Scaling Systematic Literature Reviews with Machine Learning Pipelines

Seraphina Goldfarb-Tarrant[*] and **Alexander Robertson**[*]
School of Informatics
University of Edinburgh

Jasmina Lazic
Bayes Centre
University of Edinburgh

{s.tarrant,alexander.robertson,jasmina.lazic}@ed.ac.uk

Theodora Tsouloufi and **Louise Donnison** and **Karen Smyth**
Supporting Evidence Based Interventions
Royal (Dick) School of Veterinary Studies
University of Edinburgh
{theodora.tsouloufi,louise.donnison,karen.smyth}@ed.ac.uk

Abstract

Systematic reviews, which entail the extraction of data from large numbers of scientific documents, are an ideal avenue for the application of machine learning. They are vital to many fields of science and philanthropy, but are very time-consuming and require experts. Yet the three main stages of a systematic review are easily done automatically: searching for documents can be done via APIs and scrapers, selection of relevant documents can be done via binary classification, and extraction of data can be done via sequence-labelling classification. Despite the promise of automation for this field, little research exists that examines the various ways to automate each of these tasks. We construct a pipeline that automates each of these aspects, and experiment with many human-time vs. system quality trade-offs. We test the ability of classifiers to work well on small amounts of data and to generalise to data from countries not represented in the training data. We test different types of data extraction with varying difficulty in annotation, and five different neural architectures to do the extraction. We find that we can get surprising accuracy and generalisability of the whole pipeline system with only 2 weeks of human-expert annotation, which is only 15% of the time it takes to do the whole review manually and can be repeated and extended to new data with no additional effort.[1]

1 Introduction

Systematic reviews are part of the field of evidence-based analysis, and are a methodology for conducting literature surveys, where the focus is on comprehensively summarising and synthesising existing research for the purpose of answering research questions (Higgins et al., 2019). The aim of this process is to be very broad coverage to avoid unknown bias creeping into results via the alternative of cherry-picking scientific results (Chalmers et al., 1995). Conducting systematic reviews requires trained researchers with domain knowledge. The stages of the process are time-consuming, but vary in how much physical and mental labour they require (Borah et al., 2017). As a result, systematic reviews suffer from three primary challenges (Allen and Olkin, 1999; Shojania et al., 2007):

1. they are very expensive, as they require many months of expert human labour;
2. they easily become out of date, for the same reason;
3. there is no amortised cost to human time at expanding them; human effort is linear in amount of research reviewed.

So though systematic reviews have been shown to be very effective and less prone to human biases (Mulrow, 1994), these issues often prove prohibitive.

However, these challenges are well suited to Machine Learning solutions, and there has recently been an increase in interest in applying NLP to this process (Marshall and Wallace, 2019). In this paper, we investigate the feasibility of implementing the multi-stage human process of a systematic review as a Machine Learning pipeline. We construct a systematic review pipeline which aims to assist researchers and organisations focusing on livestock health in various African countries who previously performed reviews manually (via a process visualised in Figure 1). The pipeline begins

[*] Equal contribution, order determined by coin flip.

[1] Code and links to models available at https://github.com/seraphinatarrant/systematic_reviews

Proceedings of the First Workshop on Scholarly Document Processing, pages 184–195
Online, November 19, 2020. ©2020 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17

with scraping for articles, then classifies them into whether or not to include in the review, then identifies data to extract and outputs a spreadsheet. We discuss the technical options we evaluated at each steps. Pipeline components are evaluated with intrinsic metrics as well as more pragmatic, extrinsic, considerations such as time and effort saved.

While previous work exists surveying the applicability of various Machine Learning methods and toolkits to the systematic review process (Section 6) and a few apply them, there are no extant studies that implement a full system and analyse the trade-offs between different methods of training data creation, different annotation schemas, human expert hours needed to build a system, and final accuracy. We experiment with all of these factors, as well as with a few different architectures, with the aim of informing the planning and implementation of systematic review automation more broadly.

To further this goal, we particularly experiment with low resource scenarios and with generalisability. We investigate different thresholds for training data for the document classifier and different annotation schemas for the data extraction. We additionally test the ability of the system to generalise to documents from new countries.

Key research questions are as follows:

Extraction Which techniques are best for identifying and extracting the desired information?

Data Requirements How much labelled training data is needed? Can existing resources be leveraged?

Re-usability How generalisable is a pipeline to new diseases and countries?

Performance What is the trade-off between pipeline accuracy and human time savings?

Architecture & Pre-training How important is model architecture as applied to extraction tasks? How important is embedding pre-training, and how important is pre-training on scientific literature vs. general content (domain match)?

We find that surprisingly little training data (and few human hours) are necessary to get an accurate document classifier, and that it generalises well to unseen African countries (Section 5), which enables systematic reviews to be expanded to new areas with essentially constant time. In our text extraction experiments, we find that both sentence and phrase level extraction models can each play a role in such a pipeline, but that phrase extraction, which has not previously been done for this task, performed better than expected both with baseline CNN models (Yang et al., 2016) and with BERT-based Transformers (Devlin et al., 2019), with Transformers based on scientific pre-training (Beltagy et al., 2019) performing best. We demonstrate how the creation of labelled training data can be sped up through annotation tools, and that consideration should be given to the balance of training examples present within this data, since doing so may require less data overall while still maintaining good performance. Furthermore, besides automatic information extraction, much labour in constructing systematic reviews can be saved through simply automating the process of searching and downloading documents.

We empirically demonstrate that most of the three month pipeline of a systematic review can be automated to require very little human intervention, with acceptable accuracy of results. We release our code, annotation schema, and labelled data to assist in the expansion of systematic reviews via automation.

