Journal of Informetrics 15 (2021) 101166

Contents lists available at ScienceDirect

Journal of Informetrics

journal homepage: www.elsevier.com/locate/joi

A supervised machine learning approach to author disambiguation in the Web of Science

Andreas Rehs[1]

Deutsche Bundesbank, Germany

ARTICLE INFO

Keywords:
Author name disambiguation
Machine learning
Pairwise classification
Random forest
Community detection
Web of science

ABSTRACT

Author-level scientometric indicators are an important tool in individual and institutional-based research assessment and require high-quality author-publication profiles. To address this need, our study developed a robust supervised machine learning approach in combination with graph community detection methods to disambiguate author names in the Web of Science publication database. We used the unique author identifier *Researcher ID* to retrieve true authorship data of 1,904 scientists and trained a random forest and a logistic regression classifier on 1.2 million corresponding publication pairs with authors that share the same last name and first name initial. To do this, we reviewed a vast set of paper and author characteristics and randomly included missing data to make our machine learning robust to quality changes of new publication data. In the application on an unseen test set, we achieved F1 scores of 0.82 in the random forest and 0.75 in the logistic regression model. Subsequently, we evaluate feature performance and apply the *infomap* graph community detection algorithm to identify all publications belonging to an author. The community detection results in reasonable cluster metrics (Mean K-Metric in logistic regression-based model = 0.78 and = 0.81 in random forest-based model). Finally, we test our algorithm on a large surname-initial block ("Muller, M.") and demonstrate speed and predictive performance.

1. Introduction

Author-level scientometric indicators have become an important object of research and help to understand the fundamentals of the scientific system. They are, for instance, crucial in investigating scientist productivity (Hirsch, 2005), scientific collaboration (Glänzel & Schubert, 2006) or mobility patterns in academia. Similarly, author-level scientometric indicators are being used more often to evaluate individual researchers and the scientific system in general, thereby providing a vital basis for the decision-making of university administrators and policy makers (Abbott et al., 2010; Hicks, 2012). The databases upon which these indicators are calculated should therefore be of adequate quality and represent the actual author publications as well as possible (D'Angelo & van Eck, 2020; Weingart, 2005).

One of the key challenges to the validity of indicators is author name ambiguity, also called the namesake problem (Shin, Kim, Choi & Kim, 2014). The namesake problem belongs to the universal problem of entity resolution and concerns problems of whether references to entities belong to the same entity or different entities (Talburt, 2011). One issue in author name ambiguity is called the block problem and typically occurs with common names like "Zhang, Ying", which appeared in 6124 publications in the 2017 version of the publication database Web of Science (WOS). Two papers written by "Zhang, Ying" might be written by either the same

E-mail address: andreasrehs@googlemail.com

[1] The views expressed in this paper are those of the author(s) and do not necessarily coincide with the views of the Deutsche Bundesbank or the Eurosystem.

https://doi.org/10.1016/j.joi.2021.101166
Received 3 October 2020; Received in revised form 4 April 2021; Accepted 13 April 2021

"Zhang, Ying" or two different people, each with the name "Zhang, Ying". Blocks are even more problematic when using abbreviations and initials in author names. "Zhang, Ying" becomes "Zhang, Y.", which is not distinguishable from "Zhang, Yong" or other names starting with the letter Y. This abbreviation practice increases the number of publications by "Zhang, Ying" to be disambiguated in the WOS from 6,124 to 256,554. Synonyms are also a problem in name ambiguities. Synonym problems occur when the same author appears under different names, such as the German name "Müller", which is sometimes written as "Muller" or "Mueller". The manual disambiguation of such block and synonym cases in small datasets is usually the best way to account for all problems. However, when it comes to large datasets, such as for "Zhang, Ying", manual disambiguation is impossible in a reasonable amount of time and may include errors as well (Shin et al., 2014). Accordingly, adequate publication databases that represent all publications belonging to an author cannot be built, and the validity of subsequent scientometric indicators is endangered.

A substantial amount of literature therefore deals with computational methods to solve this problem. With the advancements in methodology and the increase in computational capacities, machine learning methods came into the focus of research One can distinguish the different approaches by their requirement of training data into supervised and unsupervised approaches (Hussain & Asghar, 2017; Torvik & Smalheiser, 2009). Supervised methods (Kim, Rohatgi & Lee Giles, 2019; Louppe, Al-Natsheh, Susik & Maguire, 2016) require external labeling of training data regarding whether the papers in question are from the same author. With this information, the methods learn how paper attributes refer to the same or different authors. Unsupervised methods (Caron & van Eck, 2014; D'Angelo & van Eck, 2020; Wu, Li, Pei & He, 2014) don't require labeled training data; instead, they try to find patterns in the data by themselves.

Both supervised and unsupervised approaches to author disambiguation have certain advantages and disadvantages. Unsupervised methods are easier to implement but are often much more computationally expensive than author disambiguation (Hussain & Asghar, 2017). Supervised approaches allow researchers to determine the importance of features, but are more reliant on adequate, reliable, and representative training data (Smalheiser & Torvik, 2009). In previous author name disambiguation, both families of approaches have been shown to deliver comparably high performance (Hussain & Asghar, 2017).

In this paper, we implement a supervised machine learning approach using random forest and logistic regression. This choice is motivated by two factors. First, there is generally limited knowledge on feature importance (Tekles & Bornmann, 2020). Moreover, we develop features that are either new to literature or whose performance, also in conjunction with traditional features, has not yet been examined thoroughly. Our developed features especially add to the literature on block sizes (Caron & van Eck, 2014; Tekles & Bornmann, 2020) and name-based features (Smalheiser & Torvik, 2009) as we test novel features based on first name frequencies and ratios of first or last names to block sizes. Both machine learning algorithms we use allow us to assess features and have previously been successfully applied in author name disambiguation (Abdulhayoglu & Thijs, 2017; Gurney, Horlings & van den Besselaar, 2012; Kim & Kim, 2018; Louppe et al., 2016). Second, we choose a supervised machine learning, since supervised learning scales well and potentially allows us to disambiguate vast sets of homonyms in a reasonable amount of time.

To address the problems of adequate training data, we use the well-established Researcher ID author identifier of Clarivate Analytics and develop a sophisticated sampling and manipulating strategy to resemble the true data-generating process of authorship in the WOS. In the manipulation step, we therefore insert missing data in order to make our machine learning robust to the large amount of missing data in the WOS. Missing data has been identified as an issue in disambiguation approaches (Tang & Walsh, 2010) and is not specifically addressed in approaches that used similar datasets. Tekles and Bornmann (2020), for instance, who also rely on Researcher ID, did not manipulate their first name data. Therefore, their comparison of unsupervised approaches may overestimate the disambiguation performances.

A crucial step in author disambiguation approaches is to group or cluster the obtained predictions between every two papers. This task is complicated by mainly two problems. First, the actual number of authors K is not known in advance and must be estimated in a reliable and scalable manner (Hussain & Asghar, 2018b). Second, the clustering approach must deal with problems arising from transitivity violations in the pairwise similarities between papers (Wang et al., 2012). Transitivity violations occur, for instance, when two papers have been estimated to be dissimilar (and therefore likely to be from different authors), but are predicted to be similar to a third paper (and therefore likely to be from the same author).

With respect to the problem of the number of authors , the literature mainly uses agglomerative clustering methods, which are problematic when transitivity violation is involved (Ferreira, Gonçalves & Laender, 2012). Graph-based approaches provide a viable alternative, as graphs can naturally represent transitive relations. In graph-based approaches, papers are often considered as nodes, and similarities to other papers as edges. The previous literature has used graph-based author disambiguation methods, either in complete disambiguation approaches or as pre-processing strategies. Several complete approaches dissolve ambiguous blocks by means of co-authors networks, such as (Fan, Wang, Pu, Zhou & Lv, 2011). In newer (complete) approaches, machine learning methods directly revert on the graphs and disambiguate the authors (e.g., Qiao, Du, Fu, Wang & Zhou, 2019). A graph-based pre-estimation strategy has been used by Wang, Tang, Cheng and Yu (2011).

In this paper, we build on a graph-based post-estimation strategy to author name disambiguation. A post-estimation strategy has been adopted before by Gurney et al. (2012) and rests upon the assumption that the preceding supervised machine learning has predicted sufficient correct linkages between every two papers. Those correct linkages are required to detect graph communities or, in other words, all papers belonging to an author. The use of a graph-based post-estimation strategy brings with it the advantage that it can easily be implemented, deals with transitivity, scales well and allows researchers to visually represent the predictions from the machine learning in a network graph.

In the subsequent graph community detection, we use the infomap algorithm (Rosvall & Bergstrom, 2007). The basic idea of the infomap algorithm is that random walks from a given node are more likely to stay within the same community rather than leave the community. In comparison to other graph-based community detection methods, such as the Louvain algorithm, infomap is generally

faster and more accurate (Lancichinetti & Fortunato, 2009). Our application of the infomap algorithm may therefore outperform previous disambiguation approaches that rely on such algorithms (e.g., Gurney et al., 2012).

