Distribution Agreement

In presenting this thesis as a partial fulfillment of the requirements for a degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis in whole or in part in all forms of media, now or hereafter now, including display on the World Wide Web. I understand that I may select some access restrictions as part of the online submission of this thesis. I retain all ownership rights to the copyright of the thesis. I also retain the right to use in future works (such as articles or books) all or part of this thesis.

Signature:

Tiantian Li April 1, 2023

Improving Biomedical Abstract Screening Using Contrastive Learning

By

Tiantian Li

Joyce C. Ho Ph.D.
Advisor

Computer Science

Joyce C. Ho Ph.D.
Advisor

Judy W. Gichoya, Ph.D.
Committee Member

Li Xiong, Ph.D.
Committee Member

2023

Improving Biomedical Abstract Screening Using Contrastive Learning

By

Tiantian Li

Joyce C. Ho Ph.D.
Advisor

An abstract of
a thesis submitted to the Faculty of Emory College of Arts and Sciences of
Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Science with Honors

Computer Science

2023

Abstract

Improving Biomedical Abstract Screening Using Contrastive Learning
By Tiantian Li

Systematic review is a crucial tool for evidence-based medicine as it identifies and synthesizes published medical literature to inform prevention and intervention strategies. However, it requires intensive labor and time to identify relevant articles to include. While automating the screening process has been proposed using the abstracts, the performance is still suboptimal. Contrastive learning has achieved great success in computer vision but has not been used to expedite the systematic review process. In this thesis, we propose a new method using an autoencoder trained with contrastive loss to generate vector representation for abstracts. We apply data augmentation techniques on the abstract and train the autoencoder to generate representations for anchor and positive samples that are closer in vector space than those for anchor and negative samples. Our experiments suggest that contrastive learning can be used to help filter irrelevant articles during the abstract screening phase.

Improving Biomedical Abstract Screening Using Contrastive Learning

By

Tiantian Li

Joyce C. Ho Ph.D.
Advisor

A thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Science with Honors

Computer Science

2023

Acknowledgments

I am very thankful for Emory University for providing me with four years of high-quality education. Being in such a close-knit community, I have been exposed to a ton of opportunities to explore my interests and discover my potentials. I am especially grateful for Dr. Joyce Ho for her support and guidance through my junior and senior years of college. I have enjoyed the intellectual challenges she presented me both in her Database Systems class and in our research journey. I have grown a lot as a student and a researcher from working with Dr. Ho, who guided me through reading papers, conceptualizing ideas, to conducting experiments and reporting results. As an undergraduate student, I have had the rare and incredible opportunity to experience what research in Computer Science is like- a question constantly brought up to me by my peers. I would also like to thank Eric Lee, the Ph.D. student in Dr. Ho's group who has also provided me great support and resources along the processing of writing this thesis.

My interest in Machine Learning stemmed from the ML class I took with Dr. Xiong. Dr. Xiong not only introduced me to a variety of classification algorithms, but also trained me the fundamentals of conducting ML researches through the homework projects and the final projects where we had the chance to work with real-world datasets.

I am also truly thankful for Dr. Gichoya for her mentorship and support in me ever since I first joined the HITI lab in my Junior year. Dr. Gichoya has taught me a lot about pushing my limits and taking on responsibilities from putting me on lead on the ICU data curation project. Dr. Gichoya has also shown me great and genuine passion in healthcare research, which has motivated me to keep challenging myself and not giving up.

Contents

Chapter 1

Introduction

Systematic review is the process of identifying, selecting, and critically appraising all relevant primary research related to a well-formulated research problem. Systematic review bridges the gap between research and practice by providing all levels of decision and policy makers with comprehensive and up-to-date information. Specifically, in medicine, systematic reviews help to identify safer and more effective treatments for patients by synthesizing resources from a wide range of studies. While the systematic review is a crucial tool for evidence-based medicine, it has become increasingly challenging to conduct a comprehensive systematic review in a reasonable period of time due to the exponential growth in papers [26].

Abstract screening is one of the most laborious and resource-consuming steps in the conduction of a systematic review. To preserve the validity of the study and minimize selection bias, the primary search for relevant articles usually results in thousands of articles that need to be screened carefully in later steps. During the abstract and title screening step, researchers examine the abstract and title of each article and make decisions on whether to include the article or not based on the inclusion criteria, which can take up to months. With the exponential growth in new primary article production, it is essential for a systematic review to be conducted

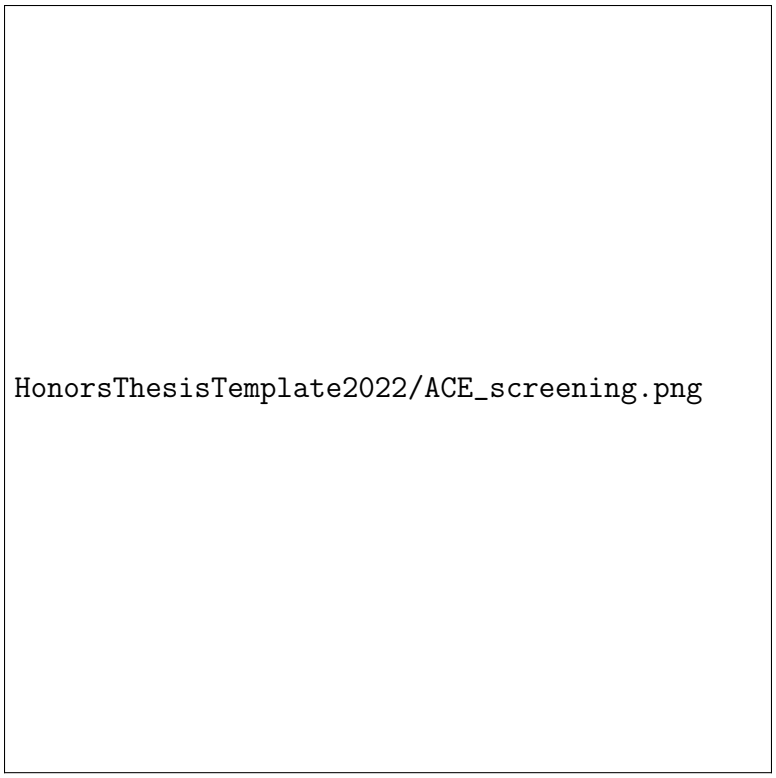


Figure 1.1: A simplified illustration of the screening process of a systematic review by Cohen *et al.*[5]

effectively and comprehensively in order to maintain timeliness and robustness. Figure 1.1 shows an example of the article screening process of a systematic review on angiotensin-converting enzyme inhibitors by Cohen *et al.*[5]. Only 7.2% of the articles collected from broad keyword searches passed abstract and title screening and only 1.6% of the total articles passed the full-text screening and were included in the final systematic review.

