Using a Neural Network-based Feature Extraction Method to Facilitate Citation Screening for Systematic Reviews

Georgios Kontonatsios[a,*], Sally Spencer[b], Peter Matthew[a],
Ioannis Korkontzelos[a]

[a] *Department of Computer Science, Edge Hill University, United Kingdom*
[b] *Faculty of Health and Social Care, Edge Hill University, United Kingdom*

Abstract

Citation screening is a labour-intensive part of the process of a systematic literature review that identifies citations eligible for inclusion in the review. In this paper, we present an automatic text classification approach that aims to prioritise eligible citations earlier against ineligible ones and thus reduces the manual labelling effort that is involved in the screening process. e.g. by automatically excluding lower ranked citations. To improve the performance of the text classifier, we develop a novel neural network-based feature extraction method. Unlike previous approaches to citation screening that employ unsupervised feature extraction methods to address a supervised classification task, our proposed method extracts document features in a supervised setting. In particular, our method generates a feature representation for documents, which is explicitly optimised to discriminate between eligible and ineligible citations. The generated document representation is subsequently used to train a text classifier. Experiments show that our feature extraction method obtains average workload savings of 56% when evaluated across 23 medical systematic reviews. The proposed method outperforms 10 baseline feature extraction methods by approximately 6% in terms of the $WSS@95\%$ metric.

[*] Corresponding author

Email addresses: georgios.kontonatsios@gmail.com (Georgios Kontonatsios), sally.spencer@edgehill.ac.uk (Sally Spencer), peter.matthew@edgehill.ac.uk (Peter Matthew), Yannis.Korkontzelos@edgehill.ac.uk (
Ioannis Korkontzelos)

Preprint submitted to Journal of Experts Systems with Applications *March 19, 2020*

Keywords: Citation Screening, Text Mining, Neural Feature Extraction

1. Introduction

Systematic reviews of the effects of interventions constitute the cornerstone of modern evidence-based medicine (Greenhalgh et al., 2014). High quality reviews, such as those produced by Cochrane, are frequently used to inform healthcare guidelines and to provide policy makers with the best and most up-to-date evidence on a specific medical topic (Volmink et al., 2004).

However, owing to the proliferation of the published literature (Bastian et al., 2010), the manual production of a systematic review has become a time-consuming process, with an average completion time of approximately 2.4 years (Bekhuis & Demner-Fushman, 2012). In addition, Shojania et al. (2007) reported that 23% of the published systematic reviews need to be updated with new relevant studies within 2 years from the time they are completed. In practice, this means that review authors are required to repeat the same resource-intensive tasks of the systematic review pipeline, such as literature searches, citation screening, data extraction and evidence synthesis, at regular intervals.

To reduce the average completion time of systematic reviews, we present a novel text mining method that semi-automates the citation screening task, i.e. a critical process of the systematic review pipeline that identifies relevant citations for inclusion in the review (O'Mara-Eves et al., 2015)). Our text mining method requires a seed of manually labelled citations to learn to discriminate between relevant/positive and irrelevant/negative instances. In succession, the trained classifier is used to automatically process the unlabelled citations, minimising the manual labelling effort that is associated with the citation screening task.

In practical application scenarios, our text mining method can be used to aid (human) systematic reviewers in screening more efficiently citations for inclusion in a review. More specifically, a human reviewer needs to manually label only a subset of the citations, i.e. a training dataset. This manually labelled subset of citations is used to train the underlying text classification algorithm, which is subsequently used to automatically label the remaining unlabelled citations. In addition to systematic reviews, our proposed method can be used in a wide range of different application areas relevant to expert and intelligent systems, such as information retrieval (Sethi & Dixit, 2015), text categorisation (Mirończuk & Protasiewicz, 2018), knowledge discovery (Bandaru et al., 2017) and recommendation systems (Wei et al., 2017). Text categorisation task, e.g. sentiment analysis (Cambria, 2016), constitutes a direct application area of our method. The conducted experiments demonstrate that the proposed method can substantially improve text classification performance. Moreover, our method could be integrated with text search engines, e.g. Apache Solr (Smiley et al., 2015), in order to learn to identify documents that are relevant to individual user preferences.

In the context of citation screening, existing text mining methods can be coarsely classified into: a) automatic text classification (Cohen et al., 2006; Frunza et al., 2010; Bekhuis & Demner-Fushman, 2012; Adeva et al., 2014) and b) automatic screening prioritisation (Cohen, 2008; Cohen et al., 2012, 2015; Howard et al., 2016) techniques. Both types of methods follow a similar approach to firstly train a supervised classification algorithm, e.g. Support Vector Machines (Wallace et al., 2010), Naive Bayes (Matwin et al., 2010), Random Forest (Khabsa et al., 2016), on a subset of the citations that are manually labelled with include/exclude codes by human reviewers. The trained classification algorithm is subsequently used to automatically process the remaining unlabelled citations. In an automatic text classification setting, the trained model is used to discriminate between eligible and ineligible citations in the unlabelled set and it can therefore be used to directly automate the underlying process. Although automatic text classification methods have been shown to achieve substantial workload savings (Cohen et al., 2006; Frunza et al., 2010), Bekhuis & Demner-Fushman (2012) noted that such methods may not always converge to a high recall performance of at least 95%, which is a key requirement of the citation screening task.

Automatic screening prioritisation techniques, including our proposed method, aim at re-ordering the citations in the unlabelled set so that citations that are likely to be eligible for inclusion in the review are ranked higher than ineligible citations (Howard et al., 2016). In contrast to automatic text classification approaches that frame the screening task as a binary classification problem, automatic screening prioritisation methods assign a classification confidence value to each citation rather than a binary label. The confidence value determines the likelihood of a citation being relevant to the review and it is used by the model to prioritise the unlabelled citations. Automatic prioritisation methods can reduce the screening workload, considering that human reviewers need to process only the top ranked citations, whereas the lower ranked citations are automatically excluded from the review (Cohen et al., 2015; O'Mara-Eves et al., 2015).

The vast majority of existing semi-automatic citation screening methods adopts unsupervised document representation techniques, such as bag-of-words, to address an inherently supervised classification task. Therefore, the induced feature representation of documents naturally ignores the readily available class-membership information of manually labelled citations. In this paper, we present a new supervised feature representation technique that leverages the class-membership information of the manually screened citations to generate informative document features. The proposed method uses a multi-layer feed forward neural network to learn a latent representation of documents that encodes discriminative and class-specific information about the citation screening task.

More specifically, our proposed feed forward neural network is trained on the manually labelled citations, while the hidden layers of the network are iteratively optimised to better discriminate between eligible and ineligible studies. We then extract an embedded feature representation of documents using the fixed weights of the hidden layers. The document embeddings can be integrated with any classification algorithm used for automatic screening prioritisation. Following previous approaches (Wallace et al., 2010; Cohen et al., 2015), we use a Support Vector Machine with a linear kernel to assign a classification confidence to each citation set and we rank the citation list in order of relevance to the review.

To further improve the performance of our neural network-based feature extraction method, we investigate pre-training techniques that aim to enhance the initialisation process of the feed forward neural network. In our approach, we employ a deep denoising autoencoder (Vincent et al., 2010), a type of unsupervised neural network that learns to denoise an artificially corrupted version of the input feature space. The reconstructed version of the input feature space is subsequently used to initialise the feed forward component of our method.

