

Classifying Medical Literature Using k-Nearest-Neighbours Algorithm

Andreas Lüscho^{*} and Christian Wartena (ORCID: 0000-0001-5483-1529)

University of Applied Sciences and Arts Hanover
Expo Plaza 12, 30539 Hannover, Germany
`christian.wartena@hs-hannover.de`

Abstract. The amount of papers published yearly increases since decades. Libraries need to make these resources accessible and available with classification being an important aspect and part of this process. This paper analyzes prerequisites and possibilities of automatic classification of medical literature. We explain the selection, preprocessing and analysis of data consisting of catalogue datasets from the library of the Hanover Medical School, Lower Saxony, Germany. In the present study, 19,348 documents, represented by notations of library classification systems such as e.g. the Dewey Decimal Classification (DDC), were classified into 514 different classes from the National Library of Medicine (NLM) classification system. The algorithm used was k-nearest-neighbours (kNN). A correct classification rate of 55.7% could be achieved. To the best of our knowledge, this is not only the first research conducted towards the use of the NLM classification in automatic classification but also the first approach that exclusively considers already assigned notations from other classification systems for this purpose.

1 Introduction

To find and eventually use documents, it is necessary to make them accessible through metadata. The result of this process can be found e.g. in the catalogues of scientific libraries. Two major fields are distinguished: descriptive cataloguing and subject cataloguing. While descriptive cataloguing uses formal aspects of the documents to describe a resource (e.g. title, author, publisher, ISBN), subject cataloguing allows topical attributions with the help of classification systems or keywords. By this, documents can be described both in form and content and thus made available for library users. Over the years, many different types of classification systems were developed, often specialised in a specific topic, adjusted for specific users or regions or with a certain degree of differentiation. Typically, these systems have tens of thousands of classes, e.g., there are ca. 38,000 classes in the Dewey Decimal Classification (DDC) and about 860,000 classes in the German Regensburger Verbundklassifikation (RVK) [15].

Due to the continuously increasing amount of publications since the widespread usage of digitization, libraries can hardly afford to manually and intellectually

^{*} This paper is based on the bachelor's thesis of Andreas Lüscho [9].

assign all these resources with adequate indexing terms and topical attributions. Especially the amount of documents existing in an online environment exceeds the capacity of library staff or catalogers, respectively. Therefore, many of these resources are solely described by the metadata their publishers provide. Libraries simply transfer and adopt this information to use it in their online catalogues. However, if classes from different classification systems are mixed up, it will become hard to systematically search for all works in a certain category. Thus, the ultimate goal is still to classify all works according to one single system [2].

The research presented in this paper is based on the assumption that the class numbers of different classification systems can be attached to each other, i.e. that there is a correlation between several systems. This further leads to the assumption that class numbers of a particular classification can be determined by analyzing existing class numbers from other available classifications with the help of machine learning algorithms. Through this process, missing metadata for single library records could be identified and added to these datasets. This can e.g. lead to the development of software that is able to support library staff in enriching catalogue data when a desired classification system is not part of the available metadata but other assignments are available for analysis. Hereby, the homogeneity of the datasets could be improved and additional metadata for retrieval purposes could be generated.

Our approach differs from other approaches (see e.g. [15]) that also use classes from other classification systems for prediction of a certain classification, but that use mappings between systems to translate the classes. We expect that a direct use of the classes has more potential than the indirect use via a static mapping. Moreover, when using the classes of classification systems as features, we can easily use several classes from several classification systems.

2 Previous Research

So far, no consolidated summaries that thoroughly treat and recapitulate existing research on the automatic classification of library stocks exist. Most of the literature explores the classification of electronic resp. online documents. These usually allow the usage of full texts or at least abstracts. Automatic classification based on classical metadata such as title, author, keywords or classification numbers is an exception, though it is usually the only available information libraries have on their works.