Numerous methods have been proposed to expedite the completion of a systematic review. Some focus on increasing the efficiency of labor workflow through crowdsourcing [26, 32], while others focus on automating the screening process using text mining approaches [41]. Under the latter paradigm, a common setup is for a human reviewer to manually label a subset of the articles to train the machine learning model which can then classify the unlabeled articles. Although deep learning approaches have be-

come the de-facto architecture in natural language processing (NLP), limited studies have explored its viability for automating the screening process [16, 40]. Even then, it remains unclear if such complex models are practical [18].

One of the major disadvantages of a deep learning model is the need for large amounts of labeled, training data. To overcome this, contrastive learning has been proposed to generate (self-supervised) augmentations of the input data to learn more robust representations. Contrastive representation learning is a technique of generating a useful representation of data by comparing similar and dissimilar samples. The goal of training a contrastive learning-based model is to learn a mapping such that representations of similar samples are pushed together in the embedding space while representations of dissimilar samples are pulled away. Contrastive learning has achieved great success in computer vision [19], but the benefits of text mining methods still remain unknown [34]. We posit that contrastive learning can be utilized to yield better screening automation tools without requiring additional labeled data.

To mitigate the need for labor-intensive manual systematic reviews as well as minimize the amount of annotated samples needed to train a deep learning model, we propose CTRL-Screener, a self-supervised contrastive learning-based model. CTRL-Screener is used to eliminate irrelevant studies based on the abstract only. CTRL-Screener consists of a pre-trained autoencoder model and a softmax activation layer. The pre-trained autoencoder model is trained with unlabelled data from a variety of systematic review topics, and the softmax layer is trained independently for each systematic review. CTRL-Screener requires a percentage of the data to be manually labeled to learn to distinguish between relevant and irrelevant samples, then it can be used in the abstract screening phase of systematic review to reduce the human effort required.

The main contributions of this thesis are:

1. We propose CTRL-Screener, a self-supervised contrastive learning-based model,

that can be used to expedite the systematic review process. To the best of our knowledge, we are the first to investigate the use of contrastive learning for screening articles.

2. We introduce text augmentation techniques that preserve the structural integrity of the abstract. The resulting article representations improve the predictive performance of the classifier.

3. We discuss the effect of different variations of the text augmentation techniques and report the results.

4. We conducted experiments on 19 systematic review topics, demonstrating the potential of CTRL-Screener towards filtering out irrelevant articles at the abstract screening phase. We compared the performance of CTRL-Screener against other state-of-the-art language models including fastText and Sentence-BERT.

The rest of this thesis is structured as follows. Chapter 2 introduces some of the work that has been done in the fields of automating systematic reviews and contrastive learning. Chapter 3 elaborates on the architecture and the implementation details of the proposed model CTRL-Screener. Chapter 4 introduces the experiment set-up, including the datasets used and evaluation methods. Chapter 5 reports and discusses the results of CTRL-Screener and baseline models. Lastly, Chapter 6 concludes our contribution to systematic review automation, and discusses future research directions.

Chapter 2

Background

In this section, we briefly review relevant works for automating systematic reviews and contrastive learning approaches in NLP.

2.1 Automating Systematic Reviews

Multiple studies have introduced text mining methods to help automate the screening process (see [41] for a recent systematic review on this topic). Most of the prior work done on automating the screening process of systematic reviews with supervised or semi-supervised machine learning models extract features from title and abstract. Full-text features are often avoided as full-text documents require a lot of natural language pre-processing steps. Even before that, reliable conversion from pdf to text is required as most full-text documents are only available in pdf format [40].

2.1.1 Word Embeddings

Existing studies have explored a variety of classifiers such as support vector machines, Naïve Bayes, decision trees, and k-nearest neighbors in conjunction with bag-of-word representations for text [4, 15, 29]. More recently, word embeddings have emerged as

a popular text representation for automating systematic reviews [20, 22, 31]. Word embeddings seek to learn a dense, low-dimensional vector representation that simultaneously captures the semantic properties of the words. Amongst neural network-based word embeddings, the three popular models include word2vec [24], Global Vectors for Word Representation (GloVe) [30], and fastText [3]

While neural network based word embedding models preserve semantic meaning of words, they do not encode contextual information. Transformer-based word embedding models have been introduced to better capture semantics on sentence level. BERT[14] was the first model to pre-train deep bidirectional representations (i.e. conditioning on both previous and latter context), leading to better performance on tasks such as natural language inference and paraphrasing. Sentence-BERT[33], as a modification on BERT, generates semantically meaningful sentence embedding where semantically similar sentences have closer vector representations in the vector space.

2.1.2 Deep Learning Models

Given the impressive performance of deep learning for various natural language processing tasks [27], two deep learning algorithms using denoising autoencoders and a multi-channel convolutional neural network were proposed to automate the systematic review process [16, 40].

In the work by Kontonatsios *et al.* [16], the authors proposed a multi-layer, feed-forward neural network that learns a latent representation of documents that preserves class-membership information (i.e., the label of the document) through supervised learning. The generated embeddings from the proposed feature extractor can be integrated with any classification algorithm to predict whether an unseen document is relevant to the topic or not. More specifically, the feature extractor in the proposed model consists of three independently trained denoising autoencoders (DAE) and a multi-layer, feed forward neural network (FF). Each DAE is trained independently to

accomplish the task of reconstructing manually corrupted bag-of-word embeddings of text documents. The purpose of training three DAEs in parallel is to generate different reconstructions of the same document. The empirical experiments showed that the combination of different reconstructions led to better results. Then, the reconstructed vector representations of the DAEs are used to initialize the first three layers (l_1, l_2, l_3) of the feed forward neural network. l_1, l_2, and l_3 are parallel layers, meaning that there are no connected units in between. The outputs of l_1, l_2, l_3 are concatenated and fed to the next layer l_4. The final layer of the feed forward network is an activation layer using softmax function to predict the probability distribution of each class. The feed forward network is trained in a supervised manner to minimize the cross entropy between the predicted probability distribution and the true class distribution. After the supervised training, the output vector of l_4 is calculated again and used as the extracted feature vector, which can be incorporated with any classification models (i.e. support vector machine) to predict the probability distribution of target classes.