While we demonstrate this system on one domain, the framework is domain independent and could be applied to other kinds of systematic reviews. New training data and annotation schemes would be necessary to switch to medical or other domains, but our findings on time saving processes for annotation would apply, and confidence thresholds that we implement are adjustable to customise to different levels of accuracy to human time trade-offs that are appropriate to different fields. Our exploration into necessary amounts of training data for accuracy and generalisability are broadly applicable.

2 Background and Motivation

As a case study, we work with the Supporting Evidence Based Interventions team at the Royal (Dick) School of Veterinary Studies at the University of Edinburgh, focusing on putting data and evidence at the centre of livestock decision-making in low and middle-income countries, predominantly in Africa. In these countries, livestock offer a path out of poverty for millions of smallholders, as well as providing vital nutrition for families and communities. While the veterinary technology and techniques required to improve livestock outcomes already exist (and are readily available to large scale commercial concerns worldwide), there is a lack of reliable information on animal health and productivity in these countries, at this scale. This data is needed not only in order to best target interventions, but to select the most efficient intervention in any particular context.

There is very little data in this area for these countries, and it is often out of date. One proxy for direct measurement of livestock health in all herds in a country is the evidence found in veterinary science research publications, which have conducted prevalence studies. Individually, these studies give an indication of the prevalence of a specific disease in a specific region of a country at a specific time, affecting a specific breed of animal. But collectively, they give a much broader understanding of livestock health.

Four strands of data are of key importance: general livestock statistics (herd size and characteristics), health (mortality and disease), production (yields and growth rates) and economics (breeding costs, feeding costs, produce sale values). Here, we focus on health, specifically the prevalence of a wide range of diseases (e.g. brucellosis, foot and mouth disease) that effect ruminants (sheep, goats and cattle), with a focus on countries such as Nigeria, Ethiopia and Tanzania.

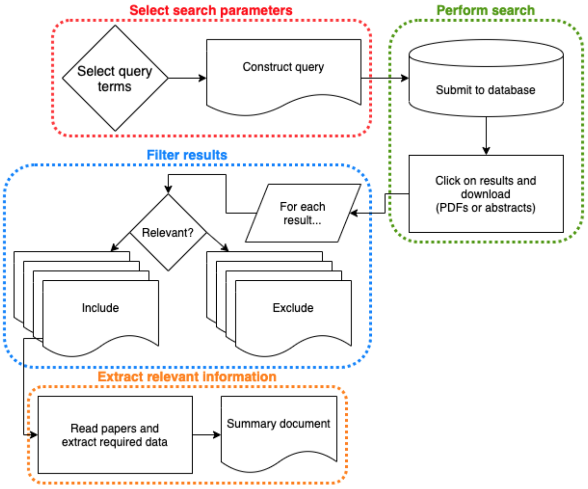


Figure 1: Human-based information extraction systematic review pipeline.

However, collecting and summarising the findings of these studies is time-consuming manual work. For example, searching databases such as Google Scholar, Pubmed and Web of Science for prevalence studies, conducted between 2010 and 2018 on 28 diseases in small ruminants in Ethiopia, returned many thousands of results. Of these, 403 papers were considered relevant and information was extracted by experienced veterinarian researchers. The completely manual process, outlined in Figure 1, produced high quality data but took approximately three months.

The target information consists of dates (the start/end of the study), numerical data (sample size, direct/percentage prevalence numbers, animal age) and small lexical terms. These terms can be veterinary (diagnostic tests used, production systems, study design, statistical analysis performed, species) or geographic (region, ecosystem).[2]

Since this summarisation process isn't abstractive, readers find it easy, if time-consuming, to identify the relevant information in research papers. The current project aimed to automate as much of the process as possible, to allow it to scale to a wider range of diseases (approximately 50) and countries. All but the initial component of Figure 1 can be entirely automated – humans are still required to define search terms.

3 Pipeline Details & Experimental Setup

In the following we detail each stage of the constructed pipeline, how it compares to the human version and the human time saved. In the succeeding Section 4 we detail metrics and evaluation for our targeted experiments with the classification and extraction pipeline components.

3.1 Document Search

The highest return on investment in terms of engineering effort vs. human time-savings was our automation of literature searches. Previously, researchers would construct a list of searches to perform, input these into various databases, then manually download the results they considered relevant. An example search is `(Livestock OR ruminants OR sheep OR goats OR cattle OR cow OR ram OR ewe OR bull) AND (Ethiopia) AND (Anthrax OR "Bacillus anthracis") AND (prevalence OR incidence)`

APIs are available for Scopus, Pubmed and Web of Science, for which we obtained institutional access. Google Scholar has no available API, so we used the SerpApi service[3] which provides paid programmatic access. The pipeline therefore maintains full coverage of paper sources. As the

[2] Appendix A contains example target extracted data
[3] https://serpapi.com/

APIs return only links to papers, PDFs still need to be retrieved. Issues to navigate here included links to websites rather than files, which requires additional negotiation through user-agent strings, parsing HTML for links to the PDF or parsing page headers to extract metadata redirecting to a PDF such as `citation_pdf_url`. A generic approach was successful in most cases but site-specific downloaders had to be constructed for 38 domains, based on trial and error.

Time spent on retrieval of potentially relevant documents to include in the systematic reviews were reduced dramatically, with the main limiting factor being rate throttles on APIs. Conducting searches on all four databases for 50 diseases in 3 countries takes approximately one hour on one machine and requires no human input. Parallel downloading of PDFs is even faster. This can be repeated at any interval to keep results up to date. By contrast, this step of the process used to take human experts 83 hours (2 weeks full-time) each time it was done, while covering fewer diseases and only one country.

