper A** to refer to the study by Kontonatsios et al. [16] and **Paper B** to indicate work by van Dinter et al. [8].

Moreover, we investigate if the models are invariant to different data features and random initialisations. 18 out of 23 datasets are available as a list of Pubmed IDs of the input papers with assigned categories (included or excluded). As we needed to recreate data collection stages for both papers, we wanted to measure if the choice of the document features would influence the final results of the replicated models.

Both papers utilise deep learning due to their claimed substantial superiority over traditional (including shallow neural network) models. We compare the models with previous benchmarks and assess to what extent do these models improve performance over simpler and more traditional models. Finally, we make

our data collection and experiment scripts and detailed results publicly available on GitHub[1].

2 Related Work

Out of all stages of the systematic review process, the selection of primary studies is known as the most time-consuming step [2,20,24]. It was also automated the most often in the past using text mining methods. According to a recent survey on the topic of automation of systematic literature reviews [9], 25 out of 41 analysed primary studies published between 2006 and 2020 addressed (semi-)automation of the citation screening process. Another, older systematic review from 2014 found in total 44 studies dealing implicitly or explicitly with the problem of screening workload [18].

Existing approaches to automation of the citation screening process can be categorised into two main groups. The first one uses text classification models [25,17] and the second one screening prioritisation or ranking techniques that exclude items falling below some threshold [10,6]. Both groups follow a similar approach. They train a supervised binary classification algorithm to solve this problem, e.g. Support Vector Machines (SVMs) [25,6], Naïve Bayes [17] or Random Forest [15]. A significant limitation of these approaches is the need for a large number of human decisions (annotations) that must be completed before developing a reliable model [24].

Kontonatsios et al. [16] (**Paper A**) was the first one to apply deep learning algorithms to automate the citation screening process. They have used three neural network-based denoising autoencoders to create a feature representation of the documents. This representation was fed into a feed-forward network with a linear SVM classifier trained in a supervised manner to re-order the citations. Van Dinter et al. [8] (**Paper B**) presented the first end-to-end solution to citation screening with a deep neural network. They developed a binary text classification model with the usage of a multi-channel convolutional neural network. Both models claim to yield significant workload savings of at least 10% on most benchmark review datasets.

A different procedure to automating systematic reviews was presented during the CLEF 2017 eHealth Lab Technology Assisted Reviews in Empirical Medicine task [13,14]. Here, the user needs to find all relevant documents from a set of PubMed articles given a Boolean query. It overcomes the need for creating an annotated dataset first but makes it harder to incorporate reviewers' feedback.

The recently published BERT model [7] and its variants have pushed the state of the art for many NLP tasks. Ioannidis [11] used BERT-based models to work on document screening within the Technology Assisted Review task achieving better results than the traditional IR baseline models. To our knowledge, this was the first use of a generative neural network model in a document screening task.

[1] https://github.com/ProjectDoSSIER/CitationScreeningReplicability

3 Experiment setup

3.1 Models

DAE-FF Paper A presents a neural network-based, supervised feature extraction method combined with a linear Support Vector Machine (SVM) trained to prioritise eligible documents. The data preprocessing pipeline contains stopword removal and stemming with a Porter stemmer. The feature extraction part is implemented as three independent denoising autoencoders (DAE) that learn to reconstruct corrupted Bag-of-Words input vectors. Their concatenated output is used to initialise a supervised feed-forward neural network (FF). These extracted document vectors are subsequently used as an input to an L2-regularised linear SVM classifier. Class imbalance is handled by setting the regularisation parameter $C = 1 \times 10^{-6}$.

Multi-Channel CNN Paper B presents a multi-channel convolutional neural network (CNN) to discriminate between includes and excludes. It uses static, pre-trained GloVe word embeddings [19] to create an input embedding matrix. This embedding is inserted into a series of parallel CNN blocks consisting of a single-dimensional CNN layer followed by global max pooling. Outputs from the layers are concatenated after global pooling and fed into a feed-forward network. The authors experimented with a different number of channels and Conv1D output shapes. Input documents are tokenised and lowercased, punctuation and non-alphabetic tokens are removed. Documents are padded and truncated to a maximum length of 600 tokens. Class imbalance is handled with oversampling. For our replicability study, we have chosen the best performing Model_2.

fastText We also test a shallow neural network model which is based on fastText word embeddings [3]. This model is still comparable to more complex deep learning models in many classification tasks. At the same time, it is orders of magnitude faster for training and prediction, making it more suitable for active learning scenarios where reviewers could alter the model's predictions by annotating more documents. To make it even simpler, we do not use pre-trained word embeddings to vectorise documents. Data preprocessing is kept minimal as we only lowercase the text and remove all non-alphanumerical characters.

