Automatically Cataloging Scholarly Articles using Library of Congress Subject Headings

Nazmul Kazi[1], **Nathaniel Lane**[1], **Indika Kahanda**[2]
[1] Montana State University, MT, USA
[2] University of North Florida, FL, USA
kazinazmul.hasan@montana.edu
nathaniel.lane@student.montana.edu
indika.kahanda@unf.edu

Abstract

Institutes are required to catalog their articles with proper subject headings so that the users can easily retrieve relevant articles from the institutional repositories. However, due to the rate of proliferation of the number of articles in these repositories, it is becoming a challenge to manually catalog the newly added articles at the same pace. To address this challenge, we explore the feasibility of automatically annotating articles with Library of Congress Subject Headings (LCSH). We first use web scraping to extract keywords for a collection of articles from the Repository Analytics and Metrics Portal (RAMP). Then, we map these keywords to LCSH names for developing a gold-standard dataset. As a case study, using the subset of Biology-related LCSH concepts, we develop predictive models by formulating this task as a multi-label classification problem. Our experimental results demonstrate the viability of this approach for predicting LCSH for scholarly articles.

1 Introduction

An Institutional Repository (IR) is the collection of scholarly work hosted and maintained by institutions such as universities. For example, "ScholarWorks[1] is an open access repository for the capture of the intellectual work of Montana State University (MSU) in support of its teaching and research goals". Repository Analytics and Metrics Portal (RAMP) is a web service that accurately counts item downloads for each article in the institutional repository (Obrien et al., 2016; OBrien et al., 2017). Besides counting the number of downloads, RAMP stores metadata of the articles such as title, abstract, and keywords. Currently, nearly 40 institutions have registered their repositories with RAMP.

[1]https://scholarworks.montana.edu/

To facilitate the easy finding of articles, the IR managers need to catalog them using different subject headings manually. One of the most popular vocabularies for cataloging is the Library of Congress Subject Headings (LCSH) (Walsh, 2011). LCSH is a subject indexing language that is actively maintained since 1898 to catalog materials in the Library of Congress and most widely adopted by large and small libraries around the world (Work, 2016). A subject heading is the most specific word or a group of words that capture the essence of a subject category. Due to the rapid growth of items in IRs, manual cataloging using LCSH or other vocabularies is becoming highly resource-consuming (Engelson, 2013).

Due to the above challenge, there have been a few previous attempts on the automatic assignment of LCSH through keyword extraction (Wartena et al., 2010; Aga et al., 2016), by collecting LCSH concepts that are assigned to similar texts (Paynter, 2005), using semantic similarity (Yi, 2010), and co-occurrence-based mapping (Vizine-Goetz et al., 2004). These techniques primarily depend on the presence of the keywords or similar words/ phrases within the actual text and do not utilize machine learning. Furthermore, one of the studies claims that the prediction of LCSH using machine learning may be infeasible due to the large size of the vocabulary leading to inadequate training data (Wartena et al., 2010). Note that machine learning has been used for a seemingly similar but actually different task of predicting Library of Congress Classification (LCC) (Frank and Paynter, 2004). However, despite the similarity in their names, LCC and LCSH are completely different vocabularies.

Semantic indexing with other vocabularies has gained traction recently (Mirowski et al., 2010; Salakhutdinov and Hinton, 2009; Wu et al., 2014). Most notably, predicting Medical Subject Headings (MeSH) for biomedical literature using machine

Proceedings of the 16th Conference of the European Chapter of the Associationfor Computational Linguistics: Student Research Workshop, pages 43–49
April 19 - 23, 2021.

learning and deep learning techniques has seen significant recent interest (Mao and Lu, 2017; Jin et al., 2018; Kehoe et al., 2017; Rios and Kavuluru, 2015; Kosmopoulos et al., 2015; Yan et al., 2016) thanks to the BioASQ challenge on Biomedical Semantic Indexing (Tsatsaronis et al., 2015).

In this work, we explore the feasibility of developing an automated pipeline for predicting LCSH for scholarly articles using machine learning. As a case study, we leverage an extensive collection of scholarly articles from RAMP and generate a gold-standard dataset by assigning Biology-related LCSH concepts to each article through web scraping and string matching techniques. Using this gold-standard data, we develop predictive models that can predict LCSH by modeling this as a multi-label classification problem. Our experimental results indicate the effectiveness of the proposed approach.