3.2 Document Classification

Once search results have been collected, retrieved PDFs must be classified for inclusion vs. exclusion in the systematic review. Human reviewers use various criteria to assess inclusion (peer-review, type of study/experiment, subject matter) which, as we observed in user studies, they determine entirely from the title and abstract. We use the PDF DOI or ISSN to retrieve the title and abstract [4] using the Wikimedia Citoid API [5]. We then train an SVM Classifier implemented in Sci-kit Learn (Pedregosa et al., 2011) with concatenated TF-IDF Vectors, which can train in under an hour on a standard Linux machine. The classification process, which previously took a human expert 20 hours for 1000 documents, now generates results in minutes. We additionally implemented human review of documents with low classifier confidence (further details in Section 5) upon consultation with our systematic review experts, as this both increases classifier accuracy and human trust in results.

3.3 PDF to Text Conversion

Relevant PDFs are converted to plain text with the pdfminer.six package[6]. The text data extracted from PDFs can be noisy – tables are especially problematic, headers/footers may end up inside main text, word and line spacing may be inconsistent, fonts may be improperly converted to text. The bulk of this can be overcome through basic pre-processing.

Once converted, the text is split into paper sections (e.g. abstract, introduction, methods) using regular expressions derived from manual inspection of 100 papers. This involves matching spans of text which appear between common section titles. For example, the abstract generally appears between ‘abstract’ and ‘introduction’, ‘abstract’ and ‘keywords’, ‘summary’ and ‘introduction’.[7]

3.4 Data Extraction

The goal of a systematic review is to output a tabular file where each column stores target information for each paper; this will then later be used to generate visualisations. Manually extracting this information is easy for knowledgeable humans: it isn’t abstractive and does not require close reading of the full text. However, it is time-consuming at scale and does require experts, so both performing the process manually and creating training data incurs a significant cost. In addition, the different kinds of target information pose different technical challenges. Consider the sentence *Rose Bengal Plate Test found 1.72% (5/291) of the samples to be sero-positive*, which contains information about the diagnostic test used, the prevalence rate and sample size, all of which we want to capture. The phrase associated with the diagnostic test can be understood out of context but numbers generally cannot. Simply extracting all percentages from a text will be uninformative, rendering rule-based extraction approaches unsuitable. We therefore explored two machine learning approaches to automatic extraction to balance the difficulty in creation of annotated training data with suitability of the extraction approach.

A *sentence-based* classifier can be used to label sentences as containing target information, and has been the method of extraction used in previous work (Marshall et al., 2017; Kiritchenko et al.,

[4] PDFs are converted to text in the following step and we could use the title and abstract from extraction, but PDF extraction is noisy and so we chose to rely on reference database lookups where available.

[5] https://en.wikipedia.org/api/rest_v1

[6] https://github.com/pdfminer/pdfminer.six

[7] PDF processing is documented fully in released code.

2010; Schmidt et al., 2020). But this does not fit in well with the desired tabular output for all target fields: the same information can appear in multiple sentences and the same sentence can contain multiple targets. However, this approach is much easier for a human annotator. It should also work well for numerical targets: the context is preserved in the output and non-relevant numerical targets will be ignored or scored low. Alternatively, a *phrase-based* classifier can apply labels to individual words and phrases within sentences. The extracted information will be more focused and should work well for phrase-based targets. The results will not require rule-based and human post-processing, as with the results of sentence-based extraction, but training data creation is more onerous. So given a fixed amount of human expert hours available, this approach may be less desirable, since it will generate much less training data.

We test the difference between both approaches using CNN-based text classification and named entity recognition models implemented in Prodigy[8]. This tool combines data annotation and model training. We created an annotation schema with 16 labels taken from manually created gold standard systematic review output.[9]

Creating training data for a sentence-based classifier is mechanically simple: the Prodigy annotation tool allows non-technical users to quickly assign labels using an interface with keyboard shortcuts, and we can display one sentence at a time. Prodigy also allows phrase-level labelling, but this is a more involved process as the user must mark the start/end boundaries of a span and then apply the appropriate label. A single veterinarian labelled 4600 items at the sentence level in 56 hours, reporting the process to be easy and straightforward. The same veterinarian labelled 4200 items at the phrase level in 70 hours, reporting it to require much more physical and mental effort.

4 Methodology

We performed detailed evaluation of the different classification and extraction components.

Document Classification We investigate the trade-off between training data volume and performance, and how generalisable a model is. For training volume, we fix a test set and reduce training data in chunks of 20% of total. We test generalisability by training models on country-specific data and evaluating on unseen data from other countries. We report Accuracy overall, as well as Precision, Recall, and F1 on the *include* label in this binary classification task. Finally, we investigate the effect of thresholding classifier confidence, and sending low confidence documents for manual human review, on both the accuracy of the system and on human time cost.

Data Extraction We evaluate the sentence classifier and sequence-labelling approach with our CNN models. We also consider the impact of using document representations constructed with embeddings trained entirely on the source data, versus general purpose GloVe embeddings (Pennington et al., 2014) trained on web data, versus general purpose GloVe embeddings fine-tuned on the source data[10]. As the sentence-level classifier is multi-label and multi-class, we report AUC (Area Under Curve).[11] For the phrase-level sequence labelling approach, we report F1 score.

4.1 Training Data Creation

Document Classification veterinarian experts manually labelled papers as include/exclude: 608 papers from searches for 50 diseases for the countries of Ethiopia, Nigeria and Tanzania. We experimented with labelling 100 test documents: half via a reference manager/document reader[12] and via a simple spreadsheet interface where one column contained the paper title, one contained the paper abstract, and the expert filled in a third column for the include/exclude label.[13] The spreadsheet method was 3 times faster than using a reference manager, enabling experts to complete the 608 papers of training data in 5 hours. Half the data contains country information, so we use only that half for our generalisability experiments.

