Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 222 (2023) 114–126

Procedia Computer Science

www.elsevier.com/locate/procedia

International Neural Network Society Workshop on Deep Learning Innovations and Applications (INNS DLIA 2023)

Evaluation of Attention-Based LSTM and Bi-LSTM Networks For Abstract Text Classification in Systematic Literature Review Automation

Regina Ofori-Boateng[a,*], Magaly Aceves-Martins[b], Chrisina Jayne[c], Nirmalie Wiratunga[a], Carlos Francisco Moreno-Garcia[a]

[a]*School of Computing, Robert Gordon University, Aberdeen, Scotland, UK*
[b]*The Rowett Institute, University of Aberdeen, Aberdeen, Scotland, UK*
[c]*Teeside University, Middlesbrough, England, UK*

Abstract

Systematic Review (SR) presents the highest form of evidence in research for decision and policy-making. Nonetheless, the structured steps involved in carrying out SRs make it demanding for reviewers. Many studies have projected the abstract screening stage in the SR process to be the most burdensome for reviewers, thus automating this stage with artificial intelligence (AI). However, majority of these studies focus on using traditional machine learning classifiers for the abstract classification. Thus, there remain a gap to explore the potential of deep learning techniques for this task. This study seeks to bridge the gap by exploring how LSTM and Bi-LSTM models together with GloVe for vectorisation can accelerate this stage. As a further aim to increase precision while sustaining a recall >= 95% due to precision-recall trade-off, attention mechanics is added to these classifiers. The final experimental results obtained showed that Bi-LSTM with attention has the capacity to expedite citation screening.

Keywords:
Systematic literature review; abstract screening; artificial intelligence; machine learning; deep learning; LSTM; Bi-LSTM

1. Introduction

Systematic Review (SR), also known as *Knowledge Synthesis* (KS) or *Evidence-Based Medicine* (EBM) plays a crucial role by providing the highest form of evidence to guide policies and inform decision-making in medical research and beyond [1]. Thus, making it the "heart of evidence-based medicine research" [2]. This results from the

* Corresponding author. Tel.: +44 (0) 1224 262000
E-mail address: r.ofori-boateng@rgu.ac.uk

10.1016/j.procs.2023.08.149

orderly and structured steps involved in the SR process. In summary, the SR process begins with 1) the development of a protocol that highlights in detail how the succeeding steps in the process should be carried out. For example, the protocol outlines the number of researchers to be involved in the study, which databases will be queried, the type of checklist to be used for critically appraising potential studies, the exclusion and inclusion criteria, among others [2]. Following this is, 2) the definition of the research question using standardised models, 3) searching for all potential studies from sources or databases already penned down in the protocol. Upon retrieval of these studies from the various sources, 4) the title and abstract of each study found is screened to find all studies relevant to the research question based on the inclusion criteria, also known as *citation screening*, which is then followed by, 5) a full-text screening of these prospective relevant studies. To appraise the quality of the methodology of these studies, 6) risk of bias (RoB) assessment is performed on studies that successfully pass the full-text screening phase using a standard checklist [3], 7) data is extracted and synthesised from studies obtained from the RoB assessment and, 8) the results are interpreted, published, and reported.

This robust approach of an SR makes it transparent, reproducible, and comprehensive to include all potentially relevant studies [3]. However, the detailed structure constrains the process to be burdensome and time-consuming for researchers. They have an enormous duty to follow the SR's rigorous steps to publish a review. From literature, it has been reported that it takes about 15 months for an SR to be completed and published [4]. Another area of concern in medical research is the increasing daily rate of published articles, looking at the limited time researchers have for the SR process to be followed [5]. This has led to the issue of "missing data" [6], in that the majority of these recently published articles, which might have been relevant to a particular research question, will not be encompassed during the search phase. Thus, most SRs published become obsolete before they are completed [6].

Contrarily, this upsurge has led to the application of artificial intelligence techniques such as natural language processing (NLP), machine learning (ML), and deep learning (DL) to reduce the burden associated with SRs [7]. Among all the SR stages, the abstract screening step has been reported to be the most tedious [6]. As an illustration, studies have reported that it typically takes an experienced researcher between $30 - 90$ s to screen a single abstract [8] and between $8 - 125$ hrs generally to review an estimate of $5,000$ publication [9]. According to Shemilt et al. [10], when using double screening method (which is advised), it takes an estimated $4 - 5$ mins for researchers to resolve a debate over whether to include or exclude a piece of abstract. As a result, screening $10,000$ worth of publications can take between $100 - 150$ hrs [8].

On the other hand, the majority of recent studies that focus on citation/abstract screening automation approach it as a binary task, *text classification*, categorising abstracts as relevant or not to the research topic at hand. These approaches always aim at achieving a *recall* $\geq 95\%$ in order for the algorithms to include all potentially relevant literature [11]. However, as it is well-known in classification problems, a rise in recall results in a fall in precision, and vice versa. Nonetheless, achieving high precision is similarly important because it assures that the articles flagged as relevant are indeed pertinent to the research study. Another essential metric in screening automation that is affected by a high recall is *Work saved oversampling* (WSS) [11], which measures how much human burden the classifier can lessen at a particular recall.

Several approaches have been put forth in some text classification citation screening tasks to increase precision and WSS aside from training the model with abstract and/or titles. These include feature enrichment techniques e.g. addition of keywords, references, bibliometric features, Medical Subject Headings (MesH) [12]; integration of knowledge graphs such as Unified Medical Language System (UMLS) [13] and sampling techniques [14]. Though some techniques impacted precision, these research findings revealed a trend for precision to decrease as recall rose. Thus, there still remains a gap in exploring other techniques to improve these metrics. Additionally, from recent SR studies done on SRs automation techniques by van Dinter et al. [15] and O'Mara et al. [16], the study pinpointed that supervised ML is the most popular method for citation screening with the major algorithms being Support Vector Machine (SVM) and Naive Bayes (NB) classifiers. Also, their study identified Bag of Words (BoW) as the most deployed feature extraction method in SR citation automation. In conclusion, the SR studies revealed an aperture in the application of DL methods to automate abstract screening.