The often cited and comprehensive article “Machine Learning in Automated Text Categorization” by Sebastiani can be seen as the most important source for the introduction to data mining methods for automatic classification tasks [14]. Beyond that, hardly any literature dealing with the automatic classification of books and no significant studies or implementations concerning this matter in the library sector existed until the year 2005, as Oberhauser pointed out [12]. In contrast to this, a wide number of research investigating the automatic assignment of keywords or descriptors from a thesaurus (i.e. the automatic indexing) can be found, also in the medical sector [3, 7, 8].

Solely the research from Larson has to be emphasized, which looked at the automatic assignment of classes from the Library of Congress Classification (LCC) to bibliographic datasets [6]. The research was based on ca. 30,000 MARC datasets from the holdings of the University of California Berkeley Library School Library. Each record had an attributive class from the LCC, in total, 5,765 different classes existed. Larson used different combinations of attributes extracted from the metadata and a nearest centroid classifier, i.e. a classifier that compares the document that has to be classified with the most typical document of each class. The best combination of attributes and attribute weights resulted in a correct classification rate of 46.6%. Considering the ten best fitting clusters, a recall of 74.4% could be achieved. Larson stated that automatic classification is not possible but semi-automatic approaches lead to satisfying results.

Cheng used the main title and chapter titles from a small collection of books and achieved a correct classification rate of 85–90% into classes from the DDC [1]. Ishida used the Nippon Decimal Classification (NDC), which is based on the DDC. He classified 1,000 books using different extraction and weighting methods into the first 1,000 sections (i.e. classes) of the NDC. A correct classification rate of 55.9% was achieved [4]. Pong et al. analyzed problems that occur while processing bibliographic data in general and when trying to automatically generate class numbers from the LCC in particular. The authors compared the k-nearest-neighbours (kNN) and Naive Bayes algorithm and presented a self-developed automatic document classification system called WADCS. To improve the classification performance, they used a preprocessed and edited version of the LCC. They concluded that kNN is more suitable to support the classification process than Naive Bayes [13]. Wang investigated the automatic assignment of DDC class numbers by using supervised machine learning methods. After a thorough analysis of the distribution of the training documents within the DDC, a new structure for the DDC was designed to reduce intrinsic problems of this classification system. A semi-automatic system was proposed to achieve an acceptable quality of correct classifications. Using a maximum of three user interactions, a correct classification rate of 90% could be achieved with this system [16]. Joorabchi examined links between documents in terms of citations and thereupon tried to assign electronic documents to DDC classes [5].

Publications that treat the automatic assignment of documents to classes from the classification of the National Library of Medicine (NLM) were not found during research for this paper. Rather, the general impression was confirmed: many publications concerning the automatic indexing, i.e. the assignment of keywords from the Medical Subject Headings (MeSH) to documents, exist (see e.g. [11] and the NLM Medical Text Indexer (MTI) as described in [10]), but no works that explicitly deal with the automatic classification of medical literature to classes from the NLM classification system.

Summing up, diverse approaches to study automatic classification in a library context can be found in literature. But they often are not comparable since they use e.g. different classification systems or methods. Even if the prerequisites are similar, most researches differ in data structure, data processing and documenta-

tion. Another indication that not much relevant literature in this field of research exists is that many of the above mentioned authors describe their work as unique to date.

3 Methods and Dataset Analysis

The experiments presented in this paper differ from most previous experiments since the automatic classification is based on already assigned classes from other classification systems instead of using book titles or keywords as the content representation for each document. Furthermore, the classification system whose classes will be predicted is the NLM classification.

The library of the Hanover Medical School (*Medizinische Hochschule Hannover*, MHH) arranges its stock by using the classification of the NLM. Therefore, most of its books are provided with a so called *local notation* (taken from that classification) that indicates affiliation to a certain topic or science. For example, a resource with the local notation WB 300 belongs to the class “General Therapeutics” and therefore stands in a row with all other resources from that class. In July 2016, all data base entries from the library catalogue containing such a local notation were exported in comma-separated values format (CSV). This file contained 45,350 datasets resp. records.