Data Extraction 52 documents were randomly sampled from the set of documents manually classified for inclusion. The sampled documents covered 13 diseases for studies in Ethiopia, Nigeria and Tanzania. Only abstracts, results and methods sections were annotated.

[8] `https://prodi.gy/`

[9] Annotation schema included in code repository.

[10] Implemented in spacy `https://spacy.io/`

[11] AUC ̄true positive vs false positive rate over a range of discrimination thresholds

[12] Zotero (`zotero.org`) was chosen as it is the only service which provides an API

[13] Recall that criteria for inclusion in the study are fully determinable via these fields.

Data	Description	Phrases	Sentences
disease	Animal disease	3307 (31.5%)	778 (31.4%)
species	Species studied	2002 (19.1%)	518 (20.9%)
region	Area within country	1487 (14.2%)	298 (12.0%)
individual_prevalence	Number of infected animals	743 (7.1%)	172 (6.9%)
diagnostic_test	Test used to detect disease	729 (6.9%)	172 (6.9%)
reference	Reference to another study	591 (5.6%)	137 (5.5%)
sample_type	Biological samples used	486 (4.6%)	117 (4.7%)
statistical_analysis	Analysis performed	261 (2.5%)	63 (2.5%)
age	Ages of animals tested	228 (2.2%)	65 (2.6%)
sample_size	Number of animals tested	161 (1.5%)	24 (1.0%)
production_system	Type of farm	141 (1.3%)	43 (1.7%)
ecosystem	Geography of farm	141 (1.3%)	44 (1.8%)
study_design	Type of study used	120 (1.1%)	28 (1.1%)
study_date	Date study was conducted	64 (0.6%)	8 (0.3%)
herd_prevalence	Number of herds infected	28 (0.3%)	7 (0.3%)
mortality	Animals killed by disease	5 (0.0%)	1 (0.0%)

Table 1: Proportion of target items identified during data annotation.

To select a manageable volume of data for annotation, and avoid including noisy data from the PDF extraction process, we applied some restrictions. For the sentence-based task, all sentences of at least 9 words within the abstract were included, along with a random sample of 150 sentences (between 9 and 25 words long) from the results and methods sections. Sentence length was based on the fact that very short/long sentences were generally noisy due to the PDF conversion process. For the phrase-based task, sections were split into chunks of three sentences to preserve some context. The entire abstract was used, plus a random sample of 25 chunks from each of the methods and results sections.

Table 1 briefly describes each item and the breakdown of label frequency in our annotated data. There is a clear imbalance in label frequency – some are not commonly reported in general (e.g. mortality, herd prevalence) while others are reported very few times per paper (e.g. study date).

4.2 Experimental Conditions

Data Volume We trained document classification models using proportions from 20% to 100% of all data.

Generalisability Three document classification models were each trained on two of the three countries, with the final country held out. We included data volume ablations in these experiments as well.

Sentence vs. Phrase Models We trained the CNN-model on sentence labelled vs. phrase labelled data to assess the feasibility of using each annotation approach.

Architecture & Pretraining We experiment with five different architectures for the phrase-based models. We use the Prodigy CNN with randomly-initialised embeddings, the Prodigy CNN with frozen pre-trained embeddings, the Prodigy CNN with pre-trained embeddings fine-tuned on our data, distilBERT (Sanh et al., 2019), and SciBERT (Beltagy et al., 2019). The CNN is easy to implement out of the box, as it is built into the annotation tool, can be trained without access to a GPU, and could potentially be less data-hungry than a transformer - all important considerations in our resource constrained setting. Adding pre-trained embeddings allows us to isolate the effect of pre-training from the effect of architecture. Since the phrase-labelling task is well suited to the masked language modelling objective, we additionally experiment with fine-tuning distilBERT (which is reasonably sized for our small amount of data) and SciBERT, to test whether the domain match of pre-trained data matters.

5 Results

Results for document classification experiments are shown in Figure 2. The upper left quadrant of Figure 2 contains data for 608 documents with an 85-15 train-test split across all 3 countries, showing an expected increase in classifier performance as data increases, but levelling off slightly by 80% of the full training volume. The other quadrants