Nevertheless, recent studies have revealed the advantages of DL classifiers and word embedding techniques over these supervised ML methods in achieving higher performance metrics and capturing the contextual meaning of texts [17]. For example, in a comparative study by Meger et al. [18] on comparing SVM with BoW to recurrent neural networks (RNN) with word embedding techniques, the latter was found to achieve better performance metrics.

Thus, this paper aims to bridge the gap by exploring and evaluating the potential of DL models for citation screening, improving precision and WSS whilst maintaining a high recall. To the best of knowledge, one study by Moreno-García et al. [19] proposes DL techniques (zero-shot classification) for abstract screening. However, the study does not take into consideration measures to achieving a high recall, WSS and maintaining improved precision. To enable the deduction of the research questions **(RQ)**, we follow similar questions by Timsina et al. [14] in their study:

1. *Which deep learning models can be investigated to automate abstracts as compared to the most used ML techniques?*
   Variants of RNN, Long Short Term Memory (LSTM) [20], Bi-directional Long term Short Term Memory (Bi-LSTM) [21] will be explored. RNN is chosen because of its effortless ability to retrieve semantic information from the input data [17, 18]. LSTMs and Bi-LSTMs overcome the issue of long-term dependency in the original RNN model and have proven effective for sequential data tasks [22], of which the abstract is an example of data that occurs sequentially. Also, these models have achieved remarkable results in text classification tasks such as sentiment analysis, hate speech detection, and disaster prediction [23, 24].
2. *How can these models be used to achieve a recall >= 95% whilst having an improved precision and WSS score?*
   To do this, a threshold at which this recall would be attainable will be set. Additionally, the concept of attention mechanism [25] will be investigated on how its addition to the variants of RNN selected can improve precision and other metrics.

As a recommended approach to compare proposed methods to existing ones, this research uses the study by Bannach-Brown et al. [26] as the benchmark model. The study is selected because its proposed method was evaluated on a similar dataset to be used in this study. Furthermore, considering that class imbalance is one of the main challenges associated with SR automation [16], we propose using a cost-sensitive learning technique [27]. Finally, we perform and evaluate the proposed methodology on six health-related datasets. The results from the comparative experiments show that the proposed Bi-LSTM method with attention mechanics can noticeably aid in SR screening automation and lower the volume of items that need to be manually reviewed while maintaining a high recall. The outline of this paper is as follows. Section 2, highlights some related text classification approaches and the research gap; Section 3, describes the proposed methodology. In Section 4, we provide the results from our experiments and discuss these results, with Section 5 concluding the paper.

2. Related Works and Research Gap

Several initiatives have been made to approach citation screening as a text classification task using supervised ML. Cohen et al. [11], one of the earliest studies to have proposed recall >= 95% in SR automation, suggested the use of a voting perception-based classifier with a linear kernel as the classifier and BoW technique for feature extraction. To increase recall, the learning rate of the kernel was adjusted at different values. However, due to the precision-recall trade-off, the precision on some datasets were as low as 0%, as well as for WSS. One possible factor for these low scores could have been not addressing the issue of class imbalance in the dataset because a comparative study by Timsina et al. [14], using the method by Cohen et al. [11] as a benchmark presented employing data sampling techniques to improve precision using the same dataset. In their experiment, UMLS was used for vectorisation and softMax SVM was proposed as the best-performing classifier as opposed to the BoW presented in their benchmark model. The results obtained from the experiment showed that the combination of Synthetic Minority Oversampling (SMOTE) [28] with the undersampling technique obtained better precision and WSS score compared to the benchmark model.

Another supervised ML approach proposed by Bekhuis & Demner-Fushman [29] was to investigate how kNN, NB, and SVM with BoW can aid in the automation of SRs. To improve precision, they suggested combining each citation's title, abstract and metadata in the dataset. As an attempt to handle class imbalance, they proposed using a cost-sensitive classifier, Contemporary Naive Bayes(cNB). The experimental results showed that adding additional information to the abstract could reduce the human screening workload. Similarly, Almeida et al. [30] also explored how feature enrichment techniques and feature selection techniques could expedite citation screening. The researchers proposed using keywords and MeSH in addition to the title or abstract to train an ensemble method of logistic regression and decision tree. Their results showed that BoW with MesH and IDF (Inverse document frequency) as a feature extraction technique could aid citation screening. Furthermore, Olorisade et al. [31] also explored how references and bibliographies can improve performance metrics on nineteen datasets. For feature extraction, BoW and word2vec were explored. This study is perhaps one of the first to have explored a word embedding technique compared to the most popular BoW or TF-IDF. They proposed an SVM as a classifier evaluated using a 5×2 fold cross-validation. The results of their proposed method confirmed the study's aim and objective, having the potential to automate screening.

Likewise, Rúbio & Gulo [12] also explored how bibliometric features obtained during the search phase could help in automation. In their experiments, these features (the citation number, media type, publication number, etc.) were used to train a wide range of classifiers such as decision tree, kNN, SVM, and NB. Their experimental results proved that the addition of bibliometric features trained on the classifiers had the potential to reduce the human workload by improving precision. Additionally, Frunza et al. [13] also proposed using BoW and NB classifier for classification. To increase precision whilst attaining a high recall, the research question associated with each dataset was added to train the classifier alongside the UMLS knowledge graph. Their results proved that the addition of the research question was an alternative to improving precision. Finally, our benchmark study, Bannach-Brown et al. [26], also presented the use of a tri-gram with TF-IDF for vectorisation trained on an SVM with stochastic gradient descent (SGD). To prevent over-fitting, they used a five-fold cross-validation technique. Like all experiments, the results of their evaluation metrics highlight how their proposed method could enable screening automation. However, considering that the various techniques presented for citation text classification focus on supervised ML classifiers, there remains a gap in exploring DL techniques for citation classification tasks.

3. Methodology

This section describes the proposed approach intended to be implemented. This includes the datasets, preprocessing method, vectorisation, training, and evaluation metrics deployed. Pictorially the proposed pipeline is summarised in Figure 1. A detailed explanation of each of these steps is described in the subsequent subsections.

