Automated Dewey Decimal Classification of Swedish library metadata using Annif software

Koraljka Golub
iInstitute, Linnaeus University, Vaxjo, Sweden
Osma Suominen
Library Network Services, The National Library of Finland, Helsinki, Finland
Ahmed Taiye Mohammed
iInstitute, Linnaeus University, Vaxjo, Sweden, and
Harriet Aagaard and Olof Osterman
National Library of Sweden, Stockholm, Sweden

Received 17 August 2022
Revised 20 January 2023
Accepted 31 January 2024

Abstract

Purpose – In order to estimate the value of semi-automated subject indexing in operative library catalogues, the study aimed to investigate five different automated implementations of an open source software package on a large set of Swedish union catalogue metadata records, with Dewey Decimal Classification (DDC) as the target classification system. It also aimed to contribute to the body of research on aboutness and related challenges in automated subject indexing and evaluation.

Design/methodology/approach – On a sample of over 230,000 records with close to 12,000 distinct DDC classes, an open source tool Annif, developed by the National Library of Finland, was applied in the following implementations: lexical algorithm, support vector classifier, fastText, Omikuji Bonsai and an ensemble approach combing the former four. A qualitative study involving two senior catalogue librarians and three students of library and information studies was also conducted to investigate the value and inter-rater agreement of automatically assigned classes, on a sample of 60 records.

Findings – The best results were achieved using the ensemble approach that achieved 66.82% accuracy on the three-digit DDC classification task. The qualitative study confirmed earlier studies reporting low inter-rater agreement but also pointed to the potential value of automatically assigned classes as additional access points in information retrieval.

Originality/value – The paper presents an extensive study of automated classification in an operative library catalogue, accompanied by a qualitative study of automated classes. It demonstrates the value of applying semi-automated indexing in operative information retrieval systems.

Keywords Automated subject indexing, Automatic classification, DDC, Annif, Libris, Supervised machine learning, Lexical algorithm, Ensemble approach, Qualitative evaluation

Paper type Article

Introduction

With more and more information resources in the online world, often overstretched budgets of libraries hardly suffice in providing quality subject access points especially in new digital collections and cross-search services (see, e.g. Golub, 2016, 2018). Large libraries are more and

Thanks to student volunteers: Linnéa Bengtsson, Nic Olsson and Joel Tjerneld.

Journal of Documentation
Vol. 80 No. 5, 2024
pp. 1057-1079
Emerald Publishing Limited
0022-0418
DOI 10.1108/JD-01-2022-0026

more exploring (semi)-automated approaches to subject indexing and classification. However, generally accepted approaches that are functional in operative library systems are lacking and even early adopters are still in the process of identifying the possibilities. A rare example of a well-researched approach used in an actual library is machine-aided indexing software of the National Library of Medicine (US National Library of Medicine, 2019) which has been developed for decades. Other libraries have implemented automated subject indexing more recently; for example, the German National Library developed a fully automated system for subject classification using Dewey Decimal Classification (DDC) (Junger, 2017). The fully automated approach has been criticised (Wiesenmüller, 2017). This is linked to the problematics of evaluating automated subject indexing and classification, often conducted outside of the context of operative information systems (Golub *et al.*, 2016).

In order to help further identify the potential of applying an automated approach to DDC classification of textual documents in Swedish Union Catalogue Libris, this paper builds on earlier work (Golub, 2021) and introduces a new suite of algorithms through Annif (Suominen, 2019), an open source automated subject indexing and classification tool developed by the National Library of Finland. The sample comprises circa 230,000 Swedish-language catalogue records. The article further illuminates the challenge of evaluation based on a study of how a group of two senior DDC professionals and three students who took a course on DDC judged the assigned DDC classes.

The remainder of the paper is structured as follows: the Background section outlines approaches to automated subject classification, related work and key challenges with a focus on evaluation. The Methodology section presents the DDC, data collection, different Annif backend algorithms and evaluation. The Results section in its first part shows performance of the different algorithms using the common approach of training and testing documents and in its second part it describes the evaluation study involving four sets of expert opinions. Implications and further research are given in the Conclusion section.

Background

Automated subject indexing and classification

There are different approaches to automated subject indexing and classification, based on the purpose of application, but also coming from different research fields and traditions, accompanied by varied terminology. In the context of libraries, the current ISO indexing standard, ISO 5963:1985, confirmed in 2020 (International Organization for Standardization, 1985) defines subject indexing performed by the information professional as a process involving three steps: (1) determining the subject content of a document; (2) a conceptual analysis to decide which aspects of the content should be represented; and, (3) translation of those concepts or aspects into a controlled vocabulary such as a classification system like DDC or a subject headings system like Library of Congress Subject Headings (LCSH).

Automated subject indexing is then a machine-based subject indexing where human intellectual processes of the above three steps are replaced by, for example, statistical and computational linguistics techniques. A common general approach to automated subject indexing or classification, often called text categorisation or text classification, is the application of *supervised machine learning* algorithms. Here the algorithm "learns" about characteristics of target index terms or classes based on characteristics of documents that had been manually pre-assigned those index terms – these documents are called training documents. The output of the training process is a *model*, a set of data representing what has been inferred from the training documents. The model is then tested with a new set of documents from the collection – called test documents. Often a third set of documents – called the validation set – is used to evaluate models during experimentation, for example to select the best performing hyperparameters (configuration settings) for algorithms before

performing the final evaluation on the test documents. Text classification tasks can be further divided into multi-class and multi-label problems; in multi-class problems the task of the algorithm is to predict a single correct class or category for a document and the classes are mutually exclusive, while in a multi-label problem there can be more than one correct answer and the algorithm has to predict a set of classes or index terms.

There are many different ways to build supervised machine learning classifiers, for example support vector machines (SVM) (e.g. Lee *et al.*, 2012; used by Annif as described below), artificial neural networks (e.g. Ghiassi *et al.*, 2012; You *et al.*, 2019), linear models with some additional tricks such as the fastText algorithm (Joulin *et al.*, 2016, used by Annif as described below), tree-based methods (e.g. Khandagale *et al.*, 2020; and the Omikuji library used by Annif as described below), and most recently, deep learning approaches such as transformer models (Chang *et al.*, 2020). Also, two or more different classifiers can be combined to make a classification decision, called ensembles (see, e.g. Toepfer and Seifert, 2020).

However, the supervised machine learning approach to text classification requires the existence of a relatively large number of training documents per each target class or subject index term. In many document collections there will be too few or no training documents available to train and test the classifier. The above mentioned challenge of the many detailed DDC classes for automated classification is further exacerbated by the fact that class numbers are typically built from several components. The first part of a DDC number indicates the main subject; this may then be followed by additional facets such as place and time, where the available choices are listed in auxiliary DDC tables. For example, the class number 929.209485, representing Swedish family histories, is built from class 929.2 (family histories) from the main DDC schedules and additional facets; from auxiliary Table 1, number 09 for geographical treatment as well as from auxiliary Table 2, number 485 for Sweden. The rules for building numbers are quite complex and often specific to individual parts of the hierarchies. Due to number building, there is potentially a very large number of possible class numbers; perhaps around 10^9, if all the permitted combinations were pre-built and made searchable in a computer interface (Brattli, 2012). The fact that it is both possible, and expected according to the DDC rule about being specific, to build a new number makes automated classification very challenging because the algorithms would need to predict all the different combinations and be able to suggest classes from a potentially extremely large vocabulary.

Related research

Earlier research of automating DDC assignment starts with the Online Computer Library Center's (OCLC) project Scorpion (OCLC, 2004). Experiments to automatically assign DDC classes to web documents were conducted within the Wolverhampton Web Library, at the time a manually maintained library catalogue of British web resources (Jenkins *et al.*, 1998).

Joorabchi and Mahdi (2014) explored the possibility to improve automated DDC classification based on external resources: references from the document to be classified and Wikipedia. Khoo *et al.* (2015) created DDC terms and numbers from pre-existing Dublin Core metadata.