Hyperparameters Paper A optimised only the number of training epochs for their DAE model. In order to do so, they used two datasets: Statins and BPA reviews and justified this choice with differences between smaller datasets from Clinical and Drug reviews and SWIFT reviews. Other hyperparameters (including the minibatch size and the number of epochs for the feed-forward model) are constant across all datasets. Paper B used the Statins review dataset to tune a set of hyperparameters, including the number of epochs, batch size, dropout, and dense units.

3.2 Data

All 23 datasets are summarised in Table 1, including the dataset source, number of citations, number and percentage of eligible citations, maximum WSS@95% score (Section 3.3) and the availability of additional bibliographic metadata. Every citation consists of a title, an abstract, and an eligibility label (included or excluded). Moreover, 18 datasets contain also bibliographic metadata. The percentage of eligible citations (includes) varies between datasets, from 0.55% to 27.04%, but on average, it is about 7%, meaning that the datasets are highly imbalanced.

Table 1. Statistics of 23 publicly available datasets used in the experiments on automated citation screening for Systematic Literature Reviews.

	Dataset name	Introduced in	# Citations	Included citations	Excluded citations	Maximum WSS@95%	Bibliographic metadata
1	ACEInhibitors		2544	41 (1.6%)	2503 (98.4%)	93.47%	Yes
2	ADHD		851	20 (2.4%)	831 (97.6%)	92.77%	Yes
3	Antihistamines		310	16 (5.2%)	294 (94.8%)	89.84%	Yes
4	Atypical Antipsychotics		1120	146 (13.0%)	974 (87.0%)	82.59%	Yes
5	Beta Blockers		2072	42 (2.0%)	2030 (98.0%)	93.07%	Yes
6	Calcium Channel Blockers		1218	100 (8.2%)	1118 (91.8%)	87.20%	Yes
7	Estrogens	Drug	368	80 (21.7%)	288 (78.3%)	74.35%	Yes
8	NSAIDs	(Cohen et al.,	393	41 (10.4%)	352 (89.6%)	85.08%	Yes
9	Opioids	2006)	1915	15 (0.8%)	1900 (99.2%)	94.22%	Yes
10	Oral Hypoglycemics		503	136 (27.0%)	367 (73.0%)	69.16%	Yes
11	Proton PumpInhibitors		1333	51 (3.8%)	1282 (96.2%)	91.32%	Yes
12	Skeletal Muscle Relaxants		1643	9 (0.5%)	1634 (99.5%)	94.45%	Yes
13	Statins		3465	85 (2.5%)	3380 (97.5%)	92.66%	Yes
14	Triptans		671	24 (3.6%)	647 (96.4%)	91.57%	Yes
15	Urinary Incontinence		327	40 (12.2%)	287 (87.8%)	83.38%	Yes
	Average Drug		1249	56 (7.7%)	1192 (92.3%)	87.67%	15/15
16	COPD	Clinical	1606	196 (12.2%)	1410 (87.8%)	83.36%	No
17	Proton Beam	(Wallace et al.,	4751	243 (5.1%)	4508 (94.9%)	90.14%	No
18	Micro Nutrients	2010)	4010	258 (6.4%)	3752 (93.6%)	88.87%	No
	Average Clinical		3456	232 (7.9%)	3223 (92.1%)	87.45%	0/3
19	PFOA/PFOS		6331	95 (1.5%)	6236 (98.5%)	93.56%	Yes
20	Bisphenol A (BPA)	SWIFT	7700	111 (1.4%)	7589 (98.6%)	93.62%	Yes
21	Transgenerational	(Howard et al.,	48638	765 (1.6%)	47873 (98.4%)	93.51%	Yes
22	Fluoride and neurotoxicity	2016)	4479	51 (1.1%)	4428 (98.9%)	93.91%	No
23	Neuropathic pain — CAMRADES		29207	5011 (17.2%)	24196 (82.8%)	78.70%	No
	Average SWIFT		19271	1206 (4.6%)	18064 (95.4%)	90.66%	3/5
	Average (All datasets)		5454	329 (7.0%)	5125 (93.0%)	88.29%	18/23

Cohen et al. [5] was the first one to introduce datasets for training and evaluation of citation screening. They constructed a test collection for 15 different systematic review topics produced by the Oregon Evidence-based Practice Centre (EPC) related to the efficacy of medications in several drug classes.

Another three datasets for evaluation of automated citation screening were released by Wallace et al. [25]. These systematic reviews are related to the clinical outcomes of various treatments. Both drug and clinical reviews contain a small number of citations (varying from 310 to 4751).

The third group of datasets was introduced by Howard et al. [10] and consists of five substantially larger reviews (from 4479 to 48 638 citations) that have been used to assess the performance of the SWIFT-review tool. They were created using broader search strategies which justifies a higher number of citations.

