Automated Class Numbers Prediction for Books: an AI/ML Based Approach Using Annif

Soumik Kerketta[1] and Parthasarathi Mukhopadhyay[2]

[1]Junior Research Fellow (JRF), Department of Library and Information Science, University of Kalyani, Kalyani- 741235, Nadia, West Bengal, India, email: soumik.kerketta.bdy@gmail.com

[2]Professor, Department of Library and Information Science, University of Kalyani. Kalyani- 741235, Nadia, West Bengal, India, email: psm@klyuniv.ac.in

Abstract

In this research study as reported here, we endeavor to explore the possibilities of an AI/ML-based automated indexing system for the vast collections in a library. Library classification systems are considered pre-coordinated indexing approaches a while ago. Various machine learning techniques are applying to synthesizing classification numbers. A recently popular technique involves using a supervised learning algorithm to train a model on a set of documents that have been manually indexed/classified by their corresponding annotations using standardized terminology by trained library professionals' experts using controlled vocabularies. The trained model learns patterns from the reference data and then predict the subject and class number for new documents. In the preliminary phase, we gathered a substantial collected around 2 lacks MARC-21 formatted bibliographic records where Tag 082 (DDC Call Number), Tag 245 (Title of Document), Tag 520 (Summary Note), and Tag 650 (Subject Descriptors) are contained in the datasets. After that processed this data using the data wrangling software named OpenRefine. Then dataset was subsequently divided into three sections: (i) a training dataset, (ii) a validation dataset and (ii) a test dataset. Here We usedAnnif, an open-source AI environment to analyze the dataset using the Dewey Decimal Classification (DDC) Scheme. Training Annif involved utilizing a substantial set of bibliographic records, based on the MARC-21 tags mentioned previously. In the next stage, the framework was trained using a various of backend algorithms, such asOmikuji, fastText, SVC (associative group), and simple and neural network (ensemble)based on neural network model. In order to assess the effectiveness of these models, all of these machine learning backends were finally compared using two crucial retrieval metrics: F1@5 and NDCG. When it comes to automated class number building, we have discovered that the neural network model outperforms rather than all other backends. This overall framework based on open-source software, an open dataset, and open standards.

Keywords: Annif, Automated indexing, DDC,F1@5, Library classification, Neural network, NDCG.

1. Introduction

The exponential rise in bibliographic records in libraries of all shapes and sizes, together with the expanding workloads associated with handling this massive amount of material, especially in intellectual pursuits like subject indexing and classification, are significant global trends. This growing amount of processing work presents opportunities and difficulties for libraries worldwide. On

B. Rautaray et al. (eds.), *Proceedings of the International Conference on Marching Beyond the Libraries (ICMBL): Leadership, Creativity, and Innovation (ICMBL 2024)*, Advances in Economics, Business and Management Research 326,
https://doi.org/10.2991/978-94-6463-712-0_12

the one hand, a robust technological infrastructure is necessary for efficiently managing and analysing the massive volume of bibliographic data. On the other hand, it also presents chances for enhanced searchability, accessibility, and organisation of library resources, all of which will eventually benefit academics, students, and regular users. Indeed, the use of artificial intelligence (AI) and machine learning (ML) technologies has increased recently, with the promise of more effective management and processing of massive amounts of data. These days, an automated indexing system based on AI/ML can completely change how information workers manage enormous document collections. Such a system can do away with human indexing by using sophisticated algorithms to automatically analyse and classify articles according to their content.In addition to saving labour and time, this guarantees correctness and consistency in indexing. The area of library and information science (LIS) has already undertaken a number of semi-automated classification attempts. These projects usually use a mix of human and automatic procedures to categorise materials according to preset standards or vocabularies.

The use of supervised machine learning algorithms is the most popular method for automated subject indexing, also known as text categorization or text classification (Sebastiani, 2002). The German National Library is one of the national libraries that has created a completely automated subject classification system utilizing Dewey Decimal Classification (DDC) (Junger, 2017).The National Library of Finland developedAnnif (https://www.annif.org/) framework for automated subject indexing is an example of an open-source AI/ML application. Annif has developed a novel and useful method for the library and information science community that combines open-source technologies to anticipate topic headings or class numbers for documents based on popular knowledge organisation systems (KOSs) including LCSH, UDC, MeSH, and Agrovoc.