For evaluation, we conduct a series of experiments to investigate the performance of our supervised feature induction method when applied to the citation screening task of 23 publicly available systematic review datasets from the medical domain (Cohen et al., 2006; Howard et al., 2016). Experimental results demonstrate that our proposed feature extraction method can reduce the number of items that need to be manually screened without decreasing the sensitivity of the review, i.e. at least 95% of relevant studies are identified by the semi-automatic screening method. Moreover, our neural network-based feature extraction method shows substantial performance improvements when compared to 10 baseline feature extraction methods. The contributions of this paper can be summarised as follows:

1. We develop a new neural network-based feature extraction method to accelerate the citation screening task of systematic reviews.
2. We conduct large-scale experiments across a total number of 23 medical systematic reviews datasets to evaluate the effectiveness of the proposed method.
3. Our feature extraction method yields significant workload savings of at least 10% in 22 out of 23 review datasets.
4. Our method outperforms 10 baseline feature extraction methods by approximately 6%, in terms of the average workload saving (Cohen et al., 2006).
5. We make the source code of our tool publicly available at: `github.com/gkontonatsios/DAE-FF`.

2. Related Work

Prior work to semi-automatic citation screening, concerning both document classification and document ranking techniques, has investigated the use of different document representation techniques, such as bag-of-words (BoW), topic modelling, bibliographic metadata or a combination of the above, to improve the performance of the underlying text classification algorithm. Moreover, existing document representation techniques used for semi-automatic citation screening have been evaluated across a number of domain topics, including clinical medicine (Cohen et al., 2006; Wallace et al., 2010), social science (Miwa et al., 2014) and software engineering (Marshall & Brereton, 2013).

The BoW model is a standard document representation technique that has been widely adopted by previous semi-automatic citation screening methods (Cohen et al., 2006; Frunza et al., 2010; Kim & Choi, 2012; Wallace et al., 2010). In the BoW model, each document is represented as a sparse, high-dimensional feature vector, wherein the dimensions of the vector correspond to words or phrases that occur in the document. Bekhuis & Demner-Fushman

(2012) demonstrated that an automatic text classification method trained on BoW features achieved substantial workload savings of 35%-46% on two medical systematic reviews. Moreover, the authors showed that single-word features yielded an optimal performance when compared to bi-gram or tri-gram features, i.e. phrases consisting of two or three words, respectively. However, a limitation of the BoW model is that the resulting feature space consists of a large number of word-features and therefore the model is associated with increased memory and computational costs when applied to large-scale systematic review datasets (Forman, 2003).

Feature selection methods, e.g. forward feature selection (Cohen et al., 2015) or information gain filters (Bekhuis & Demner-Fushman, 2012), have been previously used to reduce the size of the BoW space, although Adeva et al. (2014) reported that such feature selection methods result in insignificant performance improvements.

Several studies proposed using bibliographic metadata to enhance the BoW space with additional features. Wallace et al. (2010) presented an automatic text classification system, trained via active learning, that used multiple feature types, including BoW, the publication type and indexing keywords. Each feature type was firstly used to train a text classification model. The screening decisions made by individual classification models were subsequently combined using a voting scheme. The experiments that they conducted showed that their ensemble classification model that exploited multiple feature types obtained a robust performance by reducing the screening workload of 4 medical reviews by 40%-50%. Although, bibliographic metadata can be used to improve upon the performance of the BoW feature space, Miwa et al. (2014) noted that such bibliographic features may not always be available for every citation or domain topic (e.g. social science). In response, the authors used an unsupervised topic modelling method, namely Latent Dirichlet Allocation (LDA) (Blei et al., 2003), to automatically identify latent topics in a collection of documents. Experimental results demonstrated that automatically identified topics can be used to complement potentially missing bibliographic metadata.

One-hot encoding feature extraction methods (e.g. as BoW features and bibliographic metadata) are strong baselines which yield a robust performance across a wide range of different classification tasks(Mirończuk & Protasiewicz, 2018). Moreover, one-hot encoding methods are easy to implement while the underlying feature model is highly interpretable(Wang & Manning, 2012). However, one-hot encoding methods are known to discard the order and the semantics of words and phrases (Mikolov et al., 2013; Le & Mikolov, 2014). In practice, this means that the text classifier, trained on one-hot encoding features, may yield a decreased performance due to the high ambiguity of the technical terminology used in complex, multi-disciplinary topics (e.g., public health) (Miwa et al., 2014; Hashimoto et al., 2016).

In a study closely related to our work, Hashimoto et al. (2016) presented a variation of the widely popular paragraph vectors (PV) model (Le & Mikolov, 2014), a document representation technique for extracting informative document features. The PV model is a neural network-based feature extraction method that follows a distributional semantics approach to better account for words and documents semantics. More specifically, the PV model trains a shallow neural network, consisting of one hidden layer, by maximising the conditional probability of a word given its context and the document that it appears. Hashimoto et al. (2016) modified the original implementation of the PV method in order to model each document as a distribution of latent topics. The authors further showed that their proposed PV method achieved a superior performance on complex, multi-disciplinary reviews when compared to the LDA topic modelling method. However, a limitation of the PV model is that it follows an unsupervised approach to feature representation and therefore the generated feature space is not explicitly optimised to discriminate between eligible and ineligible studies.

The main advantage of our method when compared to previous feature extraction methods is that it follows a supervised approach to extract discriminative document features. Moreover, our method generates a dense and low-dimensional feature space which is easier to manage when compared to the sparse and high-dimensional feature space produced by the BoW model. We further show that our supervised feature extraction method can enhance the performance of semi-automatic citation screening when compared to previously used unsupervised feature extraction methods, including BoW, bibliographic metadata, a low dimensional projection of the BoW space using the Singular Value Decomposition, a topic-based feature extraction method based on Latent Dirichlet Allocation (Bekhuis & Demner-Fushman, 2012; Miwa et al., 2014; Mo et al., 2015; Howard et al., 2016) and a topic-based feature induction method which exploits a shallow neural network (Hashimoto et al., 2016). Moreover, we report statistical significant improvements over the baseline methods in several review datasets.

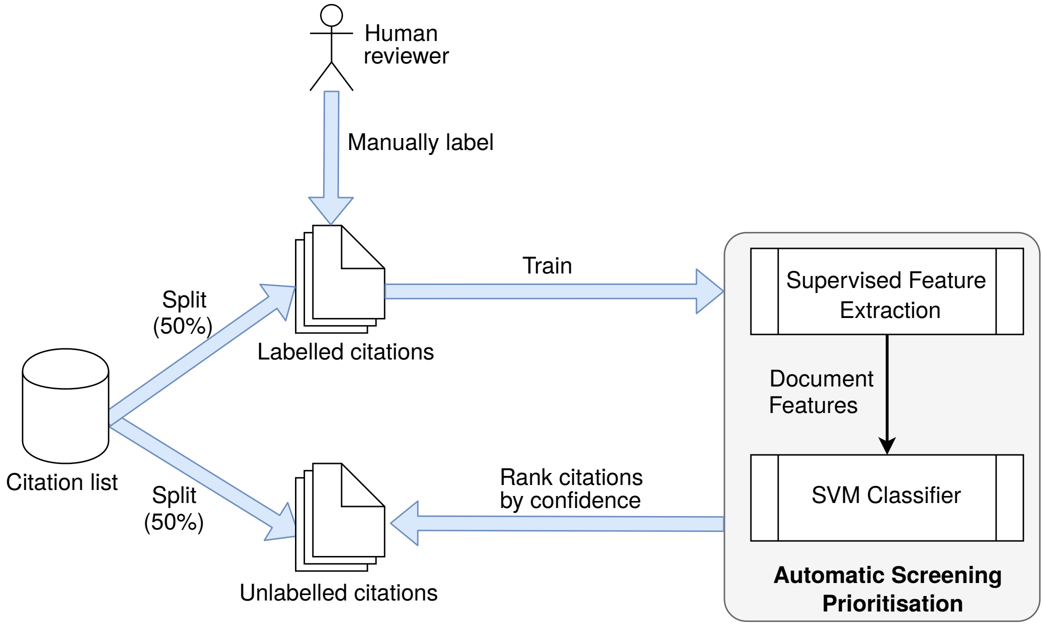


Figure 1: Architecture of the automatic screening prioritisation framework

3. Methodology

In this section, we detail the methodology that we follow to semi-automate the citation screening process of systematic reviews. Firstly, we describe the automatic screening prioritisation framework that we use to evaluate different feature representation methods. We then provide implementation details of our proposed neural network-based feature extraction method.