Paper A trained and evaluated their model on all 23 datasets coming from three categories. Paper B used 20 datasets from the Clinical and SWIFT cat-

egories. Paper B states that, on average, 5.2% of abstracts are missing in all 20 datasets, varying between 0% for *Neuropathic Pain* and 20.82% for *Statins*. Compared to previous papers, Paper B reports fewer citations for three datasets (Table 6 in the original paper): *Statins*, *PFOA/PFOS* and *Neuropathic Pain*. This difference is insignificant compared to the dataset size, e.g. 29207 versus 29202 for *Neuropathic Pain*, so it should not influence the model evaluation.

3.3 Evaluation

Evaluation of automated citation screening can be very challenging. Traditional metrics used for classification tasks like precision, recall, or F-score cannot capture what we intend to measure in this task. For an automated system to be beneficial to systematic reviewers, it should save time and miss as few relevant papers as possible. Previous studies suggested that recall should not be lower than 95%, and at the same time, precision should be as high as possible [5].

Work saved over sampling at r% recall (WSS@r%) is a primary metric for evaluation of automated citation screening. It was first introduced and described by Cohen et al. [5] as "the percentage of papers that meet the original search criteria that the reviewers do not have to read (because they have been screened out by the classifier)." It estimates the human screening workload reduction by using automation tools, assuming a fixed recall level of r%. WSS@r%, given a recall of r%, is defined as follows:

$$WSS@r\% = \frac{TN + FN}{N} - (1 - r)$$

where TN is the number of true negatives, FN is the number of false negatives, and N is the total number of documents. Based on previous studies, we fix the recall at 95% and compute the WSS@95% score.

One drawback of this metric described by [5] is that it does not take into account time differences caused by varying lengths of documents and also the time needed to review a full-text article compared to only reading the title and the abstract.

A further drawback of WSS is that the maximum WSS value depends on the ratio of included/excluded samples. A perfectly balanced dataset can achieve a maximum value of WSS@95% = 0.45, whereas a highly imbalanced dataset with a 5%/95% split can obtain a maximum WSS@95% score of 0.9. Consequently, it does not make sense to compare the results nor average them across different datasets (as done in Paper A and B).

For our replicability study, we decided to use the implementations of the WSS metric provided by Papers A and B.

Cross-validation Both papers use a stratified 10×2 cross-validation for evaluation. In this setting, data is randomly split in half: one part is used to train the classifier, and the other is left for testing. This process is then repeated ten times, and the results are accumulated from all ten runs. We also use this approach to evaluate the quality of all three models.

3.4 Code

The authors of both papers uploaded their code into public GitHub repositories: [2,3]. Both models were written in Python 3 and depend primarily on TensorFlow and Keras deep learning frameworks [1]. The whole implementation was uploaded in four commits for Paper A and one for Paper B (excluding commits containing only documentation). Except for the code, there is no information about versions of the packages used to train and evaluate the models. This missing information is crucial for replicability, as, for TensorFlow alone, in 2020, there were 27 different releases related to 6 different MINOR versions[4].

The model prepared by Paper B uses also pre-trained 100-dimensional GloVe word embeddings which we downloaded separately from the original authors' website[5] according to the instructions provided by the Paper B GitHub Readme.

Both papers did not include the original datasets they used to train and evaluate their models. Paper A provided sample data consisting of 100 documents which presents the input data format accepted by their model, making it easier to re-run the experiments. Paper B does not include sample data but describes where and how to collect and process the datasets.

4 Results and discussion

4.1 Replicability study

WSS@95% scores from older benchmarks and original papers, along with our replicated results, are presented in Table 2. For all datasets, both Paper A and B provide only mean WSS@95% score from cross-validation runs. Therefore, we were not able to measure statistical significance between our replicated results and the original ones. To quantify the difference, we decided to calculate the absolute delta between reported and replicated scores: $|x-y|$. Both models report a random seed for the cross-validation splits but not for the model optimisation. Usage of different seeds for model optimisation might be one of the reasons why we were not able to achieve the same results.

For two datasets (*Bisphenol A (BPA)* and *Triptans*), Paper A reports two different results for the DAE-FF model (Tables 5 and 6 in the original paper). We suppose this was only a typing mistake, as we managed to infer the actual values based on the averaged WSS@95% score from all datasets available in the original paper.

The average delta between our replicated results and the original ones from Paper A is 3.59%. Only for three datasets is this value higher than 10%. If we consider different seeds used for training models, these results confirm the successful replication of Paper A's work.