3.1 Distribution of the Class Numbers

A first data analysis showed a wide spreading of the single datasets across the main classes from the NLM classification. The medical classes QS to QZ and W to WZ comprised 34,705 datasets (76.5%). In total, 4,768 different class numbers existed throughout the data, whereof 2,368 (49.7%) were located in the medical sciences and 2,400 (50.3%) in other sciences. Since our experiments focused on medical literature, the data were reduced to the appropriate classes only. These remaining classes were also distributed unequally, with 656 classes holding only one document, 323 holding two documents and 195 holding three documents – hence resulting in 1,174 classes (49.6%) with only three or less assigned documents. On top of that, the 24 largest classes (representing 1% of the total number of classes) held 7,774 documents, which corresponded to 22.4% of all documents. Thus, the data were highly characterized by a skewed distribution of its classes (see Fig. 1). This is a common characteristic in real-world applications [16].

The skewed distribution is a problem for automatic classification. In classes with only a few available documents it is hard (or even impossible) for an algorithm to analyze and learn the characteristics of this class, especially when the differences between two classes are marginal, which often is the case in library classification systems.

The examination of the largest and smallest classes of the data lead to further insights into the data structure: The largest classes mostly were superordinate main classes, only a few specific classes were assigned notably often. So most of

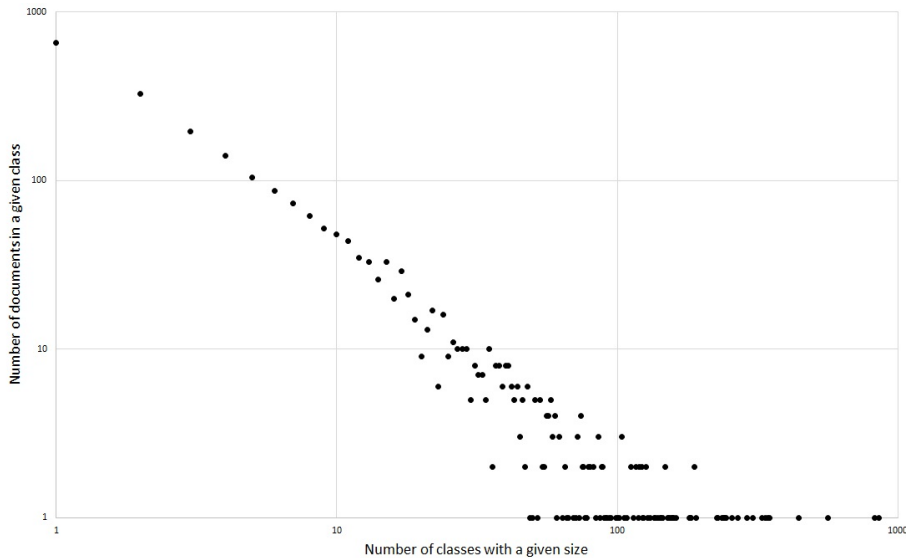


Fig. 1. Frequency distribution of classes

the examined documents were series or journals, since the MHH library assigns these to main classes from the NLM classification, or general, often unspecific resources like textbooks and synopses.

3.2 Attribute Selection

After exporting the datasets from the library catalogue, all attributes except for the assigned classifications were removed. The following classification systems were used in the data from the MHH library:

- Dewey Decimal Classification (DDC)
- Classification of the National Library of Medicine (NLM)
- Regensburger Verbundklassifikation (RVK)
- Classification of the Library of Congress (LCC)
- Basisklassifikation (BK)

Nearly all records showed assignments to more than one classification system, and in many cases, even within a single system more than one notation existed. This trait needed to be taken into account during the following steps. Plus, several datasets showed deficient entries (e.g. unnecessary blanks, spelling mistakes, commutation of classifications) which needed to be cleaned before the data mining process.

