A Hybrid Approach to Assignment of Library of Congress Subject Headings

Christian Wartena and Michael Franke-Maier

Abstract Library of Congress Subject Headings (LCSH) are popular for indexing library records. We studied the possibility of assigning LCSH automatically by training classifiers for terms used frequently in a large collection of abstracts of the literature on hand and by extracting headings from those abstracts. The resulting classifiers reach an acceptable level of precision, but fail in terms of recall partly because we could only train classifiers for a small number of LCSH. Extraction, i.e., the matching of headings in the text, produces better recall but extremely low precision. We found that combining both methods leads to a significant improvement of recall and a slight improvement of F1 score with only a small decrease in precision.

Christian Wartena
Hochschule Hannover
Expo Plaza 12, 30539 Hannover, Germany
✉ christian.wartena@hs-hannover.de

Michael Franke-Maier
Freie Universtät Berlin, Universitätsbibliothek
Garystraße 39, 14195 Berlin
✉ franke@ub.fu-berlin.de

Archives of Data Science, Series A
(Online First)
KIT Scientific Publishing
Vol. 4, No. 1, 2018

DOI: 10.5445/KSP/1000085951/22
ISSN 2363-9881

1 Introduction

Library of Congress Subject Headings (LCSH) are popular for indexing documents. There are over 400 000 different subject headings with a very limited number of structuring relations between the headings. In any collection of annotated documents, most LCSH will never be used. Most headings at the bottom of the hierarchy cannot be understood as classes that should be used to classify documents but rather as normalized names for an entity or concept. We find examples of very specific headings like those given in Table 1.

Table 1: Examples of very specific subject headings. Only the last part of the URI is displayed, while the prefix `http://id.loc.gov/authorities/subjects/` is left out.

URI	Label
sh00000172	Halle 13 (Expo, International Exhibitions Bureau, 2000, Hannover, Germany)
sh2005002460	Brown versus Board of Education of Topeka
sh85120114	Septets (Piano, flute, zither, percussion)

Thus, it seems to be nearly impossible to train a classifier to assign Library of Congress Subject Headings. The fact that many labels of subject headings are ambiguous and the fact that labels of subject headings in many cases are highly frequent words occurring in many texts also makes it hard to extract labels like named entities from a text, as we have shown in previous research (Aga et al, 2016).

In the following, we propose a hybrid approach. For the most frequent headings, we train a classifier, whereas we use extraction to create specific headings. We show that we can achieve good results that way. Nevertheless, the resulting headings differ significantly from manually assigned headings.

2 Related Work

Research on automatic keyword extraction came up in the context of automatic information retrieval starting with the question of what words are suited as index terms (Salton and Buckley, 1988). In 1972, Spärck Jones (reprinted as Spärck Jones (2004)) proposed a term-weighting scheme defined by the relation

between *exhaustivity* and *specificity* that became known as tf.idf weighting. The tf.idf weighting has proven to be a relevance measure that is hard to beat.

However, there are further factors that determine whether a word is likely to be viable as a keyword respectively keyphrase. Thus, Frank et al (1999) and Turney (2000) proposed a supervised-learning approach to combine tf.idf weighting with multiple other features. Both studies pointed to the improved performance of the extraction algorithm if there is domain-specific knowledge.

If keywords are to be selected from a structured thesaurus, the thesaurus's hierarchical structure and the relations between potential keywords can be useful sources for improving algorithmic results (Brussee et al, 2010), e.g. by counting the number of thesaurus relations (Tiun et al, 2004). The core concept behind these approaches is that all appropriate keywords have to be related to the main topic of the text and thus to each other. In that manner, completely unrelated keywords are identified and filtered out. Problems of those approaches are caused by the limited number or restricted nature, respectively, of available keywords or, strictly speaking: *descriptors*.

Pouliquen et al (2006) distinguish between conceptual thesauri (CT) and natural-language thesauri (NLT). Since most concepts of CT will never be found in text, they argue that the expression "keyword assignment" should be used rather than "keyword extraction". To solve the key challenge in assigning not literally present keywords, Pouliquen, Steinberger and Ignat build topic signatures for each concept. A signature is a vector of words for each concept. The words with a high weight in such vectors are typical for texts annotated with the respective concept. Pouliquen and his coauthors note that the amount of training material needed is a problem, even for the EUROVOC thesaurus (7041 active descriptors in June 2017), which is much smaller than the LCSH vocabulary.

Wartena et al (2010) use vectors of latent features for representing both the potential keywords and the text (however constructed in a completely different way). By computing distributional similarity between keywords and abstracts, they measure the importance and discriminative powers of candidates or the suitability of the keyword, respectively.