[2] https://github.com/gkontonatsios/DAE-FF
[3] https://github.com/rvdinter/multichannel-cnn-citation-screening
[4] https://pypi.org/project/tensorflow/#history
[5] https://nlp.stanford.edu/data/glove.6B.zip

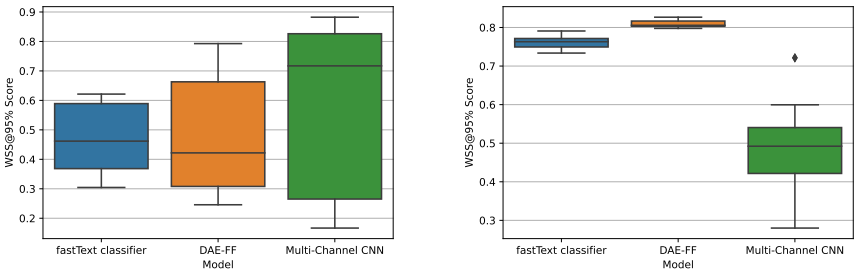
Table 2. WSS@95% results for replicated models compared with original results and benchmark models. WSS@95% scores are averages across ten validation runs for each of the 23 review datasets. Underlined scores indicate the highest score within the three tested models, **bold** values indicate the highest score overall.

Dataset name	Cohen (2006)	Matwin (2010)	Cohen (2008/ 2011)	Howard (2016)	**Paper A**	Paper A replicated	Absolute delta	**Paper B**	Paper B replicated	Absolute delta	fastText classifier
ACEInhibitors	.566	.523	.733	**.801**	.787	.785	0.16%	.783	.367	41.59%	.783
ADHD	.680	.622	.526	**.793**	.665	.639	2.58%	.698	.704	0.57%	.424
Antihistamines	.000	.149	.236	.137	**.310**	.275	3.48%	.168	.135	3.32%	.047
Atypical Antipsychotics	.141	.206	.170	.251	**.329**	.190	13.92%	.212	.081	13.15%	.218
Beta Blockers	.284	.367	.465	.428	**.587**	.462	12.52%	.504	.399	10.51%	.419
Calcium Channel Blockers	.122	.234	.430	**.448**	.424	.347	7.66%	.159	.069	9.03%	.178
Estrogens	.183	.375	.414	**.471**	.397	.369	2.80%	.119	.083	3.56%	.306
NSAIDs	.497	.528	.672	**.730**	.723	.735	1.18%	.571	.601	2.98%	.620
Opioids	.133	.554	.364	**.826**	.533	.580	4.71%	.295	.249	4.58%	.559
Oral Hypoglycemics	.090	.085	**.136**	.117	.095	.123	2.80%	.065	.013	5.21%	.098
Proton PumpInhibitors	.277	.229	.328	.378	**.400**	.299	10.13%	.243	.129	11.38%	.283
Skeletal Muscle Relaxants	.000	.265	.374	**.556**	.286	.286	0.04%	.229	.300	7.14%	.090
Statins	.247	.315	.491	.435	**.566**	.487	7.93%	.443	.283	16.03%	.409
Triptans	.034	.274	.346	.412	.434	.412	2.24%	.266	**.440**	17.38%	.210
Urinary Incontinence	.261	.296	.432	**.531**	**.531**	.483	4.81%	.272	.180	9.21%	.439
Average Drug	.234	.335	.408	**.488**	.471	.431	5.13%	.335	.269	10.37%	.339
COPD	—	—	—	—	**.666**	.665	0.07%	—	.128	—	.312
Proton Beam	—	—	—	—	**.816**	.812	0.39%	—	.357	—	.733
Micro Nutrients	—	—	—	—	.662	**.663**	0.08%	—	.199	—	.608
Average Clinical	—	—	—	—	**.715**	.713	0.18%	—	.228	—	.551
PFOA/PFOS	—	—	—	.805	**.848**	.838	0.97%	.071	.305	23.44%	.779
Bisphenol A (BPA)	—	—	—	.752	**.793**	.780	1.34%	.792	.369	42.31%	.637
Transgenerational	—	—	—	.714	.707	**.718**	1.14%	.708	.000	70.80%	.368
Fluoride and neurotoxicity	—	—	—	.870	.799	.806	0.68%	**.883**	.808	7.48%	.390
Neuropathic pain	—	—	—	**.691**	.608	.598	1.03%	.620	.091	52.89%	.613
Average SWIFT	—	—	—	**.766**	.751	.748	1.03%	.615	.315	39.38%	.557
Average (all datasets)	—	—	—	—	**.564**	.537	3.59%	—	.273	17.63%	.414

For Paper B, the average delta is 17.63%. For 10 out of 20 datasets, this delta is more than 10%. For the two largest datasets: *Transgenerational* and *Neuropathic Pain* we were not able to successfully train the Multi-Channel CNN model. All of these results raise concerns about replicability.