Not every record was assigned to each classification system. Figure 2 illustrates in how many of the datasets each classification was represented. A total of 4,695 datasets (13.5%) that had no assignment to one of the examined classification

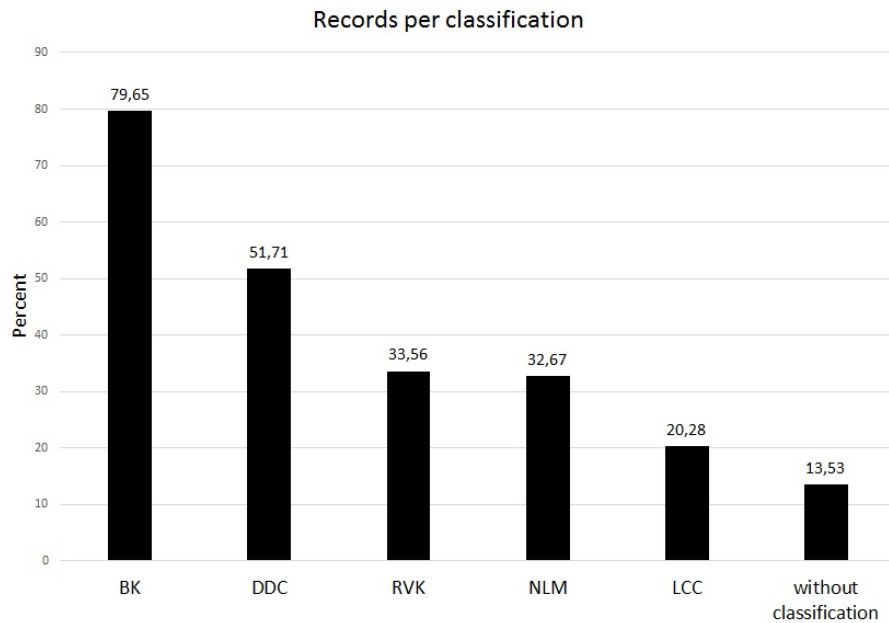


Fig. 2. Records per classification system

systems were removed from the data because they can not offer any information that can be used by a classifier. Thus, 30,010 records remained.

3.3 Data Preprocessing

To eliminate the above mentioned impurity of the data, every single classification was adjusted and the data were corrected as much as possible. This consisted e.g. in removing multiple entries within a single classification (only leaving the first mentioned notation in the dataset) or in trimming and normalizing the data. However, the information for the Basisklassifikation (BK) was treated differently: Here, all notations were left inside the dataset, only being trimmed to their class number (e.g. “44.01\$;Geschichte der Medizin [History of Medicine]” became “44.01”). This enabled to include all the BK classes assigned to a certain dataset by transforming this attribute to a vector in a later process step.

After this data cleansing, 29,946 datasets that were assigned a local notation and at least one notation from another classification system were left over. The percentage of classes with a maximum of three documents changed from 49.6 % to 45.6 %, with holding 5.4 % of all documents before the preprocessing and 4.9 % after it (see Table 1). The ten most assigned class numbers were still the superordinate main classes. To receive meaningful results in the experiments, the data were reduced once again: All records with one of these main classes as their local notation were removed, which affected 6,475 (21.6 %) of the 29,946

records. Thus, a considerable part of the data were excluded from the automatic classification process. This is justifiable though because it does not lead to a discrepancy with practical needs of subject cataloguing: The main classes are mainly used for series, journals (not their individual articles) and handbooks. Series and journals usually do not need to be documented as often as other publications (e.g. books or articles). Moreover, they usually can be assigned easier to a topical area so that support from a computer is not essential in classifying these resources.

After this first preprocessing, the data were nevertheless still sparse and the classes distributed unevenly: A small number of classes held a majority of the documents and most of the classes were assigned to only a few documents. To reduce the data once more, all datasets from classes that contained less than a total of ten documents were removed. This applied to 4,143 datasets. Most of these records can be accurately assigned to other, frequently used class numbers and as mentioned above, the characteristics of uncommon classes can hardly be learned. So this removal should not have a large impact on the experiments' results.

In addition, three attributes were added to the data: The class attribute "LNfull" (which represents the "correct" classification of a single dataset) was initially truncated to the first four digits ("LN1-4"), then to the first three digits ("LN1-3") and finally to the first two digits ("LNmain"). Since the NLM classification system uses medical notations that consist of two letters and one up to three numbers, the latter editing resulted in an attribute that represents the main class (e.g. *WB* if the full notation was *WB 300*) of the notation. Thus, we established the possibility to evaluate the accuracy of our machine learning algorithm on these four different hierarchies.