2. Literature Review

Since the 1990s, the development of artificial intelligence and machine learning (AI/ML) systems for knowledge organisation has been a key area of research in the discipline of library and information science (LIS) (Golub, 2021). Recent research in our library area has demonstrated the effectiveness of deep learning, neural network techniques, and convolutional neural networks as tools for collection discovery, search, and analysis (Golub et al., 2024). In their study, Desale& Kumbhar (2013) examined the earliest attempts made in the 1970s to automate the process of subject classification. The potential of a semi-automated indexing system based on AI and ML in a library that can handle large volumes of materials was investigated by (Ahmed et al., 2023) in their study. Using the Python virtual environment, it installs and configures an open-source AI environment called Annif and feeds it datasets from the Library of Congress Subject Headings (LCSH) and Linked Open Data (LOD) as a traditional KOS (Knowledge Organisation System). Golub et al. (2024) made an effort to determine the use of automatically generated DDC classes for Swedish digital collections as well as the efficacy of six machine learning techniques and a string-matching algorithm based on DDC features. Modern machine learning methods require at least 1,000 training instances for each class. In their research, Halder & Biswas (2023) demonstrated how to use CCLitBox to create class numbers for Indian literature using the Colon Classification, Sixth Edition. Using faceted classification and linked open data (LOD), the Wikidata tool CCLitBox is used to automatically classify literary writers and works (Bianchini, 2023). Several studies have reported that the Annif framework can be used to assign subject headings to new documents (Suominen, 2019;Suominen et al., 2022).In another research Agris offers the training dataset and the Agrovoc thesaurus is used as a vocabulary. Annif will be used in the study to automatically produce subject keywords and descriptors for agricultural documentary resources (Ahmed, 2023). According to Mukhopadhyay (2023), datasets obtained from MARC records collected from many libraries worldwide can be used to train the Annif framework, which can handle the linked open data format of LCSH. Through comparison with a set of retrieval measures,

this study assessed the applicability of three Annif machine learning backends: TF-IDF, Omikuji, and Neural Network.Using Homosaurus as the vocabulary backend, Mitra & Mukhopadhyay (2023) created a REST/API call-based method for indexing a sizable number of documents pertaining to the LGBTQIA+ domain.

3. Objectives

The primary objectives of this research endeavor followed:

- To create a formatted bibliographic dataset for Annif that consists of MARC Bibliography records with notes summarising the entries (tag 520 $a), subject descriptors (tag 650 $a), titles (tag 245$a and 245$b), and DDC notation (tag 082 $a).
- To implement an AI/ML framework (here, Annif) to import subject descriptors from LCSH in Turtle (.ttl) format along with the DDC major class dataset up to 3rd summary of Dewey or 1000 divisions, in order to support vocabulary control and facilitate vocabulary maintenance.
- To compare and measure effectiveness and performances between different machine learning backends in Annif, such as associative models (FastText, Omikuji, SVC), and ensemble models (Simple and Neural Network), by comparing and examining their output.

4. Methodology

This study previously mentioned section addresses the general duties that had to be completed in order to achieve the objectives of this research. Methodology is divided into two parts. The group I methodology involved gathering and organising MARC-formatted bibliographic records from various libraries, putting them in an Annif-compatible structure as training, validation, and test datasets and group II methodology involved creating a vocabulary that complies with SKOS standards, in this case DDC 1000 divisions along with subject labels from LCSH, training various machine learning backends using the training dataset, and assessing the effectiveness of these deployed models using retrieval metrics like primarily F1@5 and NDCG.