Paper B also tried to replicate the DAE-FF model from Paper A. They stated that *"(...) we aimed to replicate the model (...) with open-source code via GitHub. However, we could not achieve the same scores using our dataset. After emailing the primary author, we were informed that he does not have access to his datasets anymore, which means their study cannot be fully replicated."*. Our results are contrary to findings by Paper B: we managed to replicate the results of Paper A successfully without having access to their original datasets. Unfortunately, Paper B does not present any quantitative results of their replicability study. Therefore, we cannot draw any conclusions regarding those results as we do not know what Paper B authors meant by *"cannot be fully replicated"*.

Figure 1 presents results for *ADHD* and *Proton Beam* datasets for all three models. The Multi-Channel CNN model has the widest range of WSS@95% scores across cross-validation runs. This is especially evident on the datasets from the Clinical group (i.e. *Proton Beam*), for which the DAE-FF and fastText models yield very steady results across every cross-validation fold. This could mean that the Multi-Channel CNN model is less stable, and its good performance is dependant on random initialisation.



(a) ADHD review dataset. (b) Proton Beam review dataset.

Fig. 1. Example boxplots with WSS@95% scores for three models. Input features are titles and abstracts.

Next, we compare our replicated results and the original ones from Paper A and B to previous benchmark studies. Paper A only compares their model to custom baseline methods and does not mention the previous state of the art results. None of the tested neural network-based models can improve on the results by Howard et al. [10], which uses a log-linear model with word-score and topic-weight features to classify the citations. This means that even though deep neural network models can provide significant gains in WSS@95% scores, they can still be outperformed by classic statistical methods.

4.2 Impact of input features

As we encountered memory problems when training the Paper B model on *Transgenerational* and *Neuropathic pain* datasets, we exclude these two datasets from our comparisons in the remaining experiments.

None of the papers provided the original input data used to train the models. We wanted to measure if the results depend on how that input data was gathered. We implemented two independent data gathering scripts using the biopython package as suggested by Paper B to obtain 18 out of 23 datasets. One implementation relied on the Medline module, where a document was represented as a dictionary of all available fields. The second implementation returned all possible fields (title, abstract, author and journal information) concatenated in a single string. Furthermore, we examined how robust the models are, if the input data contained only titles or abstracts of the citations. Results are presented in the Table 3.

The best average WSS@95% results are obtained for all three models when they use all available features (Figure 2). All models achieved better results when using just the abstract data compared to the titles alone. This reaffirms our common sense reasoning that titles alone are not sufficient for citation screening. However, there are some specific datasets for which best results were obtained when the input documents contained only titles or abstracts. While this experiment does not indicate why this is the case, we can offer some potential reasons: (1) it could be that eligible citations of these datasets are more similar in terms

Table 3. Influence of input document features on the WSS@95% score for three tested models. "All features" means a single string containing all possible fields. For each row, **bold** values indicate the highest score for each model, underlined scores are best overall.

	DAE-FF				Multi-Channel CNN				fastText classifier			
Dataset name	All features	Title and Abstract	Abstract only	Title only	All features	Title and Abstract	Abstract only	Title only	All features	Title and Abstract	Abstract only	Title only
ACEInhibitors	.785	.709	.658	**.806**	.367	.461	.648	.525	**.783**	.776	.765	.441
ADHD	.639	.500	.404	**.651**	**.704**	.528	.692	.580	.424	**.470**	.444	.200
Antihistamines	**.275**	.168	.265	.016	.135	**.204**	.114	.105	.047	.124	.175	**.192**
Atypical Antipsychotics	.190	.221	**.230**	.046	.081	**.086**	.050	.013	**.218**	.188	.185	.095
Beta Blockers	**.462**	.451	.390	.408	**.399**	.243	.134	.211	**.419**	**.419**	.407	.262
Calcium Channel Blockers	**.347**	.337	.297	.137	.069	.083	.004	**.117**	.178	.139	.060	**.244**
Estrogens	**.369**	.358	.331	.145	.083	.076	.051	**.092**	**.306**	.199	.108	.241
NSAIDs	**.735**	.679	.690	.658	**.601**	.443	.358	.225	**.620**	.506	.512	.535
Opioids	**.580**	.513	.499	.280	.249	**.420**	.413	.287	**.559**	.558	.534	.245
Oral Hypoglycemics	.123	**.129**	.107	.019	.013	**.021**	.004	.005	**.098**	.049	.042	.016
Proton PumpInhibitors	**.299**	.291	.153	.285	**.129**	.121	.059	.118	.283	.228	.174	**.360**
Skeletal Muscle Relaxants	.286	.327	**.430**	.125	.300	**.329**	.242	.202	.090	.142	.180	**.210**
Statins	**.487**	.434	.392	.255	**.283**	.231	.120	.082	**.409**	.376	.281	.228
Triptans	**.412**	.253	.320	.199	**.440**	.404	.407	.129	.210	.205	**.211**	.075
Urinary Incontinence	.483	**.531**	.482	.372	**.180**	.161	.046	.099	**.439**	.310	.170	.434
Average Drug	**.431**	.394	.373	.293	**.269**	.254	.223	.185	**.339**	.313	.283	.252
COPD	.665	.665	.676	**.677**	.128	**.372**	.087	.093	.312	**.553**	.546	.545
Proton Beam	**.812**	.810	.790	.799	.357	.489	.408	**.559**	.733	.761	**.771**	**.771**
Micro Nutrients	.663	.648	.665	**.677**	.199	.255	.251	**.268**	**.608**	.602	.605	.601
Average Clinical	**.713**	.708	.670	**.718**	.228	**.372**	.249	.307	.551	.638	**.640**	.639
PFOA/PFOS	.713	.839	**.847**	.696	.305	**.405**	.391	.109	.779	**.796**	.778	.292
Bisphenol A (BPA)	**.780**	.754	.715	.631	.369	.300	**.612**	.182	**.637**	.630	.499	.079
Fluoride and neurotoxicity	.806	**.838**	.758	.726	**.808**	.688	.654	.452	**.390**	.375	.292	.250
Average SWIFT	.766	**.782**	.774	.684	.494	.464	**.552**	.247	**.602**	.600	.523	.207
Average (All datasets)	**.520**	.498	.481	.410	.295	**.301**	.274	.212	**.407**	.400	.368	.301