The second study [40] focuses on using a multi-channel convolutional neural network (CNN) to automate the title and abstract screening process. In the data preprocessing stage, title and abstract of a document are concatenated together and passed to pre-trained GloVe word embeddings [30] to generate vector representations. The CNN structure is used as it can obtain important text features from the document, such as finding keywords and phrases, and is more cost-efficient than Long Short-Term Memory (LSTM) models. The word embeddings are passed to a series of CNN blocks of CNN layers followed by global max pooling. The study experimented with different numbers of channels and different kernel sizes and concluded that a 2-channel CNN with kernel sizes of 2/4 achieved the best performance.

A recent literature review study [18] replicated the experiments performed in the aforementioned two studies and reported the performance comparison between the two proposed models. In addition, the paper presented a simpler model using the

average of the fastText word embedding [3] which yields higher performance on 18 out of the 23 datasets than the performance of the replicated model from the second paper and comparable performance against the replicated model from the first paper. The fastText model is also, on average, 72 times faster than the multi-channel CNN model proposed in the second study and 8 times faster than the DAE-FF model proposed in the first paper.

2.2 Contrastive Representation Learning

Contrastive learning models learn better representations through the comparison process. The main goal of contrastive representation learning is to generate representations of sample documents so that pre-defined 'similar' samples are closer together in vector space while 'dissimilar' samples are farther away from each other. The idea behind contrastive learning is to use similar and dissimilar samples (often generated by data augmentations) to learn a more robust representation [19]. Depending on the end goal, the notion of similarity and dissimilarity can be very different from case to case. In a multi-modal setting, similar samples can be different views of the same context; in object recognition tasks, similar samples can be different rotations of images of the same object.

Contrastive learning self-supervised training objectives have yielded state-of-the-art image representations in computer vision [11, 25]. Yet the benefits of contrastive learning in NLP are still relatively understudied. Examples of relevant work include investigating sentence augmentation techniques [23, 46], using back translation [7], or contrasting text spans [9].

A recent application of contrastive self-supervised learning in NLP field was introduced in [7]. The study proposed a contrastive self-supervised learning model CERT, which fine-tunes established language representation model (e.g. BERT [14]) to bet-

ter capture sentence-level semantics. Back-translation technique, first introduced in [6], was used to augment the dataset and generate positive and negative pairs. For a given sentence s_1 in English, we can use an English to Spanish translation model to generate s'_1 in Spanish. Then we pass s'_1 through a Spanish-to-English translation model and get s''_1 in English. The choice of intermediate language is arbitrary as long as there exists a developed translation model between the source language and the intermediate language. We can augment the dataset by repeating the back-translation technique for all sentences in the training set. s_i and s''_j are a similar pair if and only if $i = j$. s_i and s''_j are a dissimilar pair if and only if $i \neq j$. The study reported the performance of CERT, BERT model fine-tuned with contrastive representation learning, on 11 natural language understanding tasks, and CERT achieved better average performance than BERT.

The study also experimented with a different data augmentation technique. Easy Data Augmentation techniques (EDA), first introduced in [45], augments a text dataset by randomly performing one of the four operations: synonym replacement, random insertion, random swap, and random deletion. Back translation outperforms EDA in almost all of the experimental settings except in one case where its performance is 0.1% lower than that of EDA. The study attributes back-translation's better results to the fact that it performs augmentation at sentence level while EDA performs at word/phrase level, thus preserving better global semantics.

A recent survey illustrates that despite the outstanding performance of back translation compared to EDA, it is computationally expensive and non-robust [34]. There still exist NLP challenges including generating better positive and negative samples and introducing scalable, high-quality, and robust text augmentations.

Chapter 3

CTRL-Screener

In this section, we introduce CTRL-Screener, a self-supervised contrastive learning-based model that can be used to expedite the systematic review process. We present the model architecture and report the implementation details.

3.1 Problem Formulation

Given the abstract of an input document, D_i, which contains n sentences (i.e., $D_i = d_i^1, d_i^2, ..., d_i^n$), CTRL-Screener gives a binary classification that predicts whether the abstract is related to the given systematic review topic or not.

3.2 Model Architecture

CTRL-Screener consists of three neural network layers: (1) an abstract representation that is obtained using a sentence embedding, (2) a self-supervised autoencoder trained with a contrastive loss that further compresses the abstract representation and learns a more robust abstract representation, and (3) a softmax layer that uses the autoencoder representation to predict whether the article is relevant to the systematic review topic. 3.1 shows the training process of CTRL-Screener. A subset of

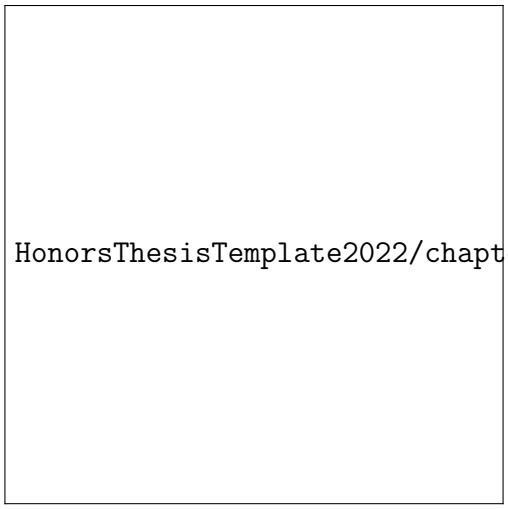


Figure 3.1: An illustration of the training process of CTRL-Screener.

documents needs to be manually labeled by the reviewers and is used by the softmax layer to learn the classification. However, the abstract representation generator and the autoencoder do not encode class membership information into the representation and thus can be learned in an unsupervised fashion.

3.2.1 Abstract Representation

Various model architectures have been proposed to learn sentence representations such as pooling word embeddings [33, 37], using the CLS token from BERT [14], or more recently Sentence-BERT, a BERT-based Siamese network architecture [33]. For the purpose of our experiments, CTRL-Screener uses Sentence-BERT and fastText [18] but we note any sentence or paragraph embedding can be used. If the abstract is longer than the supported input length, we truncate from the start of the abstract until we reach the maximum supported length.