Most closely related work is that by Golub (2021), which evaluated six machine learning algorithms as well as a string-matching algorithm based on characteristics of DDC. DDC classes found in the data set of circa 140,000 Swedish library catalogue records had to be reduced to top three DDC hierarchical levels in order to provide sufficient training data, resulting in 802 classes in the training and testing sample. Evaluation showed that Support Vector Machine with linear kernel outperformed other five machine learning algorithms as well as the string-matching algorithm on average; the string-matching algorithm outperformed machine learning for specific classes when characteristics of DDC were most

suitable for the task. Word embeddings combined with different types of neural networks (simple linear network, standard neural network, 1D convolutional neural network, and recurrent neural network) produced worse results than Support Vector Machine, but reached close results, with the benefit of a smaller representation size. Impact of features in machine learning shows that using subjects or combining titles and subjects gives better results than using only titles as input. Stemming only marginally improves the results. Removed stop-words reduced accuracy in most cases, while removing less frequent words increased accuracy marginally. The greatest impact is produced by the number of training examples: 81.90% accuracy on the training set is achieved when at least 1,000 records per class are available in the training set, and 66.13% when too few records (often less than 100 per class) on which to train are available – and these hold only for top 3 hierarchical levels (803 classes only). Another possible reason for improved performance in the latter case could be that choosing only classes with a large number of training documents resulted in a smaller number of classes to choose from, and thus increasing the potential to guess right.

Application in operative systems

When it comes to use of (semi-)automated solutions in actual library systems, they have still not been widely adopted in libraries. Reported examples include Medical Text Indexer (U.S. National Library of Medicine, 2019), a well-researched machine-aided indexing (MAI) or computer-assisted indexing (CAI) approach in which it is the human indexer who decides, based on a suggestion provided by the computer, which by 2017 was consulted by indexers in over 60% of articles indexing (Mork *et al.*, 2017). Another example is NASA's MAI software which was shown to increase production and improve indexing quality (Silvester, 1997). In the past decade some major efforts have been under way to allow implementation of automated approaches in libraries. For example, the U.S. National Agricultural Library has been using a fully automated indexing workflow since 2013, where human indexers only do spot checks on the index terms provided by the automated indexing tool (Finch, 2014). More recently, International Nuclear Information System (INIS), operated by the United Nations' International Atomic Energy Agency (IAEA), resorts to automated methods to generate subject terms from their multilingual thesaurus (Hakopov *et al.*, 2018). ZBW, the Leibniz Information Centre for Economics in Germany, is implementing an in-house solution for automated subject indexing after many years of research on methods (Kasprzik, 2020). The German National Library developed a fully automated system for subject classification using Dewey Decimal Classification (DDC) (Junger, 2017). Annif, an open source tool for automated subject indexing developed by the National Library of Finland (https://annif.org/) is used by some Finnish university repositories and for processing electronic deposits (Suominen *et al.*, 2022).

But to what degree can automated solutions be applied in operative information systems? Software vendors and experimental researchers speak of the high potential of automated indexing tools. While some claim to entirely replace manual indexing in certain subject areas (e.g. Roitblat *et al.*, 2010), others recognise the need for both manual (human) and computer-assisted indexing, each with its (dis)advantages (e.g. Anderson and Perez-Carballo, 2001; Jonasen and Lykke, 2013). The fully automated approach applied by the German National Library has thus been criticised (Wiesenmüller, 2017; Conradi, 2017). Hard evidence on the success of automated indexing tools in operating information environments, however, is scarce; research is usually conducted in laboratory conditions, excluding the complexities of real-life systems and situations, making it hard to judge the practical value of automated indexing tools (Lancaster, 2003, p. 334).

To explain this further, a common evaluation approach is to measure indexing quality directly. One method of doing so is to compare automatically assigned subject classes or index

terms against existing human-assigned classes or terms (as a "gold standard"), but this method has problems. When indexing, people make errors which could be related to: (1) exhaustivity (too many or too few subjects assigned in relation to indexing policy, e.g. the same book will have an extensive list of topics in a national bibliography, but only one in a school library); (2) specificity, usually because the assigned subject is not the most specific available esp. in a large classification system such as DDC; (3), they may omit important subjects, or assign an obviously incorrect subject. Therefore, existing metadata records should not be used as the sole "gold standard': the classes assigned by algorithms (but not human-assigned) might be wrong or might be correct but omitted during human indexing by mistake or by abiding to a certain indexing policy. However, in reality metadata records and existing test collections are used as the gold standard which can be helpful for at least initial selection of algorithms and fine tuning them for preliminary evaluation. A comprehensive approach involving expert reviews to evaluation is needed to gather a more complete picture of the success of the chosen software; for an in-depth discussion of the topic, see Golub *et al.* (2016).

Methodology

The section first describes the data collection from the Swedish Union Catalogue, then the DDC as the target classification system and the Annif tool with its five different backend algorithms. It ends with describing the evaluation approach taken in this work.

Data collection

The complete Swedish National Union Catalogue, Libris, dated 21 March 2021, was obtained from the Swedish National Library as a data dump in JSON-LD format. The full dump contained over 51 million records, of which 1,9 million records contained at least one DDC class. A large proportion of these records were English language records which had been imported, along with DDC classes, from other bibliographic databases. Out of these 1,9 million records 286,394 records with following characteristics were extracted:

(1) The record represents a bibliographic work (contains both a work and an instance, with an "instanceOf" relationship connecting them) rather than an authority record such as a person or a subject heading.

(2) The record contains a title (a "hasTitle" relationship is present).

(3) The language of the work is Swedish ("swe").

(4) The record has at least one DDC class which has one of the edition codes "22" or "23" when DDC edition 22 or 23 is applied by a library from outside of Sweden, as well as "22/swe" or "23/swe" when DDC code is entered by a Swedish library; or no edition code which implies imported records likely using either 22nd or 23rd edition.

(5) The DDC class number from the record is found in the DDC file either directly or by truncating the class number starting from the end character until a match is found. In this way facet information added from auxiliary tables is discarded from built numbers and the result is one of the 27,188 available classes (see below).

In the next step, from those 286,394 catalogue records duplicates were removed (i.e. records which had an identical main title and subtitle). Also removed were 60 records used in the expert evaluation study described below. This resulted in 232,599 records with 11,704 distinct DDC classes, showing that less than half of available DDC classes (27,188) were used by Libris at the time. For approximately 42% of the records, the DDC number class had to be truncated in order to match one of the available classes.

Finally, the following fields were extracted into a TSV file suitable for use with the Annif tool: title (main title and subtitle), subject information (74.5% records had at least one subject–usually subject headings, but also, e.g. people or organisations used as subjects) and the DDC classes. The data collection was then randomly split into train (90%), validation (5%) and test (5%) subsets, stored as separate TSV files.

Also important to mention is that while the great majority of records were classified with exactly one DDC class (98.8%), 2,859 of the records (1.2%) included more than one DDC class. We decided to include the records with multiple DDC classes, in contrast to earlier work (Golub, 2021) where these were excluded from the sample. This turns the classification task into a multi-label problem and has some implications for the classification algorithms as well as the evaluation metrics that will be explained below.

Dewey Decimal Classification

A dump of the 23rd edition of the Swedish Dewey Decimal Classification (DDC) was obtained from Pansoft, the maintainers of the WebDewey system, dated 2021-02-15. The dump consisted of a MARCXML file with the DDC classes with Swedish language headings represented using the MARC 21 Format for Classification Data.

Extracted were all DDC records with a class number of at least three digits which met the following criteria:

(1) Contains a 153 field comprising the class number and optionally a class heading.

(2) It is not an auxiliary table record (containing 153 subfield $z or $y).

(3) It is not an internal summary record having one of the tags ess = si1, ess = si2, ess = i2 or ess = se3 in the 153 subfield $9.

(4) It represents a single class number, not a range of classes.

(5) The class number has at least three digits (one-digit and two-digit class numbers are used only at the summary level, not for classifying individual documents).

This resulted in the total of 27,188 DDC classes. For each of these classes, the following information was extracted: the class heading (153 subfield $j), all Relative Index terms (from a group of fields beginning with 7xx) and any notes (from the fields 253, 353 and 680).

Then, two TSV files were constructed: (1) in order to allow loading the vocabulary to Annif, a *vocabulary file* listing the class number, a URI identifier constructed from the class number because Annif requires that all subjects are identified by a URI, and the class heading; and (2) in order to build a lexical model from the DDC data, a *term file* containing the class URIs as well as any terms from the class description: the heading, Relative Index terms and notes.

Full DDC and three-digit DDC classes

The number of classes in the target classification system is proportionally related to the difficulty of the automated classification task and also to the computational resources required for performance of individual algorithms. We decided to perform the classification experiments on two different variations of the DDC classification:

(1) The full DDC with 27,188 classes, although excluding a large number of numbers that can be potentially built; and,

(2) A much smaller version of 1,000 classes restricted to the top three levels represented by the first three digits of the class numbers, following the example of earlier work (Golub, 2021) which reduced classes with four or more digits to their three-digit root in order to increase the number of available training documents per class.