After the data preprocessing, 19,348 datasets remained which were assigned to a total number of 514 different classes from the NLM classification system. The sparseness of the data was reduced significantly, only 8.5 % of the datasets were located in the first 1 % of the largest classes (see Table 1), which still is a rather high proportion.

Table 1. Data sparseness before, during, and after data cleansing (with full local annotation)

	Before	During	After
Classes with 1 document	656	458	
Classes with 2 documents	323	254	
Classes with 3 documents	195	171	
Classes with max. 3 documents	1,174 (49.6 %)	883 (45.6 %)	
Classes (total)	2,368	1,935	514
Documents in classes with max. 3 documents	1,887 (5.4 %)	1,479 (4.9 %)	
Documents in the 1 % largest classes	7,774 (22.4 %)	6,290 (21.0 %)	1,646 (8.5 %)
Documents (total)	34,705	29,946	19,348

3.4 Data Transformation

Technically speaking, all attributes are nominal attributes. The k-nearest-neighbours-classifier from WEKA, a widely-used machine learning tool, handles nominal attributes. Thus, there is no need to change the data format, except for the classes from the BK. For the BK we often found combinations of several classes, so we changed the BK attribute to a vector of binary attributes, one for each possible class.

3.5 Automatic Classification

Classification Method. For classification, we used the k-nearest-neighbours (kNN) algorithm, an instance-based learning method. This method is characterized by solely memorizing all available training examples during the training phase. During the test phase, the documents to-be-classified are compared to these examples based on a beforehand defined distance measure. The most similar document is called the “nearest neighbour” but it is also possible to include the k nearest neighbours into the calculation. By this, the influence of potential runaway values can be limited, although considering more than one neighbour does not lead inevitably to more precise classifications. In our experiments, we set $k = 1$. kNN was chosen because of its simplicity and comprehensibility which allows first insights into our classification approach without having to deal with complex or advanced algorithms.

Several distance measures to define the similarity of two documents exist. We used the Euclidean distance that calculates the square root of the sum of all squares of the attributes’ value differences:

$$\sqrt{(a_1^{(1)} - a_1^{(2)})^2 + (a_2^{(1)} - a_2^{(2)})^2 + \dots + (a_y^{(1)} - a_y^{(2)})^2} . \quad (1)$$

The first document has the attribute values $a_1^{(1)}, a_2^{(1)} \dots a_y^{(1)}$ (with $y =$ total number of attributes), hence the second document has the values $a_1^{(2)}, a_2^{(2)} \dots a_y^{(2)}$. For nominal attributes, WEKA defines the distance by setting:

$$a^{(1)} - a^{(2)} = \begin{cases} 0 & \text{if } a^{(1)} = a^{(2)} \\ 1 & \text{if } a^{(1)} \neq a^{(2)} \end{cases} . \quad (2)$$

Evaluation. For evaluation we used ten-fold cross validation: The data were automatically divided into ten equally-sized parts and the training and testing process was conducted ten times, with each of these parts being the test data once and the remaining parts being used for training. After training and evaluation, WEKA gives the most likely class for each test record. We also evaluated the recall when modifying the output to giving the n most likely classes for each test record. By this means, it is possible to see if the correct classification, i.e. the correct notation from the NLM, can be found among these n classes.