4.1. Group I Methodology

Group-I methodology primarily deals with gathering, organising and curating of data. Approximately 500,000 MARC records have been obtained for this research project from several sources, including the Library of Congress and Harvard Dataverse. After data curation, 213,879 MARC 21-formatted bibliographic records are ultimately chosen from the initial dataset of 500,000 records. These records contain all necessary data elements, such as tag 082 (DDC Call Number), tag 245 (title and subtitle of documents), tag 520 (summary note), and tag 650 (subject descriptors from LCSH). The next stage is to combine MARC files (from many sources) into a single, consolidated file with a structure that works with the Annif framework. For these data-intensive tasks, an open-source data wrangling program called OpenRefine has been used in conjunction with the MARC data management application MarcEditsoftware. After that, the final dataset (215,238 records) is split into three groups for the following three uses:

• A validation dataset which consisting of 2% recordsfrom the final dataset to obtain the hyperparameter optimization-based weightage formula for various machine learning backends from the Simple Ensemble model, which can be applied during the construction of the neural network based automated indexing/classification system; and
• A training dataset consisting of 96% records from the final dataset to conduct training for different machine learning backends as adopted by this research study.
• A test dataset with 2% of the final dataset's records included to assess each deployed machine

learning backend's effectiveness against a set of built-in retrieval metrics in the Annif framework (specifically, F1@5 and NDCG).

4.2. Group II Methodology

The Annif framework must be installed and configured before the chosen machine learning backends can be deployed. Annif is the primary tool which used in the group II methodology part. This open-source AI/ML framework supports a variety of machine learning backends under three groups (Table-1) like associative group (FastText, Omikuji, SVC) and ensemble group (Simple, and Neural Network).

Table 1: Machine learning backends of Annif

Group	Backend Algorithm	Scope
Associative group	omikuji	Used for multi-lebel classification, including Parable and Bonsai.
	fasttext	Text classification.
	SVC	For handling interactions between the front-end and the back-end components of the application.
Ensemble	NN	Recognize underlying relationship in a set of data through a process that is closest to the human brain operates.

The associative approaches employed in conventional backends assist the establishment of associations between vocabulary items and words in a document, hence enabling the machine learning techniques to create linkages between lexical elements. Moreover, TensorFlow's neural network model implementation integration requires the use of an organised common vocabulary. This uses a vocabulary file in a Tab-Separated Values (TSV) file format encoded with UTF-8 (Figure1). The subject Uniform Resource Identifier (URI) is in the first column, the corresponding subject descriptor is in the second, and the class number (notation) is in the third column.

Table 2: Structure of Vocabulary

URI	Subject (label_en)	Class (notation)
<http://dewey.info/class/709/e23/>	Art-History	709

```
<http://dewey.info/class/111.85/e23/>   Aesthetics.                          111.85
<http://dewey.info/class/612/e23/>      Human physiology.                    612
<http://dewey.info/class/286/e23/>      Baptists -- Soviet Union -- Biography.  286
<http://dewey.info/class/371.3/e23/>    Ethnography.                         371.3
<http://dewey.info/class/152.1/e23/>    Intersensory effects.                152.1
<http://dewey.info/class/306.461/e23/>  Social medicine.                     306.461
<http://dewey.info/class/158.7/e23/>    Psychology, Industrial.              158.7
<http://dewey.info/class/158.3/e23/>    Counseling.                          158.3
<http://dewey.info/class/200/e23/>      Religion.                            200
```

Figure 1: Vocabulary Creation

5. Results

Measuring the efficacy of machine learning backends is crucial for creating accurate and efficient information retrieval systems, such an automatic subject indexing system. One common technique for evaluating machine learning backends is retrieval metrics. Here, we tested the ability of several backends (SVC, FastText, Omikuji) to run and evaluate experiments to determine if Annif backends can accurately predict subject automatically. Automated indexing framework can be used in two ways: 1) via a Web UI micro-service running at port http://127.0.0.1:5000 (Figure 2) and also in command prompt (Figure 3).

