



Disambiguation of author entities in ADS using supervised learning and graph theory methods

Helena Mihaljević¹ · Lucía Santamaría²

Received: 25 May 2020 / Accepted: 12 January 2021
© The Author(s) 2021

Abstract

Disambiguation of authors in digital libraries is essential for many tasks, including efficient bibliographical searches and scientometric analyses to the level of individuals. The question of how to link documents written by the same person has been given much attention by academic publishers and information retrieval researchers alike. Usual approaches rely on publications' metadata such as affiliations, email addresses, co-authors, or scholarly topics. Lack of homogeneity in the structure of bibliographic collections and discipline-specific dissimilarities between them make the creation of general-purpose disambiguators arduous. We present an algorithm to disambiguate authorships in the Astrophysics Data System (ADS) following an established semi-supervised approach of training a classifier on authorship pairs and clustering the resulting graphs. Due to the lack of high-signal features such as email addresses and citations, we engineer additional content- and location-based features via text embeddings and named-entity recognition. We train various nonlinear tree-based classifiers and detect communities from the resulting weighted graphs through label propagation, a fast yet efficient algorithm that requires no tuning. The resulting procedure reaches reasonable complexity and offers possibilities for interpretation. We apply our method to the creation of author entities in a recent ADS snapshot. The algorithm is evaluated on 39 manually-labeled author blocks comprising 9545 authorships from 562 author profiles. Our best approach utilizes the Random Forest classifier and yields a micro- and macro-averaged BCubed F_1 score of 0.95 and 0.87, respectively. We release our code and labeled data publicly to foster the development of further disambiguation procedures for ADS.

Keywords Author name disambiguation · Record linkage · Supervised learning · Label Propagation · Information retrieval · Digital libraries

✉ Helena Mihaljević
helena.mihaljevic@htw-berlin.de

Lucía Santamaría
lucsan@amazon.de

¹ Hochschule für Technik und Wirtschaft, Wilhelminenhofstraße 75A, 12459 Berlin, Germany

² Amazon Development Center, Charlottenstr. 4, Berlin 10969, Germany

Introduction

Bibliographic databases contain large compilations of research articles' metadata as released by academic publishers. This includes fields such as title; authors' names, affiliations, and email addresses; publication venue and date; among others. Only infrequently are records additionally consolidated to assign a unique identity to their authors, a process known as "author name disambiguation" or "record linkage". Yet the availability of author profiles is essential to perform effective literature research and discovery, given that direct author search is the most frequently used feature in digital libraries (Xie and Matusiak 2016). It is also crucial in bibliometrics and scientometrics, since it enables analyses of scholarly data to the level of individuals, for instance studies on academic careers (Mihaljević-Brandt et al. 2016), credit attribution (Caplar et al. 2017), research networks (Newman 2004; Jadidi et al. 2018), or migration (Moed and Halevi 2014; Sugimoto et al. 2016).

Academic publishers are naturally invested in resolving the author ambiguity problem in scholarly communication. The Open Researcher and Contributor ID (ORCID), a non-proprietary, persistent digital identifier launched in 2009, has been progressively introduced in many major publishers' submission procedures as a means to uniquely identify scientific authors and contributors. However, its usage is not yet as extended within the academic community as the equivalent DOI is for publications. Moreover, ORCIDs can not be added retroactively to all existing bibliographic records. For the purposes of comprehensive scientometrics studies spanning multiple decades, some sort of author disambiguation of past publications needs to be achieved by methods other than self-identification.

Not all digital libraries and databases offer author disambiguation. The majority of those that do have traditionally resorted to the manual labeling of authorships, an approach that is not scalable at the current volume of scholarly output. The full disambiguation of authors by means of algorithmic methods becomes an ambiguous and challenging task in practice. Usually, metadata associated to authors' names, affiliations, and email addresses is incomplete (e.g. abbreviated first names), vague (e.g. abridged institutional information), or outdated (e.g. past contact details). As a consequence, automatic authorship attribution suffers from well-known issues like the amalgamation of authors with frequent names or missing name parts, as well as the fracture of profiles due to spelling variability through transliteration or name changes throughout life events. The inclusion of further data facets, such as co-author graphs, subject field, or manuscript keywords makes it possible to cluster authorships more accurately. Nevertheless, the characteristics and availability of this type of metadata as well as diverging field-specific publication practices in different academic communities have a large impact on the modeling approaches. As a result, many practical AND approaches are developed with a particular use case, namely one database, in sight.