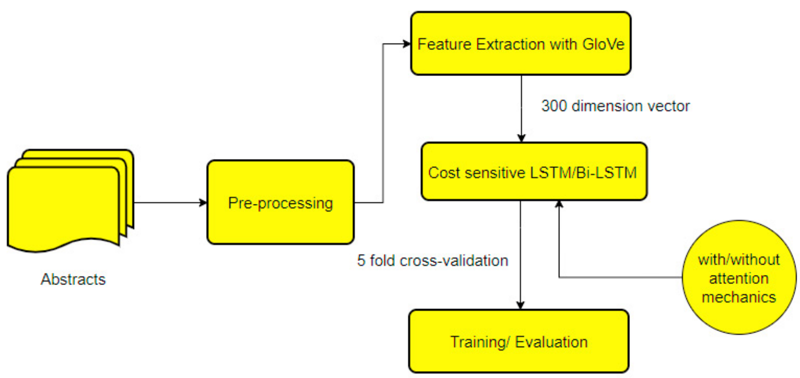


Fig. 1: Outline of the proposed method

Table 1: Overview of datasets used for training and evaluation

Name of dataset	**Topic**	**Total_papers**	**Included_papers**	**Excluded_papers**	**IR(Included:Excluded)**
Aceves-Martins2021 (AM)	Oral Health	807	18	789	1:44
Appenzeller-Herzog_2020 (AH)	Wilson Disease	3453	29	3424	1:118
Bannach-Brown_2019 (BB)	Animal Model of Depression	1993	280	1713	1:6
Cohen_AtypicalAntipsychotics_2006 (CAA)	ACEInhibitors	2544	41	2503	1:61
Cohen_ACEInhibitors_2006 (CACE)	Atypical Antipsychotics	1120	146	974	1:7
Cohen_OralHypoglycemics_2006 (COH)	Oral Hypoglycemics	503	136	367	1:3

3.1. Dataset and Pre-processing

To train and evaluate the proposed model, six health-related datasets were used. Five of the datasets are publicly available on Github [1]. These are the Bannach-Brown dataset [26] developed by authors of the benchmark study focusing on animal depression, the Appenzeller-Herzog dataset [32] on Wilson disease, the ACE Inhibitors dataset, Atypical Antipsychotics dataset, and Oral Hypoglycemics dataset developed by Cohen et al. [11]. The private dataset used is the Aceves-Martins dataset [33] focusing on oral health. An overview of all the datasets is recapitulated in Table 1.

Following the pipeline, from the dataset acquisition, the abstracts of each dataset were pre-processed by tokenising, removing stop words, and punctuation. Like the benchmark study [26], stemming/lemmatisation was not used since these could lead to the loss of some vital information in the data. To cite an example, stemming/lemmatisation will remove "s" from the word "trails" which gives a different meaning to the word in a randomised control trial (RCT) study like the Bannach-Brown dataset. This is because while "trails" is located in reports of SR of an RCT, "trail" mean the report of an RCT [26]. Thus, to prevent such an issue, this pre-processing method was avoided.

Compared to existing methods that deploy traditional feature extraction techniques, this study sought to explore how the use of word embedding techniques could aid in citation screening considering the advantages these embedding methods have over the traditional methods, such as semantics consideration and dimensionality reduction representation [34]. Though there are many state-of-the-art word embedding techniques [34], Global Vectors for Word Representation(GloVe) [35] is selected in this study because it is one of the most common techniques and offers word pairs the appropriate weights so that no word dominates the training process. Additionally, comparative studies have revealed its potential in text classification tasks [34, 31].

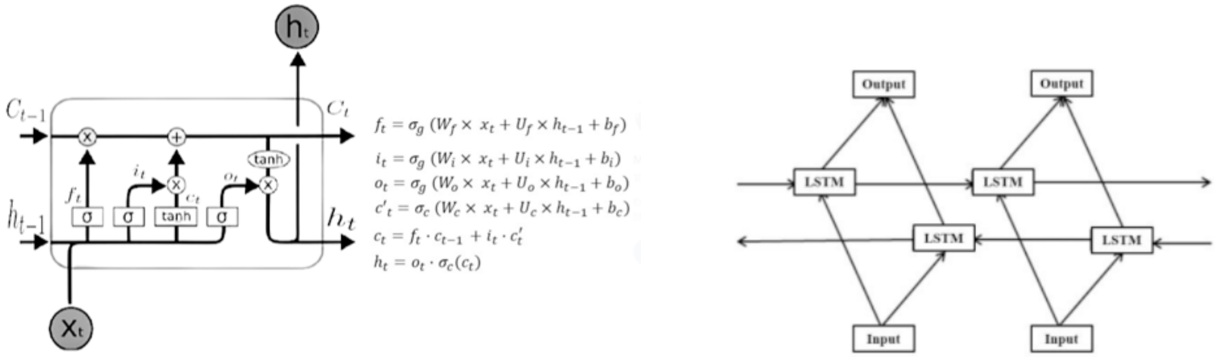
3.2. Vectorisation: GloVe

GloVe is an unsupervised statistical frequency-based technique for distributed word vector representation [35]. A detailed explanation of how this word embedding technique works is done in the study by Pennington et al. [35]. It creates these vector representations of text using a co-occurrence matrix. Mathematically, the co-occurrence is calculated using:

$$P(j|i) = \frac{X_{ij}}{X_i} \tag{1}$$

where P_{ij} is the likelihood that a word X_j will frequently appear in the context of a phrase X_i of interest. In this study, glove6b.zip [2], a pre-trained word vector was used. In selecting the vector's dimension size, a vital consideration made was that the acceptable length of abstracts is within 250-300 words. Thus, a 300-dimensional size vector is selected.