The remainder of the paper is as follows: We start by investigating the characteristics of previously applied disambiguation approaches. In our subsequent data analysis, we prepare over 1.2 million paper pairs with authors that share the same last name–first initial combination, but who are distinguishable by their Researcher ID (Enserink, 2009). Using the Researcher ID authors can assign papers to their user account in the WOS. The Researcher ID has previously been shown to be useful in providing true authorship information by Tekles and Bornmann (2020) and is amply available in the WOS database. We proceed by comparing the retrieved paper pairs by their attributes and reviewing a vast set of common and novel author and paper characteristics. We find that the complete WOS includes up to 95% missing first names before 2006 and up to 25% missing first names after 2006. Therefore, we randomly insert missing first names and second initials to make our training and test set, which is of higher quality, more similar to unlabeled publication data in the WOS.

In the next section, we describe our methodological approach using the machine learning algorithms of random forest and logistic regression. The results on the more than 53,000 pairwise paper comparisons of the test set follow, which yield an F1 score of 0.82 in the random forest and 0.75 in the logistic regression. To aggregate the pairwise predictions into author clusters—in other words, all papers belonging to a single author—we apply the infomap algorithm which is described in the section “Graph-based author community detection”. The clustering results suggest that different authors rarely appear in the same cluster and that same authors are rarely split into different clusters. The subsection “Application to full block ’Muller, M.” addresses the external validation of our approach. We use the large block “Muller, M.” with 11,665 papers and analyze the clustering results for the subset of papers containing Researcher IDs. The clustering results are reasonable and suggest the large-scale application potential of our approach. Finally, we discuss the results and conclude our approach.

2. Characteristics of disambiguation approaches

2.1. Disambugiation methods

The numerous disambiguation approaches in the literature differ in methods, data, scope and features used. Hussain and Asghar (2017) provide a rich survey of these approaches and distinguish on the method-level non-machine learning-based approaches and machine learning-based approaches. Machine learning techniques break down into supervised techniques, unsupervised techniques, and semi-supervised techniques. Non-machine learning-based techniques are split into graph-based and heuristic-based methods. In addition to Hussain and Asghar’s categories, we include the third category probabilistic approaches. In the following paragraphs, we try to categorize the literature according to these techniques, and we also discuss the features used.

Graph-based methods use papers and their attributes as node and edge representations to detect author communities in a graph. As a result, papers with the same author block, but different real authors can be separated into connected graph-communities. Graph-based methods have been applied extensively—for example, by Fan et al. (2011), On, Lee and Lee (2012) and Shin et al. (2014)—are visually interpretable and have been shown to disambiguate authors accurately (see Fan et al., 2011).

Heuristic approaches (e.g., De Carvalho, Ferreira, Laender & Gonçalves, 2011) use paper attributes to construct simple rules with which authors and their papers can be distinguished. For example, all papers from “Zhang, Y.” that share at least one attribute, such as common co-authors or institution names, are assigned to be from the same “Zhang, Y.” in heuristic approaches.

Probabilistic approaches (e.g., Tang et al., 2011; Torvik & Smalheiser, 2009; Torvik, Weeber, Swanson & Smalheiser, 2005; Wang et al., 2012) try to set up a linkage function that gives the probability that two articles belong to the same author. Torvik et al. (2005), for example, used a reference set of true and false matches of author articles with the same name and calculated the matching probability between two articles based on several paper characteristics.

Machine learning approaches are characterized by their different requirement levels for training data. Supervised and semi-supervised methods (Kim et al., 2019; Louppe et al., 2016) require external labeling of training data regarding whether the papers in question are from the same author. With this information, the methods learn how paper attributes refer to the same or different authors. Unsupervised methods (Caron & van Eck, 2014; D’Angelo & van Eck, 2020; Wu et al., 2014) don’t require labeled training data; instead, they try to find patterns in the data by themselves but are therefore more computationally expensive.

Machine learning approaches can detect complex relationships in paper attributes and adequately handle missing data. Because of their good predictive performance, they generally achieve high precision and recall rates and are therefore superior to graph-based and heuristic-based methods. In comparison to probabilistic approaches, machine learning approaches are more comfortable to implement and allow for low-cost comparison and tuning of different algorithms. For predictive power, machine learning algorithms can implicitly discover the same statistical characteristics of papers and their attributes as probabilistic approaches do.

2.2. Features for disambiguation: Bibliographic characteristics

All the methods or databases described above require some paper or author characteristics with which the disambiguation is performed. Bibliographic information is most frequently used, as it presents the most important characteristics of a paper and is almost always available in publication platforms, such as the WOS and SCOPUS. This information typically includes journal title and issue, co-authors, keywords, abstract, publication title, subject classifications and year of publication. Bibliographic characteristics are used to either directly perform author disambiguation, or generate other disambiguation measures Treeratpituk and Giles (2009),

for example, determined journal languages in order to check if two articles of the same author block were published in the same language.

2.3. Features for disambiguation: Citation characteristics

The second most frequently used type of characteristic is citation data (e.g., Louppe et al., 2016; Onodera et al., 2011; Torvik et al., 2005). Onodera et al. (2011) evaluated citation data, especially self-citation data, as highly effective for author disambiguation. However, citation-based features require adequate time to gather citations, and that the paper under examination be recognized by the scientific community. Citation-based features do not scale very well for large publication platforms since the citation databases upon which the platforms are based go well beyond 9 digits, such as the more than 600 million citation relationships in the 2017 version of the WOS.

2.4. Features for disambiguation: Country and institution-based characteristics

Country, institution and department name(s) are used as characteristics in numerous studies and are similar to bibliographic information in that they are often available for at least one author of a paper. Torvik et al. (2005) reported that institutional names are extraordinarily good predictors for disambiguating author names, but require tedious preprocessing to check for stopwords and abbreviations (Rimmert, Schwechheimer & Winterhager, 2017). International or interinstitutional researcher mobility is a key challenge for country and institution-based measures, as the same author will change addresses multiple times in a scientific career. Therefore, country, institution, and department name(s) may be used only in conjunction with invariable or inert features, such as the first name or thematic characteristics.

2.5. Features for disambiguation: Name-based characteristics

Finally, features can be generated based on the author's name, frequency, and related statistical properties. Torvik et al. (2005) present a probabilistic model based on PubMed data. They define a similarity profile between a pair of articles in the same block, based on name attributes (middle initial, and suffix) and other paper characteristics. The similarity profile distribution is then computed from gold-standard reference sets. This reference set consists of pairs of articles that almost exclusively contain author matches versus nonmatches. In analyzing the reference set, one would, for instance, conclude that a rare similarity profile in a given block (e.g., {*non-frequent block, same suffix, same journal, same keyword}*) represents a high likelihood of matching between two articles. In turn, {*frequent-block, same keyword, same journal, different suffix*} may result in a frequent profile and low matching probability. Louppe et al. (2016) and Strotmann and Zhao (2012) used another related statistical property of names in this regard to show how the determination of ethnicity and related statistical properties of a name can improve name disambiguation.

3. Data

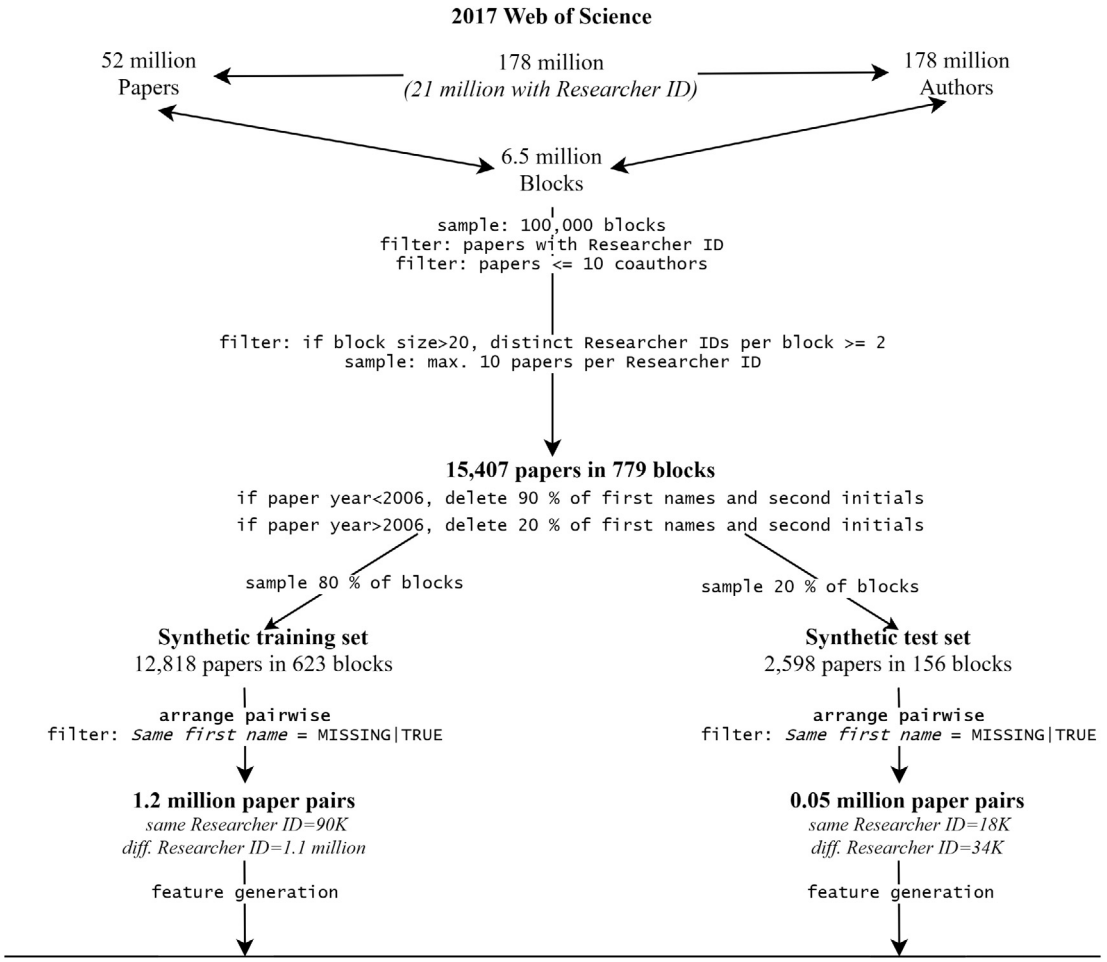
3.1. Data sources, sampling strategy and preprocessing