3.1. Automatic Screening Prioritisation Framework

Figure 1 shows the overall architecture of the automatic screening prioritisation framework that we use in our experiments. We follow the same experimental settings reported elsewhere in the literature (Cohen et al., 2006), by randomly partitioning the initial citation list into two equal sized sets, namely labelled and unlabelled. Both sets consist of 50% of the citations, whereas there is no overlap between the citations in the labelled set and the citations in the unlabelled set.

The labelled set is manually annotated by a human reviewer with include/exclude codes and it is used by the text classification method to learn to discriminate between eligible and ineligible studies. It should be noted that in our experiments we use publicly available datasets which were manually annotated with include and exclude codes in prior work Cohen et al. (2006); Wallace et al. (2010); Howard et al. (2016).

The text classification method firstly uses a feature extraction component to transform the textual content of citations into a numerical representation, i.e. feature vectors. In our approach, we develop a new supervised feature extraction method that uses a neural network model to generate a discriminative feature representation of documents. Document features extracted by our method are then used as input to a linear SVM classifier. The proposed supervised feature extraction method is described in the following section, Section 3.2.

The linear SVM classifier is trained to discriminate between eligible and ineligible citations, given the document features extracted previously. More specifically, a linear SVM constructs a linear hyperplane to best separate eligible from ineligible citations. After training the linear SVM model, we use the trained model to prioritise the citations in the unlabelled set, so that higher ranked citations are more likely to be eligible for inclusion in the review than lower ranked citations. More specifically, we rank the unlabelled citations according to the classification confidence of a citation being relevant to the eligible class. The classification confidence of a citation is computed based on the signed-margin distance of the feature vector for that citation to the SVM hyperplane, i.e. the higher the distance, the higher the classification confidence. Once the citations are prioritised in order of relevance to the eligible class, the top ranked citations are included in the review, whereas the lower ranked citations are deemed ineligible and they are thus automatically excluded from the review. Following previous studies Howard et al. (2016) we fix a cut-off threshold (i.e. minimum confidence value that discriminates between eligible and ineligible studies) at a recall level of 95%.

3.2. Supervised Feature extraction

Figure 2 illustrates the architecture of our supervised feature extraction method. The proposed method coordinates two types of neural networks: a) a denoising autoencoder and b) a feed forward network.

A denoising autoencoder (Vincent et al., 2010) aims to reconstruct the input BoW feature space given an artificially corrupted version of the BoW space. More specifically, consider $X = \{x^{(1)}, \cdots, x^{(i)}, \cdots, x^{(k)}\}$ a set of k input BoW feature vectors where $x^{(i)} \in \mathbb{R}^d$ is the BoW vector of the i-th citation. Each BoW feature vector consists of d word-dimensions, where each dimension corresponds to a word that appears in the title or in the abstract of a citation. The value of a word-dimension is the raw frequency of that word in a given citation.

Previous studies have demonstrated how a denoising autoencoder learns meaningful data representations by learning to remove the input noise in the data, in contrast to conventional autoencoders which are trained on cleaned input data (Vincent, 2011). Based on this, we artificially corrupt the input BoW feature using additive Gaussian noise of a standard deviation $\sigma = 0.5$, so that $\tilde{x}^{(i)}$ is the corrupted version of $x^{(i)}$.

The goal of an one-layer denoising autoencoder is to firstly *encode* the corrupted feature vector $\tilde{x}^{(i)}$ into a lower dimensional representation $y^{(i)} \in \mathbb{R}^h$ using the encoder mapping function:

$$y^{(i)} = f(W\tilde{x}^{(i)} + b) \tag{1}$$

where f is a non-linear activation function, such as the logistic sigmoid function, W is the weight matrix and b is the bias vector.

The encoded representation $y^{(i)}$ is then mapped back, i.e. decoded, into a BoW reconstruction $z^{(i)} \in \mathbb{R}^d$ through the decoder mapping function:

$$z^{(i)} = f(W'\tilde{y}^{(i)} + b') \tag{2}$$

The parameters $\{W, b\}$ and $\{W', b'\}$ of the encoder and decoder function, respectively, are optimised using the Adadelta optimiser (Zeiler, 2012), a variation of the stochastic gradient descent algorithm, by minimising the cross entropy of the reconstruction error according to:

$$L_H(X, Z) = -\sum_{i=1}^{k} x_i \log z_i + (1 - x_i) \log(1 - z_i) \tag{3}$$

In our approach, we use a straightforward variation of the one-layer denoising DAE, namely a deep DAE, which simply adds additional intermediate hidden layers into the network to learn more complex non-linear projections of the input data (Hinton & Salakhutdinov, 2006). Moreover, we use three different DAEs to learn potentially different reconstructions of the BoW space. The experiments that we conducted, presented in Section 4.5.3, demonstrate that a multi-branch model architecture that uses multiple DAE components obtains a statistically significantly better performance, in comparison to a single-branch architecture that uses only a single DAE component. Each DAE consists of 5 hidden layers, whereas we vary the dimensionality of the first and last hidden layer across the three DAEs to obtain

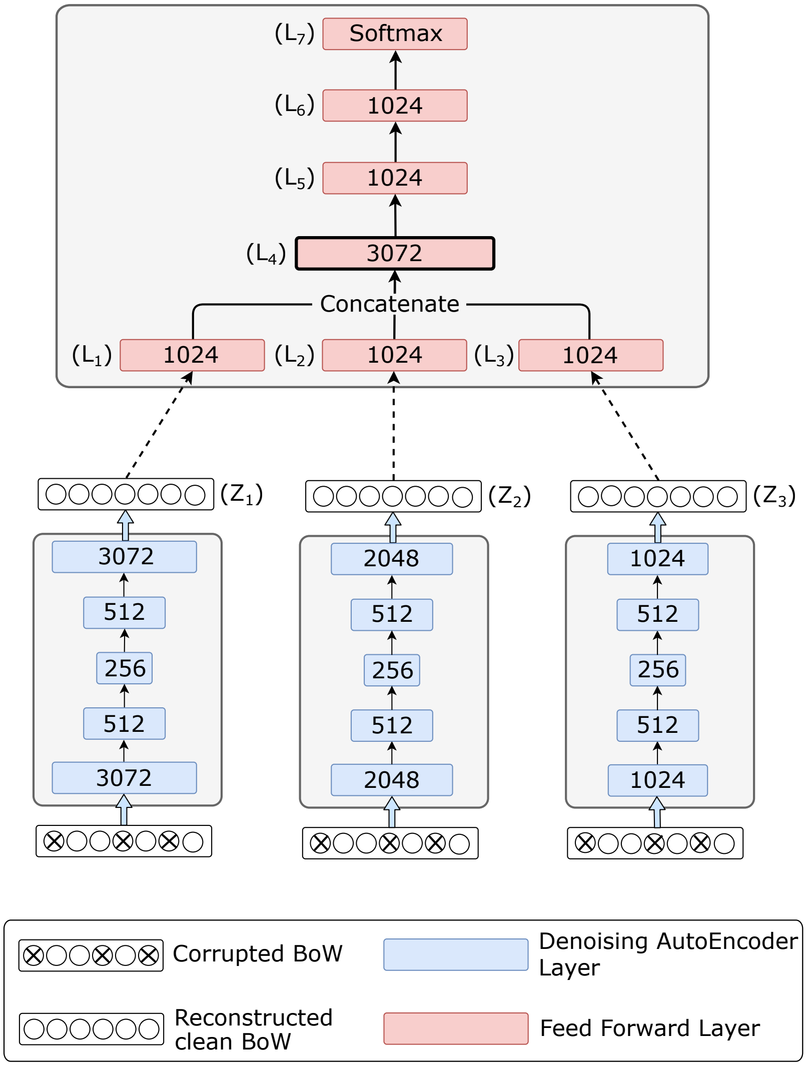


Figure 2: Architecture of the supervised feature extraction method

different reconstructions of the BoW space. The reconstructed output of each DAE is then used to initialise the supervised feed forward neural network. This type of unsupervised pre-training, where the feed forward neural network is initialised by deep DAEs, has been previously shown to substantially improve the performance of the feed forward network (Erhan et al., 2010).