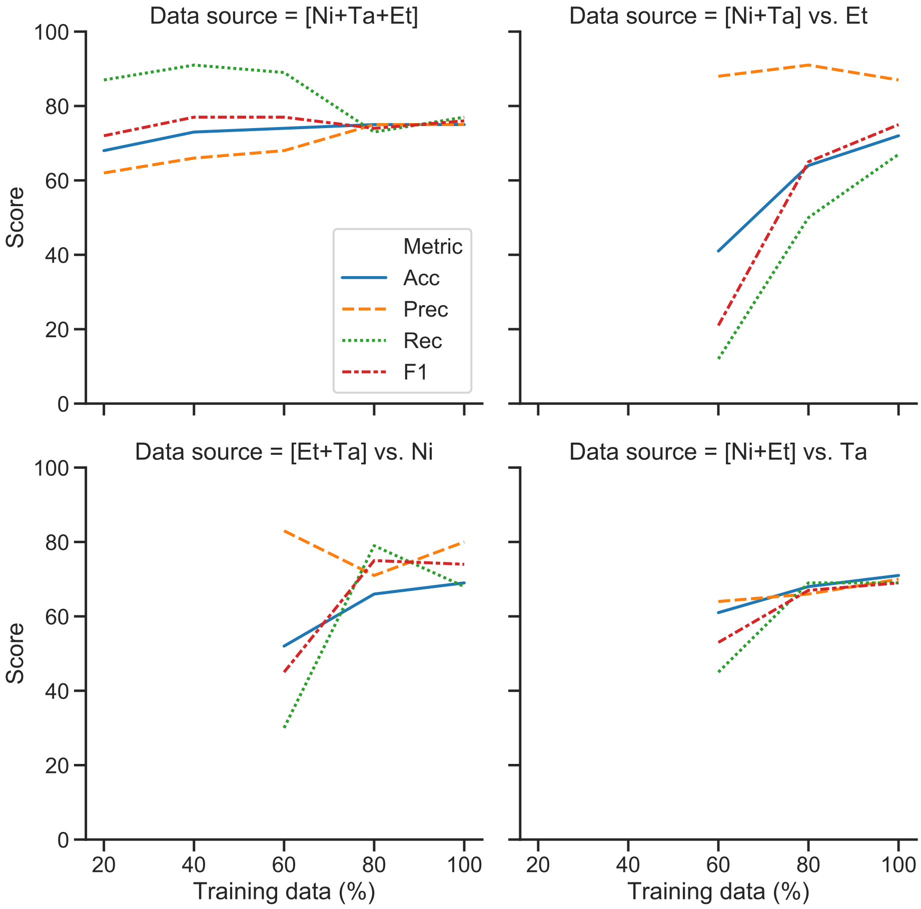


Figure 2: Accuracy, Precision, Recall, and F1 for the document classification model showing performance changes as training data is decreased for the full set of 608 documents (85-15 train-test split), as well as generalisability to held out countries. Et=Ethiopia, Ta=Tanzania, and Ni=Nigeria. Note that country experiments have only 200 train and 100 test documents (100 per country, with test held fixed).

show the same data for 100 documents per country (200 train, 100 test) but with a minimum of 60% of total data, as with less than 100 training samples the model does not converge.

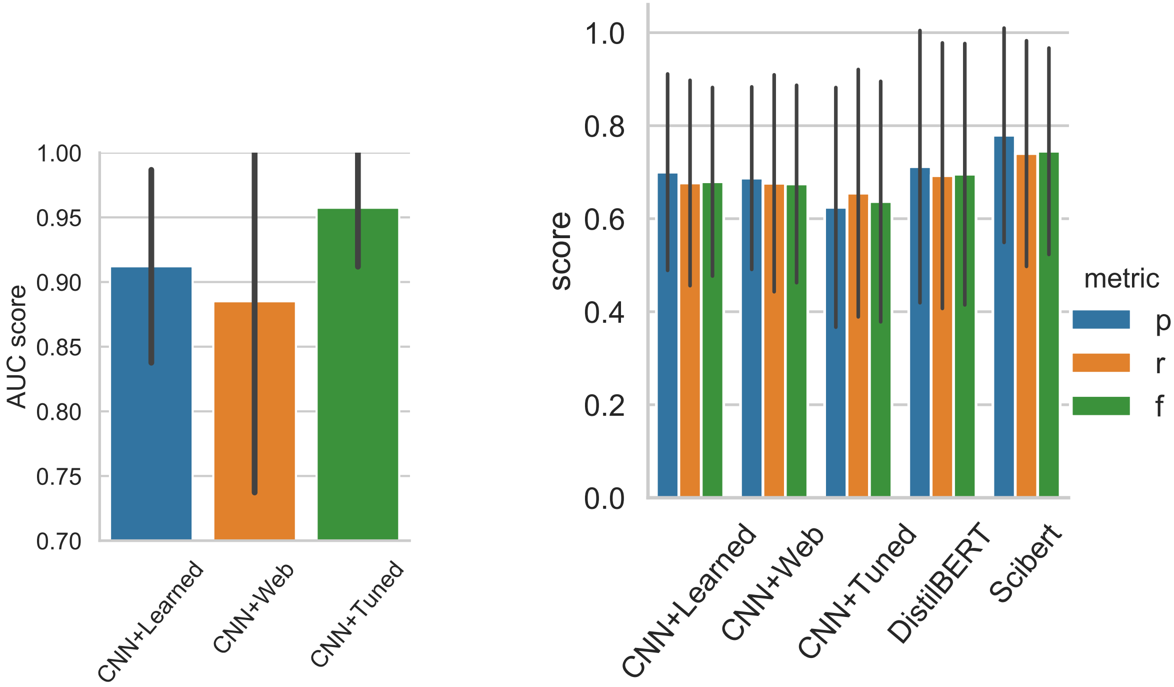
For the data volume studies on the full dataset, a notable trend is that recall is quite high even with very little training data (100 documents), and that what the classifier learns with additional data is predominantly a better precision-recall balance. For the held-out-country generalisability studies, the amount of training data is more important, and recall is no longer high immediately.

This suggests that for a fixed country with a semi-automated system that has resources for a secondary human-filtering, very little training data is necessary. However, extensibility to new countries does require more data. Given that additional more data, performance on unseen countries is equal to that of known countries of equivalent training set volumes. This suggests an important new extensibility opportunity for systematic review systems.

In practice, our experts needed slightly higher accuracy than the best combined accuracy. To address this we implemented confidence-thresholding, such that documents below a user-set threshold are uploaded to a *needs review* folder, which generates a weekly email. 15% of test documents require review at our final confidence threshold, which reduces human time to 20 min per 100 documents classified, but allows for an increased accuracy to 88%. Human reviews are then fed back into classifier training, which should incrementally improve confidence and reduce human labour over time. We leave that longitudinal study for future work.

Results for three CNN-based sentence-level data extraction models are shown in Figure 3a. We report mean AUC score on all labels (with standard deviation shown) with an 80/20 train/test split.