The three-digit DDC file was constructed by truncating all class numbers in the original term file to the first three digits while all the terms (headings, subjects and notes) of deeper classes were retained with their respective higher-level three-digit class. Accordingly, alternative versions of the Libris train, validation and test files were created where the class numbers were truncated to the top three digits, resulting in a total of 841 distinct classes (this is how many of the 1,000 three-digit DDC classes are used in Libris).

As in earlier research (see Background), distribution of classes is heavily imbalanced in the Libris data set. In the full DDC data set, the most frequent class is 839.738 (Swedish fiction, 21st century), which appears in 5.36% of records, followed by 839.7374 (Swedish fiction, 1945–1999) present in 1.90% of records. Other common classes are 948.6 (Southern Sweden (Götaland); 1.09%), 839.72 (Swedish drama; 1.02%) and 823 (English fiction; 0.91%). In the three-digit DDC data set, the most frequent class is 839 for Germanic literature and appears in 13.13% of records, followed by 948 (Scandinavia; 3.06%), 362 (Social security; 2.18%), 658 (Administration; 1.95%) and 782 (Vocal music; 1.84%).

In order to be inclusive of the entire classification system and especially since fiction represents such a high proportion of classes, the dataset has initially also included fiction. However, it was later shown that records representing fiction were often wrongly classified, likely due to the fact that fiction themes are largely classified by language or culture of origin rather than actual topics like love, relationships, crime etc.; the former are hard to tease out based on title information only. Future tests may consider this and exclude fiction from such experiments, while new approaches need to be experimented with in which to address the challenge of automatically classifying works of fiction.

Classification algorithms

Annif is a toolkit for training, evaluating and applying machine learning models for automated subject indexing and classification of textual documents. It is a software package that incorporates many specific algorithms in modules called *backends*. There are backends for base algorithms as well as ensemble backends that combine the output of other backends, as the best results are often obtained by combining multiple algorithms. For more about Annif, please see Suominen (2019) and Suominen *et al.* (2022). Annif version 0.53, released in June 2021, was used for the experiments.

All Annif experiments were performed on a single computer with 512 GB RAM (Random Access Memory) and two AMD EPYC 7401 processors, totalling 48 cores (96 threads). GPU computing (the use of a Graphical Processing Unit to speed up heavy computation) was not used as none of the tested algorithms support it.

Table 1 below provides an overview of four base algorithms and their configurations. These choices were informed by initial experimentation during which different algorithms and configurations had been explored and evaluated on the validate set. In preprocessing, bigrams were used for those algorithms that supported them. Snowball stemming was used with the lexical and Omikuji algorithms. Stop-words removal is currently not supported in Annif but this was not considered problematic because it has not been helpful in earlier research (Golub, 2021).

For each of the four base algorithms, two models were trained in parallel, one for the full DDC task and another for the three-digit DDC task. In most cases the same configuration was used for both tasks, but there were some configuration differences between the tasks for the SVC (Support Vector Classifier) and fastText algorithms. The algorithms and their configurations are defined in the following sections.

Algorithm	Input data	Stemming	Other settings for 3-digit DDC	Other settings for full DDC
Lexical	Terms from DDC	Snowball	–	–
SVC	Libris train set	–	ngram = 2	ngram = 2 min_df = 2
fastText	Libris train set	–	wordNgrams = 2 minn = 5 maxn = 5 loss = softmax dim = 150 epoch = 50 lr = 0.4234 minCount = 4	wordNgrams = 2 minn = 5 maxn = 5 loss = softmax dim = 150 epoch = 45 lr = 0.9740 minCount = 3
Omikuji	Libris train set	Snowball	ngram = 2 cluster_balanced = False cluster_k = 100 max_depth = 3	ngram = 2 cluster_balanced = False cluster_k = 100 max_depth = 3

Source(s): Authors' own creation

Table 1.
Base algorithm configurations

Lexical algorithm

For the lexical algorithm, we used the simple TFIDF (Term Frequency Inverse Document Frequency) backend in Annif using text (headings, subjects and notes) from the DDC term file as the target data against which to match the catalogue records. The text was tokenized into words (unigrams only) and Snowball stemming was applied. The TFIDF backend projects the text from each DDC class into a TFIDF normalised vector space and searches for the nearest matches for the text of individual documents, similar to how text search engines perform text retrieval such as the Apache Solr system used in previous work (Golub, 2021).

Initially the DDC term file included all 27,188 classes, but since many of the classes are never actually used in Libris, we decided to include only the DDC classes that were used at least once in the Libris training set. This improved the precision of the lexical algorithm as it no longer could suggest classes that were never used in practice. This matches the behaviour of the machine learning algorithms, which can only suggest classes that were included in their respective training sets. The effective number of classes that the algorithm can suggest was thus reduced to 833 for the three-digit DDC classification task and 11,398 for the full DDC classification task.

Support vector classifier (SVC)

We used the linear SVM classifier (LinearSVC) implemented in the scikit-learn toolkit (Pedregosa *et al.*, 2011) that is integrated with Annif. The document text was first tokenized into both unigrams and bigrams (*ngram = 2* setting) and then converted into a numeric TFIDF matrix. No stemming was used. In the case of full DDC, only tokens that appeared in at least two records were included (*min_df = 2* setting), as otherwise the training time and model size would have been prohibitively large. The SVC algorithm cannot handle more than one class per document; thus, only the first DDC class of each training document was used and the others were ignored during training.

fastText

The fastText algorithm (Joulin *et al.*, 2016) is a machine learning method for text classification inspired by neural networks and created at Facebook Research. The fastText algorithm performs its own tokenization; no stemming was used, and both unigrams and bigrams

(*wordNgrams* = *2* setting) as well as character n-grams (length range defined my *minn* and *maxn* settings) were used. The algorithm provides several *loss functions* which guide the learning process towards its goal, with different properties. We selected *softmax* as the loss function – although it is much slower than *hs* (hierarchical softmax), it can usually achieve much higher precision.

The fastText algorithm is very sensitive to the choice of hyperparameters. We performed automated hyperparameter searches using the hyperopt library (Bergstra *et al.*, 2013), separately for the three-digit DDC and the full DDC tasks. For each candidate set of hyperparameters, a model was trained on the train set and evaluated on the validate set. In total 400 trials were performed for the three-digit classification task and 80 trials for the full DDC classification task, trying to achieve the highest possible precision@1 score (defined below) for the Libris validate set. The hyperparameters that achieved the highest precision, shown in Table 1 above, were chosen; these include the size of the vector representation (*dim* setting), the number of training epochs (repetitions over the training data set; *epoch* setting), the learning rate (*lr* setting) and the minimum number of occurrences for a token to be used for learning (*minCount* setting). The hyperparameter searches were computationally expensive, taking several days on the powerful computer we used for the experiments.

Omikuji Bonsai

Omikuji is a reimplementation of a family of efficient tree-based machine learning algorithms for multi-label classification, including Parabel (Prabhu *et al.*, 2018) and Bonsai (Khandagale *et al.*, 2020). Both unigrams and bigrams (*ngram* = *2* setting) were included in the TFIDF vectorisation and Snowball stemming was used. Based on initial experimentation, we selected hyperparameters emulating the Bonsai algorithm using the hyperparameters *cluster_balanced* = *False, cluster_k* = *100* and *max_depth* = *3* which constrain the shape of the classifier tree.

Ensemble approach

Classifier algorithms have their own strengths and weaknesses. A common strategy for improving the performance of such algorithms is to combine the output of several algorithms, which has the potential to reduce unwanted bias and overfitting in individual classifiers. Using the ensemble backend of Annif, which combines the output of included classifiers using weighted averaging, we combined all the four classification algorithms described above. We started by creating an ensemble consisting of all four base algorithms that were initially assigned equal weights; we call it LSFO, a term constructed from the initials of the four algorithms. We then re-tuned the ensemble weights using the built-in hyperparameter optimisation of the Annif ensemble backend. For each task (full DDC and three-digit DDC), we tested 200 different weight combinations evaluating them against the Libris validate set. The weights that achieved the highest nDCG score (see below) were then selected. In practice, the Omikuji Bonsai algorithm was assigned the highest weight in both settings (see Tables 4 and 5).