[1] https://github.com/asreview/systematic-review-datasets
[2] https://nlp.stanford.edu/projects/glove/



(a) LSTM structure with equations [22]

(b) Bi-LSTM [36]

Fig. 2: Architecture of LSTM and Bi-LSTM model

3.3. Learning models; LSTM and Bi-LSTM

Describing the LSTM and Bi-LSTM, the LSTM architecture consists of three main gates and a collection of cell states (c_t), enabling the network to overcome long-term dependency issues. These gates are the input (i_t), the forget (f_t) and the output (o_t) as seen in Figure 2a with their associated equations which govern the training of the model, where U is the hidden state matrix, W the weight matrix and b bias of the gates. These three gates control the flow of data that passes through the hidden layers of the cell to note the values from the preceding periods. Each c_t in the LSTM comprises of input X_t which in this study is the features extracted, the previously hidden state or output h_{t-1} and the preceding cell state c_{t-1}. The inital step of the gate is to decide what data from the cell state should be ignored using a *sigmoid*, σ_g function. At c_{t-1}, f_t reads h_{t-1} and new X_t, where the output is 0 (forget) or 1 (retain). The next stage of the LSTM is to locate which new data is stored in the cell. Thus, i_t determines what values to reform using σ_g. Additionally, the *tanh*, σ_c produces a vector $c_t^{'}$. c_{t-1} is then updated to a new c_t. The last step is for the output h_t of the LSTM to be determined by o_t using σ_g as seen in Figure 2a. Together these three gates help solve the long-term dependency of the original RNN model.

To further improve the performance of RNNs, the Bi-LSTM which is a hybrid of LSTM in both the forward and backward direction as seen in Figure 2b to represent a given sequence of data, was proposed [21]. The purpose of the reverse directional LSTM is to capture patterns that may have been neglected by the LSTM which fits data in only one direction [22].

3.4. Attention Mechanics

Attention mechanics is a concept in AI intended to mimic the human cognitive process by fastidiously focusing on a specific aspect of information in a given data. A useful application of RNN that led to the introduction of attention mechanics is *machine learning translation*. This kind of task is handled with sequential learning or encoder-decoder structure [37]. RNN encoder-decoder consists of two RNNs, one acting as the encoder and the other as the decoder. The purpose of the encoder is to convert information from the input sequence into a numeric representation known as the *hidden state* or *context vector*, passed unto the decoder to produce the output. This architecture's final hidden state results in a problem called an "information bottleneck" because the hidden state compresses the entire input sequence into a single, fixed representation. As such, in cases of an extremely long sentence, information at the beginning of the sequence may be lost since all that the decoder has access to when producing the output will just be a part of the hidden state. Hence, the introduction of attention mechanics enables the model to focus on the most vital information in the text as such can learn nontrivial alignments between the words concentrating on which input tokens are most important at each time step [38].

In this experiment, the attention output that governs the LSTM and Bi-LSTM training is generated using Equation 2 to Equation 5. In Equation 2, c_i denotes the context vector, h_j is the global features, α_{ij} is the weights calculated using a softmax function in Equation 3, the attention output score e_{ij} is calculated using Equation 4, where f is the function that encapsulates the alignment between the input x_t and the output, h_t is the hidden state, h_s is target and h_{t-1} is the previous hidden state [39].

$$c_i = \sum_{i=1}^{n} \alpha_{ij} h_j \tag{2}$$

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{n} exp(e_{ij})} \tag{3}$$

$$e_{ij} = f(h_t, h_s) \tag{4}$$

$$h_t = RNN(x_t, h_{t-1}) \tag{5}$$

3.5. Class Balancing Technique

Class imbalance is one of the major issues associated with SR citation automation, which significantly impacts the performance of the classifier [16]. Due to the enormous representation of excluded documents in the dataset compared to the number of documents belonging to the included class, classification models may learn more from the exclusion examples which biases the prediction. The most popular techniques deployed in citation automation tasks are data resampling, and cost-sensitive classifiers [40]. Whilst re-sampling techniques are applied at the data level, cost-sensitive techniques are applied at the algorithmic level [30]. In this study, cost-sensitive classifiers are used because of proven potential[30]; thus, weighted LSTM and Bi-LSTM by assigning weights to the majority and minority classes to fit the LSTM and Bi-LSTM layers with respect to the various dataset IR as seen in Table 1. For example, using the Aceves-Martin dataset, the IR is 1:44, thus the assigned weight to the exclusion (majority) was 1 whilst that of the inclusion (minority) was 44, hence assigned weight = {0:1, 1:44}.

3.6. Hyper-parameters

The following final hyper-parameters were selected for the experiments in this study. For word embedding, a 300-dimensional size was used. The hidden units found best for both LSTM and Bi-LSTM were 100 units with Adam as the optimiser for both models. The best performing learning rate for the Adam optimiser in the LSTM was 3×10^{-4} whilst that of the Bi-LSTM was 1×10^{-4}. These values were selected based on a series of experiments with different values yielding the best results. To regularise the classifier to prevent over-fitting, a recurrent dropout of 0.2 for both the LSTM and the BI-LSTM but a final dropout of 0.5 for the LSTM model and 0.02 for the Bi-LSTM was selected from series of values trails before passing it to the dense layer with sigmoid activation function for the binary classification. The best performing batch size was 64 across 10 epochs. This is recapitulated in Table 2.

3.7. Performance metrics

To evaluate the performance of the proposed model, the most common metrics in citation screening automation were used [16]. These are precision, recall, WSS@R, and F_2 score (in contrast to the most used F_1 score). F_2 is used considering the fact that recall is essential and needs more weight in citation screening [12]. Thus, the F_2 assigns more weight to recall and lesser weight to precision in the calculation. WSS @R gives an estimate of the reduction of the number of irrelevant articles the researcher won't have to manually screen because the model identified those. In SRs, the acceptable recall for WSS is 95% [11], despite the possibility of some "relevant" studies being absent (5%). Another rationale Yu et al. [41] offer for a recall of 0.95 is that no algorithm can guarantee a 100% recall prior to looking at all potential papers. Thus, this study reports WSS@95. This does not, however, disprove that WSS@100 has been reported in some citation screening studies [6]. To enable the evaluation, the underlying concepts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP is the relevant citations in the dataset that the classifier will correctly identify; TN is the number of irrelevant/negative citations correctly identified by the classifier. On the other hand, FP is the number of irrelevant citations that will be wrongly classified as relevant and

Table 2: Hyper-parameters for training

Hyper-Parameter	**Value**
Embedding dimension	300
LSTM units	100
Bi-LSTM units	100
Optimiser	Adam
Learning rate (optimiser)	3×10^{-4} for LSTM
Learning rate (optimiser)	1×10^{-4} for Bi-LSTM
Dropout for LSTM	0.2, 0.5
Dropout for Bi-LSTM	0.2, 0.02
Batch size	64
Epochs	10

the inverse is true for FN. The calculation of the evaluation metrics based on these underlying concepts is summarised in Table 3. Following the benchmark study's approach, the average of these metrics is reported over a 5-fold cross-validation.