The LCSH vocabulary is not often used for automatic keyword extraction. Medelyan et al (2010) report on a software that can work with LCSH, but they did not evaluate the results. Larson (1992) uses LCSH as one of several other clustering elements for the automatic selection of Library of Congress Classification. Paynter (2005) automatically assigns LCSH to a text by collecting

LCSH terms that are assigned to similar texts. In the project *Machine-based subject indexing of English language online publications* (MAEN) the German National Library seems to use commercial software developed by Averbis (`https://averbis.com/`) to assign LCSH (Betz, 2017).

3 Data

3.1 Bibliographic Records

As underlying dataset we used the bibliographic records of the B3KAT, a union catalog shared by libraries in the German States (Bundesländer) Bavaria, Berlin and Brandenburg. The Freie Universität Berlin libraries share their cataloging data as a B3KAT partner. The B3KAT is hosted by the head office of the Bavarian Library Network, a department of Bayerische Staatsbibliothek. It includes about 26 million bibliographic records and more than 61 million associated library items. The B3KAT also has a Linked Open Data Representation that we could use for our purposes. It contains about 980 elementary statements (triples) in RDF (Resource Description Framework) language (Manola et al, 2004); the data is licensed under a Creative Commons Zero (CC0) license[1] and can be downloaded or queried through a SPARQL endpoint (a public interface for an RDF-database using the SPARQL query language; Prud'hommeaux and Seaborne, 2008; Bayerische Staatsbibliothek, 2015). We retrieved over 12 000 records via the SPARQL endpoint fulfilling the following criteria:

1. Abstract of at least 200 characters;
2. abstract written in English, according to metadata and language detection;
3. metadata containing Library of Congress Subject Headings (that can be used as ground truth for evaluation).

The dataset is the same one as used in Aga et al (2016). Since most records are also classified according to the Dewey Decimal Classification (DDC) we know for most records what disciplines they belong to. Figure 1 shows the distribution of the records in the dataset over the 10 DDC main classes.

[1] `https://creativecommons.org/share-your-work/public-domain/cc0/`

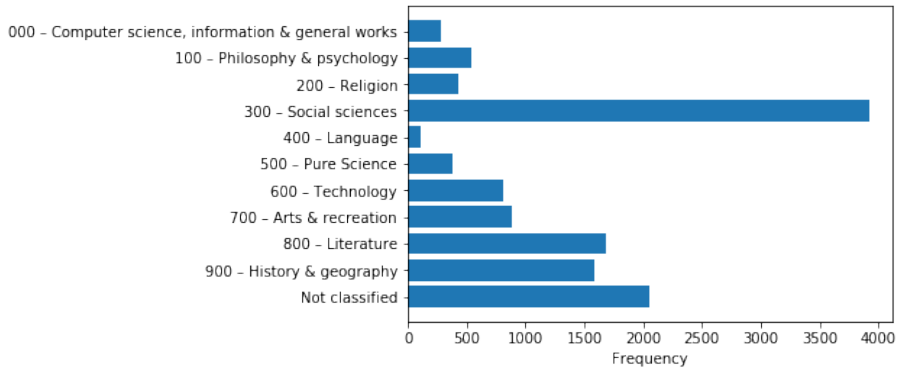

Figure 1: Main classifications of the records in the dataset.

We split the data into three sets: Training Set 1, consisting of 11 544 records used for training classifiers, Training Set 2, consisting of 500 records used for finding optimal thresholds, and a Test Set of 500 records. To represent the records for the classifiers, we use title, subtitle and abstract. We extract all words occurring in at least 10 documents and in at most 2 000 documents after lemmatization. This results in a set of 6 660 words that are used as features for classification.

3.2 Library of Congress Subject Headings (LCSH)

Library of Congress Subject Headings (LCSH) are a dynamic collection of standardized terms used since 1898 for cataloging materials at the Library of Congress. LCSH is widely adopted in the anglophone world and approximately 40 000 headings are cross-referenced to the subject headings collections of the German and French national libraries, GND and Rameau. The system is easily and freely available in a linked open data representation from `http://id.loc.gov/authorities/subjects.html`.

LCSH have IDs and labels. It is important to remember that subject headings are standardized forms of arbitrary terms and not like classes of an ontology or thesaurus. Given the huge number of subject headings (over 400 000) it is almost impossible to train a classifier for all subject headings. For many LCSH we do not have training data at all.