For sentence-level models, fine-tuned web embeddings give better performance overall. Mean



(a) Sentence-level classification: Mean AUC score, over all labels, for three embedding sources. Error bars denote standard deviation.

(b) Phrase-level classification: Mean Precision/Recall/F1, over all labels, for three embedding sources and two Transformer-based models. Error bars denote standard deviation.

AUC score was 0.96 (stdev 0.05) using fine-tuned general purpose web embeddings; 0.91 (stdev 0.08) using learned embeddings; 0.89 (stdev 0.15) using general purpose web embeddings.

For phrase-level models, this is no longer true: mean F1 score was 0.67 (stdev 0.22) using pre-trained general purpose web embeddings; 0.68 (stdev 0.2) using learned embeddings; 0.64 (stdev 0.27) using fine-tuned general purpose web embeddings. Transformer-based models performed more strongly: F1 for DistillBERT was 0.70 (stdev 0.29), SciBERT 0.75 (stdev 0.23).

Focusing on the items considered most important by the veterinarian researchers (disease, species, region, individual prevalence, diagnostic test, sample type, sample size, study date), results in an increase of 0.03 AUC for each sentence-level model. Phrase-based models F1 score increases by 0.10.

These results suggest that pre-training is important for the sentence-based classifier, and that the BERT-based Transformer architecture with the masked language modelling objective can do very well on phrase-level extraction and bring performance high enough to make this approach feasible. However, they show that domain-specific pre-training data has a larger effect than architectural differences. While Transformer-based models for phrase-level labelling out-performed CNN-based models, it was the SciBERT model trained on academic papers, then fine-tuned on our specific task, which gave the best performance, and a larger performance boost than the initial jump from BERT. The best type of pre-training does vary based on type of extraction: general purpose embeddings perform worst for sentence-level labelling, though are on par with those learned from the training data for phrase-level labelling.

We analyse per-label performance for the SciBERT model to verify phrase-level feasibility, and include this data in Figure 4 in Appendix B. Performance is generally high, even for some low-frequency items. Some of these were uncommon in our training data (due to appearing only once or twice per paper) but naturally appear in *many* academic papers in general, which goes towards accounting for the success of SciBERT on this task. For example, SciBERT was the only model to correctly identify *any* instances of herd prevalence.

6 Related Work

The application of NLP to systematic reviews is relatively new, but has been recently receiving more attention. There is a growing body of work that assesses the potential for automation in systematic reviews, but little that builds systems for the purpose and tests them empirically.

Marshall and Wallace (2019) review available tools that can be used to automate each element of the systematic review pipeline. Marshall et al. (2020) further review opportunities for semi-automation and assess opportunities and risks. Marshall and Brereton (2015); O'Mara-Eves et al. (2015) conduct systematic reviews of automation for systematic reviews. Thomas et al. (2017) analyse the systematic review pipeline to find ways that human-machine collaboration can be applied and improve the speed.

Marshall et al. (2017) create a PDF viewer that humans can use to make the systematic review process easier and faster, by training a CNN to assess risk of bias in a document (an important part of evidence-based analysis in the medical domain, though not for our particular task) and identifies and displays sentences to the user that contain a subset of the information necessary for a systematic review. Kiritchenko et al. (2010) create an extraction system that identifies sentences and then post-processes them to extract data, but operate only on structured HTML & XML. Schmidt et al. (2020) apply fine-tuned BERT-based Transformers to the task of to sentence classification for semi-automated systematic review. Goswami et al. (2019) build a PDF retrieval system for systematic reviews for psychology and use a random forest classifier to identify sentences for extraction.

As far as we are aware, no other work builds a phrase-based system, tests data volume and generalisability, or applies a diverse set of modern architectures to the task.

7 Conclusion & Future Work

We investigated the application of automation to all stages of the systematic review pipeline for our veterinary research case study. We found that with two weeks (80 hours) of human expert annotation we can automate a systematic review that previously took 3 months, and still maintain high levels of accuracy. Our classification system generalises well, enabling it to be applied to new countries for additional systematic reviews with no additional human annotation cost. Sentence-based and phase-based data extraction both perform well, and the creation of phrase-based training data can still fit within a small amount of human annotation hours and avoids the need for extensive post-processing. Fine-tuned BERT-based Transformers perform best at data extraction, with BERT pre-trained on scientific data giving the largest boost in performance, though a baseline CNN still performs surprisingly well. In future work, we plan to test generalisability cross-lingually, expand the generalisability tests to extraction as well as classification, and study the performance improvements of continuous training of classifiers on human corrections of low-confidence output.

References

I Elaine Allen and Ingram Olkin. 1999. Estimating time to conduct a meta-analysis from number of citations retrieved. *Jama*, 282(7):634–635.

Iz Beltagy, Arman Cohan, and Kyle Lo. 2019. Scibert: Pretrained contextualized embeddings for scientific text. *ArXiv*, abs/1903.10676.

Rohit Borah, Andrew W Brown, Patrice L Capers, and Kathryn A Kaiser. 2017. Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the prospero registry. *BMJ open*, 7(2):e012545.

Iain Chalmers, Douglas G Altman, et al. 1995. *Systematic reviews*. BMJ Publishing London.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Shubhaditya Goswami, Sukanya Pal, Simon Goldsworthy, and Tanmay Basu. 2019. An effective machine learning framework for data elements extraction from the literature of anxiety outcome measures to build systematic review. In *BIS*.

Julian PT Higgins, James Thomas, Jacqueline Chandler, Miranda Cumpston, Tianjing Li, Matthew J Page, and Vivian A Welch. 2019. *Cochrane handbook for systematic reviews of interventions*. John Wiley & Sons.