Table 3: Summary of performance metrics

Experiment 1 (RQ1)	**Experiment 2(RQ2)**
SVM with SDG + TF-IDF (Baseline)	
LSTM +GloVe	LSTM + GloVe + Attention
Bi-LSTM + GloVe	Bi-LSTM + GloVe + Attention

3.8. *Experimental setup*

Two main experiments were performed in all. All codes were written in Python. The first experiment (Experiment 1) comprised two main methods (TF-IDF vectorisation and GloVe vectorisation). The main objective of Experiment 1 was to address **RQ1** by comparing the results of the baseline model and this study's proposed classifiers, LSTM and Bi-LSTM. In the TF-IDF vectorisation method, the baseline study's methodology was replicated on the six datasets. As stated in Section 2, the researchers in [26] proposed using tri-grams with TF-IDF as the feature extraction technique. These features were passed as input to a linear SVM with SGD. Recollecting, a high recall >= 95% is essential for SR citation automation classifiers. Thus, to implement this, a threshold at which this recall would be achievable was found in the original studies using the scikit-learn [3] package.

On the other hand, with the GloVe vectorisation method, which is this study's proposed method, GloVe was used for extracting 300-dimension size features from the abstracts which were passed as inputs to the LSTM network. The same approach was repeated, but the features were passed through a Bi-LSTM layer this time. These were implemented with the Keras [4] package together with the hyper-parameters as stated in Section 3.6. To achieve a high recall, a threshold at which the 0.95 recall was attainable using the Keras classification metrics module [5] was found

[3] https://scikit-learn.org/stable/modules/sgd.html
[4] https://keras.io/api/layers/recurrent_layers/
[5] https://keras.io/api/metrics/classification_metrics/

Table 4: Summary of results obtained from both Experiment 1 (TF-IDF and GloVe Vectorisation) and Experiment 2 (Attention Mechanics)

Dataset	**Classifier**	**TP**	**FP**	**TN**	**FN**	**N**	**P**	**R**	**F2**	**WSS@95**
	SVM with SDG + TF-IDF	2	81	64	1	149	2.64%	61.11%	11.27%	**39.09%**
Aceves-Martins2021 (AM)	Bi-LSTM + GloVe	4	145	0	0	149	2.68%	**100.00%**	12.12%	-5.00%
	LSTM + GloVe	4	145	0	0	149	2.68%	**100.00%**	12.12%	-5.00%
	Bi-LSTM + GloVe + Attention	4	116	29	0	149	**3.34%**	**100.00%**	**14.75%**	14.52%
	LSTM + GloVe + Attention	4	139	6	0	149	2.80%	**100.00%**	12.58%	-0.97%
	SVM with SDG + TF-IDF	5	423	44	0	472	1.17%	**100.00%**	5.58%	4.28%
Appenzeller-Herzog_2020 (AH)	Bi-LSTM + GloVe	5	329	138	0	472	1.50%	96.15%	7.05%	24.25%
	LSTM + GloVe	4	468	0	0	472	0.81%	**100.00%**	3.90%	-5.00%
	Bi-LSTM + GloVe + Attention	5	228	239	0	472	**2.15%**	96.15%	**9.86%**	**45.66%**
	LSTM + GloVe + Attention	4	270	198	0	472	1.32%	94.74%	6.23%	37.01%
	SVM with SDG + TF-IDF	48	235	35	2	320	16.90%	95.22%	49.42%	6.57%
Bannach-Brown_2019 (BB)	Bi-LSTM + GloVe	50	232	38	0	320	17.76%	99.60%	51.82%	6.94%
	LSTM + GloVe	50	270	0	0	320	15.70%	**100.00%**	48.21%	-5.00%
	Bi-LSTM + GloVe + Attention	49	112	157	2	320	**30.43%**	96.08%	**67.12%**	**44.72%**
	LSTM + GloVe + Attention	48	190	80	2	320	20.34%	96.41%	55.15%	20.58%
	SVM with SDG + TF-IDF	27	159	12	1	200	14.67%	95.14%	45.36%	1.51%
Cohen_AtypicalAntipsychotics_2006 (CAA)	Bi-LSTM + GloVe	29	156	15	0	200	15.58%	**100.00%**	48.00%	2.51%
	LSTM + GloVe	29	171	0	0	200	14.41%	**100.00%**	45.71%	-5.00%
	Bi-LSTM + GloVe + Attention	28	147	25	0	200	**15.79%**	**100.00%**	**48.39%**	**7.51%**
	LSTM + GloVe + Attention	28	149	22	1	200	15.90%	97.92%	48.19%	6.21%
	SVM with SDG + TF-IDF	8	311	124	0	443	2.39%	95.00%	10.84%	23.05%
Cohen_ACEInhibitors_2006 (CACE)	Bi-LSTM + GloVe	8	409	26	0	443	1.87%	97.50%	8.69%	0.92%
	LSTM + GloVe	8	435	0	0	443	1.81%	**100.00%**	8.42%	-5.00%
	Bi-LSTM + GloVe + Attention	8	234	202	0	443	3.11%	**100.00%**	13.82%	40.57%
	LSTM + GloVe + Attention	7	191	244	1	443	**3.64%**	90.00%	**15.65%**	**50.28%**
	SVM with SDG + TF-IDF	25	61	5	1	92	28.94%	95.42%	65.38%	1.49%
Cohen_OralHypoglycemics_2006 (CAOH)	Bi-LSTM + GloVe	26	60	5	1	92	30.39%	96.32%	67.18%	1.51%
	LSTM + GloVe	26	66	0	0	92	28.35%	**100.00%**	66.43%	-5.00%
	Bi-LSTM + GloVe + Attention	26	44	22	0	92	37.14%	**100.00%**	**74.71%**	18.91%
	LSTM + GloVe + Attention	23	35	31	3	92	**39.45%**	87.02%	70.11%	**32.31%**

through a series of experimentation similar to the benchmark study, thus addressing the first part of **RQ2** in Section 1. TP, TN, FP, FN were obtained at this threshold to aid in the calculation of the evaluation metrics.