Besides the large number of headings, we have the problems of ambiguous labels and pre-combined headings: Headings can have various labels and, in several cases, headings share labels. Especially when we remove scope notes from the labels, this problem arises. This makes it hard to find LCSH by matching labels in the text. Most headings are used once in our dataset. Figure 2 shows the frequency distribution of LCSH in our dataset, the most frequent headings in Training Set 1 are displayed in Table 2.

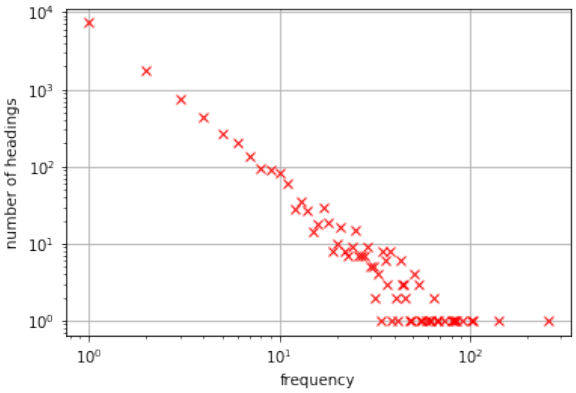

Figure 2: Frequency distribution of LCSH: number of headings with given frequency in the dataset.

Table 2: Most frequent LCSH in Training Set 1 (11 544 records).

LCSH	Label	Frequency
sh85056605	Great Britain	239
sh85147430	Women and literature	100
sh85045631	Europe	98
sh85043777	English literature	90
sh85009808	Authors, American	90

In Training Set 1 we find 10 944 different headings (2,6 % of all available headings) with a total of 28 818 assignments. 451 headings are used over 10 times with a total of 9 458 occurrences. For each of these headings, we will train classifiers.

4 Methods

We will use two methods for automatic assignment of subject headings: We train a classifier for the most common headings and we will use extraction for the rare ones. In our training data, there are 451 subject headings that occur at least 10 times. The average number of occurrences of these headings in Training Set 1 is 21. We will train a classifier for those subject headings.

4.1 Classification

The first question we want to answer is what the best possible result of the classifier would be. If the classifier were never to make a mistake, we would have a precision of 1. However, it will miss a lot of headings, since it can only predict 451 out of over 400 000 headings. We implemented a classifier doing exactly this to find the upper bound that a classifier in this setting can reach. We will refer to this classifier in the results as *Oracle*.

We train classifiers on Training Set 1 using logistic regression for each of the 451 headings occurring at least 10 times. Each classifier is thus a one-versus-the-rest (ovr) classifier (Bishop, 2006, p. 338). We used the standard implementation of the ovr scheme for logistic regression from the Scikit-learn package (Pedregosa et al, 2011). The ovr scheme attributes a probability of suitability to every subject heading that a classifier can assign. Usually, these probabilities are quite low. Now we can either assign the n most probable headings (with different values for n), or we can assign all headings with a probability above t (again with different values for t). Finally, we also can combine both methods: we assign the most probable n headings and, additionally, all headings with a probability over t. Those approaches will results in a varying number of headings assigned to each record.

We use Training Set 2 to find optimal values for n and t. Results for increasing values of n on Training Set 2 are given in Figure 3. We see that we get optimal F1 score if we assign just 1 subject heading. Similarly, we test the optimal threshold when the assignment is based on a threshold. Results are given in Figure 4. Here we see that precision increases fast with an increasing threshold while recall decreases only slightly.

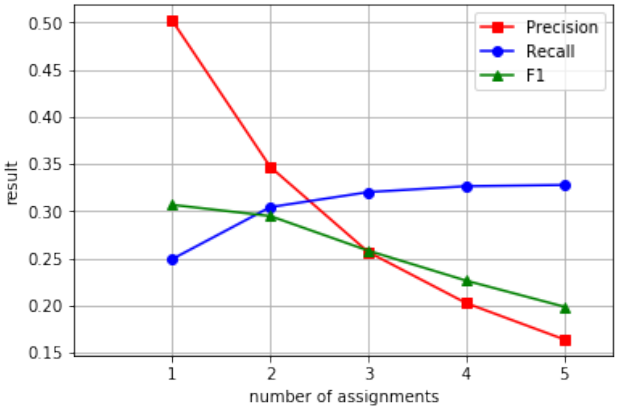

Figure 3: Results of classification (averaged precision and recall) on Training Set 2 for increasing number of assignments. The number of assignments n is displayed on the horizontal axis.