Depending on the goal and need for completeness in each particular application, the author disambiguation task can place its focus on the pure disambiguation of authorships or rather on the creation of author profiles. Regardless, the problem can be addressed on the basis of manual work, which as already mentioned suffers from non-scalability, or else by employing rule-based algorithms, collaborative input (i.e. community-based), or Machine Learning (ML) techniques, all three of which can be automatized and scaled. This work focuses on the latter.

Related work

A comprehensive overview of the field of author name disambiguation until 2009 is presented in Smalheiser and Torvik 2009; more recent techniques and algorithmic approaches, with a taxonomy, are found in Ferreira et al. 2012 and Hussain and Asghar 2017. Rule-based approaches are commonly employed in digital libraries and perform comparatively well in small databases. In this regard, a relatively simple to implement yet fairly effective method consists of clustering authors by last name and initial (Milojević 2013), which we describe in more detail below. Whereas its performance is acceptable in last-author analyses, this initials-based algorithm still introduces many potential errors in first- and all-authors studies (Strotmann and Zhao 2012). Moreover, performance gains by algorithmic disambiguation methods are more pronounced when many first names are abbreviated, as is the case in ADS, or when homonyms predominate (Kim and Kim 2020).

Most recently, several works have investigated ML-based approaches to author name disambiguation. A frequent pattern that we also follow in this article consists of two steps: (i) compute similarities between pairs of authorships, and (ii) cluster authorships based on them. Since pairwise comparisons on a large database might be computationally unfeasible, many approaches pre-process the authorship records. Typically, they are first grouped into *blocks* built on stable name features, such as some normalized form of the surname and the first initial. Similarities among pairs of publications are then computed within those blocks only. Additionally, certain post-processing procedures might be applied, for instance to deal with unassigned authorships or to double check disjoint blocks that might belong to an individual that has changed name in the course of their career.

There are various possibilities for computing pairwise comparisons and for the subsequent clustering. Caron and van Eck (2014) construct a set of rules to calculate similarity scores between authorships in Web of Science based on the idea that every bibliographic element shared between two authorships increases the evidence that the corresponding publications are written by the same author. In a subsequent step, all authorship pairs with a similarity score above a certain, block-size-dependent threshold are assigned to the same cluster and later post-processed by another set of rules. The authors report an average precision of ~ 0.97 and an average recall of ~ 0.91 on their test data set. The advantage of this method is a clear set of rules that can be easily implemented on any other database, provided that comparable attributes are available. However, email addresses, which carry a high weight in the mentioned study, are only rarely available in our ADS data set; similarly, citation data is missing. Instead, various other content-based features could be added to this approach, such as keywords and subject classifications.

It is exactly this kind of data source variability that calls for training an ML model to compute similarity scores. Cen et al. (2013) model the pairwise similarity within blocks using logistic regression and then apply a version of Hierarchical Agglomerative Clustering. The stopping criterion is itself learned from the data by means of an additional regression model. The approach is demonstrated on the DBLP database using features mainly derived from the authors' names and article titles, and yields an F_1 score of ~ 0.83 . The work of Bastrakova et al. (2016) goes along the same lines and exploits authors' names and institutions as well as article-related attributes such as keywords, subjects, and references, achieving precision and recall values of ~ 0.98 . Huang et al. (2006) apply a similar approach to the disambiguation of over 700,000 records from CiteSeer. The particularity is that they use the online SVM algorithm LASVM to learn a supervised distance function, which is then used for clustering with density-based method DBSCAN. It is claimed that