In the following, we want to present our data sources and methodology. Our disambiguation approach is based on pairs of papers in the same surname-initial block. For these blocks we know which paper-pairs belong to the same and different authors. The to be presented machine learning methodology then learns the relation between paper pairs from the same and different authors upon paper characteristics. Fig. 1 shows the schema of our data processing.

In the first step, we use a 2017 copy of the WOS processed by the Kompetenzzentrum Bibliometrie (Rimmert et al., 2017). The WOS has previously been used for disambiguation purposes by D'Angelo and van Eck (2020) and Tekles and Bornmann (2020) and is considered one of the leading publication platforms. The 2017 WOS includes about 52 million publications from 1980 to 2017 that relate to 178 million author names. Those 178 million author names again relate to 6.5 million different blocks. There is a Researcher ID available for 21 million paper-author relationships, which makes them distinguishable from other authors. The Researcher ID builds the basis for generating paper pairs of the same and different authors in a block.

To retrieve blocks and subsequently paper pairs that contain the Researcher ID, we randomly select 100,000 blocks and search for papers with Researcher IDs. In this step, we only consider papers with ten or fewer co-authors because of the computational power required and the presumably different author-characteristics of papers with more than ten co-authors. We also add restrictions on the number of distinct Researcher IDs in a block to address the unknown number of true authors in a block set. Our thought is that very infrequent blocks that have a Researcher ID available represent only one real author. For example, if a block set contains 500 papers, but only 10 of them have a Researcher ID that refers to one real author, it is not reasonable to say that block sizes of 500 papers generally relate to one author. Machine learning with that data would get the data generating process wrong and may underestimate the number of real authors.

We address this problem by requiring block sets associated with more than 20 papers to have at least two distinct Researcher IDs included. For block sets associated with 20 or fewer papers, one distinct Researcher ID in a block set is sufficient. In this way, we try to mimic the true number of authors in a block set, presuming that a block set size of 21 is a reasonable threshold to indicate block sets that may represent only one real author.



Block	Paper ID 1	Paper ID 2	*Same Researcher ID*	*Same first name*	...	Training
Wang, Z	123	456	1	1	...	1
Wang, Z	123	789	1	missing	...	0
...	...	...	...	...	...	...

Fig. 1. Data processing schema.

In total, our dataset now includes 154,092 papers and allows us to generate paper pairs. However, our data set is imbalanced considerably towards sets of very frequent blocks and sets of single scientists who have published many papers. In the machine learning approach, this could result in a performance bias in favor of productive authors and high computational costs. We accordingly restrict the number of papers that are written by a single author to a random set of 10 papers, resulting in 15,407 papers.

When splitting paper pairs into the test and training sets, we randomly sample by block, assigning 80% of the blocks to the training set and 20% to the test set. Table 1 depicts the descriptive statistics for the preliminary test and training set in comparison to the full WOS dataset. A remarkable characteristic is the number of missing first names. While our datasets have a negligible number of missing first names, the WOS includes up to 95% missing of first names before 2006 (see Fig. 2). As our goal is to provide a robust model that can handle a variety of cases, especially those with missing data, we therefore randomly delete 90% of the first names prior to 2006 in our dataset to mimic the WOS structure as well as possible. After 2006 we include 20% missing first names. We refer to our training and test sets as synthetic test and training sets because of this manipulation.

In the following, we focus on paper pairs as the basis of our analysis. We only use paper-pairs where the first name is either missing or the same. This constraint reduces the number of comparisons since presumably irrelevant comparisons of papers with different first names are left out. There is a considerable difference between the test and training set in the number of paper pairs that can be generated; the training set with its 1.2 million paper pairs is more than 23 times as large as the test set with its 53,501 pairs. Table 2 depicts the most frequent names in the training and test set. In the random sampling of blocks, we assigned by chance

Table 1
Characteristics of the test and training set.

	Train set	Test set	WOS 2017
Paper characteristics			
Number of distinct authors (Researcher IDs)	1,904	381	> 322,686
Number of papers	12,818	2,589	52,055,209
Mean number of papers per Researcher ID	6.73	6.79	59.50
Mean paper year	2008	2008	2004
Mean number of authors per paper	4.96	5.06	4.74
First name characteristics			
Number of papers < 2006	2,991	707	58%
Number of papers > 2006	8,468	1,585	42%
Number of missing first names	360	31	~50%
Number of missing first names <2006	99	12	~90%
Number of missing first names <2006 after generating missing first names	2,846	675	–
Number of missing first names >2006	213	16	~20%
Number of missing first names >2006 after generating missing first names	1,870	318	–
Block characteristics			
Number of blocks	623	156	6,450,190
Mean number of papers per block	20.57	16.60	35.92
Mean number of Researcher IDs in block	3.05	2.44	Unknown
Paper pair characteristics			
No of paper pairs where same block and same first name = MISSING|TRUE	1,252,075	53,501	–
No of paper pairs where same block and same Researcher ID and same first name = MISSING|TRUE	91,949	18,660	–
No of paper pairs where same block and different Researcher ID and same first name = MISSING|TRUE	1,160,126	34,841	–

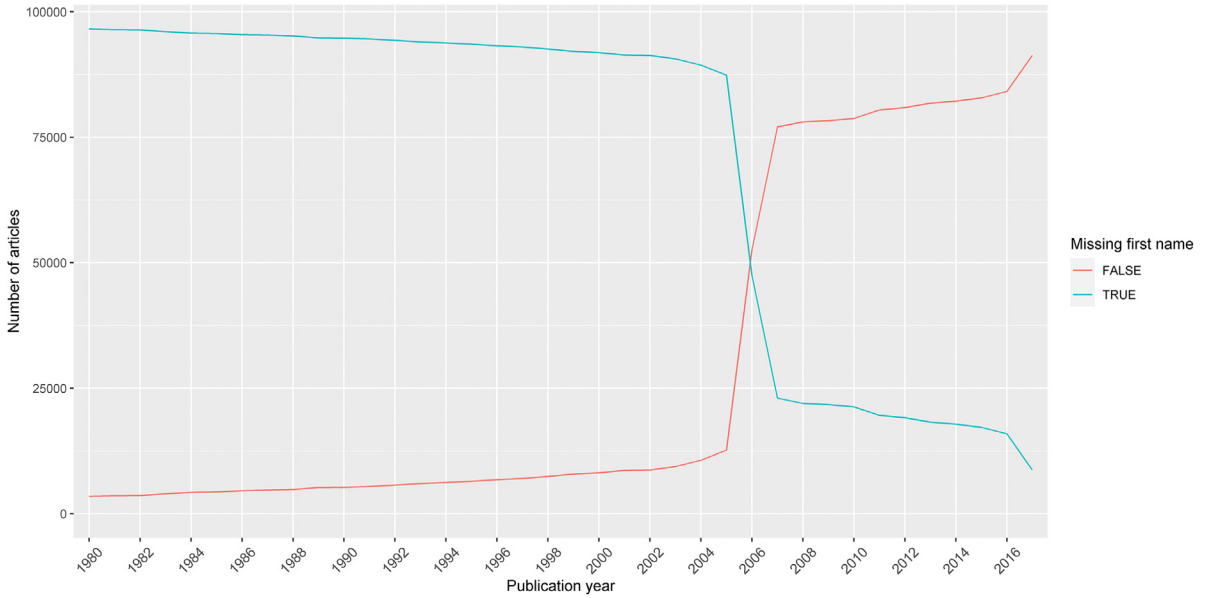


Fig. 2. Number of missing first name in 100 K paper-author sample of WOS.