3.2.2 Contrastive Autoencoder

Autoencoders are often used to generate compressed representations from their input data. First introduced in [35], autoencoder is a type of neural network that has been

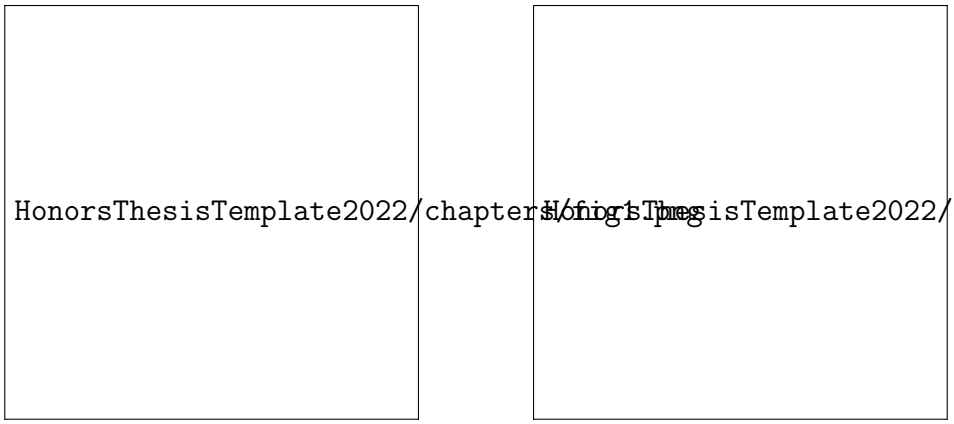
used to generate informative represenatation of data for various applications such as clustering[2][38]. Although Sentence-BERT already condenses the abstract to a 768-dimensional numeric vector, an additional autoencoder can potentially yield more robust abstract representations. Standard autoencoders are pre-trained to reconstruct the input data using the mean-squared error loss function, shown in Figure 3.2(a). However, one limitation is that minor semantic modifications of the abstract may yield drastically different representations. This is important as biomedical abstracts are generally longer due to a common pre-defined structure that consists of background, method, results, and conclusion sections. As a result, the abstracts often exceed the input length supported by convolutional neural networks, recurrent neural networks, and transformer-based embedding models.

To address this limitation, CTRL-Screener adopts a self-supervised contrastive loss to train the autoencoder. Let D_i denote the document of interest (i.e., anchor document). Given a positive sample, P_i, and a negative sample, N_i, associated with D_i, the triplet loss is then defined as:

$$\mathcal{L} = \sum_{i=1}^{n} \max(||D_i - P_i||_2^2 - ||D_i - N_i||_2^2, 0). \tag{3.1}$$

The natural question is then how to generate a positive sample that is similar to the anchor instance and a negative sample that is dissimilar. Although a contrastive learning framework has been proposed for sentence representation [46], the four sentence augmentation techniques (word deletion, span deletion, reordering, and synonym) may not alleviate lengthy abstracts.

Given the pre-defined structure in biomedical documents, we propose two mechanisms for generating high-quality positive augmentations, P_i. First, we subsample the sentences d_i^k within D_i but ensure the order remains fixed. In this fashion, there is likely at least 1 sentence that reflects each of the structured sections while maintain-



(a) Baseline (b) Contrastive

Figure 3.2: Comparison between the baseline autoencoder and the contrastive-based encoder in CTRL-Screener.

ing the overall structure of the abstract. Figure 3.3 shows an example of an abstract from the dataset in [17]. The abstract consists of 11 sentences with indices from 1 to 11, divided into 5 structured sections. Figure 3.4 illustrates an augmented sample using the first mechanism. Sentences $[1, 3, 4, 5, 7, 8, 10]$ were randomly sampled and concatenated together, preserving relative order, as an augmented sample of the abstract shown in Figure 3.3. The second mechanism subsamples sentences d_i^k within D_i and shuffles the ordering.

Negative samples are constructed by randomly selecting a subsample of sentences from any other document within the corpus, D_j, such that $j \neq i$. Figure 3.5 illustrates how positive and negative samples are defined using the augmented datasets. With respect to the document $D1$ framed in blue, $D'1$, framed in green, is the positive sample since it was generated using $D1$. Negative sample $D'2$ is framed in red as it is generated from a different abstract.

As shown in Figure 3.2(b), the contrastive autoencoder will learn to generate representations that preserve the distance between two abstract representations of the same document in the new vector space while ensuring that it is somewhat different from the other abstract representations. Since this representation does not require

labeled information and can be pre-trained on a larger corpus that may or may not include articles from the systematic review topic.

3.2.3 Classification

The resulting representation from the contrastive autoencoder is passed to a softmax layer whose weights are learned using the training data for the systematic review topic.

3.3 Implementation Details

CTRL-Screener is implemented in PyTorch v1.12.0.[1] We use Adam as the optimizer and weights to each layer were initialized using the Xavier Uniform function. Both the hidden layer and the output layer have input and output dimensions of 768.

3.3.1 Data augmentation

Positive and negative samples are generated by subsampling sentences at different percentages (50%, 60%, 70%, and 80%) from the original documents. We generated multiple positive samples, P_i, by subsampling from each anchor document and either maintaining the relative order of sentences or shuffling the sampled sentences. A negative sample, N_j, is obtained by subsampling the same percentage from another randomly selected document, $D_j, j \neq i$, and either preserving the order in the generated sample or shuffling the sampled sentences.

[1] Our code is available at `https://drive.google.com/drive/folders/1z7fpBRCcHNqZidecbS4M1kTAbQF3lRew?usp=sharing`.