this method helps overcome the transitivity problem. The authors quote a pairwise F_1 score of 0.91 on the 10 most ambiguous name blocks from CiteSeer, with 63.8% of author clusters being completely correct. It is worth mentioning that the application of a clustering procedure, which requires additional external rules such as a stopping criterion or the size of neighbourhoods, usually requires more labeled data in order to optimize corresponding hyperparameters. Another approach that partly overlaps with Huang et al. (2006) and with the procedure presented in this manuscript is that of Ackermann and Reitz (2018). With the aim of identifying homonymous author profiles, i.e. those that comprise more than one author identity, the authors train a multilayer perceptron with different feature groups using connected components of co-author graphs and embeddings of words and phrases in titles, among others. Nearly 25,000 DBLP profiles are used for training and testing, of which nearly 3,000 are reliably labeled as homonymous. As expected, the combination of all feature groups yields the best results, with precision higher than recall, while the geometric features of the aggregated journal titles turn out to be the single most helpful feature group. Wang et al. (2012) propose a multi-stage method to assign true papers to a set of focal scientists that are known from an external database: after filtering name and affiliation, a similarity score is constructed, authors are screened and finally passed to a boosted tree algorithm that decides whether the publication belongs to the focal scientist or to a homonymous author. The method achieves a low misclassification rate and an impressive performance, nevertheless the reliance on an external data base of focal scientists adds an extra requirement that might not always be available.

Since the creation of labeled data to train supervised machine learning algorithms is very costly and time-consuming, some lines of research have turned into completely unsupervised approaches, often using graph-based methods similar to those employed in our work. For instance, Fan et al. (2011) propose the GHOST method, which considers only co-authorships while excluding other attributes such as publication title or authors' affiliations. For the co-authorship graph (after removing the author's "root node") a simple similarity formula is developed that is then used for building clusters of the graph using affinity propagation. The algorithm is evaluated on documents corresponding to twelve small- to mid-size author name blocks in DBLP and eight author names in Pubmed, yielding an average F_1 -score of 0.86 and 0.98, respectively. The DISC algorithm developed in Hussain and Asghar (2018) follows a similar approach. Again, the co-authorship information is used to construct a graph. An additional similarity score is computed using the Jaccard coefficient on title word stems. The density-based community detection algorithm gSkeltonClu is used to detect outliers as well. The approach is shown to overperform three other comparable graph-based methods on a subset of the Arnetminer dataset (Wang et al. 2011). In Shin et al. (2014), another framework is constructed based on graph operations such as vertex splitting and vertex merging of co-authorship graphs and shown to mostly outperform three existing unsupervised methods on various standard evaluation metrics. A recent addition to the literature on unsupervised methods, (Ma et al. 2020), claims to offer a superior performance than GHOST and other state-of-the-art methods by employing representation learning, evaluated on AMiner, which integrates data from popular databases with several academic bibliographic collections. They use word2vec for document representation, followed by a Graph Auto-Encoder. Finally, Hierarchical Agglomerative Clustering is in charge of partitioning documents into different clusters.

For the most part, the algorithms mentioned above make use of the co-authors and, possibly, the title of the bibliographic entries and the authors' affiliations for clustering. Thus, they are applicable to any database and domain, as these attributes are always available, at least to a certain degree of completeness. Almost all approaches, possibly with the notable

exception of Ma et al. (2020), assume that co-authors are the most important information for author disambiguation. A peculiarity of astronomy in this regard is a generally larger average number of authors per article than in other disciplines, due to the existence of populous collaborations, often comprising hundreds of members. In our ADS data set there are more than 1600 documents with 100 or more authors whose first names are only available in abbreviated form. The largest collaboration in our dataset has 3674 authors. Thus, it is not rare to find the same author name, sometimes even with the same affiliation, more than once among the authors of a paper. For this reason, we believe that AND approaches for ADS should extend beyond co-authorship graphs.