In the second experiment (Experiment 2), which was designed to address the second aspect of **RQ2**, the addition of attention mechanics was explored on how it could help improve metrics such as precision and WSS of the DL classifiers whilst maintaining a high recall. Thus, in Experiment 2, attention biases with weights based on the output layers of the LSTM and Bi-LSTM model were defined, which were later added to the various layers of the DL models before the dense layer with sigmoid activation. In summary, the results from Experiment 1 and Experiment 2, respectively are summarised in with Table 4

4. Results and Discussion

From the results shown in Table 4, discussing Experiment 1, it was noticed that across all the six datasets, the proposed DL models were able to achieve a recall >= 95%. Also, from the table, it was observed that across five of the datasets, all the classifiers explored achieved a high recall except for the AM dataset, where the benchmark classifier earned a recall of 61.11%. In terms of the best-performing model for recall, the LSTM model proved to be the best, obtaining a recall of 100%. Though the LSTM achieved such an excellent recall, identifying all the positive examples in each dataset, it failed to determine the actual negative examples in each dataset, making it have a high value of FP, which greatly affected the WSS@95 (-5%). This implies that the LSTM model will fail to reduce human effort. Another inference that can be drawn could be that the cost-sensitive learning approach did not work so well for the LSTM model, making it highly biased. Moving on to the Bi-LSTM, it was also observed that the classifier achieved the highest score for precision, F_2 and WSS@95 across four of the six datasets (AH, BB, CAA, COH) performing better than the LSTM and benchmark model. For instance, with the AH dataset, the Bi-LSTM obtained 24.25% WSS@95; 19.97%higher than the benchmark model and 0.33% precision and 1.47% F_2 higher score than the benchmark method. Additionally, for the BB dataset, the Bi-LSTM obtained 0.37% higher WSS@95, 2.40% F_2, 4.38% recall and 0.85% precision compared to the baseline methodology. A similar trend was seen across the other two remaining datasets. On the other hand, the baseline classifier outperformed the LSTM and Bi-LSTM on the AM and CACE datasets obtaining a high WSS@95, precision, and F_2. In summary, considering the performance metrics results for the TF-IDF and GloVe vectorisation as summarised in Table 4, the Bi-LSTM model performs best in Experiment 1.

Having looked at the potential of LSTM and Bi-LSTM to automate SRs in Experiment 2 (Attention Mechanics), an observation from this Table 4 is that the addition of attention mechanics notably improved the performance metrics of the DL models, especially that of the original LSTM model. For example, Table 4 shows that the attention weights increased the WSS@95 value of both the LSTM and Bi-LSTM to a significantly higher value. For instance, with the AM dataset, the Bi-LSTM with attention increased the WSS@95 from -5% to 14.52% whilst maintaining a recall of 100% with improved F_2 and precision. A similar pattern is seen across the other datasets. Another example still on the AM dataset, is that the addition of the attention to the LSTM was able to improve the precision slightly, F_2 and WSS@95 (-5% to -0.97%) with the same improvement particularly in WSS@95 and precision across all the other datasets. Though the recall results of the LSTM for some of the datasets like COH and CACE were less than the expected recall, it was observed that the number of FP was lower than the original LSTM model and had higher TN, which aided in an improved WSS@95 value.

Additionally, from Experiment 2, it was also noticed that the addition of the attention mechanics to the Bi-LSTM was able to achieve a recall of approximately >= 95% across all the datasets. Overall, it can be concluded that the addition of attention mechanics addressed **RQ2**. Summarising the overall results for both Experiments 1 and 2, it can be generally concluded that the best-performing classifiers are the attention model-based models. Recalling from Experiment 1, where the benchmark model outperformed the DL model for the AM and CACE datasets, it was detected that the attention-based models gave comparative results. For example, with the AM dataset, the precision, recall, and F_2 score of the Bi-LSTM model with attention were higher than the results obtained with the baseline model with the exception of WSS@95. Likewise for the CACE, interestingly, the LSTM with attention model performs best in terms of precision, F_2 and WSS@95 though it misses one relevant example out of the eight. Reiterating and concluding the results obtained in Table 4, it can be highlighted that Bi-LSTM has the potential to aid in SR citation automation and that adding attention mechanics to the Bi-LSTM layers indeed helps improves essential performance metrics such as precision and WSS@95 in SR automation.

5. Conclusion

Putting it all together, the results from the experiments suggest that Bi-LSTM with attention mechanics has the potential to automate abstract text screening classification. From the theoretical point of the proposed methodology, this research inspects the likelihood of DL techniques for SR development. In prior studies, the most used vectorisation technique for extracting features from citations are BoW and TF-IDF. On the other hand, this study explores a word embedding method that captures semantics, GloVe. In addition, LSTM and Bi-LSTM are examined, with the addition of attention mechanics to improve evaluation metrics. The results obtained from experimentation are expected to create awareness to the public of the efficacy of diving into DL models for citation automation.

Moving to the practical perspective of the experiments, this study is anticipated to lower the cost of developing and maintaining SRs significantly. The high expense of selecting articles for SRs at the expense of the increasing rise in daily published literature prevents the development and update of SR from staying up with advancements in medical research, which in turn makes it more challenging to translate the most recent studies into healthcare practices. Thus the experimental methodology of this study has the potential to accelerate the implementation of SRs in evidence-based medicine by drastically lowering the number of papers that need to be manually reviewed by reviewers during the SR preparation process.

Some of the future works that can be explored are as follows: 1) investigating other potential class imbalance techniques, 2) exploring prospective word embedding techniques that are on a sentence level such as Doc2Vec [42] etc. to aid extract embedding from citations instead of word-level embedding technique used in this experiment, 3) exploring the addition of biomedical knowledge graphs terms such as UMLS to the proposed classifier for training and further exploring how domain knowledge i.e the content of the SR topic at hand can be incorporated into the classifier 4) evaluating method on a much larger dataset.

In conclusion, this research may impact the way best evidence-based medical research is conducted and ultimately contribute to improving society's health and well-being.

Acknowledgements

The authors would like to thank members of the COMO project [6] for supporting this research.