2 Methodology

2.1 Data

In this approach, we build a gold-standard dataset by scraping RAMP data from 27 institutional repositories (IRs). A high-level overview of our approach is shown in Figure 1.

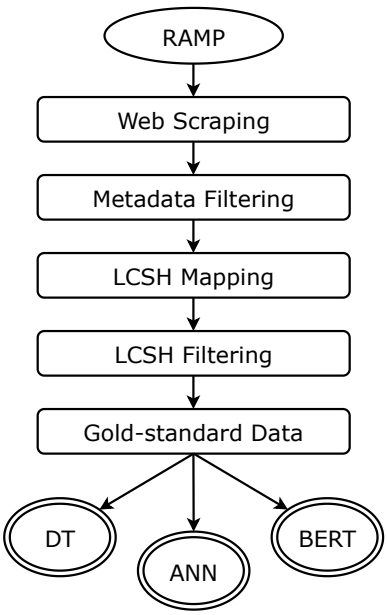

Figure 1: A high-level overview of our approach.

We identify the citable content downloads (CCD) from each institutional repository (IR) between July 2017 and July 2018 . Then, we scrape all metadata of each CCD from RAMP for the subset that includes all unique CCDs.

The raw data (scraped from RAMP) contains 457,879 articles and 270 different metadata types. However, we use only *title* concatenated with *abstract*, *article type*, and *keywords* for this study, and discard other metadata. There are many reasons why some of the metadata are empty. For example, items such as newspapers do not include abstracts, and sometimes IR managers add items into repositories without populating metadata. Therefore, we first discard articles without a title, an abstract, or keywords, which reduces the dataset to 126,655 articles that have a title, an abstract, and at least one keyword. Then, we map each keyword to the subject names from the 41^{st} edition of LCSH[2] using full string matching (case insensitive). If a keyword does not match with any subject, we ignore that keyword.

Any article without at least one assigned subject heading is discarded. This results in a smaller set of articles with annotated subject headings. Then, we filter out any subjects not related to Biology by only retaining the concept *Biology* (sh85014203)[3] and its descendants. Finally, we remove subject headings that are annotated to less than 100 articles. After all the above, we have a dataset composed of 17,367 articles with 66 Biology-related subject headings. This LCSH-annotated dataset is used as the gold-standard dataset for developing predictive models. Note that while the string matching technique used in this study itself can potentially be used for "predicting" LCSH terms, we are assuming that unseen items that need to be annotated with LCSH in real-life may not necessarily come with keywords (and hence we resort to developing predictive machine learning models). The distribution of articles across IRs in this dataset is shown in Table 1.

2.2 Models

We model the task of predicting LCSH concepts as a multi-label classification problem and develop three supervised machine learning models using the above generated gold-standard data. These models are 1) Decision Tree (DT), 2) Artificial Neural Networks (ANN), and 3) Bidirectional Encoder Representations from Transformers (BERT). All the models are implemented using scikit-learn[4], Ten-

[2]https://loc.gov/aba/publications/FreeLCSH/freelcsh.html
[3]http://id.loc.gov/authorities/subjects/sh85014203.html
[4]https://scikit-learn.org/

	IR Name	# Articles
1	Deep Blue	7,820
2	DRUM	1,578
3	EASP	1,171
4	UWSpace	1,063
5	OpenBU	960
6	MacSphere	917
7	Texas ScholarWorks	849
8	Mountain Scholar	631
9	Epsilon Open Archive	576
10	K-REx	464
11	MSU ScholarWorks	405
12	OAKTrust	380
13	MD-SOAR	245
14	SHAREOK	192
15	Others	116
	Total:	17,367

Table 1: Number of articles per institute in the gold-standard dataset.

sorFlow[5], Transformers[6] and PyTorch[7] libraries. In our preliminary work, We also train models using Support Vector Machines and Random Forest classifiers, but none of them perform better than the models reported in this paper (data not shown).

We choose standard but varying pre-processing steps independently for each model since certain pre-processing techniques work well for some models over the others. For example, removing stopwords is a common practice for Decision Tree models but not for BERT since stopwords typically can act as noise for the former.