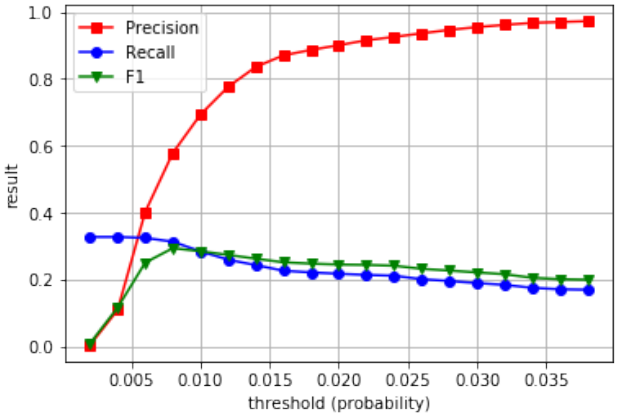

Figure 4: Results of classification (averaged precision and recall) on Training Set 2 for increasing number threshold. The horizontal axis here is the threshold.

Optimal results were achieved when the single most probable heading was combined with all headings above a certain relatively high probability. We found 0.03 to be an optimal value. The values for the parameters thus are $n = 1$ and $t = 0.03$.

4.2 Extraction

For extraction, we take all preferred and alternative labels of the subject headings, but we exclude all ambiguous labels. If the label is found literally in the title or abstract, we assign the corresponding subject heading. Note that, for many reasons, a label of a subject heading might be present as a word in the abstract or title, even though the corresponding subject heading is not correct and would not be selected in manual annotation (Aga et al, 2016).

4.3 Combining Classification and Extraction

In order to get a higher recall than possible with a classifier trained on just a small fraction of all headings, we combine both methods. For extraction, we now exclude all headings that are found at least 10 times in Training Set 1. These are of course the 451 headings for which we trained classifiers, but also all headings that were never assigned but that have a label occurring frequently in the training data. Thus we exclude most labels from extraction that are common English words, like *example*, *impact* or *research*.

Finally, we simply combine the terms produced by applying both the classification and the extraction methods, using the reduced label set as described above. We will refer to this strategy as the *combined* method. Thus we follow the steps below in assigning a subject heading to a record r:

1. Assign the most probable of 451 frequent LCSH found by the classifier.
2. subsequently, assign all other LCSH with a confidence rating of t or higher. We will use $t = 0.03$.
3. assign all LCSH that occur less than 10 times in the training data and that have an unambiguous label that is found literally in the text of r.

5 Results

For evaluation, we compare the assigned labels with the given labels in the test set. For each record, we compute precision, recall, and F1 score and report the average result over all records in the test set.

The results for all methods are displayed in Table 3. We see that recall and F1 of our classifier is quite close to the optimal result. Furthermore, we see that results of classification are clearly superior to those of extraction. Most interestingly, adding extraction results to the classification results improves recall significantly while the harm to precision is limited. This method gives the best results of all methods used.

Table 3: Results (averaged precision and recall) of all classification methods on the test set (500 records).

	Precision	**Recall**	**F1**
Extraction	0.071	0.30	0.10
Oracle	1.0	0.28	0.34
Log. Regr. (n=1)	0.47	0.22	0.28
Log. Regr. (n=1 OR p > 0.03)	0.46	0.26	0.31
Combined	0.40	0.37	0.33

6 Discussion and Conclusion

LCSH are popular for subject indexing. Due to the overwhelming amount of possible headings, it is both hard to train classifiers for assigning LCSH and to evaluate the results since, in many cases, several differnt headings can be considered correct.

We have shown that a classifier using titles and abstracts trained for frequent headings achieves a good precision while matching labels in the abstracts gives a higher recall. Combining both approaches gives the highest overall results.

Further experiments have shown that the results get worse when we train more classifiers, even when more training data are available. This is not a surprise since we have to choose from an increasing number of headings. Furthermore,

we see that results are especially bad for disciplines that are underrepresented in our dataset, such as physics and medicine, while results for politics and others are much better.

A F1 score of 0, 33 might seem very low. Taking into account the low inter-annotator agreement in subject indexing in general, the result is not that bad. Nevertheless, it is not so convincing that we would completely rely on this system. Also, for suggesting terms to human indexers, the system seems not ideal, given the low recall.

In order to improve the results in the context of the B3KAT, we could include subject headings from other vocabularies as features for the classifiers, as Lüschow and Wartena (2017) did for a similar catalog including medical subject headings.

References

Aga RT, Wartena C, Franke-Maier M (2016) Automatic Recognition and Disambiguation of Library of Congress Subject Headings. In: Research and Advanced Technology for Digital Libraries: 20th International Conference on Theory and Practice of Digital Libraries (TPDL 2016), Fuhr N, Kovács L, Risse T, Nejdl W (eds). Springer International Publishing, pp. 442–446. ISBN: 978-3-319439-97-6, DOI: 10.1007/978-3-319-43997-6_40.