The feed forward neural network consists of 6 hidden fully connected layers, i.e. $\{L_1, L_2 \cdots, L_6\}$, and an output softmax layer L_7 that is used to compute the probability distribution over the eligible and ineligible class for a given citation. The first three hidden layers of the network $\{L_1, L_2, L_3\}$ are parallel to each other, i.e. there is no connection between the units of the three layers, and they are initialised by the output reconstructions of the three DAEs, namely $\{z_1, z_2, z_3\}$, respectively. The three parallel layers are subsequently concatenated into a wide fully connected layer L_4 of 3072 units. Following the wide layer, we coordinate two additional hidden fully connected layers, i.e. L_5 and L_6, of 1024 units. We should further note that the size of the hidden layers is empirically defined. In section 4.5.3, we report the performance of the DAE component when using hidden layers of varying dimensionality.

The feed forward neural network is trained in a supervised manner by minimising the cross entropy between the probability distribution of the gold standard classes and the probability distribution of classes estimated by the softmax layer. The weights of the feed forward network are fine-tuned during training using vanilla stochastic gradient descent.

After training the feed forward network, we extract supervised feature vectors, that correspond to the whole set of the learned data, using the weight matrix of the wide fully connected layer L_4 according to:

$$h(z) = W_{L_4} \cdot [\sigma_{L_1}(z); \sigma_{L_2}(z); \sigma_{L_3}(z)] \tag{4}$$

where W_{L_4} is the weight matrix of the wide fully connected layer L_4, $[\cdot;\cdot]$ denotes feature concatenation and $\sigma_{L_1}(z)$, $\sigma_{L_2}(z)$ and $\sigma_{L_3}(z)$ are Rectified Linear Unit (ReLU) activation functions (Nair & Hinton, 2010) of the L_1, L_2 and L_3 hidden layers, respectively.

The extracted document vectors, i.e. the output of the connected layer L_4 of our proposed neural network-based feature extraction method, are subsequently used as input to a linear SVM text classifier. We further note that the feature extraction step is not dependent on the text classification model and similarly the text classification step does not rely upon the feature extraction method. Consequently, different feature extraction methods can be used with the same text classifier, and different text classifiers can be used with the same feature extraction method. In the context of this study, we seek to assess the performance of our novel feature extraction method and we therefore evaluate different baseline feature extraction methods against our proposed method using the same linear SVM text classifier.

4. Experiments

4.1. Data

We evaluate our proposed supervised feature extraction method on 23 publicly available systematic review datasets from the medical domain. Table 1 summarises the descriptive characteristics for each dataset, including a) the publication source of the dataset, b) the size of the dataset in terms of number citations that need to be screened, c) the percentage of eligible citations and d) the availability of bibliographic metadata. Each citation in the review datasets consists of a title, abstract and a classification label, i.e. eligible or ineligible, associated with that citation. Moreover, 18 out of 23 review datasets include additional bibliographic metadata for each citation in the form of Medical Subject Heading (MeSH) tags [1].

We further organise the 23 review datasets into the following 3 groups according to their publication source: a) clinical reviews (Wallace et al., 2010), b) drug reviews (Cohen et al., 2006) and c) SWIFT reviews (Howard et al., 2016). Both the clinical and the drug review datasets consist of a relatively small number of citations ranging from 310, for the *Antihistamines* review, to $4,751$ citations, for the *Proton Beam* review). The 5 SWIFT review datasets are substantially larger in size in comparison to the clinical and drug datasets, containing between $4,479$ and $48,637$ citations. Howard et al. (2016) noted that the SWIFT review datasets were constructed using broad search strategies, whereas the eligibility criteria of the reviews include multiple study designs (e.g. human/animal/in vitro clinical trials), which explains the large size of these reviews. The 3 clinical review datasets are relevant to clinical or health outcomes of different treatments (e.g. clinical outcomes of *Proton Beam* radiation treatment), while the 15 drug review datasets investigate the efficacy of drug therapies (e.g. *Skeletal Muscle Relaxant* treatment).

In order to tune the hyper-parameters of our feature extraction method, we used two development reviews, namely the *Statins* and the BPA dataset that consist of $3,465$ and $7,699$ citations, respectively.

4.2. Evaluation Settings

As evaluation metric, we use the widely adopted Work Saved over Sampling at $r\%$ recall (WSS@$r\%$) (Cohen et al., 2006; Frunza et al., 2010; Howard

[1] indexing terms used by the Medline bibliographic database

Source	**Dataset**	**# citations**	**(%) eligible citations**	**Bibliographic metadata**
Clinical (Wallace et al., 2010)	COPD	1,606	12.2	✗
	Proton Beam	4,751	5.1	✗
	Micro Nutrients	4,010	6.4	✗
Drug (Cohen et al., 2006)	ACEInhibitors	2,544	1.6	✓
	ADHD	851	2.4	✓
	Antihistamines	310	5.2	✓
	Atypical Antipsychotics	1,120	13.0	✓
	Beta Blockers	2,072	2.0	✓
	Calcium Channel Blockers	1,218	8.2	✓
	Estrogens	368	21.7	✓
	NSAIDs	393	10.4	✓
	Opioids	1,915	0.8	✓
	Oral Hypoglycemics	503	27.0	✓
	Proton PumpInhibitors	1,333	3.8	✓
	Skeletal Muscle Relaxants	1,643	0.5	✓
	Statins	3,465	2.5	✓
	Triptans	671	3.6	✓
	Urinary Incontinence	327	12.2	✓
SWIFT (Howard et al., 2016)	PFOA/PFOS	6,330	1.5	✓
	Bisphenol A (BPA)	7,699	1.4	✓
	Transgenerational	48,637	1.6	✓
	Fluoride and neurotoxicity	4,479	1.1	✗
	Neuropathic pain	29,207	17.2	✗

Table 1: 23 publicly available review datasets used in the experiments of this paper

et al., 2016; Kanoulas et al., 2017), which estimates the reduction of the (human) screening workload at a fixed recall level of r%. According to the WSS@r%, the workload reduction achieved by an automatic prioritisation method is equivalent to the percentage of citations that are ranked lower in the prioritised citation lisT, i.e. citations that are automatically excluded from the review and thus reviewers do not need to manually read those citations, than the cut-off threshold which is fixed at a recall level of 95%. The recall performance of the method is the proportion of eligible studies out of the total number of eligible studies that is ranked higher in the prioritised list. Thus, for a given recall performance of r%, the WSS@r% can be computed as follows:

$$WSS@r\% = \underbrace{\frac{TN + FN}{N}}_{\text{(\%) excluded citations}} - \overbrace{(1 - r)}^{\text{penalty term}} \tag{5}$$

where TN is the number of true negative predictions, FN the number of false negative predictions and N the total number of citations. The penalty term, i.e. $1 - r$, determines the proportion of citations that is falsely excluded from the review, i.e. eligible citations that are falsely ranked lower in the prioritised list.