References

[1] Burns, Patricia B., Rohrich, Rod J., and Chung, Kevin C. (2011) ”The Levels of Evidence and Their Role in Evidence-Based Medicine.” *Plastic and Reconstructive Surgery*, **128**(1): 305–310. https://doi.org/10.1097/prs.0b013e318219c171

[2] Stevens, Kathleen. (2001) “Systematic Reviews: The Heart of Evidence-based Practice.” *AACN Clinical Issues* **12** (December): 529–538. https://doi.org/10.1097/00044067-200111000-00009

[3] Khan, K. S., Kunz, R., Kleijnen, J., and Antes, G. (2003) ”Five steps to conducting a systematic review.” *JRSM*, **96**(3): 118–121. https://doi.org/10.1258/jrsm.96.3.118

[4] Borah, Rohit, Brown, Andrew W., Capers, Patrice L., and Kaiser, Kathryn 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): 1–7. https://doi.org/10.1136/bmjopen-2016-012545

[5] Bornmann, Lutz, and Mutz, Rüdiger. (2015) “Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references.” *Journal of the Association for Information Science and Technology* **66** (11): 2215–2222. https://doi.org/10.48550/arXiv.1402.4578

[6] van de Schoot, Rens, de Bruin, Jonathan, Schram, Raoul, Zahedi, Parisa, de Boer, Jan, Weijdema, Felix, Kramer, Bianca, Huijts, Martijn, Hoogerwerf, Maarten, Ferdinands, Gerbrich, Harkema, Albert, Willemsen, Joukje, Ma, Yongchao, Fang, Qixiang, Hindriks, Sybren, Tummers, Lars, and Oberski, Daniel L. (2021) ”An open source machine learning framework for efficient and transparent systematic reviews.” *Nature Machine Intelligence*, **3**(February): 125–133. http://dx.doi.org/10.1038/s42256-020-00287-7

[7] Marshall, Iain J. and Wallace, Byron C. (2019) ”Toward systematic review automation: a practical guide to using machine learning tools in research synthesis.” *Systematic Reviews*, **8**(1). https://doi.org/10.1186/s13643-019-1074-9

[8] Howard, Brian E., Phillips, Jason, Tandon, Arpit, Maharana, Adyasha, Elmore, Rebecca, Mav, Deepak, Sedykh, Alex, Thayer, Kristina, Merrick, B. Alex, Walker, Vickie, Rooney, Andrew, and Shah, Ruchir R. (2020) “SWIFT-Active Screener: Accelerated document screening through active learning and integrated recall estimation.” *Environment International* **138**: 105623. https://doi.org/10.1016/j.envint.2020.105623

[9] Przybyła, Piotr, Brockmeier, Austin J., Kontonatsios, Georgios, Le Pogam, Marie Annick, McNaught, John, von Elm, Erik, Nolan, Kay, and Ananiadou, Sophia. (2018) “Prioritising references for systematic reviews with RobotAnalyst: A user study.” *Research Synthesis Methods* **9** (3): 470–488. https://doi.org/10.1002/jrsm.1311

[10] Shemilt, Ian, Khan, Nada, Park, Sophie, and Thomas, James. (2016) “Use of cost-effectiveness analysis to compare the efficiency of study identification methods in systematic reviews.” *Systematic Reviews* **5** (1). https://doi.org/10.1186/s13643-016-0315-4

[11] Cohen, A. M., Hersh, W. R., Peterson, K., and Yen, P.-Y. (2006) “Reducing Workload in Systematic Review Preparation Using Automated Citation Classification.” *Journal of the American Medical Informatics Association* **13** (2): 206–219. https://doi.org/10.1197/jamia.m1929

[12] R'ubio, Thiago RPM, and Gulo, Carlos ASJ. (2016) “Enhancing academic literature review through relevance recommendation: using bibliometric and text-based features for classification.” In *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6. https://doi.org/10.1109/cisti.2016.7521620

[13] Frunza, Oana, Inkpen, Diana, Matwin, Stan, Klement, William, and O'blenis, Peter. (2011) “Exploiting the systematic review protocol for classification of medical abstracts.” *Artificial intelligence in medicine* **51** (1): 17–25. https://doi.org/10.1016/j.artmed.2010.10.005

[14] Timsina, Prem, Liu, Jun, and El-Gayar, Omar. (2015) “Advanced analytics for the automation of medical systematic reviews.” *Information Systems Frontiers* **18** (2): 237–252. https://doi.org/10.1007/s10796-015-9589-7

[15] van Dinter, Raymon, Tekinerdogan, Bedir, and Catal, Cagatay. (2021) “Automation of systematic literature reviews: A systematic literature review.” *Information and Software Technology* **136**: 106589. https://doi.org/10.1016/j.infsof.2021.106589