4 Results

The attained results are shown in Table 2. More than half of the documents, 55.7%, were classified correctly when the full notation (“LNfull”) was the target class of the automatic classification process. The other hierarchical levels, i.e. the truncated notations as described above under “Data Preprocessing”, allowed correct classification rates of 58.0%, 66.0% resp. 81.4%, meaning that in these cases, the notation as determined by the algorithm conformed to the notation that was manually assigned to the datasets by the MHH library.¹

Table 2. Recall for ten-fold cross validation. Note that the recall @1 is the correct classification rate, since we always have exactly one true positive value.

n	Recall			
	LNfull	LN1-4	LN1-3	LNmain
1	55.7	58.0	66.0	81.4
2	64.9	68.1	76.9	88.8
3	69.1	72.4	81.4	90.7
4	71.5	75.0	83.6	91.6
5	73.3	76.6	85.3	92.3
8	77.0	79.8	87.0	93.4
10	78.0	81.0	87.7	94.6

Table 2 also illustrates the results that were achieved by looking at the n most likely classes as determined by the algorithm. This led to a correct classification rate of e.g. 73.3% when considering the five most likely classes of a test dataset and using “LNfull” as target class. As can be seen, up to 94.6% of correct classifications were possible by using different combinations of target classes and the number of most probable classes.

Figure 3 shows the improvement that can be achieved when taking n most similar classes into consideration. The improvement first is considerable but flattens with an increasing value for n . Nevertheless, the curve shows that classification accuracy can be enhanced highly by embracing adjacent classes from the training datasets. These might not have the highest similarity (as calculated by the algorithm) but are still appropriate enough to represent the test dataset.

Finally, Table 3 illustrates the correct classification rates that can be achieved when using only one of the available classification systems or when using all systems except for the NLM, respectively. Obviously, the combination of all attributes, as we did in our experiments, improves the correct classification rate considerably. None of the single classification systems reaches results nearly as

¹ The bachelor’s thesis on which this paper is based comes to slightly better results. This is due to a different form of data preprocessing that left more impure data in the datasets.

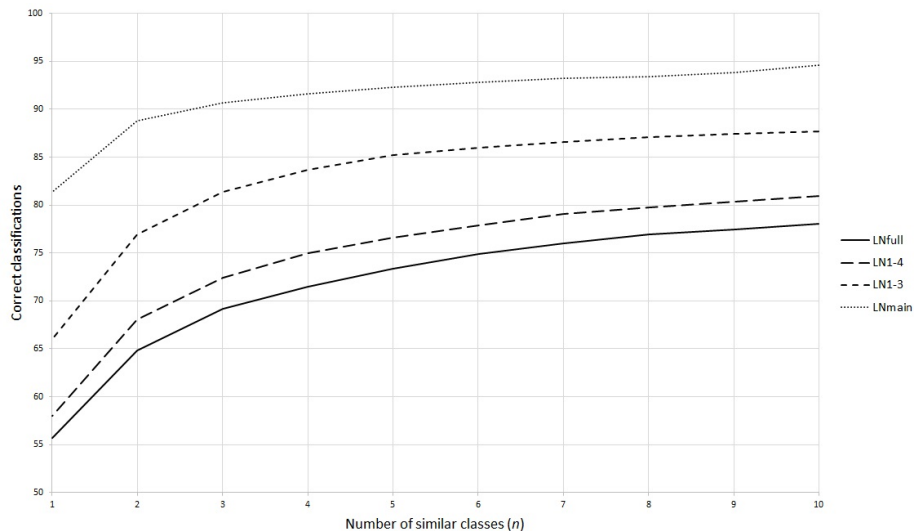


Fig. 3. Results when evaluating the n most similar classes

good as the combination of them. Moreover, this table shows that the NLM and the BK attribute seem to be the most important ones with the highest impact on the classification algorithm. Regarding the NLM, it is no wonder that the appearance of this classification system in the training data leads to good results, since this is exactly the same system whose classes the algorithm has to predict. What is all the more surprising is that the quite roughly classifying BK seems to have an even stronger impact on the algorithm than the NLM. This may be due to the fact that we left all the BK classes that were assigned to the datasets in the training and test data (in contrast to the other classification systems from which we only took one notation in each case). Like that, many different combinations of BK classes exist that probably allow to distinct the single datasets more clearly from each other – in a way that leads to more precise predictions when using an instance-based algorithm. Additionally, as can be seen in Fig. 2, the BK is present in more than twice the datasets as is the NLM classification.