Previous studies (Cohen et al., 2006; O'Mara-Eves et al., 2015; Wallace et al., 2010; Bekhuis & Demner-Fushman, 2012) noted that an acceptable recall performance of an automatic prioritisation method needs to be at least 95%. A lower recall performance of less than 95% may impact the quality of the underlying review considering that a substantial proportion of eligible studies is falsely excluded during the screening process. Based upon this, we fix the recall performance of our automatic prioritisation method at 95% and we compute the obtained work saved (i.e. $WSS@95\%$) according to:

$$WSS@95\% = \frac{TN + FN}{N} - (1 - 0.95) = \frac{TN + FN}{N} - 0.05 \tag{6}$$

In addition to the $WSS@95\%$ metric, we further compute the precision performance of our method at a fixed recall level of 95% according to:

$$precision@95\%recall = \frac{TP}{TP + FP}, \text{ given that recall=95\%} \tag{7}$$

For all evaluation tests, we report average values of the $WSS@95\%$ and $precision@95\%recall$ metrics over 10 cross-validation folds. More specifically, we follow a stratified 10 × 2 cross-validation setting. Each cross-validation round firstly partitions the initial dataset into two equally sized subsets. One subset is used for training and the second subset is used for computing the evaluation metrics. Both the training and the evaluation subsets consist of the same ratio of ineligible to eligible citations. We repeat this cross-validation process 10 times and we then average the 10 $WSS@95\%$ and $precision@95\%recall$ scores to obtain a final estimate.

4.3. Automatic prioritisation system

The automatic prioritisation system that we use in our experiments employs an L2-regularised linear SVM classifier to rank the citations according to the signed-margin distance between the citation feature vectors and the SVM hyperplane. We developed the linear SVM classifier using the Scikit-learn python library (Pedregosa et al., 2011). In order to better account for the class imbalance between eligible and ineligible citations, we used a reduced misclassification cost, i.e. a trade-off between maximising the margin between the two classes and minimising classification errors, by setting the regularisation parameter $C = 1 \times 10^{-6}$. We used the same hyper-parameter settings for the SVM classifier across all review datasets and across all feature extraction methods.

4.4. Baseline methods

Table 2 shows the hyper-parameter settings for 5 baseline feature extraction methods. With regard to the BoW method, we apply basic preprocessing steps following recommendations by Matwin et al. (2010). Firstly, we remove stop words found in NLTK's stop word list (Bird & Loper, 2004) and then we convert the original surface form of the words (e.g. *therapies*) into their corresponding base forms (e.g. *therapi*) using the Porter stemmer (Porter, 2001). After pre-processing the words that occur in the title and in the abstract of each citation, we construct BoW feature vectors consisting of the $10,000$ most frequent words in the collection. As feature values, we consider the frequency of occurrence of a word-dimension in a given citation. Feature weighting techniques, such as term frequency-inverse document frequency (tf-idf) weighting, could be potentially used to normalise word frequency values. However, Matwin et al. (2010) showed that such feature weighting techniques yield approximately the same performance as the unnormalised word frequencies on the drug development reviews.

The singular value decomposition (SVD) method is a dimensionality reduction technique that projects an input high dimensional feature space into a dense, lower dimensional space. SVD is different than the widely used Principal Component Analysis (PCA), as it computes eigenvalues and eigenvectors directly on the input data matrix, whereas PCA computes eigenvalues and eigenvectors on the covariance matrix of the input data (Wall et al., 2003). In the context of this study, we use the SVD baseline method to derive a low dimensional projection of the BoW feature space. The SVD baseline method is implemented using the Scikit-learn library. It should be

Baseline method	**Hyper-parameters**
BoW	Top (Stemmed) words: $10,000$
SVD	eigenvalues: 300
LDA	topics: 300, iterations: 500
PV	topics: 300, iterations: 500, document vector: $1,000$, word vector: 300
MeSH tags (bibliographic metadata)	—

Table 2: Baseline feature extraction methods with selected parameter settings used in the experiments of this paper.

noted that no prior work has previously evaluated SVD derived features for semi-automatic citation screening. We therefore identify optimal parameter settings, i.e. dimensionality of the projected space according to the top K eigenvalues, for the SVD method using a grid search method on the same two development reviews that we used to fine-tune our supervised feature extraction method. Experimental results showed that an SVD feature space of 300 dimensions (K=300) yields an optimal WSS@95% performance on both development reviews.

The two topic modelling methods, namely the Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and Paragraph Vectors (PV) (Hashimoto et al., 2016) model, represent each citation as a mixture of M latent topics. The LDA method is a popular feature extraction technique among existing semi-automatic citation screening systems. Here, we implement a baseline LDA method using the MALLET library (McCallum, 2002). We further tune the parameters of the LDA method by setting the number of latent topics to 300 and the number of iterations to 500 as in Miwa et al. (2014). The PV method (Hashimoto et al., 2016) is an alternative topic modelling feature extraction technique that it was previously shown to outperform the LDA method on three public health reviews. In our experiments, we use the publicly available implementation of the PV method[2]. Moreover, we use the same parameter settings as in (Hashimoto et al., 2016) by setting the

[2] nactem.ac.uk/pvtopic

dimensionality of the word embeddings to 300, the dimensionality of the document embeddings to $1,000$, the number of latent topics to 300 and the number of training iterations to 500.

MeSH tags are single word or multi-word keywords that are manually assigned to every citation indexed by the Medline bibliographic database (Lipscomb, 2000). MeSH tags aim at summarising the textual content of citations using a set of descriptive keywords. Considering that MeSH keywords may not always appear in the title or in the abstract of a citation, MeSH-based features can potentially provide complimentary information to BoW features (Trieschnigg et al., 2009). In order to retrieve MeSH tags from the Medline database, we use the Biopython library (Cock et al., 2009). We then construct binary feature vectors, where each dimension of the vectors corresponds to a different MeSH tag, while feature values determine the presence or absence of a MeSH tag in a given citation.

Previous studies investigated the performance of composite features consisting of a column-wise concatenation of different single-view feature spaces (e.g. BoW-LDA). As an example, Cohen et al. (2015) experimented with a combination of BoW and MeSH features and showed that such composite features achieve statistical significant improvements over single-view features (e.g. BoW features alone). Similarly, Howard et al. (2016) reported that BoW-LDA composite features enhance the $WSS@95\%$ performance of an automatic prioritisation method by approximately 4.4% when compared to single-view BoW features. Based upon this, in addition to the five baseline methods that extract single-view features, we report the $WSS@95\%$ performance of the following 5 composite features : BoW-LDA, BoW-SVD, BoW-LDA, BoW-PV, BoW-MeSH and BoW-SVD-LDA-PV.

4.5. Results

4.5.1. Hyper-parameter settings

In this section, we present experiments that we conducted to optimise the hyper-parameters and the network architecture of our supervised feature extraction method[3]. We do not perform any dataset-specific tuning of the hyper-parameters with the exception of the number of epochs required to train the DAEs. We show that the number of DAE epochs is sensitive to the size of the underlying review. Based on this, we use the *Statins* de-

[3] All experiments are performed using an NVIDIA TITAN Xp GPU.

Hyper-parameter	**Value**
size of minibatch (DAE)	32
dropout regularisation	0.7
number of training epochs (FF)	100
size of minibatch (FF)	128

Table 3: Hyperparameter settings of the supervised feature extraction method

velopment review to fix the number of DAE epochs across the 18 review datasets (i.e. clinical and drug reviews) which are relatively small in size while the BPA development review is used to tune the number of DAE epochs across the 5 SWIFT review datasets which are larger in size. The remaining hyper-parameters, which are summarised in Table 3, are constant across all datasets. After optimising the number of DAE epochs, we investigate the WSS@95% performance of our method when using different network architectures.