Svetlana Kiritchenko, Berry de Bruijn, Simona Carini, Joel D. Martin, and Ida Sim. 2010. Exact: automatic extraction of clinical trial characteristics from journal publications. *BMC Medical Informatics and Decision Making*, 10:56 – 56.

Christopher Marshall and Pearl Brereton. 2015. Systematic review toolbox: a catalogue of tools to support systematic reviews. In *Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering*, pages 1–6.

Iain Marshall, Joël Kuiper, Edward Banner, and Byron C. Wallace. 2017. Automating biomedical evidence synthesis: RobotReviewer. In *Proceedings of ACL 2017, System Demonstrations*, pages 7–12, Vancouver, Canada. Association for Computational Linguistics.

Iain J Marshall and Byron C Wallace. 2019. Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. *Systematic reviews*, 8(1):163.

Iain James Marshall, Blair T. Johnson, Zigeng Wang, Sanguthevar Rajasekaran, and Byron C. Wallace. 2020. Semi-automated evidence synthesis in health psychology: current methods and future prospects. *Health Psychology Review*, 14:145 – 158.

Cynthia D Mulrow. 1994. Systematic reviews: rationale for systematic reviews. *Bmj*, 309(6954):597–599.

Alison O'Mara-Eves, James Thomas, John McNaught, Makoto Miwa, and Sophia Ananiadou. 2015. Using text mining for study identification in systematic reviews: a systematic review of current approaches. *Systematic reviews*, 4(1):5.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Lena Schmidt, Julie Weeds, and Julian P. T. Higgins. 2020. Data mining in clinical trial text: Transformers for classification and question answering tasks. *ArXiv*, abs/2001.11268.

Kaveh G Shojania, Margaret Sampson, Mohammed T Ansari, Jun Ji, Steve Doucette, and David Moher. 2007. How quickly do systematic reviews go out of date? a survival analysis. *Annals of internal medicine*, 147(4):224–233.

James Thomas, Anna Noel-Storr, Iain Marshall, Byron Wallace, Steven McDonald, Chris Mavergames, Paul Glasziou, Ian Shemilt, Anneliese Synnot, Tari Turner, et al. 2017. Living systematic reviews: 2. combining human and machine effort. *Journal of clinical epidemiology*, 91:31–37.

Zichao Yang, Diyi Yang, Chris Dyer, X. He, A. Smola, and E. Hovy. 2016. Hierarchical attention networks for document classification. In *HLT-NAACL*.

A Target Extracted Data

In Table 2 is the first 15 lines of a sample gold standard target data from a human systematic review (broken into two tables for display) that we use as a template for building our system. Note that some fields are blank because information was not found in or relevant to a given entry.

B Detailed Analysis of Phrase-level Classification Performance

Displayed in Figure 4 are the per label performance breakdowns for SciBERT, the strongest phrase-level extraction model. Performance remains high across many individual labels, with changes in performance mostly tracking with commonness of the information (and thus, how much training data is available for a fixed set of annotated documents). The exceptions to this trend are the *region* and *sample_size* labels, which have lower performance compared to equivalently common labels

ROW_NUMBER	IDENTIFIER	YEAR_PUBLICATION	REFERENCE	START_DATE_DATA	END_DATE_DATA	STATE	ECOSYSTEM	PRODUCTION_SYSTEM	SPECIES	AGE	AGE_DETAIL
1	Nigussie et al; 2010	2010	Nigussie et al	2007	2008	Oromia		Mixed farming	Cattle		
2	Regassa et al; 2010	2010	Regassa et al	2007	2008	SNNPR			Cattle		
2	Regassa et al; 2010	2010	Regassa et al	2007	2008	SNNPR			Cattle		
3	Regassa et al; 2010	2010	Regassa et al	2007	2008	SNNPR			Cattle		
4	Bekele et al; 2010	2010	Bekele et al	2008	2003	SNNPR			Cattle		
5	Shiferaw et al 2013; 2010	2010	Shiferaw et al 2013	2007	2008	Afar			Cattle		
6	Shiferaw et al 2011; 2010	2010	Shiferaw et al 2011	2007	2008	Afar			Cattle		
7	Shiferaw et al 2010; 2010	2010	Shiferaw et al 2010	2007	2008	Afar			Cattle		
8	Shiferaw et al 2014; 2010	2010	Shiferaw et al 2014	2007	2008	Afar			Cattle		
9	Shiferaw et al 2012; 2010	2010	Shiferaw et al 2012	2007	2008	Afar			Cattle		
10	Kumsa et al; 2010	2010	Kumsa et al	2006	2006	SNNPR			Sheep		
11	Kumsa et al; 2010	2010	Kumsa et al	2006	2006	SNNPR			Sheep		
12	Kumsa et al; 2010	2010	Kumsa et al	2006	2006	SNNPR			Sheep		
14	Amenu et al; 2010	2010	Amenu et al	2007	2007	Oromia		Mixed farming	Cattle		
14	Amenu et al; 2010	2010	Amenu et al	2007	2007	Oromia		Mixed farming	Cattle		