Table 3. Baselines

Classification	Target class			
	LNfull	LN1-4	LN1-3	LNmain
DDC	10.6	12.2	19.8	26.6
NLM	34.5	34.9	39.7	41.4
RVK	12.1	12.9	19.9	22.5
LCC	7.3	7.9	15.2	17.8
BK	39.2	42.3	53.5	75.4
all, except NLM	44.0	47.0	57.9	77.7

5 Discussion

The easiest way to classify previously unseen documents consists in identifying the largest class of a training set during the training phase (i.e. the class that contains most of the documents) and assigning this class to all documents to-be-classified. In some cases, this approach already leads to satisfying results, especially when only a few classes exist and the documents are distributed unevenly across these classes. But in our case the application of this method does not lead to good results since none of the classes can represent the whole data in a satisfactory manner. In the presented data, only 2.8% of the documents are classified correctly when assigning the most frequently appearing class (which is *W 50*) to all documents. Compared to this value, the results of our experiments are already considerably more precise. Below, we illustrate some factors and aspects that favoured these results resp. anticipated better results.

A quite high proportion of the datasets is represented by only one classification system (6,680 datasets, which is 34.5%). Due to a lack of sufficient information, it is hard for machine learning systems to detect differences between the classes of these documents. Moreover, if this single classification system is a rather vague one (like in our case the DDC and the Basisklassifikation), hardly any correlation between these systems and the target class can be established.

Additionally, in most cases there is no objective “correct” or “false” for the assignment of a notation. In fact, this decision depends on the cataloger, so that factors like experience, time, place, or expertise influence the classification process. Hence, (semi-)automatic systems that try to learn regularities out of manually analyzed datasets always rely on these discrepancies. Inconsistent, faulty or even contradictory data therefore limit the exactness of such systems.

Possible Optimizations. Several stages in the data mining process offer the potential for optimizing the results of the experiment. In the phase of data preprocessing, we decided to take only the first value of each attribute into account. The consideration of more values (if existent) could lead to more precise results, since the datasets would be represented in a more detailed and differentiated way. Moreover, the data sparseness is a fundamental problem that could be minimized by grouping similar classes together. This would on the one hand lead to less specific classes but on the other hand could allow easier assignments for the machine learning system.

The usage and analysis of other data mining algorithms could also lead to better results, especially when these algorithms allow to compare the similarity of datasets in a more specific way than the kNN algorithm. An important aspect here is the determination of the term “similarity”. In our experiment, we treated two different notations as completely diverse, i.e. the distance between the vectors representing these documents is maximal. But of course a document from the class *WN 190* is probably very similar to one from *WN 195* but rather different to one inside the class *WC 534*. If these hierarchical relations could be taken into account, the results would probably be more accurate. However, it is a non-trivial

challenge to include these hierarchies of a classification system into the machine learning process. Usually, external sources need to be taken into account because the classification system itself does not offer a satisfying way to identify relations between its classes by solely looking at its notations.

Besides, the method presented in this paper assumes that the importance of the datasets' attributes is equivalent, which in fact is not the case: The attribute representing the NLM classification is more meaningful than the others since this is exactly the classification system that is also used for the local notation assigned by the library of the MHH. By weighting the individual attributes differently, this attribute could gain more influence during the data mining process. The importance of the other classification systems used in the datasets could also be evaluated so that some kind of "hierarchy of importance" could be taken into consideration.

6 Conclusion

Little research is engaged in the automatic classification of physical library resources. The few available experiments mostly focus on online resources and common classification systems like the DDC or the LCC. In fact, no research explicitly treating medical literature or rather the NLM classification system could be detected. Additionally, the research on hand usually considers datasets that are represented by attributes such as book titles, keywords, abstracts, or even full-text documents. We took a different approach and applied the kNN machine learning algorithm to datasets that represent medical literature by solely adhering to notations from other library classifications systems. Despite our data being characterized by many different categories and a sparse distribution of its datasets, it was possible to achieve satisfactory results with a rather simple algorithm. Improvements could be obtained by adjusting the algorithm and by pursuing a semi-automatic classification strategy that not only examines the most similar class as determined by the algorithm, but that also takes additional possibilities into account. Further research in this area will show how far the machine learning approach can lead in automatically classifying medical literature.