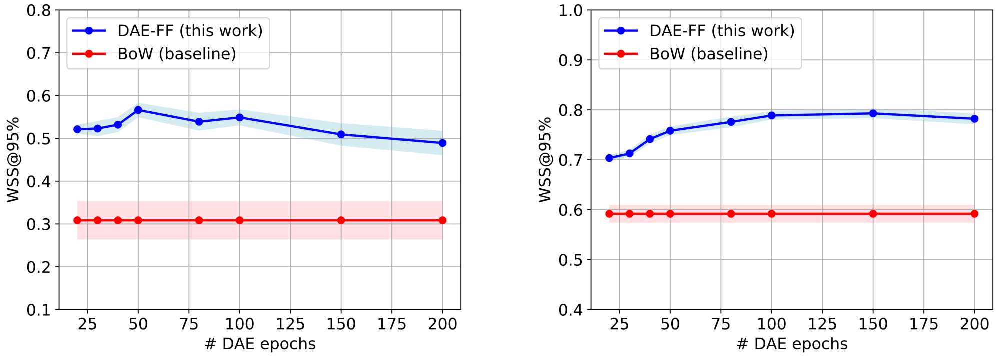
4.5.2. Effect of number of DAE epochs

Figures 3a and 3b illustrate the WSS@95% performance of our method, i.e. DAE-FF, on an increasing number of DAE epochs across the two development reviews, namely Statins and BPA, respectively. We further report the WSS@95% performance of the BoW baseline method. With regard to the Statins development review, we observe that the DAE-FF method yields a maximum WSS@95% score of 0.566 when using 50 DAE epochs. However, the performance of the method substantially decreases for a larger number of epochs, e.g. WSS@95% of 0.549 and 0.489 when using 100 and 200 epochs, respectively.

The WSS@95% performance of the DAE-FF method shows a different pattern on the larger BPA development review, when compared to the performance recorded on the smaller Statins review. Here, the performance of the method continuously improves as the number of DAE epochs increases. An optimal WSS@95% score of 0.792 is observed when training the DAE components for 150 epochs, whereas for 200 epochs the performance of the method slightly decreases to 0.782.

4.5.3. Effect of model architecture

We next investigate the performance of our method when using different model architectures. More specifically, we compare the performance of



(a) WSS@95% performance on the Statins development review

(b) WSS@95% performance on the BPA development review

Figure 3: WSS@95% performance of the proposed method (i.e. DAE-FF) on an increasing number of DAE epochs across the Statins and BPA development reviews. The figures also illustrate the WSS@95% performance of the BoW baseline method. The thick lines are average WSS@95% values. The bands surrounding the thick lines represent the 95% confidence interval of the mean WSS@95% values across 10 validation rounds.

a baseline model architecture (i.e. $model\_1$) that does not exploit unsupervised pre-training using deep DAE components against 6 model architectures that use different combinations of the three DAE components (i.e. DAE_1, DAE_2 and DAE_3) to initialise the feed forward network. We further evaluate both single-branch model architectures that use a single DAE component ($model\_2$, $model\_3$ and $model\_4$) and multi-branch architectures that use two ($model\_5$ and $model\_6$) or three ($model\_7$) DAE components. With regard to the single-branch model architectures, we co-ordinate 4 fully connected layers of 1024 units each following the single DAE component of the network. The two-branch model architectures consist of 5 fully connected layers: a) two fully connected layers of 1024 units which are parallel to each other and they are initialised by the two DAE components of the network, b) a wide fully connected layer of 2048 units (i.e. concatenation of the first two parallel layers) and c) two subsequent layers of 1024 units. Finally, our proposed three-branch model architecture (i.e. $model\_7$) co-ordinates 6 fully connected layers: a) 3 parallel layers which are initialised by the three DAE components, b) a wide fully connected layer of 3072 units and c) two layers of 1024 units.

Table 4 shows the WSS@95% performance of the 7 model architec-

Model	*DAE_1*	*DAE_2*	*DAE_3*	**WSS@95%**	
	(1024,512, 256,512,1024)	(2048,512, 256,512,2048)	(3072,512, 256,512,3072)	Statins	BPA
model_1	—	—	—	$.414^{**}$	$.687^{**}$
model_2	✓	—	—	$.514^{**}$	$.709^{**}$
model_3	—	✓	—	$.488^{**}$	$.697^{**}$
model_4	—	—	✓	$.492^{**}$	$.703^{**}$
model_5	✓	✓	—	.534	.786
model_6	✓	—	✓	.555	$.773^{*}$
model_7	✓	✓	✓	**.566**	**.792**

Table 4: WSS@95% performance of 7 different network architectures of our method (i.e. *model_1* to *model_7*) on the two development reviews. The superscript ** shows that the corresponding model obtained a statistically significant lower performance when compared to the WSS@95% performance of *model_7* according to a two-tailed paired t-test at $p < .01$ level. The superscript * denotes statistically significant difference at $p < .05$ level.

tures on the Statins and BPA development reviews. It can be observed that *model_1* (i.e. baseline architecture) obtains the lowest WSS@95% performance across the two development reviews. Our proposed three-branch model architecture, i.e. *model_7*, improves upon the performance of the baseline architecture by $\sim$ 10% to $\sim$ 15%. Moreover, *model_7* achieved a statistically significant improvement over the single-branch model architectures on both development reviews, although performance improvements over the two-branch model architectures were small and statistically insignificant in most cases.

4.5.4. Comparison with baseline methods

We evaluate our proposed three-branch DAE-FF method against 5 single-view baseline methods, i.e. BoW, SVD, LDA, PV and MeSH, on 23 review datasets. The results in Table 5 show that the DAE-FF method yielded an optimal WSS@95% performance in 16 out of the 23 review datasets, while the performance improvements over the baseline methods were statistically significant in 9 datasets. Moreover, our method obtained the best overall performance, i.e. the average WSS@95% scores across all 23 datasets, and improved upon the performance of the 5 baselines by $\sim$ 10% to $\sim$ 28%. The MeSH baseline method achieved the lowest performance, because MeSH terms are sparsely distributed across the different citations. The remaining 4

Dataset	**BoW**	**SVD**	**LDA**	**PV**	**MeSH**	**DAE-FF**
COPD	.458	.605	.555	.633	—	**.666**
Proton Beam	.746	.722	.787	.709	—	**.816****
Micro Nutrients	.510	.597	.430	.590	—	**.662***
ACEInhibitors	.752	**.791**	.548	.708	.375	.787
ADHD	**.744**	.712	.485	.481	.567	.665
Antihistamines	.048	.053	.042	.211	.192	**.310**
Atypical Antipsychotics	.136	.038	.076	.150	.199	**.329****
Beta Blockers	.470	.455	.507	.130	.237	**.587**
Calcium Channel Blockers	.177	.262	.234	.169	.130	**.424****
Estrogens	.288	.292	.360	.271	.238	**.397**
NSAIDs	.719	.698	.569	.593	.331	**.723**
Opioids	.304	.251	.350	.472	.116	**.533**
Oral Hypoglycemics	.081	.046	**.106**	.055	.065	.095
Proton PumpInhibitors	.239	.299	.293	**.503**	.323	.400
Skeletal Muscle Relaxants	.102	.186	.148	**.345**	.050	.286
Statins	.309	.306	.415	.293	.236	**.566****
Triptans	**.417**	.356	.331	.295	.241	.310
Urinary Incontinence	.291	.504	.443	.451	.220	**.531**
PFOA/PFOS	.773	.794	.797	.833	.405	**.848****
Bisphenol A (BPA)	.591	.709	.702	.629	.631	**.793****
Transgenerational	.619	.579	.612	.542	.432	**.707****
Fluoride and neurotoxicity	.719	.843	**.847**	.828	—	.799
Neuropathic pain	.471	.428	.534	.442	—	**.608****
Average (all datasets)	.433	.458	.442	.449	.277	**.564**

Table 5: $WSS@95\%$ performance of our method against 5 single-view feature extraction baselines. $WSS@95\%$ scores are averages across 10 validation runs for each of the 23 review datasets. The superscript ** shows that the DAE-FF method achieved a statistically significant better performance according to a two-tailed paired t-test over all 5 baseline methods at $p < .01$ level. The superscript * denotes statistically significant improvements over the 5 baselines at $p < .05$ level.

single-view baselines produced approximately the same average $WSS@95\%$ performance.