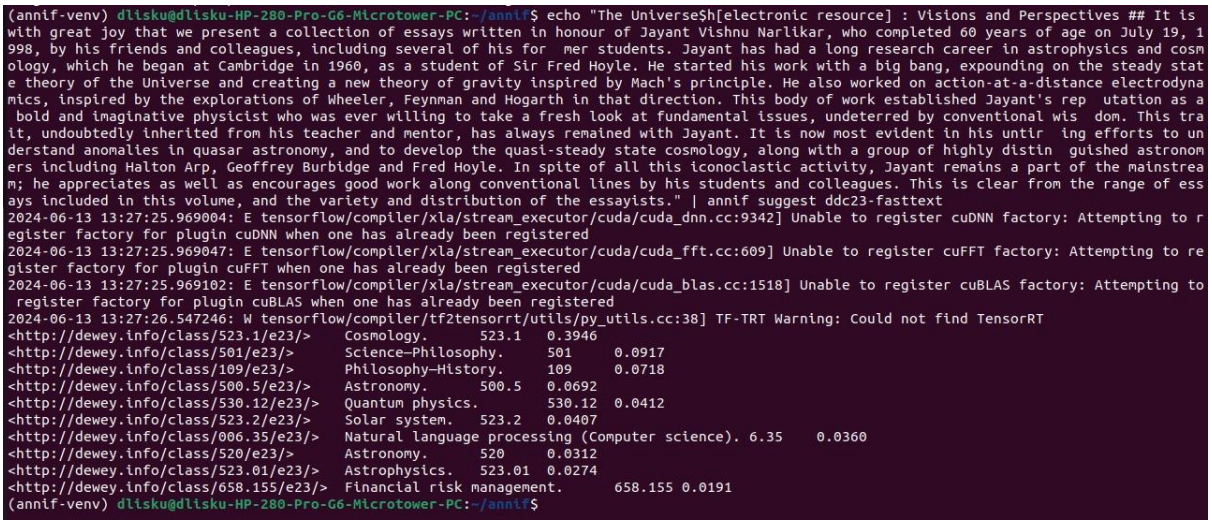

Figure 2: Predicting descriptors in Annif with accuracy scores (fasttext backend)

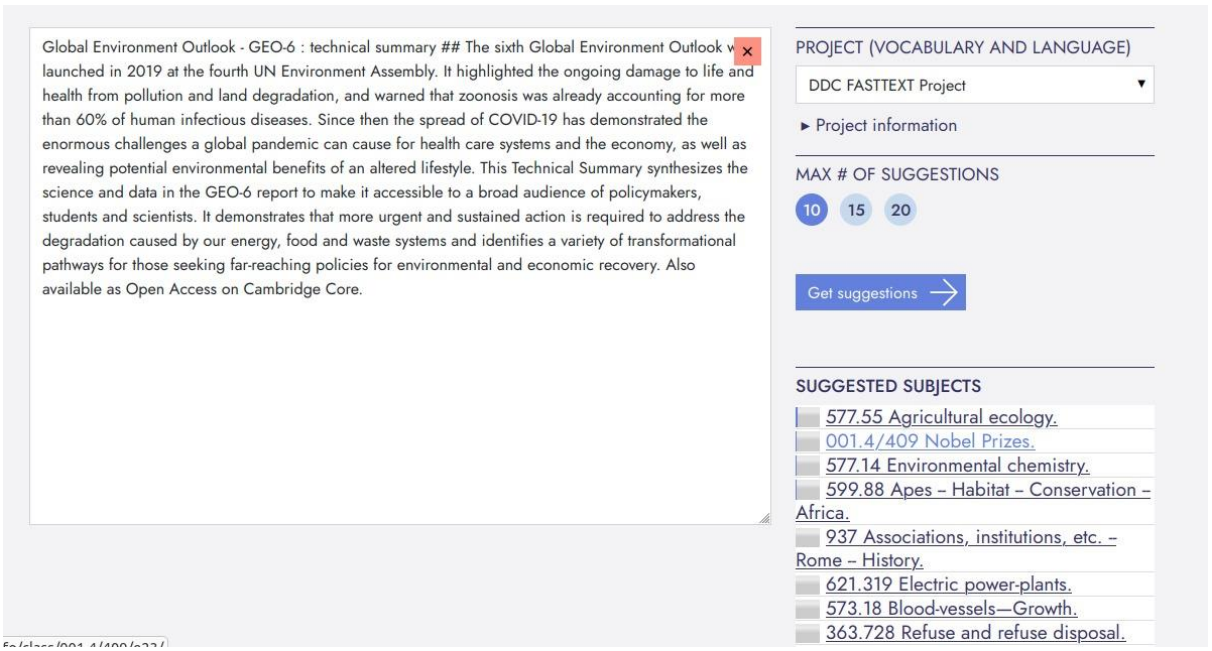

Figure 3: The framework in Web UI (fasttext backend)