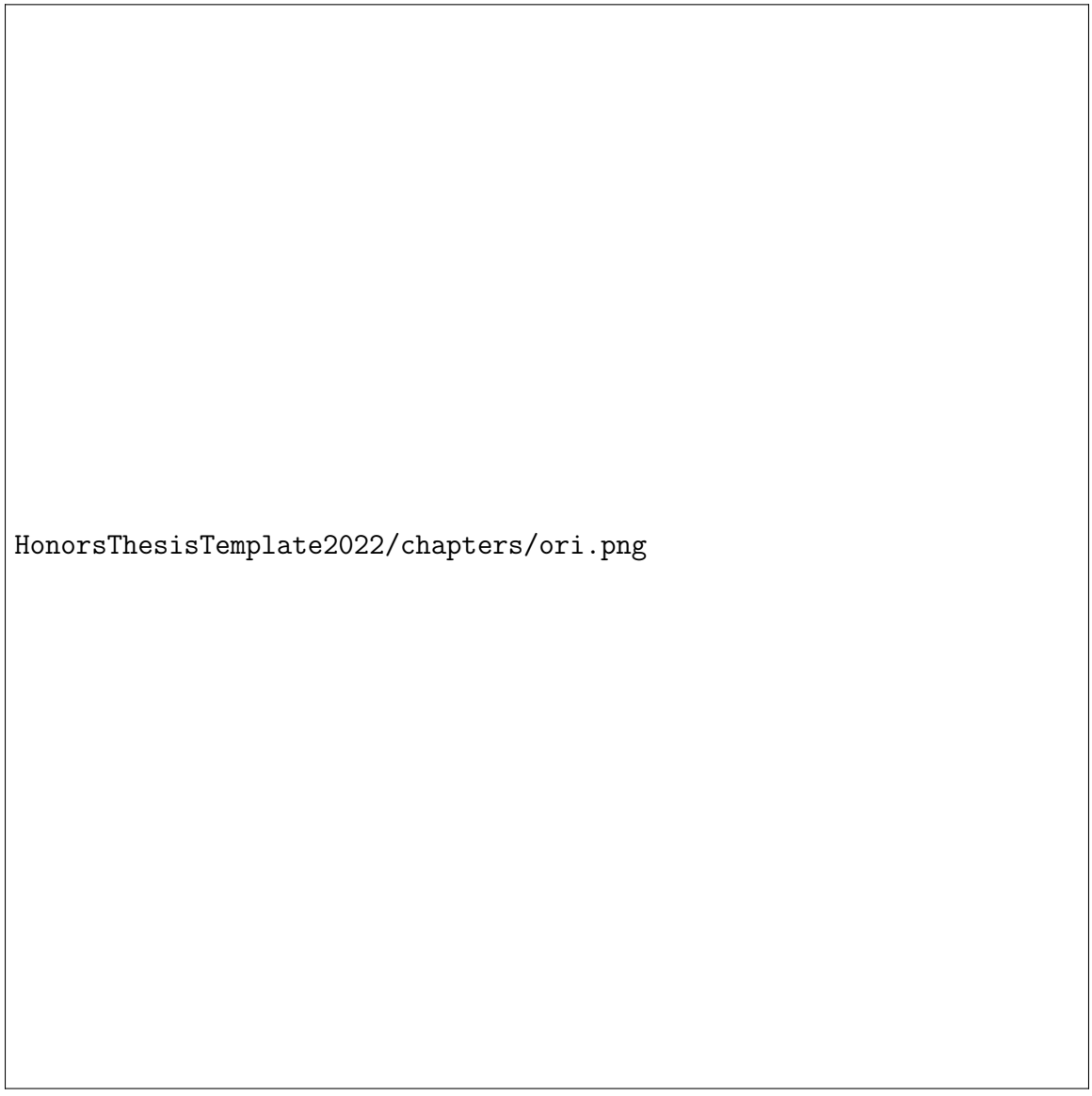


Figure 3.3: An example abstract provided in [17].

HonorsThesisTemplate2022/chapters/aug.png

Figure 3.4: An augmented sample generated from abstract shown in Figure 3.3

Figure 3.5: Illustration of the definition of positive and negative sample with respect to an anchor sample. In the figure, the anchor sample is framed with a blue square, positive sample is framed with a green square, and negative sample is framed with a red square.

Chapter 4

Experiment setup

4.1 Datasets

We evaluate our model on three datasets: (1) 15 systematic review topics related to different drug efficacy of medications in several drug classes provided by Cohen *et al.* [5], (2) 3 systematic review datasets related to clinical outcomes of various treatments released by Wallace et al. [42], and (3) 1 systematic review topic on cut-points of hyperglycemia [10]. Each SR contains the contents of the abstract and the label which indicates whether the articles are included or excluded after the abstract screening stage. The number of articles screened ranged from 310 (Antihistamines) to 7043 (Clopidrel) with an abstract screening ratio ranging from 2.07% (SkeletalMuscleRelaxants) to 32.49% (Triptans). Table 4.1 summarizes the statistics for the 19 SR topics. For each document, we constructed 4 augmented samples by subsampling $50\%, 60\%, 70\%$, and80% of the sentences in the original document while preserving relative order, and 1 augmented sample by subsampling and shuffling 70% of the sentences. In chapter 5, we discuss the impact of sample percentage and ordering on the results. The average number of sentences in the abstract for each SR is between 10 and 13. On average, positive and negative samples contain 2 to 5 fewer sentences

Table 4.1: Statistics of the systematic review topics used. Abs denotes the number of articles passing the abstract triage statuses and % shows the percentage.

SR	Abs	Total	%
ACEInhibitors	183	2544	7.23
ADHD	84	851	9.87
Antihistamines	92	310	29.67
AtypicalAntipsychotics	363	1120	32.41
BetaBlockers	302	2072	14.57
CalciumChannelBlocker	279	1218	22.90
Estrogens	80	368	21.74
NSAIDs	88	393	22.39
Opioids	48	1915	2.51
OralHypoglycemics	139	503	27.63
ProtonPumpInhibitors	238	1333	17.85
SkeletalMuscleRelaxants	34	1643	2.56
Statins	173	3465	4.99
Triptans	218	671	32.48
UrinaryIncontinence	78	327	23.85
Clopidrel	771	7043	9.29
Anemia	653	4457	11.6
Proton Beam	243	4114	5.12
Hyperglycemia	274	4026	6.81

than the anchor.

4.2 Evaluation

4.2.1 Baseline Methods

We compare CTRL-Screener against the 2 baseline embeddings and the baseline embeddings with an autoencoder pre-trained using the traditional mean squared error loss to reconstruct the abstract representations.

1. fastText[1]: A 100-dimensional representation obtained from averaging the word embeddings in the abstract. A recent replication study found that a fastText-based classifier where the shallow neural network was trained based on averaging the word embedding scores often offered the best performance compared to

[1] `https://fasttext.cc/docs/en/python-module.html`

recent deep neural networks [18].

2. Sentence-BERT[2]: A 768-dimensional representation generated using the text from the abstract as an entire sentence.

3. fastText+autoencoder: The averaged 100-dimensional fastText representation is fed into a pre-trained autoencoder using mean squared error.

4. Sentence-BERT+autoencoder: The 768-dimensional Sentence-BERT representation is fed into a pre-trained autoencoder using mean squared error.

The resulting embeddings from the above four baseline models are then fed to a softmax layer.

4.2.2 Metrics

We use work saved over sampling (WSS) and the area under the receiver operating curve (AUC). WSS measures the work saved over random sampling for a given level of recall and was introduced by Cohen *et al* [5]. WSS is defined as

$$WSS@R = (TN + FN)/N - (1.0 - R) \tag{4.1}$$

where TN denotes true negatives, FN is false negatives, N is the total number of articles, and R is the desired level of recall. Based on previous studies, we use recall at 95% (i.e., WSS@95%).

Figure 4.1 illustrates the basic idea behind WSS metric. We rank the test samples by the predicted probability of being positive in a descending order. The red square frames the samples that need to be manually screened in order to achieve a recall at R (i.e. there should be $R \cdot N$ samples with true positive label within the square), the samples outside the square can be automatically screened out. WSS measures the

[2] `https://huggingface.co/sentence-transformers/paraphrase-albert-small-v2`

Figure 4.1: An illustration of work saved over sampling metric

percentage of samples outside the red square, a larger WSS indicates fewer samples to be screened manually.