of titles or abstract; (2) it could be that these models are not able to retrieve relevant information when there is too much noise. Intra- and inter-class dataset similarity need to be further evaluated in future studies.

As presented in Table 2, the fastText classifier model was not able to outperform the original results from Paper A and B. However, compared to our replicated results of Paper B, the fastText classifier achieves higher WSS@95% scores on 18 out of 23 datasets. It is also more robust to random initialisation compared to Multi-Channel CNN.

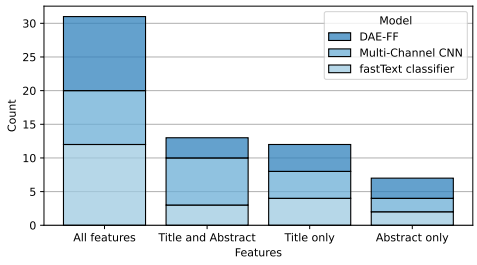


Fig. 2. A count of experiments in which a model using a specific input feature achieved the best results. Models that use all available features scored the best results 49% of times for a specific (model, dataset) combination.

4.3 Training time

We computed the training time for each of the models. The relationship between dataset size and model training time is visualised in Figure 3. For the DAE-FF model, we calculated both the training procedure of denoising autoencoder, feed-forward networks, and linear SVM. The DAE component is the most time-absorbing component as it consumes, on average, 93.5% of the total training time. For the fastText and Multi-Channel CNN models, we calculated the training procedure of the binary classifier.

For small datasets containing less than 1000 documents, one validation fold for fastText took on average 2 seconds, for Multi-Channel CNN 13 seconds, and DAE-FF 82 seconds. Training time difference increases for larger models, where the speed of fastText is even more significant. For the largest dataset, *Transgenerational*, the mean training time for fastText is 78 seconds, for Multi-Channel CNN 894 seconds and for DAE-FF, it is 18,108 seconds. On average, the fastText model is 72 times faster than DAE-FF and more than eight times faster than Multi-Channel CNN, although this dependency is not linear and favours fastText for larger datasets.

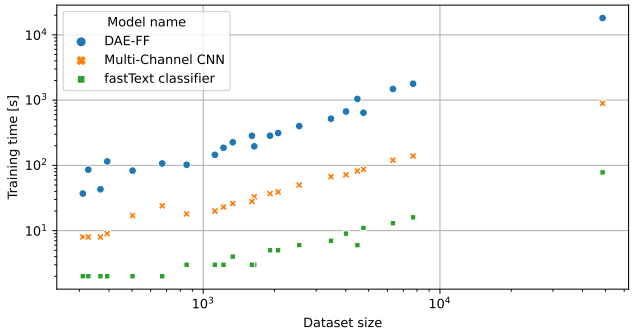


Fig. 3. The relationship between dataset size and a model training time for the three evaluated models. Both training time and dataset size are shown on a logarithmic scale.

4.4 Precision@95%recall

Finally, we measure the precision at a recall level of 95%, a metric proposed by Paper A. Table 4 shows mean scores for each model across all three review groups. Similarly to the WSS@95% metric, the best performing model is DAE-FF achieving a mean precision@95%recall on 21 datasets equal to 0.167. This method outperforms Multi-Channel CNN and fastText models by 3.2% and 4.6%, respectively. Paper A reported average precision@95%recall equal to 19% over 23 review datasets, which is comparable with our findings. Paper B does not report this score, so we cannot compare our results regarding the Multi-Channel CNN model.