To investigate the relative contribution of each algorithm to the overall performance of the ensemble, we performed an ablation analysis. In this analysis, we excluded each base algorithm from the ensemble in turn, re-tuned the weights of the remaining algorithms and evaluated the performance of the resulting three-algorithm ensembles. In this way we could see how much the performance dropped when excluding a particular algorithm and thus how much benefit that algorithm was bringing to the ensemble. We could also estimate the resource consumption of individual algorithms.

Evaluation methodology

The main evaluation was conducted following a common approach of "gold standard', the latter being original DDC classes from the Libris catalogue records. Here established evaluation measures that are implemented in the Annif toolkit were used. The most common metric for multi-class classification tasks is accuracy (proportion of records that were correctly classified by the algorithm), but since a small number of Libris records included more than one DDC class and all the tested algorithms produced not just a single prediction but a ranked list of predictions, we have mainly used precision@1 instead. This is a multi-label classification metric and is defined as the proportion of records for which the first prediction matches any of the manually assigned classes. In addition, we have measured the performance of individual algorithms using the normalised discounted cumulative gain (nDCG) metric, a ranking measure commonly used in the evaluation of information retrieval systems (Järvelin and Kekäläinen, 2002). This metric reflects the quality of suggestions in a machine-assisted indexing setting where a human indexer is provided with a ranked list of automatically generated suggestions to choose from; a larger nDCG value (up to the maximum possible 1.0) indicates that the correct class is, on average, closer to the top of the list of suggestions.

In addition, because of the challenges of aboutness discussed in the Background section above, we wanted to see to what degree automatically assigned classes originally not in Libris records could be considered accurate. To this purpose, we have conducted an additional evaluation involving a team of two senior catalogue librarians and three undergraduate student volunteers who had previously completed a 7.5 ECTS credits course on subject indexing and classification which focuses on DDC classification.

The manual evaluation study was implemented as follows. First, the 2 senior catalogue librarians from the National Library of Sweden selected 60 records including 3 sets of 20 documents, each set from a different scientific area: health, religion and natural science, all for documents published in 2019. The records were created by cataloguers at the National Bibliography Department with a good knowledge of the DDC system. Second, the 60 records were then put in a spreadsheet, one record per row, and the three highest scoring classes that had been automatically generated using the Annif LSFO ensemble model for full DDC classification were then added to each corresponding document. The order was randomised. Four spreadsheet documents were disseminated to the five evaluators: the two senior catalogue librarians formed one team while each of the students conducted individual evaluations. Over a period of three weeks, they entered their evaluations of the automated DDC classes as "correct", "partly correct" and "incorrect".

Evaluators have received instructions via email in which the spreadsheet with evaluations was attached. The spreadsheet was described as having 60 rows where each row has 1 document with classes. Each row started with an ISBN and a title column. The evaluators were specifically instructed to use those two elements to look for more information on what the book is about, anywhere but in a library catalogue or Libris; this in order to acquire independent opinions. An online bookshop such as Bokborsen was suggested as it allows searching by ISBN while any other Swedish online bookshops support searching by title.

Following columns in each row were triples "DDC_class, DDC_heading, Is this a good class". These triples are marked in blocks of colour. The heading was added to help the evaluators more easily understand the class. For each DDC class, the evaluators were instructed to enter their evaluation as "yes" for a correct class, "no" for an incorrect class or "partly" for a partly correct class. This was to be done by looking at the document title and finding out more about it in Bokborsen or another bookstore in order to get an idea what the book is about. The very last column invited for any comments such as how sure the evaluator was about the decisions and also whether they thought another class, not listed, should be the best class to assign. The evaluations took place between 18 till 31 May 2021.

Results

Comparison against the Libris test set

As explained in the preceding section, the Libris test set comprised original DDC classes used to conduct evaluation of the Annif results. The five Annif algorithms were evaluated on the test set using precision@1 and nDCG as metrics. In addition, in order to estimate the impact of subject information on classification performance, precision@1 was measured in three set-ups: (1) on 2,884 records that had title only, no subjects (Precision, title only); (2) on 8,746 records that had both title and subjects (Precision, title and subjects); and, (3) on records from both set-ups (1) and (2). The results for the three-digit DDC classification task are shown in Table 2 and the corresponding results for the full DDC task in Table 3.

In the three-digit DDC classification task, the LSFO ensemble of the four base algorithms outperformed all the base algorithms by a notable margin: the overall precision was 62.91% (66.88% for records with subjects) and the nDCG score was 0.7625. Among the four base algorithms, SVC achieved the highest precision scores for records with subjects (66.24%) and for all records (61.15%). This was closely followed by Omikuji, which achieved precisions of 65.49 and 60.88%, respectively. In terms of nDCG scores, their relative order was reversed, with Omikuji achieving a score of 0.7445 and SVC 0.7380. The fastText algorithm falls slightly behind SVC and Omikuji on both measures. The lexical algorithm was far behind the others, with an overall precision of 13.63% (16.42% for records with subjects) and an nDCG score of 0.2464.

In the full DDC classification task, differences between the algorithms are more apparent. Of the base algorithms, Omikuji achieved the highest scores both in terms of precision (43.52% overall, 48.23% for records with subjects) and nDCG (0.5706). SVC and fastText were several percentage points behind, with SVC achieving higher precision but fastText having a better nDCG score. The performance of the lexical algorithm was even worse than in the three-digit task, with an overall precision of just 7.08% (8.61% for records with subjects). Again, the ensemble of four algorithms achieved higher precision (44.87% overall, 49.50% for records with subjects) and nDCG scores (0.5744) than any of the base algorithms.

Algorithm	Precision, title only	Precision, title and subjects	Precision, all records	nDCG
Lexical	5.17%	16.42%	13.63%	0.2464
SVC	45.74%	66.24%	61.15%	0.7380
fastText	45.80%	64.01%	59.49%	0.7378
Omikuji	46.88%	65.49%	60.88%	0.7445
LSFO ensemble	*50.66%*	*66.88%*	*62.91%*	*0.7625*

Source(s): Authors' own creation

Table 2. Precision@1 of the five algorithms on the three-digit DDC classification task

Algorithm	Precision, title only	Precision, title and subjects	Precision, all records	nDCG
Lexical	2.43%	8.61%	7.08%	0.1286
SVC	24.69%	45.12%	40.05%	0.5183
fastText	26.87%	43.93%	39.70%	0.5311
Omikuji	29.23%	48.23%	43.52%	0.5706
LSFO ensemble	*30.83%*	*49.50%*	*44.87%*	*0.5744*

Source(s): Authors' own creation

Table 3. Precision@1 of the five algorithms on the full DDC classification task

Ablation analysis

In the following step we wanted to see to what degree different algorithm components contribute to performance, a process known as ablation analysis. To this purpose, we compared the performance of the LSFO ensemble against simpler ensemble configurations consisting of three of the four base algorithms on all test records. The ensembles – LSF, LSO, LFO and SFO – are named based on the initials of their constituent base algorithms. Tables 4 and 5 below show, for the three-digit DDC and the full-DDC tasks respectively, the weights of the base algorithms that were chosen based on hyperparameter optimisation, the achieved precision@1 scores as well as memory usage of the Annif process.

For the three-digit DDC task, the largest drop in precision (approximately 1.3% points) was caused by exclusion of the fastText algorithm (LSO ensemble). Somewhat surprisingly, the LSF ensemble, excluding Omikuji, achieved higher precision (by 0.3% points) than the LSFO ensemble that included all four base algorithms. Despite the poor performance of the lexical algorithm in the evaluation of individual algorithms, the overall precision score was reduced when it was excluded, indicating that the lexical algorithm does bring benefit to the ensemble, as suggested in previous research (Golub, 2021). The largest reduction in memory usage–nearly 8 GB–resulted from exclusion of the SVC algorithm, indicating that SVC requires far more memory than any of the other base algorithms.

As to the full DDC task, the original LSFO ensemble consisting of all four algorithms achieved the highest precision, followed by LFO (excluding SVC) and SFO (excluding Lexical) both of whose precision scores were approximately 0.5% points lower. The largest drop in precision (approximately 2.5% points) was with the LSF ensemble excluding the Omikuji algorithm. Like above, excluding the lexical algorithm reduced the precision score, indicating that the lexical approach brings some benefit to the ensemble even if it performs poorly on its own. Again, the largest reduction in memory usage (approximately 17 GB) was seen when the SVC algorithm was excluded, indicating that the SVC algorithm requires much more memory than the other base algorithms.