2.2.1 Decision Tree (DT) model

We apply the Decision Tree classifier to develop a tree-based one-vs-rest classification model. We use TF-IDF (term frequency-inverse document frequency) vectorizer with a word-based analyzer for feature extraction. We use lemmatization and stop word removal as standard pre-processing steps. We include both uni-grams and bi-grams as features and train our model over the top 10,000 features. Our model returns a binary value, i.e., either 0 or 1, as the prediction.

2.2.2 Artificial Neural Network (ANN) model

For the shallow artificial neural network model, we use the TF-IDF scores as input. These are generated using scikit-learn's TfidfVectorizer class. All stop words (common words such as "the" or "and") are removed before vectorization, and only the terms that appear in a minimum of 1% of all documents are kept.

Our artificial neural network has four layers: an input layer with 2,251 nodes, a dropout layer with a rate of 0.1, a hidden layer with 132 nodes, and an output layer with 66 nodes (one for each label) with a sigmoid activation function. We initially experimented with many different network structures but ultimately find that a single hidden layer with 132 nodes, double the number in the output, produces the best results (data not shown). We use 5-fold nested cross-validation to find the optimal epoch for training the networks. We train the largest network with 100 epochs and find 10 epochs as optimal as the learning curve reaches convergence. We use this optimal epoch to train all networks.

2.2.3 Bidirectional Encoder Representations from Transformers (BERT) model

We use the pre-trained BERT-Base (uncased) model (Devlin et al., 2018) and fine-tune it for multi-label text classification. The base model has 12 transformer blocks, i.e., hidden layers, a hidden size of 768, 12 attention heads, and 110 million parameters (Devlin et al., 2018). The model is pre-trained for English on uncased Wikipedia and BooksCorpus. For fine-tuning the model, we use *Adam* optimizer with a learning rate of $2e-5$, $\epsilon = 1e-8$, L2 weight decay of 0.01, learning rate warmup over the first 500 steps with linear decay and Cross-Entropy Loss function. We observe the learning curve over 5-fold nested cross-validation and find 6 epochs as the optimal number. Any example longer than the 512 token length restriction enforced by the BERT-Base model is truncated.

2.3 Experimental Setup and Metrics

In order to obtain unbiased estimations of model performance, we evaluate our models using 5-times 5-fold stratified cross-validation (Sechidis et al., 2011; Szymański and Kajdanowicz, 2017). We primarily report the performances of our models using Maximum F1-score (F_{max}), Precision at F_{max} and Recall at F_{max}. Precision reports the percentage of true samples among the samples that have been predicted as true, whereas Recall reports the percentage of true samples retrieved by the model. F1-score is the harmonic mean of precision and re-

[5]https://www.tensorflow.org/
[6]https://huggingface.co/transformers/
[7]https://pytorch.org/

Subject Frequency	# subjects	DT			ANN			BERT		
		P	R	F1	P	R	F_{max}	P	R	F_{max}
$[100, 200)$	35	0.36	0.35	0.36	0.48	0.40	0.43	0.51	0.43	0.43
$[200, 300)$	15	0.40	0.39	0.39	0.48	0.46	0.47	0.56	0.51	0.49
$[300, 400)$	6	0.31	0.30	0.30	0.42	0.44	0.43	0.55	0.55	0.54
$[400, 900)$	7	0.41	0.41	0.41	0.48	0.57	0.52	0.59	0.70	0.64
$[1700, 2600]$	3	0.40	0.40	0.40	0.46	0.67	0.54	0.57	0.71	0.63
Macro average:		0.38	0.37	0.37	0.46	0.51	0.48	0.56	0.58	0.55

Table 2: Model performance per subject frequency range. # subjects: Number of unique subjects within the range, P: precision, R: recall.

Article Type	Freq.	Length		Average Number of		F_{max}		
		Avg	Std	Keywords	Subjects	DT	ANN	BERT
Thesis	6,765	379.89	191.32	30.24	1.15	0.31	0.38	0.41
Article	1,077	225.80	99.52	21.25	1.29	0.19	0.23	0.24
Report	880	442.89	280.80	15.80	1.09	0.14	0.18	0.19
Paper	364	207.68	111.74	22.29	1.17	0.10	0.12	0.16
Book	48	221.31	194.41	20.27	1.08	0.02	0.03	0.04
Others	383	164.54	116.59	24.53	1.30	0.12	0.14	0.19
NA	7,850	253.99	147.47	21.52	1.27	0.30	0.38	0.41

Table 3: Model performance per article type. NA: Not Available, Freq: number of articles in type, Length: number of words in title and abstract, P: precision, and R: recall.