4.2.3 Evaluation Strategy

Each systematic review topic is split using the 5-fold cross-validation strategy where 1 fold is used for testing and 1 fold serves as validation for hyperparameter tuning (e.g., learning rate and the number of epochs). The average result of across 5 folds is reported.

4.2.4 Hyperparameter tuning of CTRL-Screener

We used Optuna [1] to find the optimal learning rate in the training of autoencoders and early stopping to prevent the autoencoders from overfitting on the training data. All the articles across the various systematic review topics are used to train the autoencoder. For the softmax classification layer, the validation data was used to find the optimal learning rate and the number of epochs.

Chapter 5

Results

5.1 Results

Table 5.1 and Table 5.2 summarize the AUC and WSS@95 scores across the 19 systematic review topics using fastText and Sentence-Bert as base language models, respectively. In terms of AUC, CTRL-Screener performs the best on 10 of the 19 datasets using Sentence-Bert and 5 using fastText. Similarly, for WSS@95, our model achieves the highest on 9 and 5 topics, respectively. These results highlight the added benefit of designing appropriate text augmentation techniques and utilizing the contrastive learning based approach to train the autoencoder.

From comparing across 5.1 and 5.2, we observe that Sentence-BERT embeddings generally yielded better representations than fastText embeddings on 10 of the topics in terms of AUC. However, with respect to WSS@95, fastText outperforms Sentence-BERT on 14 of the topics. This suggests state-of-the-art NLP methods may not always provide the best performance and supports the conclusion by Kusa *et al.* [18] that deep learning models do not provide substantial superiority over traditional (e.g., shallow neural network) models.

Table 5.1 also shows that passing the embedding through a traditionally trained

Table 5.1: Average performance results across the 5-folds for each of the SR topics. FT denotes the fastText-based embedding models, AE is an autoencoder pre-trained with the mean square error loss. Bold values indicate the highest score overall and underlined scores indicate the second-highest score.

	FT		FT + AE		CTRL-Screener (FT)	
Topic	AUC	WSS	AUC	WSS	AUC	WSS
ACEInhibitors	**0.782**	**0.190**	0.738	0.104	0.745	0.118
ADHD	**0.837**	**0.304**	0.764	0.190	0.783	0.165
Antihistamines	**0.620**	**0.095**	0.547	0.026	0.618	0.060
Atypical Antipsychotics	0.678	0.068	0.639	0.048	**0.685**	**0.071**
Beta Blockers	**0.709**	**0.123**	0.642	0.092	0.690	0.071
Calcium Channel Blockers	**0.744**	**0.109**	0.673	0.091	0.717	0.091
Estrogens	0.755	**0.269**	0.679	0.135	**0.810**	0.249
NSAIDs	**0.830**	**0.234**	0.763	0.119	0.810	0.233
Opiods	**0.727**	**0.277**	0.702	0.209	0.711	0.219
Oral Hypoglycemics	**0.630**	**0.067**	0.612	0.033	0.574	0.058
Proton PumpInhibitors	0.704	0.076	0.667	0.073	**0.705**	**0.102**
Skeletal Muscle Relaxants	**0.673**	0.202	0.616	0.230	0.639	**0.295**
Statins	**0.749**	0.168	0.702	0.080	0.740	0.141
Triptans	**0.823**	**0.242**	0.742	0.122	0.795	0.2178
Urinary Incontinence	0.815	0.190	0.763	0.117	**0.826**	**0.224**
Clopidrel	0.919	0.566	0.850	0.380	**0.936**	**0.597**
Anemia	**0.939**	**0.616**	0.890	0.438	0.919	0.575
Proton Beam	**0.931**	**0.645**	0.918	0.584	0.923	0.619
Hyperglycemia	**0.849**	**0.362**	0.791	0.253	0.837	0.301

autoencoder (i.e., mean squared loss) usually leads to a deterioration in performance for fastText, but can sometimes be beneficial for Sentence-Bert. However, generally speaking, the autoencoder yields slightly worse performance for abstract screening. Only in at most 5 of the topics did the contrastive learning-based autoencoder result in a performance degradation compared to the traditional autoencoder (i.e., Atypical Antipsychotics and Oral Hypoglycemics for fastText embeddings; ADHD, Opiods, Proton PumpInhibitors, Skeletal Muscle Relaxants, and Urinary Incontinence for Sentence-Bert). These systematic reviews contain a smaller number of articles ($<$ 2000 total) and may suffer from the lack of positive/negative samples to appropriately train the contrastive learning-based autoencoder.

Table 5.2: Average performance results across the 5-folds for each of the SR topics. SBERT represents the Sentence-BERT based embedding models, and AE is an autoencoder pre-trained with the mean square error loss. Bold values indicate the highest score overall and underlined scores indicate the second-highest score.

	SB		SB+AE		CTRL-Screener (SB)	
Topic	AUC	WSS	AUC	WSS	AUC	WSS
ACEInhibitors	0.790	0.188	0.783	**0.219**	**0.795**	0.187
ADHD	0.898	0.494	**0.913**	**0.585**	0.901	0.466
Antihistamines	**0.681**	**0.074**	0.620	0.064	**0.694**	0.050
Atypical Antipsychotics	0.694	**0.083**	0.680	0.060	**0.695**	0.074
Beta Blockers	0.722	0.116	0.690	0.111	**0.727**	**0.143**
Calcium Channel Blockers	0.711	0.071	0.692	0.045	**0.720**	**0.116**
Estrogens	0.811	0.246	0.803	0.279	**0.821**	**0.346**
NSAIDs	**0.791**	0.217	0.783	0.173	0.787	**0.222**
Opiods	**0.688**	**0.242**	0.674	0.212	0.667	0.166
Oral Hypoglycemics	**0.673**	0.088	0.631	0.036	0.661	**0.121**
Proton PumpInhibitors	**0.709**	0.064	0.694	**0.085**	0.687	0.079
Skeletal Muscle Relaxants	**0.816**	**0.451**	0.784	0.323	0.783	0.389
Statins	0.694	0.117	0.693	0.136	**0.753**	**0.230**
Triptans	**0.835**	**0.328**	0.808	0.237	0.816	0.297
Urinary Incontinence	0.770	0.172	**0.810**	**0.207**	0.770	0.155
Clopidrel	0.910	0.506	0.903	0.500	**0.919**	**0.551**
Anemia	0.918	0.553	0.909	0.541	**0.927**	**0.588**
Proton Beam	0.931	0.597	0.921	0.606	**0.936**	**0.639**
Hyperglycemia	**0.831**	0.324	0.814	**0.338**	0.823	0.292