Table 6 compares the performance of the DAE-FF method against 5 composite feature extraction methods. The composite baselines augment the BoW feature space with additional features derived by different single-view feature extraction methods by column-wide concatenation. More specifically, we experiment with a column-wide concatenation of two single-view feature

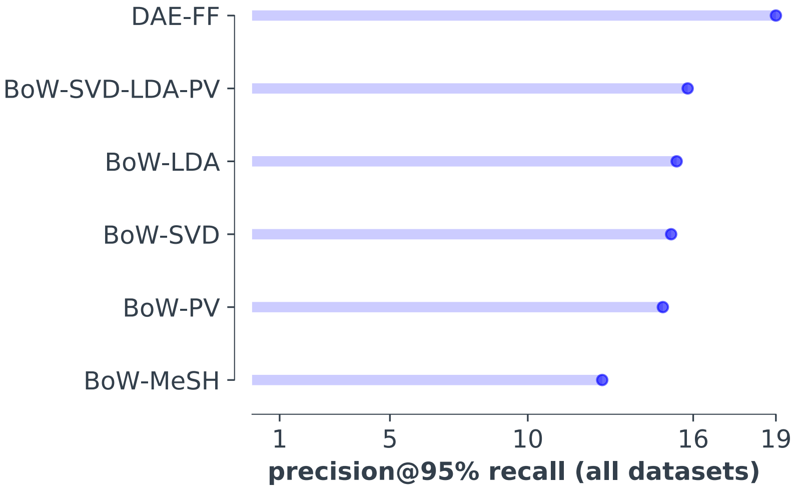


Figure 4: Average (all datasets) precision@95% recall of our method against 5 composite feature extraction methods.

spaces (i.e. BoW-SVD, BoW-LDA, BoW-PV, BoW-MeSH) and a column-wide concatenation of four single-view features spaces (BoW-SVD-LDA-PV).

The results show that the composite feature extraction methods improved upon the performance of the BoW single-view baseline. Performance gains in terms of the average WSS@95% range between $\sim$ 1% to $\sim$ 6%. The concatenation of LDA with BoW features (i.e. BoW-LDA) achieved the best average WSS@95% of 0.492 among the two-view composite baselines while the four-view composite method obtained a slightly higher average WSS@95% of 0.5 when compared to the BoW-LDA baseline. The DAE-FF method showed a superior WSS@95% score in 15 out of the 23 review datasets and a statistically significant improved performance over the 5 composite baselines in 7 datasets. Finally, our method increased the average WSS@95% score of the composite baselines by $\sim$ 6% to $\sim$ 11%.

Figure 4 shows the average precision at recall level of 95% obtained by our proposed DAE-FF across the 23 review datasets. Here, we observe that our method shows the best performance by outperforming the 5 composite feature extraction methods by 3.2% to 6.3%. The composite methods obtain approximately the same performance with the exception of the BoW-MeSH that shows a substantially lower average precision at recall level of 95% of $\sim$ 13%.

Dataset	BoW-SVD	BoW-LDA	BoW-PV	BoW-MeSH	BoW-SVD-LDA-PV	DAE-FF
COPD	.598	.609	.599	—	.640	**.666**
Proton Beam	.734	.778	.733	—	.772	**.816****
Micro Nutrients	.568	.416	.574	—	.607	**.662****
ACEInhibitors	.798	**.801**	.798	.773	.768	.787
ADHD	.719	.624	.719	**.738**	.633	.665
Antihistamines	.053	.229	.054	.273	.253	**.310**
Atypical Antipsychotics	.042	.152	.040	.134	.148	**.329****
Beta Blockers	.469	.532	.468	.552	.499	**.587**
Calcium Channel Blockers	.249	.308	.250	.398	.291	**.424**
Estrogens	.297	.300	.295	**.408**	.293	.397
NSAIDs	.699	.684	.698	.595	.692	**.723**
Opioids	.256	.318	.255	.332	.296	**.533**
Oral Hypoglycemics	.042	**.114**	.043	.112	.109	.095
Proton PumpInhibitors	.304	.302	.305	0.252	.345	**.400**
Skeletal Muscle Relaxants	.182	**.465**	.184	.318	.435	.286
Statins	.316	.364	.311	.252	.398	**.566****
Triptans	.366	.437	.361	.241	**.445**	.434
Urinary Incontinence	.500	.381	.504	.426	.362	**.531**
PFOA/PFOS	.819	.833	.796	.815	.826	**.848****
Bisphenol A (BPA)	.759	**.775**	.690	.717	.711	.758
Transgenerational	.598	.646	.576	.641	.644	**.707****
Fluoride and neurotoxicity	.835	.778	.835	—	**.849**	.799
Neuropathic pain	.484	.472	.441	—	.477	**.608****
Average (all datasets)	.465	.492	0.458	.450	.500	**.564**

Table 6: WSS@95% performance of our method against 5 composite feature extraction methods (i.e. column-wide concatenation of different single-view feature spaces).

4.6. Discussion

The results that we obtained demonstrate that our neural network-based feature extraction method substantially reduced the screening workload of 23 systematic reviews by approximately 56%. However, the workload savings varied across the 23 reviews from a low WSS@95% score of $\sim$ 9% on the Oral Hypoglycemics review to a higher WSS@95% score of $\sim$ 84% on the PFOA/PFOS review. Moreover, we observed a weak correlation ($R^2 = 0.279$)

between the $WSS@95\%$ performance and the size of the corresponding review dataset which was statistically insignificant ($p = 0.197$). This indicates that our method can obtain meaningful workload savings on both small and large review datasets.

According to Cohen et al. (2006), a significant and meaningful workload saving should be at least 10% in terms of the $WSS@95\%$ metric. This stems from the fact that the citation screening process of a systematic review, when conducted manually, requires on average 332 person hours to be completed. Therefore, a $WSS@95\%$ score of 10%, i.e. 10% of correctly excluded citations + 5% of incorrectly excluded citations, results in a workload reduction of $\sim$ 50 person hours, which according to expert reviewers is a significant reduction of their citation screening labour. The experiments that we conducted showed that our proposed feature extraction method yields significant workload savings of at least 10% in 22 out of 23 review datasets and thus it could be potentially used in practical application scenarios for accelerating the citation screening task of systematic reviews.

It should further be noted that the workload reduction (i.e. $WSS@95\%$ score) achieved by our method is relative to the size of the underlying review dataset. As an example, the DAE-FF method obtained approximately the same $WSS@95\%$ performance of 0.7 on both the NSAIDs and the Transgenerational dataset. However, the validation sample of the Transgenerational dataset consists of $24,318$ citations and it is substantially larger than the validation sample of the NSAIDs dataset (196 citations). In practice this means that a $WSS@95\%$ score of 0.7 is equivalent to a workload reduction of $18,238$ citations, which are automatically excluded from the Transgenerational review, while a $WSS@95\%$ score of 0.7 translates to a workload reduction of only 147 automatically excluded citations for the NSAIDs dataset.

4.7. Study limitations and future work

A potential limitation of our proposed method, which also applies to previous automatic screening methods, is that the $WSS@95\%$ metric assumes that an optimal cut-off threshold, i.e. the minimum value of the ranked list that discriminates higher ranked eligible studies from lower ranked ineligible studies, is pre-defined and fixed at 95% recall. However, in practical scenarios such a threshold value is difficult to define, considering that the optimal cut-off threshold varies greatly across different reviews (Howard et al., 2016). Here, threshold estimation techniques, such as the S-D rank optimisation (Arampatzis et al., 2009), can be used to approximate an optimal threshold value.

A second limitation of our method is that the underlying neural network-based feature extraction method is trained independently for each systematic review dataset. As an example, in our experiments we produced 23 neural network models corresponding to the 23 review datasets. However, different systematic reviews may share one or more more eligibility criteria (e.g. if included studies are randomised control trials) and thus learned document features could be applied to different reviews. As future work, we plan to investigate the use of domain adaptation and transfer learning in order to domain adapt a single feature extraction model across multiple reviews.