Table 4.
Ablation analysis of ensembles, three-digit DDC task

Ensemble	Lexical weight	SVC weight	fastText weight	Omikuji weight	Precision@1	RAM usage
LSFO	4.36%	41.23%	7.97%	46.43%	62.91%	10.0 GB
LSF	8.65%	83.72%	7.63%	–	*63.21%*	9.7 GB
LSO	3.37%	38.76%	–	57.86%	61.67%	8.8 GB
LFO	3.48%	–	10.00%	86.51%	62.55%	*2.1 GB*
SFO	–	1.13%	17.21%	81.66%	62.44%	10.0 GB

Note(s): The italic value represents the best accuracy in the column
Source(s): Authors' own creation

Table 5.
Ablation analysis of ensembles, full DDC task

Ensemble	Lexical weight	SVC weight	fastText weight	Omikuji weight	Precision@1	RAM usage
LSFO	2.36%	14.79%	7.06%	75.78%	*44.87%*	21.2 GB
LSF	7.50%	59.83%	32.76%	–	42.33%	20.8 GB
LSO	0.70%	0.77%	–	98.54%	43.68%	19.9 GB
LFO	0.57%	–	6.66%	92.78%	44.38%	*4.5 GB*
SFO	–	0.32%	7.34%	92.34%	44.34%	21.1 GB

Note(s): The italic value represents the best accuracy in the column
Source(s): Authors' own creation

Common classification errors

In order to identify common classification errors, we analysed the test set records in which the top prediction of the LSFO ensemble model had not matched the DDC class assigned in Libris. Here we excluded the 195 test set records with more than 1 assigned DDC class to simplify the analysis, leaving us with 11,485 records. This exclusion also allowed us to apply the classification accuracy metric. In the three-digit DDC classification task, 62.83% of records were correctly classified; for the 8,646 records with subjects, the classification accuracy was 66.82%. In the full DDC classification task, 44.74% of records were correctly classified; for the records with subjects, the classification accuracy was 49.40%.

It quickly became apparent that records representing fiction had very often been wrongly classified, likely due to the fact that fiction themes are largely classified by language or culture of origin rather than actual topics like love, relationships, crime etc.; the former are hard to tease out based on title information only. We therefore decided to exclude the 17.3% of test set records with one of fiction genres in Libris records (which Libris adopted from MARC: FictionNotFurtherSpecified, Drama, Essay, Novel, HumorSatiresEtc, Letter, ShortStory, MixedForms, Poetry) in order to concentrate on the more interesting classification errors among the 9,461 remaining non-fiction records. This improved accuracy a little, to 62.96% for the three-digit classification task (66.88% for the records with subjects) and 46.46% for the full DDC classification task (49.77% for the records with subjects).

We also analysed classification accuracy for each of the ten top level DDC classes (000–900) by further splitting the test set of 9,461 non-fiction records by the first digit of the assigned DDC class in Libris. The results are shown in Table 6. For the three-digit DDC task, the highest accuracy of 70.9% was achieved for the class 600 Technology, which comprises 17.8% of the test set records, and the lowest accuracy was 56.2% for the class 200 Religion (4.5% of records). For the full DDC task, the highest accuracy of 62.1% was achieved for the class 400 Language, which comprises only 2.4% of the test set records, and the lowest accuracy 37.2% for the class 800 Literature (4.1% of records). The variation in accuracy between top level classes was large especially in the full DDC task.

Since a full confusion matrix (a table displaying frequencies for predicted vs actual classes, showing cases where the algorithm makes errors) would be impractical for a classification having hundreds or thousands of classes, we only looked at the most frequent pairs of predicted vs assigned classes. These correspond to the non-diagonal cells of the confusion matrix with the largest occurrence numbers.

Prediction errors that affected 10 or more non-fiction records for the three-digit DDC classification task are shown in Table 7. Many of these classification errors involved

DDC top level class	Frequency in test set	Accuracy, three-digit DDC	Accuracy, full DDC
000 Computer science, information and general works	427 (4.5%)	60.0%	50.1%
100 Philosophy and psychology	320 (3.4%)	58.1%	51.6%
200 Religion	427 (4.5%)	56.2%	43.1%
300 Social sciences	*3,051 (32.3%)*	62.4%	44.1%
400 Language	224 (2.4%)	66.1%	*62.1%*
500 Science	387 (4.1%)	58.9%	46.5%
600 Technology	1,686 (17.8%)	*70.9%*	52.1%
700 Arts and recreation	1,511 (16.0%)	60.3%	43.4%
800 Literature	392 (4.1%)	66.6%	37.2%
900 History and geography	1,029 (10.9%)	60.3%	46.9%

Note(s): The italic value represents the best accuracy in the column
Source(s): Authors' own creation

Table 6. Classification accuracy for non-fiction records with a single DDC class, by DDC top level class assigned in Libris

Freq.	Predicted DDC class	Assigned libris DDC class
26	839 Other Germanic literature	782 Vocal music
24	839 Other Germanic literature	920 Biography, genealogy, insignia
23	839 Other Germanic literature	791 Public performances
22	371 Schools and their activities, special education	370 Education
20	839 Other Germanic literature	783 Music for single voices
19	948 Scandinavia	914 Geography of and travel in Europe
19	759 Painters–History, geographic treatment, biography	709 Arts–History, geographic treatment, biography
17	362 Social problems of and services to groups of people	616 Diseases
15	371 Schools and their activities, special education	372 Primary education (elementary education)
14	616 Diseases	362 Social problems of and services to groups of people
13	616 Diseases	618 Gynaecology, obstetrics, paediatrics, geriatrics
12	839 Other Germanic literature	306 Culture and institutions
12	914 Geography of and travel in Europe	948 Scandinavia
11	839 Other Germanic literature	914 Geography of and travel in Europe
11	510 Mathematics	372 Primary education (elementary education)
11	372 Primary education (elementary education)	371 Schools and their activities, special education
10	783 Music for single voices	782 Vocal music
10	839 Other Germanic literature	809 History, description, critical appraisal of more than two literature
10	839 Other Germanic literature	439 Other Germanic languages
10	362 Social problems of and services to groups of people	305 Groups of people

Note(s): The class headings are shown here in English, although Swedish DDC was used in the experiment
Source(s): Authors' own creation

Table 7.
Most common prediction errors in the three-digit DDC prediction task

improperly predicting the class number 839 (other Germanic literature, including Swedish), which is also the most frequent class in the Libris data set; although fiction was excluded from the records used for this analysis, the algorithms often suggested this class number as the training data set included many fiction records. Other classification errors are more subtle, for example distinctions within education (370 Education vs 371 Schools and their activities, special education vs 372 Primary education (elementary education)), geography (914 Geography of and travel in Europe vs 948 Scandinavia), art history (709 Arts–History, geographic treatment, biography vs 759 Painters–History, geographic treatment, biography) and healthcare and social services (362 Social problems of and services to groups of people vs 616 Diseases; 616 Diseases vs 618 Gynaecology, obstetrics, paediatrics, geriatrics; 362 Social problems of and services to groups of people vs 305 Groups of people).

For the full DDC classification task, the available DDC classes comprised various hierarchical levels, so it was possible to misclassify a record by assigning it either a too broad or a too specific class. There were 369 cases (3.90%) where the algorithm suggested a broader class than the original Libris class (103 cases by one digit; 123 cases by two digits; 75 cases by three digits and 68 cases by four to six digits) and 472 cases (4.99%) where the algorithm suggested a narrower class (112 cases by one digit; 132 cases by two digits; 108 cases by three digits and 120 cases by four to nine digits); in total, 8.90% of the records were classified with an almost correct class, but with an incorrect level of specificity.