Table 2
Top 5 blocks by frequency in training and test sets.

Train set					Test set				
Rank	Block	Distinct Researcher IDs in set	No of papers in set	Number of papers in WOS	Rank	Block	Distinct Researcher IDs in set	No of papers in set	Number of papers in WOS
1	Wang, z	218	950	157,519	1	Xie, j	26	100	16,936
2	Chen, g	68	309	54,578	2	Shi, h	17	80	13,270
3	Chen, t	43	189	39,443	3	Choi, d	16	70	11,114
…	…				…				
623	Yakushevich, n	1	1	5	156	Voytenko, v	1	1	10

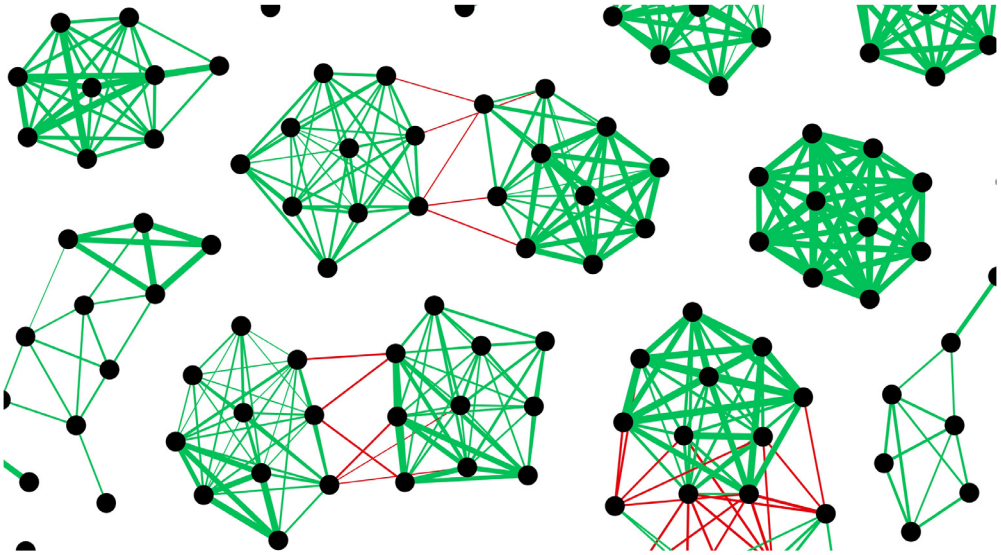


Fig. 3. Illustration of true and false predicted paper pairs

into the training set the very frequent block “Wang, Z.”, which accounts for the dominant number of papers and accordingly paper pairs.

3.2. Feature generation: Name-based features

The next major step is the creation of features that provide input to the machine learning procedure. For name-based features, we use the same approach as Torvik et al. (2005), who calculated first name priors that input their probabilistic approach to author disambiguation, and provide as information to the machine learning algorithms the overall block paper counts (*Block set size*), the overall count of the first name stated on paper 1 and paper 2 in a pair (*First name count 1* and *First name count 2)*, the second first name initials (*Second initial count set 1* and *Second initial count set 2),*the overall last name count *(Last name count*), the first name count within the block paper set (*First name count group 1* and *First name count group 2*) and several ratios[1] of these features.

First name count 1 and *First name count 2* complement the *Block set size* and provide additional, valuable information by indicating whether a given first name is expected to belong to multiple authors within a block. The first name “John”, for instance, can be found in the WOS 67,041 times, while the first name “Soesoe” can be found only five times. The probability that two papers are from the same author should be much higher for the infrequent “Soesoe” than for the frequent “John””. *First name count group 1* and *First name count group 2* allow to analyze this the first name frequency on the block level. If, for instance, in a block set “Doe, J.” with 50 observations, the first name “John” accounts for all 50 observations, we would have either a very productive John Doe or, more likely, two or more different John Does. If “John” accounts for only one observation, and there are 49 other distinct names of block “Doe, J.”, we can be certain that there is only one John Doe. *First name count group 1* and *First name count group 2* account for this and provide the first name’s frequency in the block.

3.3. Feature generation: Country and institution-based characteristics

Our second major group of features comes from regional and institutional information retrieved from papers. The WOS processes the author’s address string and extracts the country. We use this country information in the feature *Same country*. This feature might be especially helpful with “international” blocks, such as “Muller, M.”, which frequently occurs in the US and Germany. We expect that the dominant share of block paper pairs can be found within a single country and face two challenges. The first problem is that the address string is missing in 28% of our papers because the block author is not the corresponding author of the paper, as only those were mandated to fill in their address string on the paper before 2008 (Liu, Hu & Tang, 2018). The second challenge consists of little deviations in the address strings and institutional mobility of researchers. This peculiarity is addressed by using the Jaccard string similarity metric between two address strings (*Institution name similarity)*.

[1] The ratios are: *Block set size / Last name count = Ratio Block Last, First name 1 count set / Last name count = Ratio First Last Group 1, First name 2 count set / Last name count = Ratio First Last Group 2*

3.4. Feature generation: Thematic features: topic modeling

One methodological innovation of our author disambiguation approach is the exploitation of latent thematic information in paper titles and abstracts. We use topic modeling, which is a group of probabilistic methods used to discover the latent semantic structures (topics) in text collections. In topic modeling, documents, such as titles or abstracts, are considered as mixtures over K latent topics, where each topic is again considered to be a distribution over all the words that exist in the collection of documents. We use a correlated topic model as basis (Blei & Lafferty, 2007) and 50,000 random abstracts from WOS to train a model with 89 topics. The next step is the feature generation with this correlated topic model. We apply the topic model to both abstracts and titles in a block and retrieve two topic distributions for each abstract or title. We apply cosine similarity to compare all distribution pairs (Rehs, 2020). The cosine similarity measures the angle between two vectors (the topic distributions of the two abstracts or titles) projected in a multi-dimensional space. Values towards 1 indicate similarity, which allows us to see whether two abstracts or titles are semantically similar by a single number. The cosine similarity for paper pair title distributions is shown by the feature *Thematic similarity title* and for abstracts by *Thematic similarity abstract*. To complement these measures, we additionally calculate the Jaccard distance of the titles in a paper pair (*Jaccard distance title*).

3.5. Feature generation: Thematic features: classifications and keywords

We also exploit thematic information retrieved from subject classifications and self-assigned keywords in the WOS. The WOS usually makes journal-level classifications, which include five broad categories and 252 disciplines. We use the 252 disciplines to create the feature *Same classification*, which checks if two block papers have at least one classification in common. To generate a feature from keywords, we use only author-assigned keywords, as we assume they are of higher quality than automatically generated keywords. We deleted very frequent keywords (count>100.000 in the WOS) from the list of possible keywords because we assume, they don't add to the performance of the feature. Finally, the feature *Same keyword* checks if two papers have at least one common keyword.

3.6. Feature generation: Bibliographic features

Our last set of features concerns bibliographic characteristics of the block paper pairs. The feature *Diff. in number of co-authors* shows differences in co-authorship counts and addresses individual publication behavior or disciplinary differences in co-authorship counts. *Diff. in publication year* should help to disambiguate block authors who are published in different periods of time. *Publication year 1* and *Publication year 2* account for yearly effects, such as the generally lower number of papers and subsequent matching probability in past years. Finally, *Same first name* compares whether two papers share the same first name or whether the information is missing in one of the papers.

Table 3 presents statistics for the synthetic test and training set and is differentiated by authors who have the same block and are either the same or different authors according to the Researcher ID. Variation between these two groups is necessary for the random forest and the logistic regression to find meaningful rules with which the same and different authors can be distinguished. For better arrangement, we separated numeric and factorial features. For all features, we find variation between the two groups of same authors and different authors. Our features are, therefore, generally relevant and should be used in the machine learning procedure.

3.7. Machine learning approach to author disambiguation: Random forest

In our research, we used random forest and logistic regression machine learning algorithms to disambiguate author names. In a pre-test we also tested a 2-layer feed-forward neural network, which delivered comparable results, but does not allow feature assessment.

Breiman (2001) proposed the use of random forests as a method for classification and regression tasks. A random forest consists of an ensemble of decision trees, each of which uses a random sample of the data and features to find a given tree's best split. By introducing random sampling of data and features, random forests generally avoid overfitting the training data. Random forests can be applied in both supervised and unsupervised approaches. One of the major advantages of random forests is their ability to evaluate feature performance.

We use a supervised implementation of a random forest model with 200 trees and tune the hyperparameter *mtry*, which is the number of random features tried at each split of a tree. We tune *mtry* by incremental increases and choose the value with the best out-of-bag error. The out-of-bag error exploits that not all trees use every observation in training. In this way, the out-of-bag error measures the mean prediction error for all samples not included in the predictions. The out-of-bag error, therefore, allows validation of the performance without using an external test set. We obtain the best result with a *mtry* value of 10. The "winning" class for an observation in a random forest is the one with the maximum ratio of the proportion of votes to cutoff. To find the best trade-off between precision and recall, we also tuned this cutoff value and find 0.8 proportion of trees that vote for two papers in pair of being from the same author as the best cutoff value.