The creation of a reliable author name disambiguator is strongly tied to the characteristics of each particular database. The transfer of existing methods to other digital libraries relies either on the availability of open code implementations or on a higher level of technical detail than what is typically found in the literature. This becomes particularly problematic when various ML algorithms are combined, each of which is sensitive to the choice and fine-tuning of multiple hyperparameters, a problem that can be nontrivial. Finally, let us mention the need for trustworthy data sets in the problem of author name disambiguation. Supervised learning methods require a fair amount of annotated records to train and evaluate models. Yet the literature does not go into much detail about the requirements that such data sets must meet, nor are they described in depth in many publications. An exception is found in Müller et al. (2017), which reviews existing sets for author name disambiguation, formalizes desired requirements, and introduces a new resource for use in the information retrieval community.

Our contribution

In this work we combine methods from supervised learning and graph theory to perform author disambiguation on the bibliographic records contained in the SAO/NASA Astrophysics Data System (ADS). We first apply blocking by grouping all authorships that share the same surname and first initial after being pre-processed by removing diacritics and lowercasing. We then train a probabilistic classification model to decide whether two articles within a given block belong to the same author. Finally, we use the trained classifier to create author entities as subgraphs of authorship graphs. These are constructed as follows: the authorships of the block are nodes of the graph; an edge is drawn between two nodes if the classifier predicts that both are authored by the same person. Additionally, we use the classifier's class probabilities as edge weights, which results in a labeled graph. Author profiles are constructed as communities through label propagation. We note that the last step could be replaced by any other graph clustering algorithm that requires no further parameter specification or tuning (e.g. k -cliques).

We train different classification algorithms in various configurations and validate them on a dedicated set of 39 manually-labeled author blocks that comprise 9,545 authorships from 562 author profiles. To construct the test set we chose six from the largest ten blocks in ADS to account for particularly challenging cases with many homonyms; the remaining 33 blocks were randomly sampled. The data stems from a snapshot of the ADS Astronomy database from March 2018 that contained about 1 million publications. In addition, we evaluate these ML approaches against the rule-based algorithms of Milojević (2013). Our best model features a Random Forest classifier trained to maximize the average precision score on an imbalanced data set, in which authorship pairs of different authors are five times as strongly represented than those of the same author. Considering the detected

communities as author profiles results in a micro- and macro-averaged BCubed F_1 score of 0.95 and 0.87, respectively, on the test set. Using Local Interpretable Model-Agnostic Explanations (LIME) (Ribeiro et al. 2016) we showcase how the classifier decides whether two authorships belong to the same author. This interpretability layer increases the usefulness of the developed procedure and allows for a better integration as a semi-automatic decision support system.

To the best of our knowledge, this is the first published work on author name disambiguation that targets the creation of author profiles in ADS. Considering that most authors' first names are abbreviated in the bibliographic records of ADS, such an algorithm is essential to enable scientometrics analyses to the level of individuals in astronomy and astrophysics, as we have done in the study of the effects of gender and geographical location on astronomers' careers (Mihaljević and Santamaría 2020). We release our code and labeled data¹ in order to foster the development of further AND procedures for ADS and studies on the transferability of existing domain-agnostic procedures to ADS.

Data and methods

Data

The ADS is a digital library for research in Astronomy and Astrophysics operated by the Smithsonian Astrophysical Observatory under a NASA grant. It is considered to be the first resource for bibliographic search in its field. ADS provides cross-links with other sources, most prominently the arXiv, and is continuously enriched based to the input of the community. ADS does not currently provide author profiles; typical searches for the publications of an individual author are conducted querying by '*Last, F**' plus potential additional filters. An overview of the current status and functionalities offered by the service is presented in Accomazzi et al. (2018).

Academic publications are authored by one or various individuals. We consider each one-to-many pair of publication and author as one instance of *authorship*. The data set used in this work consists of 3,667,321 authorships referring to 924,329 documents. Thus, the average number of authors per article in ADS is four (with a stark increasing trend over time). This data snapshot was downloaded from the ADS servers via its API at the end of March 2018 and contains all publications indexed in the Astronomy database since 1970.

An example of an authorship from a record indexed in ADS is given in Table 1, displaying only data facets that were deemed effective for the author disambiguation task and that are broadly populated in the data set. Coverage is shown as percentage in square brackets. Because additional stored fields in ADS exhibit larger fractions of missing values, we were forced to exclude attributes such as the arXiv class, present in just a quarter of the records, or the ORCID, which is only sparsely available. Our data set does not contain information on references or citations other than the number of citations, which we consider too noisy for its usage as a training feature.