[6] https://www.comoprojectmx.com/

[16] O'Mara-Eves, Alison, Thomas, James, McNaught, John, Miwa, Makoto, and Ananiadou, Sophia. (2015) "Using text mining for study identification in systematic reviews: a systematic review of current approaches." *Systematic reviews* **4** (1): 1–22. https://doi.org/10.1186/2046-4053-4-5
[17] An, Bang, Wu, Wenjun, and Han, Huimin. (2018) "Deep active learning for text classification." In *ACM International Conference Proceeding Series*, number 37. https://doi.org/10.1145/3271553.3271578
[18] Menger, Vincent, Scheepers, Floor, and Spruit, Marco. (2018) "Comparing deep learning and classical machine learning approaches for predicting inpatient violence incidents from clinical text." *Applied Sciences (Switzerland)*, **8**(6). https://doi.org/10.3390/app8060981
[19] Moreno-Garcia, Carlos Francisco, Chrisina Jayne, Eyad Elyan, and Magaly Aceves-Martins. (2023) "A novel application of machine learning and zero-shot classification methods for automated abstract screening in systematic reviews." *Decision Analytics Journal* **6**: 100162.
[20] Hochreiter, Sepp, and Jürgen Schmidhuber. (1997) "Long short-term memory." *Neural computation* **9** (8): 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
[21] Cho, Kyunghyun and Van Merriënboer, Bart and Gulcehre, Caglar and Bahdanau, Dzmitry and Bougares, Fethi and Schwenk, Holger and Bengio, Yoshua (2014) "Learning phrase representations using RNN encoder-decoder for statistical machine translation." *arXiv preprint arXiv:1406.1078*. https://doi.org/10.3115/v1/d14-1179
[22] Wu, Kuihua, Wu, Jian, Feng, Liang, Yang, Bo, Liang, Rong, Yang, Shenquan, and Zhao, Ren. (2021) "An attention-based CNN-LSTM-BiLSTM model for short-term electric load forecasting in integrated energy system." *International Transactions on Electrical Energy Systems*, **31**(1). https://doi.org/10.1002/2050-7038.12637
[23] Wang, Jenq-Haur, Liu, Ting-Wei, Luo, Xiong, and Wang, Long. (2018) "An LSTM approach to short text sentiment classification with word embeddings." In *Proceedings of the 30th conference on computational linguistics and speech processing (ROCLING 2018)*, pp. 214–223. https://doi.org/10.1109/aiam48774.2019.00014
[24] Zhou, Peng, Qi, Zhenyu, Zheng, Suncong, Xu, Jiaming, Bao, Hongyun, and Xu, Bo. (2016) "Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling." *arXiv preprint arXiv:1611.06639*. https://doi.org/10.1109/acpr.2017.113
[25] Bahdanau, Dzmitry, Cho, Kyunghyun, and Bengio, Yoshua. (2014) "Neural Machine Translation by Jointly Learning to Align and Translate." *arXiv*. https://arxiv.org/abs/1409.0473
[26] Bannach-Brown, Alexandra, Przybyła, Piotr, Thomas, James, Rice, Andrew SC, Ananiadou, Sophia, Liao, Jing, and Macleod, Malcolm Robert. (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. https://doi.org/10.1186/s13643-019-0942-7
[27] Elkan, Charles. (2001) "The foundations of cost-sensitive learning." In *International joint conference on artificial intelligence*, volume 17, number 1, pages 973–978. https://doi.org/10.1007/springerreference_178893
[28] Chawla, Nitesh V. (2009) "Data mining for imbalanced datasets: An overview." *Data mining and knowledge discovery handbook*, Springer, 875–886. https://doi.org/10.1007/0-387-25465-x_40
[29] Bekhuis, Tanja, and Demner-Fushman, Dina. (2012) "Screening nonrandomized studies for medical systematic reviews: a comparative study of classifiers." *Artificial intelligence in medicine* **55** (3): 197–207. https://doi.org/10.1016/j.artmed.2012.05.002
[30] Almeida, Hayda, Meurs, Marie-Jean, Kosseim, Leila, and Tsang, Adrian. (2016) "Data sampling and supervised learning for HIV literature screening." *IEEE transactions on nanobioscience* **15** (4): 354–361. https://doi.org/10.1109/bibm.2015.7359733
[31] Olorisade, Babatunde Kazeem, Pearl Brereton, and Peter Andras. (2019) "The use of bibliography enriched features for automatic citation screening." *Journal of biomedical informatics* **94**: 103202. https://doi.org/10.1016/j.jbi.2019.103202
[32] Appenzeller-Herzog C, Mathes T, Heeres MLS, Weiss KH, Houwen RHJ, Ewald H (2019) "Comparative effectiveness of common therapies for Wilson disease: A systematic review and meta-analysis of controlled studies." *Liver International* **39** (11): 2136–2152. https://onlinelibrary.wiley.com/doi/full/10.1111/liv.14179
[33] Aceves-Martins, Magaly and López-Cruz, Lizet and García-Botello, Marcela and Gutierrez-Gómez, Yareni Yunuen and Moreno-García, Carlos Francisco (2022) "Interventions to Treat Obesity in Mexican Children and Adolescents: Systematic Review and Meta-Analysis." *Nutrition Reviews* **80** (3): 544–560. https://doi.org/10.1093/NUTRIT/NUAB041
[34] Dharma, Eddy Muntina, Gaol, Ford Lumban, Leslie, H., Warnars, H.S., and Soewito, B. (2022) "The accuracy comparison among word2vec, glove, and fasttext towards convolution neural network (cnn) text classification." *J. Theor. Appl. Inf. Technol.*, **100**(2): 31.
[35] Pennington, Jeffrey, Socher, Richard, and Manning, Christopher D. (2014) "Glove: Global vectors for word representation." In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*: 1532–1543. https://doi.org/10.3115/v1/d14-1162
[36] Zhang, Y, Sun, J, and Wang, J. (2020) "Detecting driver distractions using a deep learning approach and multi-source naturalistic driving data." In *99th Annual Meeting*, volume 25.
[37] Sutskever, Ilya, Vinyals, Oriol, and Le, Quoc V. (2014) "Sequence to Sequence Learning with Neural Networks." *Advances in Neural Information Processing Systems*, **4**(January), pp. 3104–3112. https://arxiv.org/abs/1409.3215v3
[38] Zhang, Hao, Zhang, Qiang, Shao, Siyu, Niu, Tianlin, and Yang, Xinyu. (2020) "Attention-Based LSTM Network for Rotatory Machine Remaining Useful Life Prediction." *IEEE Access*, **8**, pp. 132188–132199. https://doi.org/10.1109/ACCESS.2020.3010066
[39] Bahdanau, Dzmitry, Cho, Kyung Hyun, and Bengio, Yoshua. (2014) "Neural Machine Translation by Jointly Learning to Align and Translate." In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. https://arxiv.org/abs/1409.0473v7
[40] Elyan, Eyad, Moreno-García, Carlos Francisco, and Jayne, Chrisina. (2020) "CDSMOTE: Class decomposition and synthetic minority class oversampling technique for imbalanced-data classification." *Neural Computing and Applications*. https://doi.org/10.1007/s00521-020-05130-z
[41] Yu, Zhe, Nicholas A. Kraft, and Tim Menzies. (2018) "Finding better active learners for faster literature reviews." *Empirical Software Engineering* **23** (6): 3161–3186 https://doi.org/10.1007%2Fs10664-017-9587-0

[42] Le, Quoc, and Tomas Mikolov. (2014) "Distributed representations of sentences and documents." In *Proceedings of the International Conference on Machine Learning*, 1188–1196. PMLR https://doi.org/10.48550/arXiv.1405.4053