Finally, random forest performance can suffer from large class imbalances in the training data (Kim & Kim, 2018). Here, inference of the majority class is easier since there are more examples of how best to split a decision tree. In our case, the negative class (same block and different person) is prevalent and accordingly results in a biased model. To address this issue, we sample in each tree as many positive training examples as negative examples. Because we could build only trees that take at maximum 107,002

Table 3
Descriptive statistics features synthetic test and training set.

	Synthetic Train Set (1,252,075 paper pairs)						Synthetic Test Set (53,501 paper pairs)					
	Same Researcher ID pairs (91,949 pairs)			Different Researcher ID pairs (1,160,126 pairs)			Same Researcher ID pairs (18,660 pairs)			Different Researcher ID pairs (34,841 pairs)		
	Miss.	Mean	Std. dev.	Miss.	Mean	Std. dev.	Miss.	Mean	Std. dev.	Miss.	Mean	Std. dev.
Numeric features												
Jaccard distance title		0.76	0.31		0.94	0.03		0.76	0.32		0.94	0.03
Thematic similarity title	171	0.30	0.36		0.12	0.17	156	0.31	0.37	47	0.14	0.21
Thematic similarity abstract	17,780	0.28	0.36	192,519	0.12	0.18	3419	0.31	0.38	6,327	0.13	0.21
Institution name similarity		0.39	0.42		0.07	0.08		0.37	0.41		0.05	0.09
Diff. in publication year		3.24	4.34		5.61	4.80		3.18	4.26		7.10	5.77
Publication year 1		2007	6.45		2008	5.74		2007	6.48		2005	6.77
Diff. in author position		1.33	1.56		1.76	1.66		1.30	1.54		1.79	1.67
Diff. in number of co-authors		5.02	2.26		5.01	2.20		4.98	2.26		4.91	2.19
Diff in number of citations		10.22	43.23		9.39	22.13		9.28	30.11		8.79	29.95
First name count 1		20,814	42,852		11,388	12,317		20,889	45,460		13,379	25,063
First name 1 count set		2,227	14,641		6,889	29,383		441	1,873		955	2,926
Second initial 1 count set		8,977	28,755		36,511	57,593		1,792	3,406		3,481	56,306
Last name count		472,497	808,817		1,866,321	727,629		59,306	79,688		127,093	93,408
Block set size		28,276	53,081		123,568	57,499		3,995	5,007		8,489	6,183
Ratio Block Last		0.12	0.15		0.06	0.03		0.16	0.19		0.10	0.12
Ratio First Last group 1		43.99	957.15		1.89	4.97		5.74	49.99		0.12	0.32
...												
	Different	Missing	Same	Different	Missing	Same	Different	Missing	Same	Different	Missing	Same
Factorial features												
Same first name		42,581	49,368		1,119,544	40,582		9,115	9,545		32,913	1,928
Second initial	261	67,854	23,834	201,707	941,260	17,159		14,783	3,877	2,442	31,866	533
Same country	5,140	37,632	49,177	410,418	493,555	256,353	1,014	8,208	9,438	8,746	21,039	5,056
Same keyword		84,745	7204		1,160,124	2		1,7238	1,422		34,841	0
Same classification		63,771	28,178		1,147,807	12,319		12,969	5,691		34,152	689
Same co-authors		77,881	14,068		1,160,100	26		15,832	2,828		34,840	1
...												

Table 4
Pairwise prediction results synthetic test set.

	Random forest	Logistic regression
Paper pairs	53,501	53,501
True negative	33,531	34,233
False negative	4,664	7,047
False positive	1,310	806
True positive	13,996	11,613
Pairwise precision	0.91	0.95
Pairwise recall	0.75	0.62
Pairwise F1	0.82	0.75
Classification threshold	0.80	0.80

Note: Random forest parameters: *mtry* = 11, stratified sampling by block set classes, 200 trees.

observations (two times 53,501 Same Researcher IDs in the synthetic training set) into their inference procedure, which is less than 10% of the total dataset, we may undersample some small but important sub-feature classes. To make our model robust, we would need in each tree observations that refer to small, medium, and large blocks. Large blocks with more than 10,000 papers, however, account for 95% of the observations; medium sets from 500 to 9,999 papers, for 3%; and small sets of fewer than 500 papers, for the remaining 2%. Accordingly, only a very small number of observations in each sample can be drawn from medium and small paper sets. Therefore, the tree would most likely not learn meaningful rules related to those sets. Because this repeats in every tree, our random forest model would generally underperform on small and medium sets. We address this issue by implementing a stratified sampling strategy for each decision tree. We establish five block-size groups in this strategy and sample 6,000 positive and 6,000 negative paper-pairs from each group.[2]

3.8. Machine learning approach to author disambiguation: Logistic regression

In addition to the random forest model,[3] we also train a logistic regression classifier with most of the same features. Logistic regression is based on different concepts and therefore might arrive at different conclusions on feature relevance, and also different predictions. Unlike random forest, logistic regression estimates the explicit probability that two papers are from the same author. It uses a logistic function to find the model parameters via maximum likelihood estimation. For multicollinearity issues in this estimation, we cannot use highly correlated variables, such as the *Last name count* and *Block set size* in the same model. We identify these problematic correlations by running the model and then calculating each variable's variance inflation factor. A variance inflation factor greater than five generally indicates collinearity. We remove these variables and estimate the model again. The full model can be found in Appendix A along with the estimation of average marginal effects.

3.9. Machine learning approach to author disambiguation: results

Table 4 presents the results of the random forest and the logistic regression. The random forest algorithm delivers precision and recall rates of >0.75 values, beating the logistic regression in precision but not in recall. Table 5 depicts the importance of the features, as shown by the mean decrease in accuracy for the predicted class *Same author = T* and *Same author = F*. In other words, how the number of correct classifications decreases for the given class when the feature is excluded. The column Mean Decrease Accuracy summarizes this measure for both classes. The final column, Mean Decrease in Gini, indicates the average of a given variable's total decrease in node impurity, weighted by the proportion of sampled observations reaching that node in each decision tree. This measures how important a variable is for estimating whether two papers are from the same author for all of the trees that make up the forest. A higher Mean Decrease in Gini indicates higher variable importance.

Same first name is the most powerful feature for the Mean Decrease in Gini. For Mean Decrease Accuracy where *Same author = F* and *Same author = T*, the *Jaccard distance title* and *Same classification* are the most important features. The performance of the *Jaccard distance title* can be explained by its universal applicability, as every paper must have a title. Although the *Jaccard distance title* was intended to complement the cosine similarity of the title and abstract, it outperforms both measures substantially. However, the insignificance of the cosine similarity of title and abstract and other features should not be overemphasized, as there might often be a high correlation between the randomly drawn features in each tree. Random forests can capture this correlation and therefore use only the most powerful features when splitting a node. Highly correlated variables appear irrelevant although they might be only slightly worse predictors.

It is noteworthy that *Same co-authors* and *Same keyword* do not play a role, and the same keyword even impacts average accuracy negatively, perhaps because of our stratified sampling strategy. Since there are only 7206 observations with the *Same keyword* and

[2] For the smallest group of block sets with less than 100 papers, we can sample only 500 positive and negative observations.

[3] Other machine learning models, such as neural nets, Naïve Bayes classifiers and SVMs were also considered, but they do not fulfil at least two criteria of: predictive power, possibility of feature assessment and training time and scalability (prediction time).

Table 5
Feature importance random forest.

	Mean decrease accuracy Same author=*T*	Mean decrease accuracy Same author=*F*	Mean decrease accuracy	Mean decrease Gini
Same first name	26.16	37.29	38.16	8130.88
Institution name similarity	70.59	62.38	73.30	7266.78
Jaccard distance title	192.54	121.09	155.86	7100.02
Ratio First Last group 1	55.18	28.77	58.80	3281.55
Same classification	138.81	102.91	129.61	2494.70
First name count group 1	25.02	18.92	20.94	2342.46
Ratio First Last group 2	44.91	32.41	48.27	2130.46
Same second initial	89.79	26.24	38.00	1958.98
First name count group 2	69.88	39.7	45.05	1478.85
Diff. in number of citations	93.75	33.83	38.47	1333.61
Diff. in publication year	71.52	35.12	39.38	1327.15
Publication year 2	63.92	28.26	33.49	1224.83
Second initial count set 1	42.04	33.46	40.12	1202.55
Ratio Block Last	39.17	27.96	29.64	1193.52
Block set size	23.10	18.86	21.16	1129.82
Last name count	23.04	20.88	22.59	1096.66
Second initial count set 2	37.73	18.74	22.29	1021.48
Publication year 1	50.11	21.86	26.06	983.34
First name count 2	16.04	12.94	14.72	965.83
Thematic similarity abstract	63.43	25.07	31.16	953.04
Thematic similarity title	62.51	24.50	30.81	933.60
Same country	28.21	16.92	23.56	857.13
First name count 1	15.17	11.92	13.82	821.16
Diff. in number of co-authors	81.70	27.25	41.36	696.53
Diff. in author position	72.50	28.03	34.83	556.80
Same co-authors	7.55	14.05	14.18	11.25
Same keyword	−3.81	6.75	−0.35	0.76

Note: Ordered by the mean decrease in Gini.