¹ https://github.com/helenamihaljevic/ads_author_disambiguation.

Table 1 Example of an authorship record in ADS with the fields used for disambiguation. The coverage of each attribute in the entire data set is indicated as percentage in brackets

Author [100%]	Zhang, Peng-Fei
Affiliation [78%]	Key Laboratory of Dark Matter and Space Astronomy, Purple Mountain Observatory, Chinese Academy of Sciences, Nanjing 210008, China; Key Laboratory of Astroparticle Physics of Yunnan Province, Yunnan University, Kunming 650091, China
Title [99.9%]	Possible Quasi-periodic Modulation in the $z = 1.1$ Gamma-Ray Blazar PKS 0426-380
Publ. year [100%]	2017
Journal [100%]	The Astrophysical Journal
Keywords [65.3%]	Galaxies: jets : gamma rays: galaxies : radiation mechanisms: non-thermal : Astrophysics - High Energy Astrophysical Phenomena
Abstract [78.7%]	We search for γ -ray and optical periodic modulations in distant flat-spectrum radio quasar (FSRQ) PKS 0426-380 (the redshift $z = 1.1$). Using two techniques .
Co-authors [100%]	Cao, Li-Jia; Liao, Neng-hui; Wang, Jian-cheng; Yan, Da-hai; Zeng, Wei; Zhang, Peng-fei

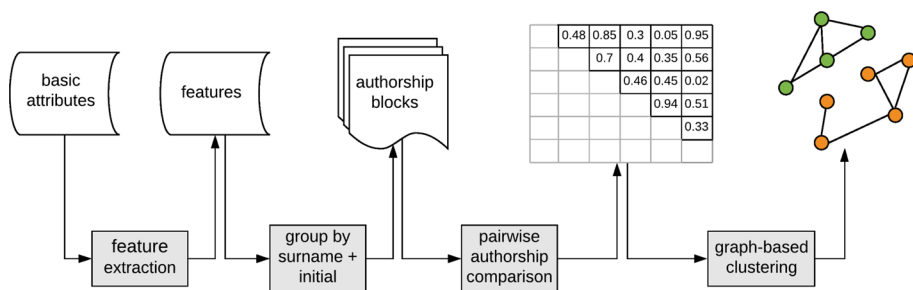


Fig. 1 Algorithmic procedure to create author profiles: after grouping authorships into author blocks, a classifier is trained to predict whether two authorships in a block belong to the same person. The results (yes/no or probabilities) are used to create clusters of authorships representing author entities

Authorship blocks

The complete procedure for the creation of author profiles is depicted in the flow diagram of Fig. 1. The two main steps are: (i) the pairwise comparison of authorships, which leads either to a Boolean decision (authored by the same person: yes=1/no=0), or to a similarity score with values between 0 and 1; and (ii) the clustering of authorships into author profiles using graphs built on results from the previous step. However, the computation of similarity scores for *any* given pair of authorships in a collection of n records grows as $\mathcal{O}(n^2)$ and is thus not feasible in most practical cases. A good compromise involves the grouping of authorships into likely non-overlapping blocks and the restriction of comparisons to pairs within them. For the construction of blocks we gather all records that share the same surname and first name initial, after removing diacritics and non-word characters. With this pre-processing, names 'Sorensen, Larry B.', 'Sørensen, L. L.' and 'Sørensen, Louise Sandberg' are grouped together in block 'sorensen.l'. In total, 166,545 blocks with at least two authorships are created; trivial blocks of size one amounting to about 3% of all authorships are discarded. The average block in the data set contains 21 authorships, but the standard deviation (~ 50) is very high. This skewness is representative of the distribution of

Table 2 Features to train a classifier that compares two authorships within the same block