Table 4. Precision at 95% recall results for the three models averaged on 21 benchmark datasets. We did not measure the precision for the two largest SWIFT datasets: *Transgenerational* and *Neuropathic pain*.

	DAE-FF	Multi-Channel CNN	fastText classifier
Average Drug	**.143**	.121	.112
Average Clinical	**.324**	.221	.230
Average SWIFT	**.127**	.091	.058
Average (21 datasets)	**.167**	.135	.121

5 Conclusions

This work replicates two recent papers on automated citation screening for systematic literature reviews using deep neural networks. The model proposed by Paper A consists of a denoising autoencoder combined with feed-forward and SVM layers (DAE-FF). Paper B introduces a multi-channel convolutional neural network (Multi-Channel CNN). We used the 23 publicly available datasets to measure the quality of both models. The average delta between our replicated results and the original ones from Paper A is 3.59%. Considering that we do not know the random seed used for the training of original models, we can conclude that the replication of Paper A was successful. The average delta for Paper B is 17.63%. In addition to that, this model is characterised by a significant variance, so we cannot claim successful replication of this method.

Subsequently, we evaluated the fastText classifier and compared its performance to the replicated models. This shallow neural network model based on averaging word embeddings achieved better WSS@95% results when compared to replicated scores from Paper B and, at the same time, is on average 72 and 8 times faster during training than both Paper A and B models.

None of the tested models can outperform all the others across all the datasets. DAE-FF achieves the best average results, though it is still worse when compared to a statistical method with the log-linear model. Models using all available features (title, abstract, author and journal information) perform best on the average of 21 datasets when compared to just using a title, abstract or both.

Availability of the code alone does not guarantee a replicable experimental setup. If the project was not documented for the specific software versions, it might be challenging to reconstruct these requirements based exclusively on the code, especially if the experiments were conducted some time ago. In the case of code written in Python, explicitly writing environment version with, for example, *requirements.txt* or conda's *environment.yml* files should be sufficient in most of the cases to save time for researchers trying to replicate the experiments.

Acknowledgements. This work was supported by the EU Horizon 2020 ITN/ETN on Domain Specific Systems for Information Extraction and Retrieval – DoSSIER (H2020-EU.1.3.1., ID: 860721).

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems (2015), https://research.google/pubs/pub45166/
2. Bannach-Brown, A., Przybyła, P., Thomas, J., Rice, A.S.C., Ananiadou, S., Liao, J., Macleod, M.R.: Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error. Systematic Reviews 2019 8:1 **8**(1), 1–12 (1 2019). https://doi.org/10.1186/S13643-019-0942-7, https://systematicreviewsjournal.biomedcentral.com/articles/10.1186/s13643-019-0942-7
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics **5**, 135–146 (7 2017). https://doi.org/10.1162/tacl_a_00051, http://arxiv.org/abs/1607.04606
4. Borah, R., Brown, A.W., Capers, P.L., Kaiser, K.A.: Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the PROSPERO registry (2 2017). https://doi.org/10.1136/bmjopen-2016-012545, http://bmjopen.bmj.com/
5. Cohen, A.M., Hersh, W.R., Peterson, K., Yen, P.Y.: Reducing workload in systematic review preparation using automated citation classification. Journal of the American Medical Informatics Association **13**(2), 206–219 (3 2006). https://doi.org/10.1197/jamia.M1929, /pmc/articles/PMC1447545//pmc/articles/PMC1447545/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC1447545/
6. Cohen, A.M.: Optimizing Feature Representation for Automated Systematic Review Work Prioritization. AMIA Annual Symposium Proceedings **2008**, 121 (2008), /pmc/articles/PMC2656096//pmc/articles/PMC2656096/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC2656096/
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference **1**, 4171–4186 (10 2018), https://arxiv.org/abs/1810.04805v2
8. van Dinter, R., Catal, C., Tekinerdogan, B.: A Multi-Channel Convolutional Neural Network approach to automate the citation screening process. Applied Soft Computing **112**, 107765 (11 2021). https://doi.org/10.1016/J.ASOC.2021.107765
9. van Dinter, R., Tekinerdogan, B., Catal, C.: Automation of systematic literature reviews: A systematic literature review. Information and Software Technology **136**, 106589 (8 2021). https://doi.org/10.1016/j.infsof.2021.106589, https://linkinghub.elsevier.com/retrieve/pii/S0950584921000690
10. Howard, B.E., Phillips, J., Miller, K., Tandon, A., Mav, D., Shah, M.R., Holmgren, S., Pelch, K.E., Walker, V., Rooney, A.A., Macleod, M., Shah, R.R., Thayer, K.: SWIFT-Review: A text-mining workbench for systematic review.