14,094 with the *Same co-author*, it is unlikely that enough of those observations are sampled in each tree, and subsequently the trees cannot build meaningful splitting rules.

The logistic regression significantly outperforms the random forest in identifying true negative cases and false-positive cases. Assessing the marginal effects in Appendix A, we again can observe that the first name plays a major role in determining whether two papers are from the same author. As indicated by the average marginal effects (sample) column of Appendix A, the average effect of the *Same first name* on matching probability is about 8 pp. Other relevant features are *Jaccard distance title* (−37 pp), *Same second initial* = *T* (11 pp) and *Institution name similarity* (13 pp). Unlike with the random forest, the feature *Same co-authors* plays a role in the logistic regression model, and its effect is about 12 pp. This is likely because the logistic regression utilizes all observations where *Same co-authors* = *T* when estimating the model. As explained above, random forests may suffer from sampling not sufficiently enough observations where *Same co-authors* = *T* in each tree.

3.10. Graph-based author community detection

At this point, we cannot draw author-paper sets from the predicted paper pairs; they must first be aggregated in order to show all papers belonging to an author. This can be done by graph-based community detection, where papers represent nodes and edges the predicted matches of our machine learning approach. Edges can be weighted by their attributes, such as *Same first name* or *Same second initial*. We determine edge weights by a score that adds one point to an edge weight when both papers have the same first name, country, classification and second initial; makes no change if the first name/country/… information is missing; and subtracts one when the two papers have different first names/countries/… . We also sum up the given values for the *Jaccard distance title, Thematic similarity title, Thematic similarity abstracts* and *Institutional distance* and add them to the score. Finally, we consider the time difference between two papers in the edge weighting by adding the time difference's reciprocal value.

Figure 3 shows an illustration of the exemplary graph communities. Red lines indicate false predictions and line thickness, the edge weight. Although many cohesive communities represent all true papers from a single author, the small number of erroneously assigned paper pairs connects different communities and therefore misclassifies all papers in both communities. Algorithms that can detect incorrect linkages are referred to as community detection methods and are well described in graph theory (Newman, 2018).

In the following section, we test the infomap algorithm (Rosvall & Bergstrom, 2007). The basic idea of the infomap algorithm is that random walks from a given node are more likely to stay within the same community rather than leave the community. This, however, only holds true when the community is cohesive, when there are many connections within the community and wrong predictions to other communities are uncommon. The included edge weights support the infomap clustering because they indicate which edges are most likely to go for the random walk. In order to provide a reference to the infomap algorithm, we assume that all connected paper pairs belong to the same author. Results for this reference clustering are depicted in Table 6.

Table 6
Cluster algorithm results synthetic test set.

	Random forest		Logistic Regression	
	Connected components (reference)	Infomap	Connected components (reference)	Infomap
Number of papers	2,589	2,589	2,589	2,589
Number of clusters	385	484	594	636
Mean number of papers per cluster	6.72	5.34	4.39	4.07
ACP	0.6487 (0.26)	0.9379 (0.09)	0.8141 (0.12)	0.7763 (0.14)
APP	0.9208 (0.08)	0.8546 (0.11)	0.9108 (0.14)	0.9580 (0.10)
K-Metric	0.7496 (0.16)	0.8915 (0.07)	0.8529 (0.09)	0.8561 (0.08)

Note: Standard deviation in parentheses.

To evaluate the clusters, we use the average cluster purity (ACP), average author purity (AAP) and the K-Metric. ACP, AAP and K-Metric are frequently used cluster metrics in author disambiguation (Kim, 2019). ACP is defined by Eq. (1) and evaluates the purity of author clusters on the block level. An ACP of 1 indicates if each of the calculated clusters contain only papers of the same Researcher ID. The AAP evaluates the fragmentation of the calculated clusters and analyzes if the papers of a given Researcher ID can be found in different clusters (Ferreira, Veloso, Gonçalves & Laender, 2014). AAP is defined by Eq. (2). An AAP value of 1 indicates no fragmentation of Researcher IDs into different clusters. K-metric is defined as the geometric mean of ACP and AAP and is defined by Eq. (3).

$$ACP = \frac{1}{N} \sum_{i=1}^{e} \sum_{j=1}^{t} \frac{n_{ij}^2}{nj} \tag{1}$$

$$ACP = \frac{1}{N} \sum_{i=1}^{t} \sum_{j=1}^{e} \frac{n_{ij}^2}{nj} \tag{2}$$

$$K = \sqrt{ACP * AAP} \tag{3}$$

In Eqs. (1) and (2), N is the number of papers in a block, t is the number of distinct Researcher IDs in a block, e is the number of detected clusters in a block and n_i is the number of papers in detected cluster i. Finally, n_{ij} is the total number of papers in the detected cluster i, which also belong to the Researcher ID j.

The clustering results shown in Table 6 indicate that, after the application of the infomap algorithm, both approaches suffer from an increase in fragmentation. Clusters are more often divided into different sub-clusters. At the same time, the new clusters contain, on average, fewer different Researcher IDs as compared to the reference clustering approach. The K-Metric, which combines both measures, suggests two different patterns. Only the random forest predictions profit from the application of the infomap algorithm. The logistic regression clusters are, on average, of the same quality when using the reference clustering of connected components.

3.11. Application to full block "Muller, M"

The test of our algorithm on real-world data is our final step. We aim to demonstrate the applicability and predictive performance using the typically German block "Muller, M." This block takes the 572nd rank of blocks in the WOS and accounts for 11,655 papers that have fewer than 10 co-authors. We process these papers and generate more than 88 million paper pairs where one or both first names are missing in a pair or where the first names are the same. We apply the random forest model and the logistic regression to the paper pairs in the next step. The processing takes 2,700 s in the random forest and 80 s in the logistic regression. Both calculations have been done in R and on one core of an Intel Xeon Bronze 3204.

In the next step, we build the graph and apply the infomap community detection. The infomap takes about 30 s to process the predictions from the random forest and the logistic regression; it delivers 1,136 clusters in the random forest and 723 clusters in the logistic regression. We measure the quality of these clusters by evaluating the ACP, AAP and K-Metric of the 851 papers that have a Researcher ID. Table 7 shows the results. For both approaches, we find perfect author cluster purity, indicating that the clusters contain only papers of the same Researcher ID. The AAP, which describes the fragmentation of a Researcher ID into different clusters, is 0.61 in the random forest and 0.66 in the logistic regression. Compared to the test set results, both algorithms perform worse in the AAP measure and create too many clusters. The K-Metric, which reports the geometric mean of AAP and ACP, decreases significantly as compared to the test set results.

4. Discussion and conclusion

In this paper, we developed a supervised machine learning approach to author disambiguation in large publication databases. We used 12,818 publications from 1,904 scientists with Researcher ID available in the WOS and compared pairwise all authors and papers that have the same last name–first initial block. In this comparison, we exploited bibliographic, country, institution, thematic and author name-based characteristics of papers to train a random forest and a logistic regression classifier. We applied the model to an unseen test set of 2,589 publications and obtained pairwise F1 scores of over 0.75. In our post-estimation, we used graph-based

Table 7
Application: Cluster results for block "Muller, M".

	Random forest	Logistic regression
Number of papers	11,655	11,655
Number of paper pairs	88,164,593	88,164,593
Number of clusters	1,136	723
Mean number of papers per clusters	10.25 (32.87)	16.12 (44.13)
Papers with Researcher ID		
Number of papers with Researcher ID	851	851
Number of different Researcher IDs	27	27
ACP	1	1
AAP	0.61	0.66
K-Metric	0.78	0.81

Note: Standard deviation in parentheses.

clustering to combine predictions to find all papers belonging to a single author. To cancel incorrect predictions from the logistic regression and the random forest, we applied infomap community detection, thereby significantly improving our results. Finally, we also applied the model on the full block "Muller, M." with 11,655 papers in the WOS. We evaluated this application by analyzing the clustering for papers that have a Researcher ID and find perfect ACP and acceptable APP cluster metrics.

Besides some exceptions, our approach does not yield the same high disambiguation performance as other current approaches (D'Angelo & van Eck, 2020; Tekles & Bornmann, 2020). There are several reasons for this, which can broadly be structured into data source and training and test set design, blocking approach, feature engineering, machine learning method and algorithms, and clustering method.

4.1. Data source and training and test set design

Our choice of test and training data allowed us to collect a rich dataset of true authorship information, which we manipulated to mimic the WOS as well as possible. This manipulation may not have been completely accurate, as we made some important assumptions about the data-generating process of the WOS. We had no information about the true number of authors in a block paper set and assumed that the number of true authors is reflected in the block set size. While this assumption may hold true for medium and large block sets, it may not for small block sets, requiring further refinement of our approach. For example, if an author with a very atypical name published 100 papers, this author would not have been included in our sample because we required block sets with more than 20 papers to have at least two different Researcher IDs, and therefore two different real authors. Therefore, we missed an important aspect of the true data-generating process.