Raw data field	Authorship attribute	Feature for pairwise comparison
Author name	First name	Equality, Jaro-Winkler, dmetaphone
	Middle name	Equality
	Middle name initial	Equality
	Signature	Compatibility
Affiliation	Affiliation string	Jaro-Winkler
	<i>Countries</i>	Intersection non-empty
	<i>Cities</i>	Intersection non-empty
Journal	Journal	Belongs to list of top journals, journal name contains <topic word>
Publication year	Age	Difference, maximum
Title, abstract	Top 5 words	Intersection non-empty
	Top 5 bigrams	Intersection non-empty
	<i>Doc embedding</i>	Cosine similarity
	Acronyms	Sum of TF-IDF scores of intersection
Keywords	Keywords	Sum of TF-IDF scores of intersection
Co-authors	Signatures of all co-authors	Intersection non-empty
Authorship block	Block size percentile	Block size percentile
	Matching signatures	Signatures compatible with s_1 are also compatible with s_2

authorships in bibliographic databases and is caused by the fact that a few blocks contain an inordinate number of records, such as ‘*wang.j*’, which constitutes our largest block with 2,170 items.

Pairwise comparisons

We compare two authorships by training a classifier to decide whether they are authored by the same individual. The similarity score is then defined as the probability of the positive event, i.e. the class 1. To train a classifier we perform a two-step feature engineering procedure. First, we pre-process the raw data to build new attributes such as the author’s first name or to extract standardized country and city names from affiliation strings. In a second stage we employ a number of functions that derive a *feature for a pair of authorships* (a_1, a_2) by comparing two attributes, such as the phonetic distance between two first names or the information whether a_1 and a_2 share the same country. Table 2 shows the full list of attributes and features for pairwise comparison extracted from raw data.

In the first step we apply standard text processing procedures. Usually, names are the most important information source and human experts compare them in various ways, also taking into account the names of other potential candidates in the block.² Consequently, we extract the first name, middle name, middle name initial and signature of every author name. A signature is simply the pair of first and middle name, i.e. ‘*first-middle*’; if a name

² For instance, if in a large block all authorships carry the name ‘*Miller, J.*’ except two with the name ‘*Miller, Jane Mary*’, then the probability is relatively high that these belong to two different authors. In contrast, a block containing only ‘*Miller, J.*’ and ‘*Miller, John*’ is more likely to involve a unique author.

part does not exist it is replaced by an empty string. We compare signatures in terms of *compatibility*: roughly speaking, the corresponding name parts of two signatures must be substrings of each other in order for them to be compatible. We further allow for typical English nicknames³ and concatenations of name parts. For instance, ‘*abbigail-joan*’ is compatible with ‘*abbigail-*’ (name variations), ‘*anthony-*’ is compatible with ‘*tony-m*’ (nickname) and ‘*pei-yu*’ with ‘*peiyu*’ (concatenation) but ‘*maria-f*’ and ‘*maria-t*’ are not compatible with each other.

Furthermore, we apply additional sophisticated algorithms to extract supplementary features, which are marked in italics in Table 2. For the extraction of country and city from affiliations we combine rule-based string searches, queries to the GeoNames⁴ database that contains entities of geographic objects, and named-entity recognition (NER) with ML-based Java library CERMINE (Tkaczyk et al. 2017). We also train two Doc2Vec models with Python library Gensim⁵ to embed the title and abstract of a document into a 25-dimensional vector space. For the titles model we have randomly sampled the title strings of 500,000 documents; for the abstract model we have used 100,000 randomly sampled texts. Pre-processing was applied consisting of removal of diacritics, lowercasing, tokenization yielding tokens that are maximal contiguous sequences of alphabetic characters, and removal of English stopwords.

We exploit the fact that journal names in ADS reflect their topical focus: astrophysics journals usually contain ‘*astrop*’ in their title; similarly, journals that focus on astronomy are named after ‘*astron*’. Conveniently, these rules are largely language independent. We have thus created additional Boolean features reflecting whether journal names contain a word that starts with (a) ‘*astro*’, (b) ‘*astron*’, (c) ‘*astrop*’, (d) ‘*geo*’, or (e) ‘*earth*’. Furthermore, we encode whether the journal is one of these six: The Astrophysical Journal, Astronomy and Astrophysics, Monthly Notices of the Royal Astronomical Society, The Astronomical Journal, Nature, or Science. These are considered to be the main publication venues in astronomy and astrophysics that “encompass the vast part of astronomical research today” (Caplar et al. 2017).