5.2 Effect of Relative Order

We study the effect of the 2 augmented sample generation mechanisms on the contrastive learning-based autoencoder: (1) order preservation (or unshuffled) and (2) random sampling of the sentences (or shuffled). Table 5.3 summarizes the results for the Sentence-BERT encoding for 4 of the systematic review topics. As the results illustrate, preserving the sentence order generally yields better performance both in terms of AUC and WSS@95. The combined column illustrates the results for training CTRL-Screenerusing both shuffled and unshuffled datasets. We observe that using both sampling mechanisms and training on a larger dataset, (i.e., CTRL-Screener (SB)) yields a more robust pre-trained autoencoder. Clopid, Anemia, and Hyperglycemia show improvements both in terms of AUC and WSS, whereas proton

Table 5.3: A study of the impact of relative and shuffled ordering of the sentences on CTRL-Screener.

	Unshuffled		Shuffled		Combined	
Topic	AUC	WSS	AUC	WSS	AUC	WSS
Clopidrel	0.917	0.497	0.914	0.499	**0.924**	**0.556**
Anemia	0.914	0.524	0.916	**0.546**	**0.917**	0.544
Proton Beam	**0.928**	**0.622**	0.926	0.611	0.927	0.590
Hyperglycemia	0.827	0.303	0.811	0.265	**0.838**	**0.313**

Table 5.4: An AUC comparison of the effect of the sampling percentage on CTRL-Screener.

Topic	50%	60%	70%	80%
Clopidrel	0.911	0.908	**0.917**	0.908
Anemia	**0.923**	0.918	0.914	0.918
Proton Beam	0.922	**0.934**	0.928	0.929
Hyperglycemia	**0.827**	0.814	**0.827**	0.817

Beam experiences a slight degradation from combining the two mechanisms.

5.3 Effect of Sampling Percentage

Next, we study the impact of the sampling percentage on CTRL-Screener. Table 5.4 summarizes the results for the Sentence-BERT embedding for 4 of the systematic review topics. As the results illustrate, there does not appear to be an obvious correlation between the percentage of sentences to sample and CTRL-Screener's performance. Anemia and Hyperglycemia yields better performance using 50% of the abstract as data augmentation, whereas Clopidrel and Hyperglycemia achieved better performance from 70% sampling. In comparison with the results in Table 5.1 and 5.2, we observe that training using all percentages yields a more robust pre-trained autoencoder.

Chapter 6

Conclusion

6.1 Conclusion

In this paper, we investigate the use of a contrastive learning approach to improve the automation of the systematic review screening process. We propose CTRL-Screener, which introduces a data augmentation technique to generate positive and negative samples that can yield better abstract representations. Our results on 19 systematic review topics demonstrate the potential of contrastive learning to obtain better predictive results for citation screening.

6.2 Future work

In this section, we discuss some of the limitations of our study, and future work that can be done to mitigate those limitations.

6.2.1 Additional datasets

Only 19 systematic review datasets were used in the study, and 18 out of the 19 datasets were produced before 2010. We plan to evaluate CTRL-Screener on addi-

tional and more recent systematic review tasks such as the CLEFT-TAR [13] and the SWIFT-Review [12] datasets.

6.2.2 Comparison to other data augmentation techniques

We plan to perform empirical experiments to compare the data augmentation technique proposed in this paper with other existing data augmentation technique in NLP such as EDA [45].

6.2.3 Different levels of automation

As reported in previous studies[8], Level 2 automation (i.e. tools enabling workflow prioritization) generally achieves better results than Level 4 automation (i.e. tools performing tasks to completely eliminate human participation)[28]. We intend to explore the contrastive learning techniques with existing ranking-based approaches for systematic review screening [21, 36, 43].

6.2.4 Continual learning

We identify that one potential impediment to efficiently applying CTRL-Screener is the time and memory required for training the contrastive learning-based autoencoder. Currently, this is trained with samples from all systematic review topics in the corpus. A solution to the problem is to pre-train the autoencoder with the large corpus of all systematic review and fine-tunes the pre-trained model for each new systematic review task. To prevent catastrophic forgetting (i.e., poor performance on sample seen early in the pre-training) and negative transfer (i.e., poorer performance on most recent tasks due to local adaptation), we plan to adopt the methodology of efficient life-long learning framework as proposed in [44], which is improved on by the Memory-Based Parameter Adaptation framework introduced in [39].

Bibliography

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[2] Dor Bank, Noam Koenigstein, and Raja Giryes. Autoencoders, 2021.

[3] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

[4] Aaron M Cohen. Optimizing feature representation for automated systematic review work prioritization. In *AMIA annual symposium proceedings*, volume 2008, page 121. American Medical Informatics Association, 2008.

[5] Aaron M Cohen, William R Hersh, Kim Peterson, and Po-Yin Yen. Reducing workload in systematic review preparation using automated citation classification. *Journal of the American Medical Informatics Association*, 13(2), 2006.

[6] Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. Understanding back-translation at scale, 2018.

[7] Hongchao Fang, Sicheng Wang, Meng Zhou, Jiayuan Ding, and Pengtao Xie.

Cert: Contrastive self-supervised learning for language understanding. *arXiv preprint arXiv:2005.12766*, 2020.

[8] Allison Gates, Samantha Guitard, Jennifer Pillay, Sarah A. Elliott, Michele P. Dyson, Amanda S. Newton, and Lisa Hartling. Performance and usability of machine learning for screening in systematic reviews: a comparative evaluation of three tools. *Systematic Reviews*, 8(1):278, 2019.

[9] John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. Declutr: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 879–895, 2021.

[10] Unjali P Gujral, Ram Jagannathan, Siran He, Minxuan Huang, Lisa R Staimez, Jingkai Wei, Nanki Singh, and KM Venkat Narayan. Association between varying cut-points of intermediate hyperglycemia and risk of mortality, cardiovascular events and chronic kidney disease: a systematic review and meta-analysis. *BMJ Open Diabetes Research and Care*, 9(1):e001776, 2021.

[11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[12] Brian E Howard, Jason Phillips, Kyle Miller, Arpit Tandon, Deepak Mav, Mihir R Shah, Stephanie Holmgren, Katherine E Pelch, Vickie Walker, Andrew A Rooney, et al. Swift-review: a text-mining workbench for systematic review. *Systematic reviews*, 5:1–16, 2016.