The prediction errors that affect five or more records in the full DDC classification task are shown in Table 8. Compared to the three-digit DDC task, the errors are more diverse and not

Freq.	Predicted DDC class	Assigned libris DDC class
15	948.6 Southern Sweden (Götaland)	914.8 Scandinavia–geography
15	839.738 Swedish fiction–2000-	791.4372 Single films
12	759.85 Swedish painting	709.2 Art–Biography
9	839.738 Swedish fiction–2000-	839.72 Swedish drama
8	641.5 Cooking	641.59 Cooking characteristic of specific geographic environments, ethnic cooking
8	948.6 Southern Sweden (Götaland)	936.8 Scandinavia to 481
7	372.7 Mathematics–primary education	510.71 Mathematics–education
6	839.738 Swedish fiction–2000-	839.7374 Swedish fiction–1945–1999
6	948.6 Southern Sweden (Götaland)	948.7 Central Sweden (Svealand)
5	613.7 Physical fitness	613.71 Exercise and sports activities
5	709.2 Biography	759.85 Swedish painting
5	839.738 Swedish fiction–2000-	439.78 Swedish language–applied linguistics

Note(s): The class headings are shown here in English, although Swedish DDC was used in the experiment
Source(s): Authors' own creation

Table 8. Most common prediction errors in the full DDC prediction task

as often related to fiction. There are cases involving geography (948.6 Southern Sweden (Götaland) vs 914.8 Scandinavia–geography; 948.6 Southern Sweden (Götaland) vs 936.8 Scandinavia before 481; 948.6 Southern Sweden (Götaland) vs 948.7 Central Sweden (Svealand)), distinctions between fiction, drama and films (839.738 Swedish fiction–2000- vs 791.4372 Single films; 839.738 Swedish fiction–2000- vs 839.72 Swedish drama; 839.738 Swedish fiction–2000- vs 839.7374 Swedish fiction–1945–1999; 839.738 Swedish fiction–2000- vs 439.78 Swedish language–applied linguistics), and difficulties to distinguish between biographies in general and those of Swedish painters (709.2 Biography vs 759.85 Swedish painting). Mathematics–education (510.71) is confused with Mathematics–primary education (372.7). There are also cases where the predicted class is less specific than the class assigned in Libris: cooking in general (641.5) vs cooking in specific geographic environments (641.59) and physical fitness in general (613.7) vs exercise and sports (613.72).

Evaluation of the 60 manually reviewed records

After collecting evaluations of the 60 documents from four different evaluators as described above (five persons of which two senior librarians worked as a team), all the evaluations were merged into one Excel file to allow analysis. Values were replaced with numbers in order to make the calculations easier: "incorrect" was assigned 1, "partly correct" was assigned 2 and "correct" was assigned 3. Not all evaluations were filled:

(1) Out of 60 original Libris records, 6 did not have a DDC class assigned.

(2) Of the student evaluators, evaluations were missing for the total of 20 classes.

The first finding was that of the 54 existing, original DDC classes in Libris records, 7 were considered incorrect by the two senior librarians. The senior librarians also added the 6 missing classes. The final set comprising 47 original DDC classes, 7 corrected classes and 6 added classes was then used as the baseline against which the subsequent analysis was conducted.

Accuracy, or the ratio of the correct automatically assigned classes against the baseline of these 60 documents was found to be 33.33%, i.e. the classifier was able to find exactly the

same classes for 20 out of 60 documents; additional 8 classes differed in the last digit and another 5 in the last two digits.

Table 9 below shows to what degree automatically assigned classes were considered correct by the four sets of evaluators. On average across all the classes and all the evaluators, 26.61% are considered correct, 34.50% partly correct and 37.32% incorrect. Of the automated classes with top score, on the three-point scale, a slight majority are considered partly correct (36.82%), followed by correct (31.38%) and incorrect (27.62%). Of the second-ranked classes, the majority is also partly correct (43.51%); there are also a larger portion of incorrect classes (38.91%), and least correct ones (17.57%). Third-ranked classes have, as could be expected, the largest portion of incorrect classes (46.26%), but also more correct (30.84%) than partly correct classes (22.47%). In summary, about one-third of them are on average considered fully correct, similar to results reported in the above section for full DDC classes; combined with partly correct ones, some 60% are relevant, which is also similar to the results we gained for this algorithm on 3-digit DDC classes (Table 2). This could indicate that using Libris as the "gold standard" seems appropriate.

A more detailed insight into what went well and what went wrong in automated classification with DDC can be discerned from an in-depth scale expanding the 3 points (correct, partly correct, incorrect) into an 11-point scale implemented by the two senior librarians:

(1) Correct

(2) Correct, missing aspect

(3) Partly correct

Correct classes	Class with top score Number	Class with top score Total number of values filled by evaluator	Class with second to top score Number	Class with second to top score Total number of values filled by evaluator	Class with third to top score Number	Class with third to top score Total number of values filled by evaluator
Experts	25	60	7	60	5	60
Student A	20	60	14	60	28	54
Student B	13	60	13	60	31	54
Student C	17	59	8	59	6	59
Average	*18.75*	*59.75*	*10.50*	*59.75*	*17.5*	*56.75*
Average in %	*31.38%*		*17.57%*		*30.84%*	
Partly correct classes						
Experts	22		21		17	
Student A	22		32		14	
Student B	19		27		14	
Student C	25		24		6	
Average	*22*	*59.75*	*26.0*	*59.75*	*12.27*	*56.75*
Average in %	*36.82%*		*43.51%*		*22.47%*	
Incorrect classes						
Experts	13		32		38	
Student A	18		14		12	
Student B	18		20		9	
Student C	17		27		47	
Average	*16.5*	*59.75*	*23.25*	*59.75*	*26.25*	*56.75*
Average in %	*27.62%*		*38.91%*		*46.69%*	

Table 9. Evaluators' judgement on automatically assigned classes

Source(s): Authors' own creation

(4) Partly correct, missing aspect
(5) Partly correct, wrong discipline
(6) Partly correct, wrong aspect
(7) Partly correct, too wide
(8) Partly correct, too specific
(9) Incorrect
(10) Incorrect, too wide
(11) Incorrect, wrong discipline

We see that 28% of the top three automated classes are considered fully correct while 7% are correct but missing an aspect. Then 4% of classes are partly correct in general, while most others are: partly correct with wrong discipline (11%); partly correct due to a missing aspect (6%); and, a smaller proportion, partly correct with a wrongly identified aspect (3%). A few examples fall into other subcategories of partial correctness: 1% "partly, too wide" and 0% "partly, too specific". Finally, 26% of classes are generally incorrect, 10% have a wrongly recognised discipline and 3% are incorrect because they are too wide. In summary, using this scale indicates that the most common problems of automated classification with DDC are related to: (1) missing a key aspect when (partly) correct, (2) wrong aspect when partly correct, (3) wrong discipline when partly correct or incorrect as well as (4) too broad of a class when incorrect, since the specificity principle demands the assignment of most specific class available.

Going back to the three-point scale where correct values are represented as 3, partly correct as 2 and incorrect as 1, Table 10 below presents average evaluation values assigned across all top-ranked, second-ranked and third-ranked classes. Classes at all three ranks are closest to 2, i.e. partly correct on average, with top-ranked considered most correct on average (1.98), followed by second-ranked (1.78) and third-ranked classes (1.74).

However, what is also obvious from Tables 9 and 10 are considerable differences among evaluators' judgements. The agreement between evaluators across all automated classes was 62.42%. Considering the complexity of DDC classification involving many rules and requiring lots of practice, experts evaluations are to be most trusted. In Table 10 they also provide highest correct values to top-ranked DDC with quite large differences from top-ranked and lower-ranked ones. Only the experts and one student (C) follow the average pattern (most correct being top-ranked classes, less correct second-ranked classes and least correct third-ranked ones), while students A and B consider third-ranked classes most correct on average. Very big differences are also seen in Table 9. Here, while the number of correct classes by experts is 36, two of the students provided considered almost twice as many classes as fully correct (62 and 57), and one student considered fewer classes to be correct (31).

Average correctness	Automated DDC classes Top-ranked DDC	Second-ranked DDC	Third-ranked DDC
Experts	2.2	1.58	1.45
Student A	2	2	2.07
Student B	1.75	1.88	2.17
Student C	1.97	1.65	1.28
Average	*1.99*	*1.78*	*1.74*

Source(s): Authors' own creation

Table 10. Average correctness of automatically assigned classes

Similar gaps are seen for incorrect classes: experts 84, two students much less (47 and 57) and one student 90 which is more similar to experts.

Of the automatically assigned classes that were *not assigned* in the baseline, 11.11% were considered fully correct by team of experts across 18 out of 60 records; 34.48% classes were considered correct by students A across 30 records; 31.60% were considered correct by student B across 33 records; and 16.95% were considered correct by student C across 23 records. More classes were considered partly correct: 60 (33.33%) were considered partly correct by the team of experts across 41 records; 68 (39.08%) by student A across 46 records; 59 (33.90%) by student B across 44 records; and 54 (30.50%) by student C across 35 records.

The students' optional comments were not many. One student wrote that information on one item was found in Libris only, about another that it was hard to judge content without having the book at hand as metadata were insufficient, in two others cases that it was also hard to judge, and on seven classes gave suggestions for another class. Student C provided more comments for their choices; below are examples for which they provided comments and belong to the set of 23 metadata records for which the student considered at least one of the automated classes fully correct while the automated class was different from the baseline class for that record. Librarians' comments to the student's comment illustrate challenges of aboutness and subject indexing using a large classification system such as DDC that demands extensive training to be applied according to the professional standards. Both the librarians' and student's comments also demonstrate the value of some automatically classes not being originally assigned.