Subject	Freq	F_{max}		
		DT	ANN	BERT
Commencement ceremonies	141	0.99	1.00	1.00
Discrimination	227	0.83	0.88	0.88
Irrigation	125	0.72	0.66	0.89
Machine Learning	260	0.68	0.71	0.75
Nanoparticles	174	0.67	0.67	0.78
Self-efficacy	112	0.64	0.69	0.71
Animal ecology	520	0.56	0.67	0.79
Autism	103	0.68	0.51	0.75
Feminism	113	0.63	0.52	0.76
Planning	245	0.50	0.65	0.69

Table 4: Top ten easiest to predict subjects. Freq: Frequency of subject in the dataset.

Subject	Freq	F_{max}		
		DT	ANN	BERT
Social psychology	157	0.05	0.14	0.02
Clinical psychology	196	0.10	0.22	0.00
Metabolism	104	0.14	0.19	0.00
Molecular biology	185	0.07	0.15	0.11
Developmental psychology	174	0.14	0.20	0.00
Cognition	109	0.18	0.20	0.00
Epidemiology	224	0.17	0.20	0.04
Zoology	242	0.13	0.24	0.05
Physiology	190	0.12	0.20	0.11
Neurology	176	0.23	0.25	0.00

Table 5: Top ten hardest to predict subjects. Freq: Frequency of subject in the dataset.

call. Unlike F1, F_{max}, which is computed across a range of thresholds, is threshold independent. More specifically, let threshold $t \in [0, 1]$, then

$$F_{max} = \max_t \left\{ \frac{2 \cdot \text{precision}(t) \cdot \text{recall}(t)}{\text{precision}(t) + \text{recall}(t)} \right\}$$

For this study, we use a step size of 0.05 for thresholds and Macro-averaging (arithmetic mean) for aggregating the performance across classes.Note that since the DT model returns binary predictions directly, without class probabilities, we report the performance of this model only using F1 instead of F_{max}.

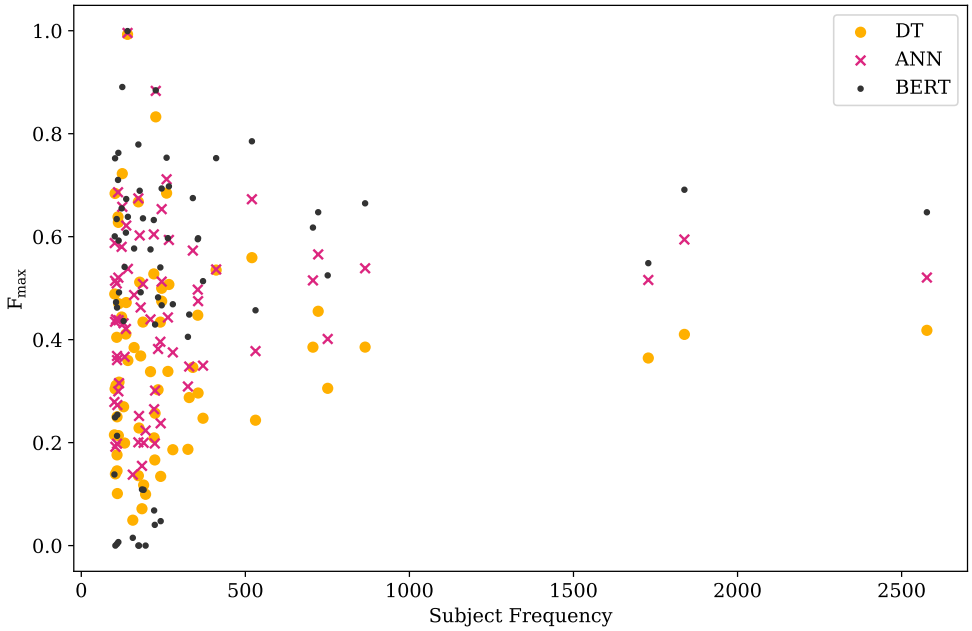

Figure 2: Model performance against subject frequency. DT: Decision Tree, ANN: Artificial Neural Network.