DISEASE	SAMPLE	DIAGNOSTIC_TEST	MEASUREMENT	NUMBER_POSITIVE	NUMBER_TESTED	PERCENTAGE	CALCULATION	COMMENTS	SOURCE
BVD	Serum	i-ELISA	Individual Prevalance	65	567	11.4638447971781	TOTAL	national surveillance/mixed altitudes (midland, highland)/zone&sex splitting/adult>young(<3y)	LITERATURE
Tb	Intraderm test	CIDT	Herd Prevalance	19	39	48.7179487179487	TOTAL	>6 MONTHS	LITERATURE
Tb	Intraderm test	CIDT	Individual Prevalance	48	413	11.6222760290557	TOTAL	>6 MONTHS	LITERATURE
Tb	PM specimen	PM	Individual Prevalance	11	1023	1.0752688172043	TOTAL		LITERATURE
TRYPs	Blood	BC	Individual Prevalance	71	323	22	TOTAL	East African zebus, >1y/T. congolense, vivax&brucei splitting	LITERATURE
FMD		Survey	Individual Mortality			0.73	TOTAL		LITERATURE
Pasteurelloses		Survey	Individual Mortality			1.5	TOTAL		LITERATURE
CBPP		Survey	Individual Mortality			2.5	TOTAL		LITERATURE
Blackleg		Survey	Individual Mortality			0.13			LITERATURE
Anthrax		Survey	Individual Mortality			1.3			LITERATURE
Endoparasites	Faeces	Floatation, Microscopy	Individual Prevalance			6.7	TOTAL	Strongyloides papillosus	LITERATURE
Endoparasites	Faeces	Floatation, Microscopy	Individual Prevalance			15	TOTAL	Trichuris spp	LITERATURE
Endoparasites	Faeces	Floatation, Microscopy	Individual Prevalance			100	TOTAL	Gi parasites/Strongyle eggs	LITERATURE
Tb	Intraderm test	SCIDT	Herd Prevalance			35	TOTAL		LITERATURE
Tb	Intraderm test	SCIDT	Individual Prevalance	27	425	6.35	TOTAL		LITERATURE

Table 2: Example target extracted data from a gold-standard human systematic review

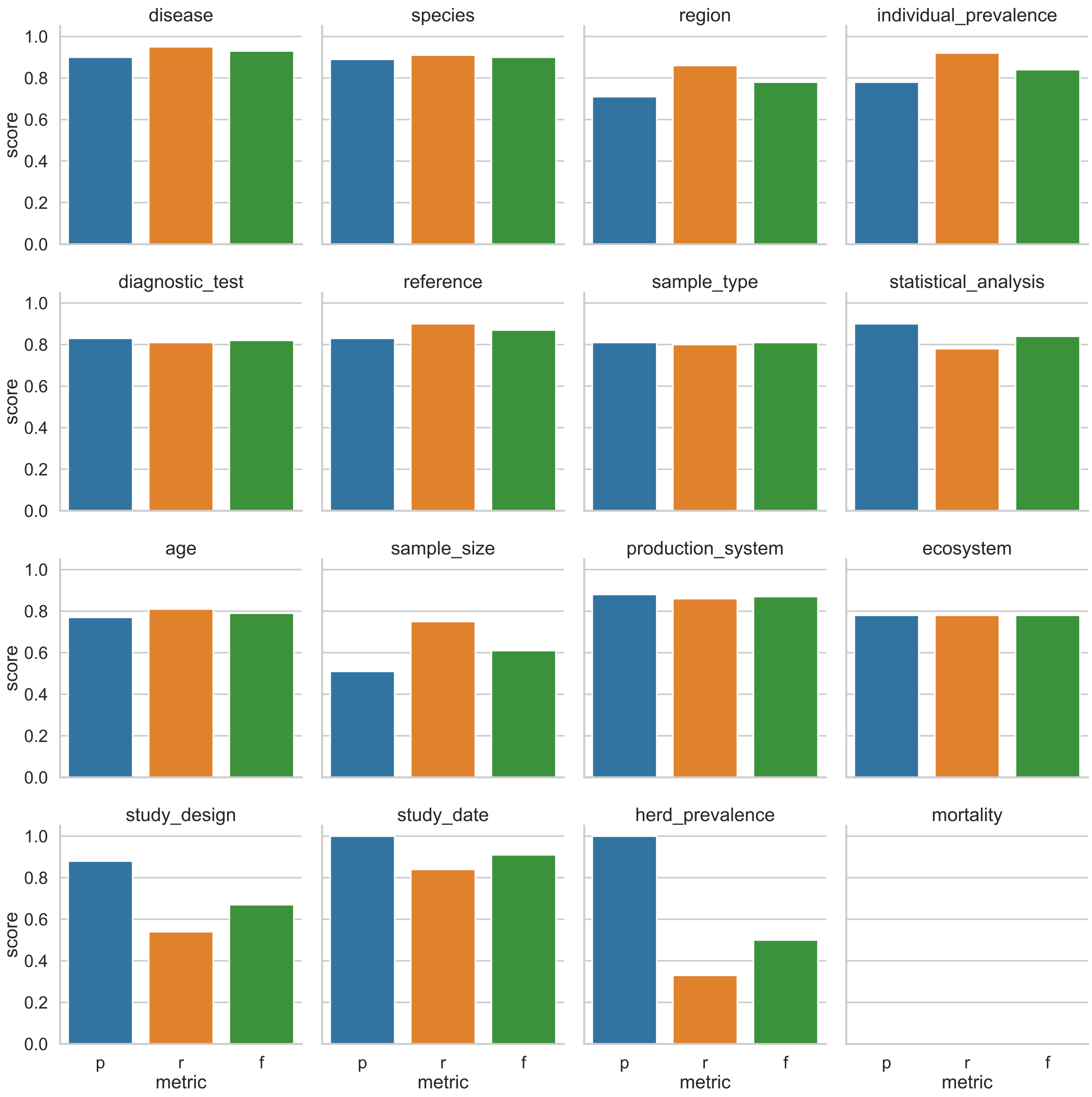


Figure 4: Phrase-level classification: Precision/Recall/F1, per label, for SciBERT. From top left to bottom right: most to fewest examples in training data.