Example 1

(1) Title: Ersättningen och e-hälsan (Compensation for digital health services)

(2) ISBN: 9789188637130

(3) Baseline DDC: 338.47362109485 "Medical economics–Sweden"

(4) Automated classes considered correct by the student:

- Automated DDC 1: 338.433621 "Medical economics"
- Automated DDC 2: 362.1028 "Health care–techniques"
- Automated DDC 3: 362.10681 "Health services–financial management"

(5) Student's comment: All the classes feel correct. They are different classes but I can't point out which one is more right.

(6) Librarians' comment: 362 is not a correct discipline. The book deals with health budgets–national economics, not the budget of a specific health institution.

Example 2

(1) Title: Sjostakovitj förändrade mitt liv (Shostakovich changed my life)

(2) ISBN: 9789188316936

(3) Baseline DDC: 780.92 "Music–biography"

(4) Automated classes considered correct by the student:

- Automated DDC 1: 615.85154 "Music therapy–medicine"
- Automated DDC 2: 781.11 "Music–psychological principles"

(5) Student's comment: DDC1 and DDC2 are both (probably) correct, DDC1 is music-therapy and DDC2 is the psychological principles of music.

(6) Librarians' comment: According to DDC rules, the main topic of the book was Shostakovich, not music therapy. If music therapy had been the main topic DDC 615.85154 would have been better.

Example 3

(1) Title: Dansa mjukt med tillvaron: om mening, mod och möjligheter (Dance lightly through life: about meaning, courage and possibilities)

(2) ISBN: 9789127826717

(3) Baseline DDC: 158.1: "Personal improvement–applied psychology"

- Automated DDC 1: 158.1 "Personal improvement–applied psychology"
- Automated DDC 3: 362.2 "People with mental disabilities–social welfare"

(4) Student's comment: DDC1 and DDC3 are both correct. Another possible class could be 158.13 (personal development through mindfulness). Mindfulness is not explicitly mentioned but it seems to capture the theme.

(5) Librarians' comment: 158.13 is quite close, but a bit too specific. People under treatment for a mental illness should be placed in 362.2

Example 4

(1) Title: Ät dig frisk: revolutionerande forskning och enkla kostråd för ett längre och friskare liv (Eat healthy: revolutionary research and simple dietary advice for a longer and healthier life)

(2) ISBN: 9789188859471

(3) Baseline DDC: 613.2 "Dietics"

- Automated DDC1: 613.2 "Dietics"

(4) Student's comment: DDC1 is correct, there doesn't seem to be a more specific but still accurate class unfortunately.

(5) Librarians' comment: We agree.

Example 5

(1) Title: "Kom igång med vetenskap" (Start working with science)

(2) ISBN: 9789198525960

(3) Baseline DDC: 500 "Natural sciences"

- Automated DDC 1: 500 "Natural sciences"

(4) Student's comment: DDC1 is correct, but could be more specific maybe.

(5) Librarians' comment: Non-fiction about general science for children. It is not specific.

Example 6

(1) Title: "Nordiska gudinnor: vardagsmagi för dagens kvinnor" (Germanic goddesses: everyday magic for women of today)

(2) ISBN: 9789198070637

(3) Baseline DDC: 133.43082 "Magic and witchcraft–women"

- Automated DDC1: 293 "Germanic religion"

(4) Student's comment: DDC1 feels most correct (Nordic ancient religion). There might be a way to include the "self-help"-aspect as well.

(5) Librarians' comment: The topic is Germanic goddesses, but about non-religious magic and women.

Conclusion

In order to address the problems of scale and to sustain established bibliographic objectives, semi-automated solutions to subject classification and indexing are necessary. This is the case for library catalogues with established cataloguing and indexing resources and even more for those emerging more recently such as repositories of academic publications and other digital collections which more or less tend to rely on full-text indexing and thus effectively prevent successful subject searching.

This paper aimed to identify the potential of applying automated subject classification on Swedish union catalogue using 1.9 million catalogue records. Compared to previous work on automated DDC classification of Libris records (Golub, 2021), we had access to over 60% more Swedish language records with DDC classes, following the natural increase of DDC records to the Swedish Union Catalogue between April 2018 and March 2021. The previous research concentrated on three-digit DDC classification as well as a smaller set of 29 major classes for which more than 1,000 training examples could be found. Both of these set-ups are too coarse-grained to be of practical value in library services; this is why in our experiments we used both the full DDC as well as three-digit DDC – the latter to increase the number of available training documents per class as well as to compare to earlier research.

In the previous work (ibid.) different algorithms had been individually tested including Naïve Bayes, SVM classification (SVC), several word embedding methods and a lexical (string matching) method; the SVC method performed best, achieving an accuracy of 66.13% on records with subjects in the three-digit DDC classification task. In this work, we tested SVC as well as two other machine learning algorithms (fastText and Omikuji) with roughly similar levels of performance to the previous work, as well as a lexical method. In addition, we combined the five individual algorithms into a weighted ensemble that achieved 66.82% accuracy on the three-digit DDC classification task on records with subjects, an improvement of almost 0.69% over the previous work (ibid.). This improvement can be partly attributed to over 60% more training data, but it is also clear from the results that the ensemble approach brings an improvement over individual classification algorithms. The ablation analysis showed that the lexical method, which overall performs poorly on its own, can still improve overall classification accuracy when combined with other algorithms. Specific in-depth analysis of when individual algorithms perform best would help improve their individual and combined performance further; this is subject to future research.

In the analysis of common classification errors, it was apparent that DDC classes for fiction are difficult to classify accurately and in the case of full DDC, the algorithms have difficulty distinguishing between fiction from different time periods. In hindsight, the performance results could be explained by the fact that the algorithms were only given the title and subjects available in the Libris records. For non-Swedish literature, those records represent the Swedish language translations of the original works, so the title is also given in Swedish. Even for a human indexer, classifying a work of fiction correctly based on its Swedish language title (and possibly subjects) alone is a near-impossible task which is why indexers must have the actual document at hand to inspect the title page, table of contents,

preface etc. More information about its origin is necessary, such as the language of the original work, the place where the original work was published or the country the author is based in. Likewise, more information such as the year the work was originally published or the date of birth of its author would be necessary to distinguish between 20th and 21st century fiction. Solving these challenges with classifying fiction could improve the overall accuracy. Furthermore, it should also be possible to find out more about the author based on authority data that now link from Libris to Wikidata.

In addition to exploring the above mentioned approaches, our future plans include further automatic improvement and testing. Since DDC is highly hierarchical, its built-in relationships could help with disambiguation, especially in lexical algorithms. However, Annif backends do not currently support hierarchical classification so this should be experimented in the future.

Looking more into the aspects of aboutness, the study of 60 records with 3 automatically assigned classes to each confirmed earlier studies in rather low inter-rater agreement. To start with, 3.78% of original DDC classes were deemed incorrect by the team of 2 senior classification experts. Of the automatically assigned classes that were not assigned in the baseline, the experts considered 20 (11.11%) fully correct across 18 out of 60 records and 60 (33.33%) partly correct across 41 records. The students were more forgiving and thought even more classes were (partly) correct. Adding additional automatically derived classes may potentially lead to improved information retrieval as it increases the number of search access points.

While evaluation approaches often assume that human indexing is best, and that the task of automated indexing is to meet the standards of human indexers, more serious scholarship needs to be devoted to evaluation in order to further our understanding of the value of automated subject assignment tools and to enable us to provide a fully informed input for their development and enhancement. Thus, in addition to further algorithm set ups and tests, future research should include testing Libris records for accuracy and inter-indexer consistency to further understand implications of using it as a "gold standard" in evaluation.

Also, while the potential of semi-automated solutions to many digital collections (e.g. web archives, new digitised cultural heritage collections) is there to help with the scale and sustainability as well as ensure interoperability with Libris; however, in order to establish the value of tools like Annif in operative information systems, it would be important to test it during DDC classification workflows and conduct evaluation to determine the value of automated suggestion as part of the operational workflow (see Golub *et al.*, 2016). Technical aspects such as RAM usage would be important to also test here. Finally, the value of automated classes should also be studied in the context of subject searching and browsing, as finding relevant documents is raison d'etre of subject indexing and classification.