Systematic Reviews **5**(1), 1–16 (5 2016). https://doi.org/10.1186/s13643-016-0263-z, https://link.springer.com/articles/10.1186/s13643-016-0263-zhttps://link.springer.com/article/10.1186/s13643-016-0263-z

11. Ioannidis, A.: An Analysis of a BERT Deep Learning Strategy on a Technology Assisted Review Task (4 2021), https://arxiv.org/abs/2104.08340v1http://arxiv.org/abs/2104.08340
12. Jo, A., Raquel, A.I., Jane, B., Duncan, C., Alison, E., Debra, F., Susanne, H., Kate, L., Stephen, R., Amber, R., Lesley, S., Christian, S., Paul, W., Nerys, W.: Systematic Reviews: CRD's guidance for undertaking reviews in health care. CRD, University of York, York (1 2009), www.york.ac.uk/inst/crd
13. Kanoulas, E., Li, D., Azzopardi, L., Spijker, R.: CLEF 2017 technologically assisted reviews in empirical medicine overview. CEUR Workshop Proceedings **1866**, 1–29 (9 2017), https://pureportal.strath.ac.uk/en/publications/clef-2017-technologically-assisted-reviews-in-empirical-medicine-
14. Kanoulas, E., Li, D., Azzopardi, L., Spijker, R.: CLEF 2018 technologically assisted reviews in empirical medicine overview. CEUR Workshop Proceedings **2125** (7 2018), https://pureportal.strath.ac.uk/en/publications/clef-2018-technologically-assisted-reviews-in-empirical-medicine-
15. Khabsa, M., Elmagarmid, A., Ilyas, I., Hammady, H., Ouzzani, M.: Learning to identify relevant studies for systematic reviews using random forest and external information. Machine Learning 2015 102:3 **102**(3), 465–482 (10 2015). https://doi.org/10.1007/S10994-015-5535-7, https://link.springer.com/article/10.1007/s10994-015-5535-7
16. Kontonatsios, G., Spencer, S., Matthew, P., Korkontzelos, I.: Using a neural network-based feature extraction method to facilitate citation screening for systematic reviews. Expert Systems with Applications: X **6**, 100030 (7 2020). https://doi.org/10.1016/j.eswax.2020.100030
17. Matwin, S., Kouznetsov, A., Inkpen, D., Frunza, O., O'Blenis, P.: A new algorithm for reducing the workload of experts in performing systematic reviews. Journal of the American Medical Informatics Association **17**(4), 446–453 (7 2010). https://doi.org/10.1136/JAMIA.2010.004325
18. O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., Ananiadou, S.: Using text mining for study identification in systematic reviews: A systematic review of current approaches. Systematic Reviews **4**(1), 5 (1 2015). https://doi.org/10.1186/2046-4053-4-5
19. Pennington, J., Socher, R., Manning, C.D.: GloVe: Global Vectors for Word Representation. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference pp. 1532–1543 (2014). https://doi.org/10.3115/V1/D14-1162, https://aclanthology.org/D14-1162
20. Sellak, H., Ouhbi, B., Frikh, B., Ben, S.M.: Using Rule-based Classifiers in Systematic Reviews: A Semantic Class Association Rules Approach. Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services (2015). https://doi.org/10.1145/2837185, http://dx.doi.org/10.1145/2837185.2837279
21. Shojania, K.G., Sampson, M., Ansari, M.T., Ji, J., Doucette, S., Moher, D.: How quickly do systematic reviews go out of date? A survival analysis. Annals of Internal Medicine **147**(4), 224–233 (8 2007). https://doi.org/10.7326/0003-4819-147-4-200708210-00179
22. Tricco, A.C., Brehaut, J., Chen, M.H., Moher, D.: Following 411 Cochrane Protocols to Completion: A Retrospective Cohort Study. PLOS ONE **3**(11), e3684 (11

2008). https://doi.org/10.1371/JOURNAL.PONE.0003684, https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0003684

23. Tsafnat, G., Glasziou, P., Choong, M.K., Dunn, A., Galgani, F., Coiera, E.: Systematic review automation technologies (4 2014). https://doi.org/10.1186/2046-4053-3-74
24. Tsafnat, G., Glasziou, P., Karystianis, G., Coiera, E.: Automated screening of research studies for systematic reviews using study characteristics. Systematic Reviews 2018 7:1 **7**(1), 1–9 (4 2018). https://doi.org/10.1186/S13643-018-0724-7, https://link.springer.com/articles/10.1186/s13643-018-0724-7https://link.springer.com/article/10.1186/s13643-018-0724-7
25. Wallace, B.C., Trikalinos, T.A., Lau, J., Brodley, C., Schmid, C.H.: Semi-automated screening of biomedical citations for systematic reviews. BMC Bioinformatics 2010 11:1 **11**(1), 1–11 (1 2010). https://doi.org/10.1186/1471-2105-11-55, https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-11-55