Finally, we have engineered a couple of features that reflect the relative size and signature diversity within the entire authorship block. In the final model we have included a Boolean feature indicating whether all signatures that are compatible with that of authorship a_1 are also compatible with authorship a_2 and vice versa. This helps the model to distinguish between the following two scenarios: assume that a block contains signatures ‘*jane-mary*’, ‘*j*’ and ‘*j-r*’. Signature ‘*j-r*’ is not compatible with ‘*jane-mary*’, indicating a higher ambiguity for the authorship associated with ‘*j*’ in this block. This is in opposition to a more homogeneous group containing signatures ‘*jack-*’ and ‘*j*’ only.

Regarding the actual training of the classifier on our pre-processed records, we split the data into various sets for training, validation and test on the level of authorship blocks, as depicted in Fig. 2. This way we achieve more reliability in terms of transfer to new authorship blocks.

³ https://en.wiktionary.org/wiki/Appendix:English_given_names.

⁴ <http://www.geonames.org/>.

⁵ <https://radimrehurek.com/gensim/>.

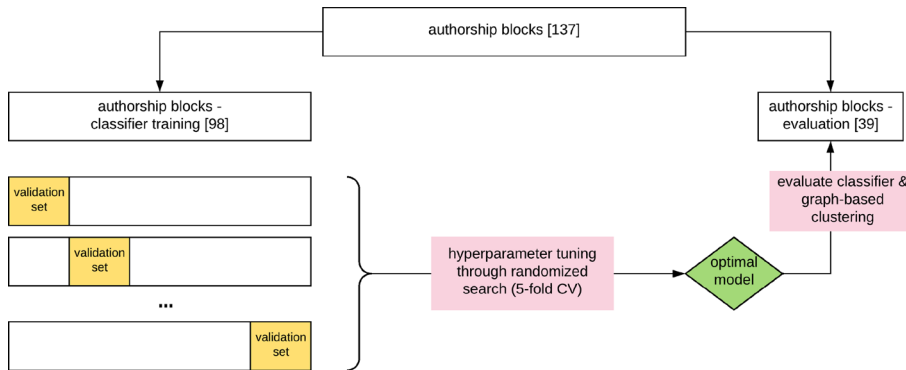


Fig. 2 Overview of training, hyperparameter tuning and evaluation of a classifier, with corresponding numbers of blocks used in the experiment in square brackets

Graph-based clustering

The pairwise classifier described in the previous subsection allows for a canonical representation of each authorship block as an undirected graph in which a node represents an authorship and an edge is drawn if the classifier has predicted the corresponding two authorships to belong to the same author. The graph becomes a weighted authorship graph by labeling the edges with the probabilities for class 1.

In order to divide the individual nodes of such a graph into disjoint groups using the edge information, we can employ methods from the analysis of real-world networks. Under the term *community detection*, numerous methods have been developed to identify groups of nodes that have an inherent or externally specified similarity to each other. The number and size of communities is not specified in advance, which is a significant advantage when applied to author disambiguation, as the number of distinct authors is not known a priori.

Although various approaches could be applied to an authorship graph, ranging from simple ones such as *k*-cliques to rather sophisticated methods, in this work we build author profiles from communities generated by the asynchronous *label propagation* algorithm. The underlying idea, which excels through simplicity and speed, is as follows: assume that a node *u* has neighbouring nodes, each of which carries a label denoting the community to which it belongs. Then *u* selects its community according to the majority of the neighbours' labels. The algorithm is initialized with a random unique labeling of all nodes and each node adopts the label shared by the majority of its neighbours iteratively, typically resolving label ties randomly. Labels propagate through the graph until an equilibrium is reached. Label propagation uses only the network structure and thus requires no prior information (such as the number of central nodes) of the communities to be detected. Moreover, it is also suitable for large networks since it exhibits near-linear time complexity in the number of edges. For more details see e.g. Raghavan et al. (2007), Šubelj (2019).