3 Results and Discussion

The overall performance for all our models is depicted in Table 2. Overall, the BERT model performs the best, and the DT model performs the worst among the three models. The DT model achieves an average F1 score of 0.37, whereas the lowest F1 score (0.30) is observed for frequency range $[300, 400)$. The performance of the DT model is seemingly immune to the frequency of subjects. The ANN model notably outperforms the DT model with an average F_{max} of 0.48. The ANN model also struggles for frequency range $[300, 400)$. However, the lowest F_{max} (0.43) of ANN is higher than the best F1 score (0.40) achieved by DT in any frequency range. Except for frequency range $[300, 400)$, we can see an increase in F_{max} of ANN as the frequency range increases. The BERT model significantly outperforms both DT and ANN models with an average F_{max} of 0.55 and shows a positive correlation between F_{max} and frequency range.

Figure 2 shows variation of performance of all three against the frequency. The subjects between range $[100, 200)$ are widely spread across the y-axis (F_{max}) for each model, which indicates that the easiest and the hardest subject to predict have similar subject frequencies. Top ten easiest and hardest subjects across all three models are listed in Table 4 and Table 5, respectively. We use macro-averaged F-score from all three models to compile these rankings. All three models show their best performance for the same subject, *Commencement ceremonies*. Both DT and ANN have a non-zero F-score for each subject. Despite being the best model, BERT shows zero F_{max} for several subjects, e.g., *Clinical psychology*.

We also assess the performance of each model per document type, as reported in Table 3. For the following analysis, we exclude the document type denoted as NA for which the corresponding metadata was missing. Same as before, BERT performs the best, and ANN outperforms DT. All three models show their best and worst performance for the same article types across all models, Thesis and Book, respectively. The frequency of each type may have played a significant role in these extremes. This is further supported by the fact that the performance across all three models follows the same trend: as the frequency decreases, the performance decreases as well.

4 Conclusions and Future Work

In this work, we explore the feasibility of using machine learning for predicting LCSH for scholarly articles. We first generate a gold-standard dataset annotated with LCSH subjects by web scraping/ string matching and utilize this data for developing multi-label classification models. Our results indicate the feasibility of our approach. We believe our approach is applicable to other data similar to LCSH concepts. This automated pipeline should

be extremely valuable to librarians for expediting the manual cataloging process. We plan to measure the efficiency gains of this method through the Montana State University Library.

While our approach displays promising results, there are many different avenues for future investigation. First, in this work, we map the web scraped keywords to subject names (instead of identifiers or IDs). However, some subject names may map to more than one identifier (e.g., Psychology: sh85108459 or sh2002011487). So, we plan to explore two different solutions to this. One approach is to develop a chain-classifier that can predict the LCSH IDs using the already predicted subjects (i.e., a second classifier for disambiguation). Another option is the improve the web scraping/ string matching pipeline so that we can generate a gold-standard dataset directly annotated with IDs.

To improve the performance of our traditional machine learning models, we plan to investigate the inclusion of hand-engineered features, other resources such as MeSH terms, metadata fields that were ignored in this study, and the hierarchical information from the LCSH. Besides, using larger more sophisticated language models (e.g., Megatron-LM), using the complete set of LCSH terms (without restricting to Biology-related), and structured output models that explicitly use the hierarchy information will likely improve performance. Moreover, Extreme Multi-Label (XML) models that are equipped to handle very large sets of classes (Kumar et al., 2019) will also likely provide better performance.

5 Acknowledgement

We would like to thank Patrick OBrien and Kenning Arlitsch from Montana State University Library and Jonathan Wheeler from University of New Mexico for providing us with the data and guidance for this project. This work was supported in part by NSF awards CNS-1730158, ACI-1540112, ACI-1541349, OAC-1826967, the University of California Office of the President, and the University of California San Diego's California Institute for Telecommunications and Information Technology/Qualcomm Institute. Thanks to CENIC for the 100Gpbs networks.

References

Rosa Tsegaye Aga, Christian Wartena, and Michael Franke-Maier. 2016. Automatic recognition and disambiguation of library of congress subject headings. In *International Conference on Theory and Practice of Digital Libraries*, pages 442–446. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Leslie Engelson. 2013. Correlations between title keywords and lcsh terms and their implication for fast-track cataloging. *Cataloging & classification quarterly*, 51(6):697–727.