We also had to make restrictions on the number of sampled papers per Researcher ID in order to keep the computational effort manageable. This is problematic in that a block set consisting of 100 papers and two authors, where author 1 has 90 papers and author 2 has 10 papers, would result in exactly 10 papers sampled for each author. Therefore, the resulting prior matching probability for a pair of papers of this set would be lower and again not reflect the true data-generating process.

More generally, the difficulties explained above are inherent to supervised machine learning in author name disambiguation. Smalheiser and Torvik (2009) indicate that any supervised approach is dependent on adequate, reliable, and representative training data. In comparison to other approaches, our training and test set design was more complex and may more accurately represent the true data-generating process. Especially the deletion of first names and second initials is an especially uncommon methodical innovation and makes a direct comparison of performance to other approaches difficult. For instance, Tekles and Bormann (2020), who provide an extensive comparison of several unsupervised approaches, did not include missing data in their similar test set based on WOS Researcher IDs. Their implementation of four different approaches may therefore have achieved higher performance in terms of (pairwise) F1 and related measures than our approach did.

One solution to these problems is manually disambiguating full block sets and using this information as training data. As described in the introduction, this is a major effort, likely includes human-coder error and is not feasible for large block sets of thousands of papers. However, these large sets are needed to design training data that resembles the true data-generating processes of the WOS and other publication platforms. Therefore, a promising direction for further research might be pre-structuring the manual disambiguation process to save resources and time. In this pre-structuring, our algorithm could be applied to publication data that does not contain identifier information. As the logistic regression predictions can be interpreted as probabilities, predictions that are close to our cutoff values can be filtered out and disambiguated manually, as they are likely more prone to errors.

4.2. Blocking approach

Finally, our approach's major drawback is the difficulty in appropriately handling synonyms and author name changes. Synonym issues are especially common for German names that include umlauts, which are either not included in English character sets or were inconsistently transformed by the authors. For instance, the German "Müller" is sometimes written "Muller" or "Mueller". Currently, both names are, for the sake of simplicity, treated as different blocks in our approach. In the future, this and similar problems could be addressed by the approaches proposed in Backes (2018) and Hussain and Asghar (2018a).

Appendix A
Logistic regression model and average marginal effects (AME).

Variable Same first name (reference = missing)	Coefficient	AME (sample)
Same first name = *T*	2.353***	0.080***
	(0.02)	(0.00)
Same class =*T* (reference = *F*)	2.482***	0.092***
	(0.03)	(0.00)
Diff. in publication year	−0.057***	−0.001***
	(0.00)	(0.00)
Diff. in number of citations	−0.000*	−0.000*
	(0.00)	(0.00)
Thematic similarity title	0.539***	0.011***
	(0.03)	(0.00)
Thematic similarity abstract	0.719***	0.015***
	(0.03)	(0.00)
Jaccard distance title	−18.038***	−0.372***
	(0.15)	(0.00)
Institution name similarity	6.427***	0.133***
	(0.05)	(0.00)
Diff. in number of co-authors	−0.093***	−0.002***
	(0.003)	(0.00)
First name count 1	−0.000***	−0.000***
	(0.00)	(0.00)
Block set size	0.000***	0.000
	(0.00)	(0.00)
First name count group 1	−0.000***	−0.000***
	(0.00)	(0.00)
Same keyword	−1.135	−0.018
	(0.96)	(0.01)
Ratio Block Last	0.748***	0.015***
	(0.06)	(0.00)
Same second initial (reference = *F*)		
Same second initial = missing	2.837***	0.035***
	(0.07)	(0.00)
Same second initial = *T*	4.805***	0.105***
	(0.07)	(0.00)
Same country (reference = *F*)		
Same country = missing	1.563***	0.030***
	(0.02)	(0.00)
Same country = *T*	0.338***	0.005***
	(0.02)	(0.00)
Same co-authors	2.968**	0.123***
	(0.25)	(0.00)
Author position difference	0.028***	0.001***
	(0.00)	(0.00)
Publication year 1	−0.053***	−0.001***
	(0.00)	(0.00)
Publication year 2	−0.053***	−0.001***
	(0.00)	(0.00)
Second initial count set 1	0.000***	0.000***
	(0.00)	(0.00)
Intercept	224.77***	
	(3.21)	

Note: Standard Errors in parentheses; significance levels:.
* $p<0.01$.
** $p<0.05$.
*** $p<0.01$. AME were calculated for a random sample of 10,000 observations.

4.3. Feature engineering

The feature engineering and assessment is one of our major achievements and adds to the literature gap on feature importance in author name disambiguation (Tekles & Bornmann, 2020). We decided not to use citation-based features since they require high computational effort and therefore don't allow us to apply the algorithms to vast sets of publication data in reasonable amounts of time. Many of our generated features are sparse, such as *Same co-authors*, or suffer from missing data, such as *Thematic similarity abstract*. This was not the case with first names. We therefore intentionally deleted 90% of first names and second initials before 2006 and 20% after 2006 to get a quota of missing data similar to that in the full WOS.

Despite their artificially created, missing observations, first names are one of the most decisive features in the logistic regression and the random forest approaches and confirm the previously discovered relevance of this feature (Torvik & Smalheiser, 2009). Some of our newly designed features, like the thematic similarity of abstracts and titles, did not lead to significant improvement and in fact underperformed compared to similar and simpler features, like the *Jaccard distance title*. Finally, the name-based features, like *First name count group 1* and *Block set size,* performed well and were the most important group of features. Our research therefore has extended the current literature on name and block frequency-dependent features (Tekles & Bornmann, 2020; Torvik & Smalheiser, 2009).

4.4. Machine learning method and algorithms

Our stratified sampling strategy in random forest was new to the literature and adds to previous applications of random forest (Louppe et al., 2016). With respect to the logistic regression, our estimation of average marginal effects is new to logistic regression applications in author name disambiguation (Gurney et al., 2012) and eases the feature importance interpretation. The random forest performed better than logistic regression in the F1 measure. This advantage, however, became smaller when community detection methods were applied. This may be because the logistic regression was more restrictive on positive classifications. This is beneficial in the later infomap community detection because it incurs fewer incorrect linkages between different author clusters, which must be canceled, but enough correct linkages between papers of the same author. The results of the community detection were not considered when tuning the random forest and the logistic regression. It is therefore uncertain if random forest disambiguation performance can be improved in more highly tuned machine learning approaches. Tuning the sampling strategy of the random forest may cause other improvements in performance.

4.5. Clustering algorithm

The infomap algorithm, in combination with edge weighting, was a decisive post-estimation technique in author name disambiguation and considerably improved performance. However, the use of a very different dataset and evaluation metric does not allow for comparison of our approach to the post-estimation clustering of Gurney et al. (2012). A future venue of research related to our proposed method could consist in the systemic evaluation and tuning of edge weighting and infomap parameters.

4.6. Applicability

Our application to a full block ("Muller, M.") in the WOS was successful. By evaluating the subset of papers with Researcher ID, we find perfect ACP and acceptable APP for both algorithms' clustered predictions. In comparison to other approaches tested by Tekles and Bornmann (2020) at varying block sizes, our results upon the large set "Muller, M." are promising. While the disambiguation quality (as measured by precision, recall, and F1) of other approaches quickly deteriorates as block sizes increase above four digits, our quality (as measured by ACP and APP) remains at perfect or acceptable levels.

The short processing durations of the 87 million comparisons in the block "Muller, M." on a small-scale server system suggest our approach's scalability. Therefore, our approach can be applied to large amounts of publication data such as the full WOS database in reasonable amounts of time. The results of such large-scale application may then be evaluated with the numerous remaining Researcher IDs that were not used in this paper. Other datasets, such as SCOPUS, DBLP or PubMed, may be similarly appropriate for testing since they include reliable identifiers other than the Researcher ID, like Orcid and PubMed IDs.

Finally, we want to emphasize the need for a high-quality disambiguated publication database. Our approach can help overcome this challenge and potentially improve bibliometric insights and subsequent science and technology policy recommendations.

Author statement

The paper has been completely carried out by Andreas Rehs.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.joi.2021.101166.

References

Abbott, A., Cyranoski, D., Jones, N., Maher, B., Schiermeier, Q., & Van Noorden, R. (2010). Metrics: Do metrics matter? *Nature, 465*, 860–862. 10.1038/465860a.

Abdulhayoglu, M. A., & Thijs, B. (2017). Use of ResearchGate and Google CSE for author name disambiguation. *Scientometrics, 111*(3), 1965–1985. 10.1007/s11192-017-2341-y.

Backes, T. (2018). The impact of name-matching and blocking on Au-thor disambiguation. In *ACM conference on information and knowledge management* (pp. 803–812). 10.1145/3269206.3271699.

Blei, D. M., & Lafferty, J. D. (2007). A correlated topic model of science. *The Annals of Applied Statistics, 1*(1), 17–35. 10.1214/07-AOAS114.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32. 10.1023/A:1010933404324.