Bayerische Staatsbibliothek (2015) Linked Open Data Service of B3Kat lod.b3kat.de. URL: http://lod.b3kat.de.

Betz F (2017) Automatic Indexing at the German National Library: Experiences and Results. Helsinki. URL: http://slideplayer.com/slide/11816283/. Presentation at the Finish National Library.

Bishop CM (2006) Pattern Recognition and Machine Learning, 1st edn. Information Science and Statistics, Springer, New York. ISBN: 978-0-387310-73-2.

Brussee R, Wartena C, Gazendam L (2010) Thesaurus based Term Ranking for Keyword Extraction. In: IEEE Proceedings of the 7th International Workshop on Text-based Information Retrieval (TIR-10). DOI: 10.1109/DEXA.2010.31.

Frank E, Paynter G, Witten I, Gutwin C, Nevill-Manning C (1999) Domain-Specific Keyphrase Extraction. In: Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI 99), 2, pp. 668–673. URL: https://www.ijcai.org/proceedings/1999-2.

Larson RR (1992) Experiments in Automatic Library of Congress Classification. Journal of the American Society for Information Science 43(2):130–142. DOI: 10.1002/(SICI)1097-4571(199203)43:2<130::AID-ASI3>3.0.CO;2-S.

Lüschow A, Wartena C (2017) Classifying Medical Literature Using k-Nearest-Neighbours Algorithm. In: Proceedings of the 17th European Networked Knowledge Organization Systems Workshop co-located with the 21st International Conference on Theory and Practice of Digital Libraries 2017 (TPDL 2017), Mayr P, Tudhope D, Golub K, Wartena C, Luca EWD (eds), CEUR-WS.org, CEUR Workshop Proceedings, Vol. 1937, pp. 26–38. URL: http://ceur-ws.org/Vol-1937/paper3.pdf.

Manola F, Miller E, McBride B, et al (2004) RDF primer. Tech. Rep., W3C. URL: https://www.w3.org/TR/rdf-primer/.

Medelyan O, Perrone V, Witten I (2010) Subject metadata support powered by Maui. In: Proceedings of the 10th annual joint conference on digital libraries, JCDL '10, pp. 407–408. ISBN: 978-1-450300-85-8, DOI: 10.1145/1816123.1816204.

Paynter G (2005) Developing Practical Automatic Metadata Assignment and Evaluation Tools for Internet Resources. In: Proceedings of the 5th ACM/IEEE-CS joint conference on digital libraries, JCDL '05, pp. 291–300. ISBN: 15-8113-876-8, DOI: 10.1145/1065385.1065454.

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al (2011) Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research (JMLR) 12(Oct):2825–2830. URL: http://www.jmlr.org/papers/v12/pedregosa11a.html.

Pouliquen B, Steinberger R, Ignat C (2006) Automatic Annotation of Multilingual Text Collections with a Conceptual Thesaurus. In: Proceedings of the Workshop "Ontologies and Information Extraction" at the Summer School "The Semantic Web and Language Technology" (EUROLAN'2003), pp. 9–28. URL: https://arxiv.org/ftp/cs/papers/0609/0609059.pdf.

Prud'hommeaux E, Seaborne A (2008) SPARQL Query Language for RDF. Tech. Rep., W3C. URL: http://www.w3.org/TR/rdf-sparql-query/.

Salton G, Buckley C (1988) Term-weighting Approaches in Automatic Text Retrieval. Information Processing and Management 24(5):513–523. DOI: 10.1016/0306-4573(88)90021-0.

Spärck Jones K (2004) A Statistical Interpretation of Term Specificity and its Application in Retrieval. Journal of Documentation 60(5):493–502. DOI: 10.1108/00220410410560573.

Tiun S, Abdullah R, Kong T (2004) Automatic Topic Identification Using Ontology Hierarchy. In: Computational Linguistics and Intelligent Text Processing, Gelbukh A, Gelbukh A (eds), Lecture Notes in Computer Science. Springer, pp. 444–453. DOI: 10.1007/3-540-44686-9_43.

Turney P (2000) Learning Algorithms for Keyphrase Extraction. Information Retrieval 2(4):303–336. DOI: 10.1023/A:1009976227802.

Wartena C, Brussee R, Slakhorst W (2010) Keyword Extraction Using Word Cooccurrence. In: Database and Expert Systems Applications, International Workshops (DEXA 2010), Tjoa A, Wagner R (eds), IEEE Computer Society, pp. 54–58. DOI: 10.1109/DEXA.2010.32.