References

1. Cheng, P.T., Wu, A.K.: ACS: An Automatic Classification System. *Journal of Information Science* 21(4), 289–299 (1995), DOI: 10.1177/016555159502100405
2. Dahlberg, I.: Why a New Universal Classification System is Needed. *Knowledge Organization* 44(1), 65–71 (2017)
3. Humphrey, S.M., Miller, N.E.: Knowledge-based Indexing of the Medical Literature: The Indexing Aid Project. *Journal of the American Society for Information Science* 38(3), 184–196 (1987), DOI: 10.1002/(SICI)1097-4571(198705)38:3<184::AID-ASI7>3.0.CO;2-F
4. Ishida, E.: An Experiment of Automatic Classification of Books Using Nippon Decimal Classification. *Library and Information Science* 39, 31–45 (1998), <http://lis.mslib.jp/pdf/LIS039031.pdf>

5. Joorabchi, A., Mahdi, A.E.: An Unsupervised Approach to Automatic Classification of Scientific Literature Utilizing Bibliographic Metadata. *Journal of Information Science* 37(5), 499–514 (2011), DOI: 10.1177/0165551511417785
6. Larson, R.R.: Experiments in Automatic Library of Congress Classification. *Journal of the American Society for Information Science* 43(2), 130–148 (1992)
7. Leung, C.H., Kan, W.K.: A Statistical Learning Approach to Automatic Indexing of Controlled Index Terms. *Journal of the American Society for Information Science* 48(1), 55–66 (1997), DOI: 10.1002/(SICI)1097-4571(199701)48:1<55::AID-ASI7>3.0.CO;2-0
8. Lu, K., Mao, J.: An Automatic Approach to Weighted Subject Indexing – An Empirical Study in the Biomedical Domain. *Journal of the Association for Information Science and Technology* 66(9), 1776–1784 (2015), DOI: 10.1002/asi.23290
9. Lüschor, A.: Automatische Klassifizierung medizinischer Literatur durch Analyse verfügbarer Notationen. Bachelor’s thesis, Hannover University of Applied Sciences and Arts (2016), <http://nbn-resolving.de/urn:nbn:de:bsz:960-opus4-10583>
10. Mork, J.G., Jimeno Yepes, A.J., Aronson, A.R.: The NLM Medical Text Indexer System for Indexing Biomedical Literature. *BioASQ* (2013), https://ii.nlm.nih.gov/Publications/Papers/MTI_System_Description_Expanded_2013_Accessible.pdf
11. Névéal, A., Shooshan, S.E., Claveau, V.: Automatic Inference of Indexing Rules for MEDLINE. *BMC Bioinformatics* 9(11), S11 (2008), DOI: 10.1186/1471-2105-9-S11-S11
12. Oberhauser, O.: Automatisches Klassifizieren. Entwicklungsstand, Methodik, Anwendungsbereiche. *Europäische Hochschulschriften, Reihe XLI Informatik* (43) (2005)
13. Pong, J.Y.H., Kwok, R.C.W., Lau, R.Y.K., Hao, J.X., Wong, P.C.C.: A Comparative Study of Two Automatic Document Classification Methods in a Library Setting. *Journal of Information Science* 34(2), 213–230 (2007), DOI: 10.1177/0165551507082592
14. Sebastiani, F.: Machine Learning in Automated Text Categorization. *ACM Computing Surveys* 34(1), 1–47 (2002), DOI: 10.1145/505282.505283
15. Voss, J., Balakrishnan, U.: The Cocoda Mapping Tool. Presentation. 14th European Networked Knowledge Organization Systems (NKOS) Workshop, Poznan, September 18th (2015), <http://hdl.handle.net/10760/28007>
16. Wang, J.: An Extensive Study on Automated Dewey Decimal Classification. *Journal of the American Society for Information Science and Technology* 60(11), 2269–2286 (2009), DOI: 10.1002/asi.21147