On the other hand, Annif provides a wide range of retrieval measures for assessing the accuracy of subject prediction, including precision, recall, F1 score, F1@5, and Normalised Discounted Cumulative Gain (NDCG). The Annif framework adds more support for evaluating a machine learning backend's efficacy through the use of array retrieval matrices. It is believed that F1@5 and

NDCG are the most important metrics out of all of them. F1@5 (F1 at 5) results indicate what precision means at a cut-off point of 5. It merely provides a way to rate the algorithm's recall and accuracy for the top five predicted subject descriptions.F1@5 only considers the precision of the top 5 documents that were retrieved, whereas NDCG considers the precision of each document that was acquired in the ranked list. Based on retrieval measures, the efficacy of the selected machine learning backends is compared (Table 3) as follows:

Table 3: Comparison of performance for different backends

Retrieval Metrics	Backends			
	FASTTEXT	**OMIKUJI B**	**SVC**	**NN**
Precision (doc avg):	**0.0982**	**0.0993**	**0.0991**	**0.1927**
Recall (doc avg):	**0.9769**	**0.9929**	**0.9912**	**0.8587**
F1 score (doc avg):	0.1783	0.1805	0.1802	0.2952
Precision (subj avg):	0.0005	0.0005	0.0005	0.0011
Recall (subj avg):	0.0031	0.0032	0.0032	0.0025
F1 score (subj avg):	0.0007	0.0008	0.0008	0.0012
Precision (weighted subj avg):	0.1322	0.1602	0.1515	0.289
Recall (weighted subj avg):	0.9769	0.9929	0.9912	0.8587
F1 score (weighted subj avg):	0.2258	0.2696	0.2557	0.0012
Precision (microavg):	0.0979	0.0993	0.0991	0.289
Recall (microavg):	0.9769	0.9929	0.9912	0.158
F1 score (microavg):	0.178	0.1805	0.1802	0.3633
F1@5:	**0.3144**	**0.3293**	**0.33**	**0.3335**
NDCG:	**0.8164**	**0.9277**	**0.9442**	**0.6926**
NDCG@5:	0.8052	0.9261	0.9438	0.6817
NDCG@10:	0.8164	0.9277	0.9442	0.6926
Precision@1:	0.6294	0.833	0.8713	0.5031
Precision@3:	0.2949	0.3264	0.3285	0.2763
Precision@5:	0.1887	0.1976	0.198	0.2191
True positives:	4433	4506	4498	3897
False positives:	40827	40874	40882	21944
False negatives:	105	32	40	641
Documents evaluated:	4538	4538	4538	4538

[Source: From Text Corpus to Dewey Number: Designing a Prototype for Automated Classification (unpublished)]

A significant conclusion can be drawn from the analysis of the machine learning backends' performance that was employed in this study

- SVC backend perform best in case of both F1@5 and NDCG score;
- In comparison to other backends, SVC and NN machine learning backends performed better across all important metrics;
- The Neural network (NN-Ensemble) backend outperformed FastText and Omikuji backends.

6. Conclusion

Here Annif an open-source AI/ML tool developed by a library for libraries, is to offer tools for automated bibliographic data classification and subject indexing. For text classification, Annif uses a variety of machine learning and natural language processing techniques. The main responsibility of Annif is to suggest subject headings and classification numbers for the bibliographic entries in order to improve the quality of processing those records quickly. Users can add their own vocabulary, classifiers, and machine learning models to the completely adaptive framework. Although a prototype for automated categorisation using machine learning techniques has been successfully demonstrated in this work, it is important to keep in mind that the scope is restricted to the principal classes, or the 1000 divisions of the Dewey Decimal categorisation system. Since DDC is actually a very big knowledge organisation model with millions of classes, more research is necessary to determine whether Annif and the machine learning backends that have been implemented are suitable in terms of scalability.