Eibe Frank and Gordon W Paynter. 2004. Predicting library of congress classifications from library of congress subject headings. *Journal of the American Society for Information Science and Technology*, 55(3):214–227.

Qiao Jin, Bhuwan Dhingra, William Cohen, and Xinghua Lu. 2018. Attentionmesh: Simple, effective and interpretable automatic mesh indexer. In *Proceedings of the 6th BioASQ Workshop A challenge on large-scale biomedical semantic indexing and question answering*, pages 47–56.

Adam K Kehoe, Vetle I Torvik, Matthew B Ross, and Neil R Smalheiser. 2017. Predicting mesh beyond medline. In *Proceedings of the 1st Workshop on Scholarly Web Mining*, pages 49–56.

Aris Kosmopoulos, Ion Androutsopoulos, and Georgios Paliouras. 2015. Biomedical semantic indexing using dense word vectors in bioasq. *J BioMed Semant Suppl BioMedl Inf Retr*, 3410:959136040–1510456246.

P. Kumar, V. K. Dubey, and M. I. H. Showrov. 2019. A comparative analysis on various extreme multi-label classification algorithms. In *2019 4th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT)*, pages 265–268.

Yuqing Mao and Zhiyong Lu. 2017. Mesh now: automatic mesh indexing at pubmed scale via learning to rank. *Journal of biomedical semantics*, 8(1):15.

Piotr Mirowski, M Ranzato, and Yann LeCun. 2010. Dynamic auto-encoders for semantic indexing. In *Proceedings of the NIPS 2010 Workshop on Deep Learning*, volume 2.

Patrick OBrien, Kenning Arlitsch, Jeff Mixter, Jonathan Wheeler, and Leila Belle Sterman. 2017. Ramp–the repository analytics and metrics portal. *Library Hi Tech*.

Patrick Obrien, Kenning Arlitsch, Leila Sterman, Jeff Mixter, Jonathan Wheeler, and Susan Borda. 2016. Undercounting file downloads from institutional repositories. *Journal of Library Administration*, 56(7):854–874.

Gordon W Paynter. 2005. Developing practical automatic metadata assignment and evaluation tools for internet resources. In *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'05)*, pages 291–300. IEEE.

Anthony Rios and Ramakanth Kavuluru. 2015. Analyzing the moving parts of a large-scale multi-label text classification pipeline: Experiences in indexing biomedical articles. In *2015 International Conference on Healthcare Informatics*, pages 1–7. IEEE.

Ruslan Salakhutdinov and Geoffrey Hinton. 2009. Semantic hashing. *International Journal of Approximate Reasoning*, 50(7):969–978.

Konstantinos Sechidis, Grigorios Tsoumakas, and Ioannis Vlahavas. 2011. On the stratification of multi-label data. *Machine Learning and Knowledge Discovery in Databases*, pages 145–158.

Piotr Szymański and Tomasz Kajdanowicz. 2017. A network perspective on stratification of multi-label data. In *Proceedings of the First International Workshop on Learning with Imbalanced Domains: Theory and Applications*, volume 74 of *Proceedings of Machine Learning Research*, pages 22–35, ECML-PKDD, Skopje, Macedonia. PMLR.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138.

Diane Vizine-Goetz, Carol Hickey, Andrew Houghton, and Roger Thompson. 2004. Vocabulary mapping for terminology services. *Journal of digital information*, 4(4):2004.

John Walsh. 2011. The use of library of congress subject headings in digital collections. *Library review*.

Christian Wartena, Rogier Brussee, and Wout Slakhorst. 2010. Keyword extraction using word co-occurrence. In *2010 Workshops on Database and Expert Systems Applications*, pages 54–58. IEEE.

Jill A Work. 2016. Legislating librarianship. *The Political Librarian*, 2(2):7.

Hao Wu, Martin Renqiang Min, and Bing Bai. 2014. Deep semantic embedding. In *SMIR@ SIGIR*.

Yan Yan, Xu-Cheng Yin, Bo-Wen Zhang, Chun Yang, and Hong-Wei Hao. 2016. Semantic indexing with deep learning: a case study. *Big Data Analytics*, 1(1):1–13.

Kwan Yi. 2010. A semantic similarity approach to predicting library of congress subject headings for social tags. *Journal of the American Society for Information Science and Technology*, 61(8):1658–1672.