References

Anderson, J.D. and Pérez-Carballo, J. (2001), "The nature of indexing: how humans and machines analyze messages and texts for retrieval. Part I: research, and the nature of human indexing", *Information Processing and Management*, Vol. 37 No. 2, pp. 231-254, doi: 10.1016/s0306-4573(00)00026-1.

Bergstra, J., Yamins, D. and Cox, D. (2013), "Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures", *International Conference on Machine Learning*, PMLR, pp. 115-123.

Brattli, T. (2012), "Why build Dewey numbers? The remediation of the Dewey decimal classification system", *Nordlit*, Vol. 16 No. 2, 189, doi: 10.7557/13.2383.

Chang, W.C., Yu, H.F., Zhong, K., Yang, Y. and Dhillon, I.S. (2020), "Taming pretrained transformers for extreme multi-label text classification", *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3163-3171.

Conradi, E. (2017), "DDC and automatic classification", available at: https://edug.pansoft.de/tiki-index.php?page=DDC+and+automatic+classification

Finch, L. (2014), "Automated indexing - a case study from the national agricultural library", *Webinar Presented on April 10, 2014*, available at: https://www.nal.usda.gov/automated-indexing-case-study-national-agricultural-library

Ghiassi, M., Olschimke, M., Moon, B. and Arnaudo, P. (2012), "Automated text classification using a dynamic artificial neural network model", *Expert Systems with Applications*, Vol. 39 No. 12, pp. 10967-10976, doi: 10.1016/j.eswa.2012.03.027.

Golub, K. (2016), "Potential and challenges of subject access in libraries today on the example of Swedish libraries", *International Information and Library Review*, Vol. 48 No. 3, pp. 204-210, doi: 10.1080/10572317.2016.1205406.

Golub, K. (2018), "Subject access in Swedish discovery services", *Knowledge Organization*, Vol. 45 No. 4, pp. 297-309.

Golub, K. (2021), "Automated subject indexing: an overview", *Cataloging and Classification Quarterly*, Vol. 59 No. 8, pp. 1-18, doi: 10.1080/01639374.2021.2012311.

Golub, K., Soergel, D., Buchanan, G., Tudhope, D., Lykke, M. and Hiom, D. (2016), "A framework for evaluating automatic indexing or classification in the context of retrieval", *Journal of the Association for Information Science and Technology*, Vol. 67 No. 1, pp. 3-16, doi: 10.1002/asi.23600.

Hakopov, Z., Mironov, D., Savic, D. and Svetashova, Y. (2018), "Automated KOS-based subject indexing in INIS", *Journal Article*, Vol. 10 No. 10.75, pp. 1-21, available at: http://ceur-ws.org/Vol-2200/paper2.pdf

International Organization for Standardization (1985), *Documentation – Methods for Examining Documents, Determining Their Subjects, and Selecting Index Terms: ISO 5963*, International Organization for Standardization, Geneva.

Järvelin, K. and Kekäläinen, J. (2002), "Cumulated gain-based evaluation of IR techniques", *ACM Transactions on Information Systems (TOIS)*, Vol. 20 No. 4, pp. 422-446, doi: 10.1145/582415.582418.

Jenkins, C., Jackson, M., Burden, P. and Wallis, J. (1998), "Automatic classification of Web resources using Java and Dewey decimal classification", *Computer Networks and ISDN Systems*, Vol. 30 Nos 1-7, pp. 646-648, doi: 10.1016/s0169-7552(98)00035-x.

Joorabchi, A. and Mahdi, A.E. (2014), "Towards linking libraries and Wikipedia: automatic subject indexing of library records with Wikipedia concepts", *Journal of Information Science*, Vol. 40 No. 2, pp. 211-221, doi: 10.1177/0165551513514932.

Jonasen, T.S. and Lykke, M. (2013), "The role of automated categorization in e-government information retrieval", *Proceedings of the ISKO UK 3rd Biennial Conference*, ISKO.

Joulin, A., Grave, E., Bojanowski, P. and Mikolov, T. (2016), "Bag of tricks for efficient text classification", *arXiv Preprint arXiv:1607.01759*, available at: http://arxiv.org/abs/1607.01759

Junger, U. (2017), "Automation first–the subject cataloguing policy of the Deutsche National bibliothek", *Paper Presented at: IFLA WLIC 2018 – Kuala Lumpur, Malaysia – Transform Libraries, Transform Societies in Session 115 – Subject Analysis and Access*, available at: http://library.ifla.org/id/eprint/2213

Kasprzik, A. (2020), "Putting research-based machine learning solutions for subject indexing into practice", In Paschke, A. *et al.* (Eds), *Proceedings of the Conference on Digital Curation Technologies (Qurator 2020)* Berlin, RWTH, Aachen, available at: http://nbn-resolving.de/urn:nbn:de:0074-2535-7 (accessed 20–21 January 2020).

Khandagale, S., Xiao, H. and Babbar, R. (2020), "Bonsai: diverse and shallow trees for extreme multi-label classification", *Machine Learning*, Vol. 109 No. 11, pp. 2099-2119, doi: 10.1007/s10994-020-05888-2.

Khoo, M.J., Ahn, J.W., Binding, C., Jones, H.J., Lin, X., Massam, D. and Tudhope, D. (2015), "Augmenting Dublin core digital library metadata with Dewey decimal classification", *Journal of Documentation*, Vol. 71 No. 5, pp. 976-998, doi: 10.1108/jd-07-2014-0103.

Lancaster, F.W. (2003), *Indexing and Abstracting in Theory and Practice*, 3rd ed., Facet, London.

Lee, L.H., Wan, C.H., Rajkumar, R. and Isa, D. (2012), "An enhanced support vector machine classification framework by using Euclidean distance function for text document categorization", *Applied Intelligence*, Vol. 37 No. 1, pp. 80-99, doi: 10.1007/s10489-011-0314-z.

Mork, J., Aronson, A. and Demner-Fushman, D. (2017), "12 years on – is the NLM medical text indexer still useful and relevant?", *Journal of Biomedical Semantics*, Vol. 8 No. 8, doi: 10.1186/s13326-017-0113-5.

OCLC (2004), "Scorpion", *OCLC Software*, available at: http://www.oclc.org/research/software/scorpion/default.htm

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. and Vanderplas, J. (2011), "Scikit-learn: machine learning in Python", *The Journal of Machine Learning Research*, Vol. 12, pp. 2825-2830.

Prabhu, Y., Kag, A., Harsola, S., Agrawal, R. and Varma, M. (2018), "Parabel: partitioned label trees for extreme classification with application to dynamic search advertising", *Proceedings of the 2018 World Wide Web Conference*, pp. 993-1002.

Roitblat, H.L., Kershaw, A. and Oot, P. (2010), "Document categorization in legal electronic discovery: computer classification vs manual review", *Journal of the American Society for Information Science and Technology*, Vol. 61 No. 1, pp. 70-80, doi: 10.1002/asi.21233.

Silvester, J.P. (1997), "Computer supported indexing: a history and evaluation of NASA's MAI system", *Encyclopedia of Library and Information Services*, Vol. 61 Supplement 24, pp. 76-90.

Suominen, O. (2019), "Annif: DIY automated subject indexing using multiple algorithms", *LIBER Quarterly*, Vol. 29 No. 1, pp. 1-25, doi: 10.18352/lq.10285.

Suominen, O., Inkinen, J. and Lehtinen, M. (2022), "Annif and Finto AI: developing and implementing automated subject indexing", *JLIS. It*, Vol. 13 No. 1, pp. 265-282, doi: 10.4403/jlis.it-12740.

Toepfer, M. and Seifert, C. (2020), "Fusion architectures for automatic subject indexing under concept drift", *International Journal on Digital Libraries*, Vol. 21 No. 2, pp. 169-189, doi: 10.1007/s00799-018-0240-3.

U.S. National Library of Medicine (2019), *NLM Medical Text Indexer (MTI)*, available at: https://ii.nlm.nih.gov/MTI/

Wiesenmüller, H. (2017), "Das neue Sacherschließungskonzept der DNB in der FAZ", available at: https://www.basiswissen-rda.de/neues-sacherschliessungskonzept-faz/ (accessed 2 August 2017).

You, R., Zhang, Z., Wang, Z., Dai, S., Mamitsuka, H. and Zhu, S. (2019), "AttentionXML: label tree-based attention-aware deep model for high-performance extreme multi-label text classification", *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 5820-5830.

Corresponding author
Koraljka Golub can be contacted at: koraljka.golub@lnu.se