Caron, E., & van Eck, N. J. (2014). Large scale author name disambiguation using rule-based scoring and clustering. In *Proceedings of the science and technology indicators conference 2014 Leiden.*

D'Angelo, C. A., & van Eck, N. J. (2020). Collecting large-scale publication data at the level of individual researchers: A practical proposal for author name disambiguation. *Scientometrics, 123*(2), 883–907. 10.1007/s11192-020-03410-y.

De Carvalho, A., Ferreira, A. A., Laender, A. H. F., & Gonçalves, M. A. (2011). Incremental unsupervised name disambiguation in cleaned digital libraries. *Journal of Information and Data Management, 2*(3) 289-289. Retrieved from https://periodicos.ufmg.br/index.php/jidm/article/view/128.

Enserink, M. (2009). Scientific publishing. Are you ready to become a number? *Science, 323*(5922), 1662–1664. 10.1126/science.323.5922.1662.

Fan, X., Wang, J., Pu, X., Zhou, L., & Lv, B. (2011). On graph-based name disambiguation. *Journal of Data and Information Quality, 2*(2), 1–23. 10.1145/1891879.1891883.

Ferreira, A. A., Gonçalves, M. A., & Laender, A. H. F. (2012). A brief survey of automatic methods for author name disambiguation. *SIGMOD Record, 41*, 15–26. 10.1145/2350036.2350040.

Ferreira, A. A., Veloso, A., Gonçalves, M. A., & Laender, A. H. F. (2014). Self-training author name disambiguation for information scarce scenarios. *Journal of the Association for Information Science and Technology, 65*(6), 1257–1278. 10.1002/asi.22992.

Glänzel, W., & Schubert, A. (2006). Analysing scientific networks through co-authorship. In H. F. Moed, U. Schmoch, & W. Glänzel (Eds.), *Handbook of quantitative science and technology research* (pp. 257–276). 10.1007/1-4020-2755-9_12.

Gurney, T., Horlings, E., & van den Besselaar, P. (2012). Author disambiguation using multi-aspect similarity indicators. *Scientometrics, 91*(2), 435–449. 10.1007/s11192-011-0589-1.

Hicks, D. (2012). Performance-based university research funding systems. *Research Policy, 41*(2), 251–261. 10.1016/j.respol.2011.09.007.

Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. *Proceedings of the national academy of sciences of the United States of America, 102*(46), 16569–16572. 10.1073/pnas.0507655102.

Hussain, I., & Asghar, S. (2017). A survey of author name disambiguation techniques. *The Knowledge Engineering Review, 32*, 2010–2016. 10.1017/s0269888917000182.

Hussain, I., & Asghar, S. (2018a). Author name disambiguation by exploiting graph structural clustering and hybrid similarity. *Arabian Journal for Science and Engineering, 43*(12), 7421–7437. 10.1007/s13369-018-3099-0.

Hussain, I., & Asghar, S. (2018b). DISC: Disambiguating homonyms using graph structural clustering. *Journal of Information Science, 44*(6), 830–847. 10.1177/0165551518761011.

Kim, J. (2019). A fast and integrative algorithm for clustering performance evaluation in author name disambiguation. *Scientometrics, 120*(2), 661–681. 10.1007/s11192-019-03143-7.

Kim, J., & Kim, J. (2018). The impact of imbalanced training data on machine learning for author name disambiguation. *Scientometrics, 117*(1), 511–526. 10.1007/s11192-018-2865-9.

Kim, K., Rohatgi, S., & Lee Giles, C. (2019). Hybrid deep pairwise classification for author name disambiguation. In *International conference on information and knowledge management, proceedings* (pp. 2369–2372). 10.1145/3357384.3358153.

Lancichinetti, A., & Fortunato, S. (2009). Community detection algorithms: A comparative analysis. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 80*(5). 10.1103/PhysRevE.80.056117.

Liu, W., Hu, G., & Tang, L. (2018). Missing author address information in web of science - An explorative study. *Journal of Informetrics, 12*(3), 985–997. 10.1016/j.joi.2018.07.008.

Louppe, G., Al-Natsheh, H. T., Susik, M., & Maguire, E. J. (2016). Ethnicity sensitive author disambiguation using semi-supervised learning. *Communications in Computer and Information Science, 649*, 272–287. 10.1007/978-3-319-45880-9_21.

Newman, M. (2018). Networks *(2)*. Oxford University Press.

On, B. W., Lee, I., & Lee, D. (2012). Scalable clustering methods for the name disambiguation problem. *Knowledge and Information Systems, 31*(1), 129–151. 10.1007/s10115-011-0397-1.

Onodera, N., Iwasawa, M., Midorikawa, N., Yoshikane, F., Amano, K., Ootani, Y., et al. (2011). A method for eliminating articles by homonymous authors from the large number of articles retrieved by author search. *Journal of the American Society for Information Science and Technology, 62*(4), 677–690. 10.1002/asi.21491.

Qiao, Z., Du, Y., Fu, Y., Wang, P., & Zhou, Y. (2019). Unsupervised author disambiguation using heterogeneous graph convolutional network embedding. In *Proceedings - 2019 IEEE international conference on big data, Big Data 2019* (pp. 910–919). 10.1109/BigData47090.2019.9005458.

Rehs, A. (2020). A structural topic model approach to scientific reorientation of economics and chemistry after German reunification. *Scientometrics, 125*(2), 1229–1251. 10.1007/s11192-020-03640-0.

Rimmert, C., Schwechheimer, H., & Winterhager, M. (2017). Disambiguation of author addresses in bibliometric databases-technical report. Retrieved from https://pub.uni-bielefeld.de/download/2914944/2914947/DisambiguationOfAuthorAddressesInBibliometricDatabases.pdf.

Rosvall, M., & Bergstrom, C. T. (2007). Maps of random walks on complex networks reveal community structure. *Proceedings of the national academy of sciences of the United States of America, 105*(4), 1118–1123. 10.1073/pnas.0706851105.

Shin, D., Kim, T., Choi, J., & Kim, J. (2014). Author name disambiguation using a graph model with node splitting and merging based on bibliographic information. *Scientometrics, 100*(1), 15–50. 10.1007/s11192-014-1289-4.

Smalheiser, N. R., & Torvik, V. I. (2009). Author name disambiguation. *Annual Review of Information Science and Technology, 43*, 1–43. 10.1002/aris.2009.1440430113.

Strotmann, A., & Zhao, D. (2012). Author name disambiguation: What difference does it make in author-based citation analysis? *Journal of the American Society for Information Science and Technology, 63*(9), 1820–1833. 10.1002/asi.22695.

Talburt, J. R. (2011). Entity resolution and information quality. *Entity Resolution and Information Quality*. Elsevier 10.1016/C2009-0-63396-1.

Tang, J., Fong, A. C. M., Wang, B., & Zhang, J. (2011). A unified probabilistic framework for name disambiguation in digital library. *IEEE Transactions on Knowledge and Data Engineering, 24*(6), 975–987. 10.1109/TKDE.2011.13.

Tang, L., & Walsh, J. P. (2010). Bibliometric fingerprints: Name disambiguation based on approximate structure equivalence of cognitive maps. *Scientometrics, 84*(3), 763–784. 10.1007/s11192-010-0196-6.

Tekles, A., & Bornmann, L. (2020). Author name disambiguation of bibliometric data: A comparison of several unsupervised approaches1. *Quantitative Science Studies, 1*(4), 1510–1528. 10.1162/qss_a_00081.

Torvik, V. I., & Smalheiser, N. R. (2009). Author name disambiguation in MEDLINE. *ACM Transactions on Knowledge Discovery from Data, 3*(3). 10.1145/1552303.1552304.

Torvik, V. I., Weeber, M., Swanson, D. R., & Smalheiser, N. R. (2005). A probabilistic similarity metric for Medline records: A model for author name disambiguation. *Journal of the American Society for Information Science and Technology, 56*(2), 140–158. 10.1002/asi.20105.

Treeratpituk, P., & Giles, C. L. (2009). Disambiguating authors in academic publications using random forests. In *Proceedings of the ACM/IEEE joint conference on digital libraries* (pp. 39–48). 10.1145/1555400.1555408.

Wang, J., Berzins, K., Hicks, D., Melkers, J., Xiao, F., & Pinheiro, D. (2012). A boosted-trees method for name disambiguation. *Scientometrics, 93*(2), 391–411. 10.1007/s11192-012-0681-1.

Wang, X., Tang, J., Cheng, H., & Yu, P. S. (2011). *ADANA: Active name disambiguation*. 10.1109/ICDM.2011.19.

Weingart, P. (2005). Impact of bibliometrics upon the science system: Inadvertent consequences? *Scientometrics, 62*(1), 117–131. 10.1007/s11192-005-0007-7.

Wu, H., Li, B., Pei, Y., & He, J. (2014). Unsupervised author disambiguation using Dempster–Shafer theory. *Scientometrics, 101*(3), 1955–1972. 10.1007/s11192-014-1283-x.