[13] Evangelos Kanoulas, Dan Li, Leif Azzopardi, and Rene Spijker. Clef 2019 tech-

nology assisted reviews in empirical medicine overview. In *CEUR workshop proceedings*, volume 2380, 2019.

[14] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

[15] Madian Khabsa, Ahmed Elmagarmid, Ihab Ilyas, Hossam Hammady, and Mourad Ouzzani. Learning to identify relevant studies for systematic reviews using random forest and external information. *Machine Learning*, 102(3):465–482, 2016.

[16] Georgios Kontonatsios, Sally Spencer, Peter Matthew, and Ioannis Korkontzelos. Using a neural network-based feature extraction method to facilitate citation screening for systematic reviews. *Expert Systems with Applications: X*, 6:100030, 2020.

[17] Ioannis Koulouridis, Mansour Alfayez, Thomas A Trikalinos, Ethan M Balk, and Bertrand L Jaber. Dose of erythropoiesis-stimulating agents and adverse outcomes in ckd: a metaregression analysis. *American Journal of Kidney Diseases*, 61(1):44–56, 2013.

[18] Wojciech Kusa, Allan Hanbury, and Petr Knoth. Automation of citation screening for systematic literature reviews using neural networks: A replicability study. In *European Conference on Information Retrieval*, pages 584–598. Springer, 2022.

[19] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020.

[20] Eric W Lee, Byron C Wallace, Karla I Galaviz, and Joyce C Ho. Mmidas-ae: multi-modal missing data aware stacked autoencoder for biomedical abstract

screening. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pages 139–150, 2020.

[21] Grace E Lee and Aixin Sun. Seed-driven document ranking for systematic reviews in evidence-based medicine. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 455–464, 2018.

[22] Ivan Lerner, Perrine Créquit, Philippe Ravaud, and Ignacio Atal. Automatic screening using word embeddings achieved high sensitivity and workload reduction for updating living network meta-analyses. *Journal of clinical epidemiology*, 108:86–94, 2019.

[23] Yu Meng, Chenyan Xiong, Payal Bajaj, Paul Bennett, Jiawei Han, Xia Song, et al. Coco-lm: Correcting and contrasting text sequences for language model pretraining. *Advances in Neural Information Processing Systems*, 34:23102–23114, 2021.

[24] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[25] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.

[26] Nassr Nama, Margaret Sampson, Nicholas Barrowman, Ryan Sandarage, Kusum Menon, Gail Macartney, Kimmo Murto, Jean-Philippe Vaccani, Sherri Katz, Roger Zemek, et al. Crowdsourcing the citation screening process for systematic reviews: validation study. *Journal of medical Internet research*, 21(4):e12953, 2019.

[27] Daniel W Otter, Julian R Medina, and Jugal K Kalita. A survey of the usages of deep learning for natural language processing. *IEEE transactions on neural networks and learning systems*, 32(2):604–624, 2020.

[28] Annette M. O'Connor, Guy Tsafnat, James Thomas, Paul Glasziou, Stephen B. Gilbert, and Brian Hutton. A question of trust: can we build an evidence base to gain trust in systematic review automation technologies? *Systematic Reviews*, 8:143, 2019.

[29] Alison O'Mara-Eves, James Thomas, John McNaught, Makoto Miwa, and Sophia Ananiadou. Using text mining for study identification in systematic reviews: a systematic review of current approaches. *Systematic reviews*, 4(1):1–22, 2015.

[30] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[31] Xuan Qin, Jiali Liu, Yuning Wang, Yanmei Liu, Ke Deng, Yu Ma, Kang Zou, Ling Li, and Xin Sun. Natural language processing was effective in assisting rapid title and abstract screening when updating systematic reviews. *Journal of clinical epidemiology*, 133:121–129, 2021.

[32] John Rathbone, Loai Albarqouni, Mina Bakhit, Elaine Beller, Oyungerel Byambasuren, Tammy Hoffmann, Anna Mae Scott, and Paul Glasziou. Expediting citation screening using pico-based title-only screening for identifying studies in scoping searches and rapid reviews. *Systematic reviews*, 6(1):1–7, 2017.

[33] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, 2019.

[34] Nils Rethmeier and Isabelle Augenstein. A primer on contrastive pretraining in language processing: Methods, lessons learned & perspectives. *ACM Computing Surveys (CSUR)*, 2021.

[35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. *Learning Internal Representations by Error Propagation*, page 318–362. MIT Press, Cambridge, MA, USA, 1986.

[36] Harrisen Scells, Guido Zuccon, and Bevan Koopman. A comparison of automatic boolean query formulation for systematic reviews. *Information Retrieval Journal*, 24:3–28, 2021.

[37] Dinghan Shen, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Chunyuan Li, Ricardo Henao, and Lawrence Carin. Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 440–450, 2018.

[38] Chunfeng Song, Feng Liu, Yongzhen Huang, Liang Wang, and Tieniu Tan. Auto-encoder based data clustering. In José Ruiz-Shulcloper and Gabriella Sanniti di Baja, editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 117–124, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[39] Pablo Sprechmann, Siddhant M. Jayakumar, Jack W. Rae, Alexander Pritzel, Adrià Puigdomènech Badia, Benigno Uria, Oriol Vinyals, Demis Hassabis, Razvan Pascanu, and Charles Blundell. Memory-based parameter adaptation, 2018.

[40] Raymon van Dinter, Cagatay Catal, and Bedir Tekinerdogan. A multi-channel convolutional neural network approach to automate the citation screening process. *Applied Soft Computing*, 112:107765, 2021.

[41] Raymon van Dinter, Bedir Tekinerdogan, and Cagatay Catal. Automation of systematic literature reviews: A systematic literature review. *Information and Software Technology*, 136:106589, 2021.

[42] Byron C Wallace, Thomas A Trikalinos, Joseph Lau, Carla Brodley, and Christopher H Schmid. Semi-automated screening of biomedical citations for systematic reviews. *BMC bioinformatics*, 11(1):1–11, 2010.

[43] Shuai Wang, Harrisen Scells, Justin Clark, Bevan Koopman, and Guido Zuccon. From little things big things grow: A collection with seed studies for medical systematic review literature search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3176–3186, 2022.

[44] Zirui Wang, Sanket Vaibhav Mehta, Barnabás Póczos, and Jaime Carbonell. Efficient meta lifelong-learning with limited memory, 2020.

[45] Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks, 2019.

[46] Zhuofeng Wu, Sinong Wang, Jiatao Gu, Madian Khabsa, Fei Sun, and Hao Ma. Clear: Contrastive learning for sentence representation. *arXiv preprint arXiv:2012.15466*, 2020.