References

Ahmed, M. (2023). Automatic indexing for agriculture: designing a framework by deploying Agrovoc, Agris and Annif. *Journal of Information and Knowledge*, *60*(2), 85–95. https://doi.org/10.17821/srels/2023/v60i2/170966

Ahmed, M., Mukhopadhyay, M., & Mukhopadhyay, P. (2023). Automated knowledge organization AI/ML based subject indexing system for libraries. *DESIDOC Journal of Library & Information Technology*, *43*(1), 45–54. https://doi.org/10.14429/djlit.43.01.18619

Bianchini, C. (2023). CCLitBox. A Wikidata gadget to classify world literature. *Journal of Information and Knowledge*, *60*(3), 133–141. https://doi.org/10.17821/srels/2023/v60i3/171024

Desale, S. K., & Kumbhar, R. M. (2013). Research on automatic classification of documents in library environment: A literature review. *KNOWLEDGE ORGANIZATION*, *40*(5), 295–304. https://doi.org/10.5771/0943-7444-2013-5-295

Golub, K. (2011). Automated subject classification of textual documents in the context of web-based hierarchical browsing. *KNOWLEDGE ORGANIZATION*, *38*(3), 230–244. https://doi.org/10.5771/0943-7444-2011-3-230

Golub, K. (2021). Automated subject indexing: An overview. *Cataloging & Classification Quarterly*, *59*(8), 702–719. https://doi.org/10.1080/01639374.2021.2012311

Golub, K., Suominen, O., Mohammed, A. T., Aagaard, H., & Osterman, O. (2024). Automated Dewey decimal classification of Swedish library metadata using Annif software. *Journal of Documentation*. *ahead-of-print*(ahead-to-print). https://doi.org/10.1108/JD-01-2022-0026

Halder, D., & Biswas, M. (2023). Machine-Generated Colon class numbers: automatic classification of Indian literary works in the wikidata environment. *Journal of Information and Knowledge*, *60*(3), 143–149. https://doi.org/10.17821/srels/2023/v60i3/171025

Jenkins, C., Jackson, M., Burden, P., & Wallis, J. (1998). Automatic classification of web resources using Java and Dewey decimal classification. *Computer Networks and ISDN Systems*, *30*(1-7), 646-648.

Junger, U. (2017). Automation first – the subject cataloguing policy of the Deutsche Nationalbibliothek. http://library.ifla.org/id/eprint/2213/

Mitra, R., & Mukhopadhyay, P. (2023). Machine learning applications in digital humanities: designing a semi-automated subject indexing system for a low-resource domain. *DESIDOC Journal of Library & Information Technology, 43*(4), 219-225. https://doi.org/10.14429/djlit.43.04.19227

Mukhopadhyay, P. (2023). Machine learning and bibliographic data universe: assessing efficacy of backend algorithms in Annif through retrieval metrics. *SRELS Journal of Information Management*, *60*(1), 39–48. https://doi.org/10.17821/srels/2023/v60i1/170891

Panigrahi, P., & Prasad, A. R. D. (2007). Facet sequence in analytico synthetic scheme: A study for developing an AI based automatic classification system. *Annals of Library and Information Studies, 54*(1),37-43.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, *34*(1), 1–47. https://doi.org/10.1145/505282.505283

Suominen, O. (2019). Annif: DIY automated subject indexing using multiple algorithms. *LIBER Quarterly: The Journal of the Association of European Research Libraries*, *29*(1), 1–25. https://doi.org/10.18352/lq.10285

Suominen, O., Inkinen, J., & Lehtinen, M. (2022). Annif and Finto AI: developing and implementing automated subject indexing. *JLIS.It, 13*(1), 265-282. https://doi. org/10.4403/jlis.it-12740