5. Conclusions

In this paper, we have presented a text classification method to accelerate the citation screening process of systematic reviews. The method aims to minimise the human workload involved in citation screening so that human reviewers need to manually label only a subset of the citations, while the remaining unlabelled citations are automatically labelled by the text classification method.

We have demonstrated that by initialising the feed forward neural network using multiple denoising autoencoders of varying dimensionality we can improve upon the performance of our feature extraction method. We have further performed a number of experiments to assess the performance of our method across 23 publicly available systematic review datasets. It was shown that for 22 out of 23 review datasets the proposed method achieved significant workload savings on at least 10%, while in several cases our method yielded a statistically significantly better performance over 10 baseline feature extraction methods.

Acknowledgements

This work has been carried out as part of the *TYPHON Project*, which has received funding from the *European Union's Horizon 2020 Research and Innovation Programme* under *Grant Agreement No. 780251.* Our research was further supported by the NVIDIA Corporation through the donation of a Titan GPU card.

References

Adeva, J. G., Atxa, J. P., Carrillo, M. U., & Zengotitabengoa, E. A. (2014). Automatic text classification to support systematic reviews in medicine. *Expert Systems with Applications*, *41*, 1498–1508.

Arampatzis, A., Kamps, J., & Robertson, S. (2009). Where to stop reading a ranked list?: threshold optimization using truncated score distributions. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval* (pp. 524–531). ACM.

Bandaru, S., Ng, A. H., & Deb, K. (2017). Data mining methods for knowledge discovery in multi-objective optimization: Part a-survey. *Expert Systems with Applications*, *70*, 139–159.

Bastian, H., Glasziou, P., & Chalmers, I. (2010). Seventy-five trials and eleven systematic reviews a day: How will we ever keep up? *PLoS Medicine*, *7*, e1000326.

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

Bird, S., & Loper, E. (2004). Nltk: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions* (p. 31). Association for Computational Linguistics.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, *3*, 993–1022.

Cambria, E. (2016). Affective computing and sentiment analysis. *IEEE Intelligent Systems*, *31*, 102–107.

Cock, P. J., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B. et al. (2009). Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, *25*, 1422–1423.

Cohen, A. M. (2008). Optimizing feature representation for automated systematic review work prioritization. In *AMIA annual symposium proceedings* (p. 121). American Medical Informatics Association volume 2008.

Cohen, A. M., Ambert, K., & McDonagh, M. (2012). Studying the potential impact of automated document classification on scheduling a systematic review update. *BMC Medical Informatics and Decision Making*, *12*.

Cohen, A. M., Hersh, W. R., Peterson, K., & Yen, P.-Y. (2006). Reducing workload in systematic review preparation using automated citation classification. *Journal of the American Medical Informatics Association*, *13*, 206–219.

Cohen, A. M., Smalheiser, N. R., McDonagh, M. S., Yu, C., Adams, C. E., Davis, J. M., & Yu, P. S. (2015). Automated confidence ranked classification of randomized controlled trial articles: an aid to evidence-based medicine. *Journal of the American Medical Informatics Association*, *22*, 707–717.

Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., & Bengio, S. (2010). Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, *11*, 625–660.

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of machine learning research*, *3*, 1289–1305.

Frunza, O., Inkpen, D., & Matwin, S. (2010). Building systematic reviews using automatic text classification techniques. In *Proceedings of the 23rd International Conference on Computational Linguistics* (pp. 303–311). Association for Computational Linguistics.

Greenhalgh, T., Howick, J., & Maskrey, N. (2014). Evidence based medicine: a movement in crisis? *Bmj*, *348*, g3725.

Hashimoto, K., Kontonatsios, G., Miwa, M., & Ananiadou, S. (2016). Topic detection using paragraph vectors to support active learning in systematic reviews. *Journal of Biomedical Informatics*, *62*, 59–65.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, *313*, 504–507.

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. (2016). SWIFT-review: a text-mining workbench for systematic review. *Systematic Reviews*, *5*.

Kanoulas, E., Li, D., Azzopardi, L., & Spijker, R. (2017). Clef 2017 technologically assisted reviews in empirical medicine overview. In *CEUR Workshop Proceedings* (pp. 1–29). volume 1866.

Khabsa, M., Elmagarmid, A., Ilyas, I., Hammady, H., & Ouzzani, M. (2016). Learning to identify relevant studies for systematic reviews using random forest and external information. *Machine Learning*, *102*, 465–482.

Kim, S., & Choi, J. (2012). Improving the performance of text categorization models used for the selection of high quality articles. *Healthcare Informatics Research*, *18*, 18.

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)* (pp. 1188–1196).

Lipscomb, C. E. (2000). Medical subject headings (mesh). *Bulletin of the Medical Library Association*, *88*, 265.

Marshall, C., & Brereton, P. (2013). Tools to support systematic literature reviews in software engineering: A mapping study. In *Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on* (pp. 296–299). IEEE.

Matwin, S., Kouznetsov, A., Inkpen, D., Frunza, O., & O'blenis, P. (2010). A new algorithm for reducing the workload of experts in performing systematic reviews. *Journal of the American Medical Informatics Association*, *17*, 446–453.

McCallum, A. K. (2002). Mallet: A machine learning for language toolkit, .

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems* (pp. 3111–3119).

Mirończuk, M. M., & Protasiewicz, J. (2018). A recent overview of the state-of-the-art elements of text classification. *Expert Systems with Applications*, *106*, 36–54.

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

Mo, Y., Kontonatsios, G., & Ananiadou, S. (2015). Supporting systematic reviews using LDA-based document representations. *Systematic Reviews*, *4*.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (pp. 807–814).

O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., & Ananiadou, S. (2015). Using text mining for study identification in systematic reviews: a systematic review of current approaches. *Systematic reviews*, *4*, 5.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. et al. (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, *12*, 2825–2830.

Porter, M. F. (2001). Snowball: A language for stemming algorithms.

Sethi, S., & Dixit, A. (2015). Design of personalised search system based on user interest and query structuring. In *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)* (pp. 1346–1351). IEEE.

Shojania, K. G., Sampson, M., Ansari, M. T., Ji, J., Doucette, S., & Moher, D. (2007). How quickly do systematic reviews go out of date? a survival analysis. *Annals of Internal Medicine*, *147*, 224.

Smiley, D., Pugh, E., Parisa, K., & Mitchell, M. (2015). *Apache Solr enterprise search server*. Packt Publishing Ltd.

Trieschnigg, D., Pezik, P., Lee, V., De Jong, F., Kraaij, W., & Rebholz-Schuhmann, D. (2009). Mesh up: effective mesh text classification for improved document retrieval. *Bioinformatics*, *25*, 1412–1418.

Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural computation*, *23*, 1661–1674.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, *11*, 3371–3408.

Volmink, J., Siegfried, N., Robertson, K., & Gülmezoglu, A. M. (2004). Research synthesis and dissemination as a bridge to knowledge management: the cochrane collaboration. *Bulletin of the World Health Organization*, *82*, 778–783.

Wall, M. E., Rechtsteiner, A., & Rocha, L. M. (2003). Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis* (pp. 91–109). Springer.

Wallace, B. C., Trikalinos, T. A., Lau, J., Brodley, C. E., & Schmid, C. H. (2010). Semi-automated screening of biomedical citations for systematic reviews. *BMC Bioinformatics*, *11*, 55.

Wang, S., & Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th annual meeting of the association for computational linguistics: Short papers-volume 2* (pp. 90–94). Association for Computational Linguistics.

Wei, J., He, J., Chen, K., Zhou, Y., & Tang, Z. (2017). Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications*, *69*, 29–39.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, .