Evaluation metrics

For every authorship block the construction of author entities consists of grouping together those records that were written by the same individual while keeping away those authored

by others. An important difference to the previous classification task is that each block is related to an unknown and variable number of authors, whence we treat this step as a clustering task. Since we have manually disambiguated a large set of authorship blocks, we can employ *extrinsic* evaluation metrics, characterized by the fact that they compare the result of the clustering procedure with a gold standard data set. Among these extrinsic measures are the BCubed metrics, which are inspired by precision and recall and are thus easily interpretable (Amigó et al. 2009).

Similar to other clustering evaluation scores, BCubed metrics favor cluster homogeneity and cluster completeness: homogeneity is high when clusters do not contain items belonging to different categories; completeness, on the other hand, requires that items from the same category are kept together. In addition, BCubed metrics satisfy the following two constraints: they favor the creation of a “leftover” cluster over spreading wrongly assigned items across otherwise clean clusters. This can be thought of as the preference to create a cluster that could be tagged as “unclassified” or “miscellaneous” instead of introducing noise into other clusters. Furthermore, the scores are higher for procedures that induce a small error in a big cluster than for those in which numerous errors happen in many small clusters.

BCubed precision of an authorship a is defined as the proportion of authorships in its cluster that belong to the author of a . The overall BCubed precision is the mean precision of all authorships. The BCubed recall of a is the proportion of all authorships that belong to the author of a that are contained in the cluster of a . The overall BCubed recall is again the averaged BCubed recall of all authorships. As known from the standard precision and recall, one can combine both into one metric by defining the BCubed F_1 -score as the harmonic mean of BCubed precision and recall.

We aggregate the BCubed scores of the blocks in the test set on a micro- and a macro-level to calculate average values: For the first we calculate BCubed scores per block and then average across all blocks. For the latter, we form one block from all authorships in the test set and calculate BCubed metrics once. Since the BCubed scores weight each authorship of a block equally, the large blocks are more influential in the macro-calculation. In said case, lower values are to be expected, as larger blocks are more difficult to disambiguate.

Baseline

As mentioned above, the traditional blocking strategy used to disambiguate authors within a single discipline consists of using information contained in the authors' names only. This approach, which is described in Milojević (2013), is usually employed as baseline for many AND studies, for instance in Backes (2018). Milojević (2013) presents three variations of a rule-based approach for name blocking. The *first initial* variation considers only the last name and the first initial, discarding any information from the middle name. For instance, all publications from ‘García, J’ and ‘García, JA’ will be considered as belonging to the same individual. The alternative method is to take into account *all initials*, which would split all occurrences of ‘García, J*’ into distinct author profiles. Which one of the two methods yields better results is highly dependent on the characteristics of the database and the relative frequency of names. A third, *hybrid* approach is proposed and recommended in Milojević (2013), which consists of applying the *all initials* method if, for a given last name, more than *one* middle initial appears for a particular name root; else the *first initial* method is used. Of the three variants, the latter is shown to be superior in the original

Table 3 Sizes and statistics of our data set, including number of labeled and unlabeled authorships and identified author profiles, the amount of pairwise comparisons built from the authorships, and the proportion of them belonging to the same author (class 1)

		All	Training set	Test set: all	Test set: large	Test set: random sample
Blocks	Total	137	98	39	6	33
Authorships	Total	14,054	2487	11,567	10,006	1561
	Unlabeled	2191	169	2022	2010	12
	% Labeled	84.4	93.2	82.5	79.9	99.2
	Mean/block (labeled)	86.6	23.7	244.7	1332.7	46.9
	Median/block (labeled)	7	5	35	1304	24
Author Profiles	Mean	5.9	2.6	14.4	79.2	2.6
	Median	1	1	2	80.5	1
Pairwise Comparisons	Total	5,708,230	204,299	5,503,931	5,423,970	79,961
	% in class 1	47.6	64.9	47.0	46.5	77.4

study, and it also outperforms the other two variants on our data; we thus choose Milojević's *hybrid* method as baseline to compare with our AND method for ADS.

Experiment data set

From the available snapshot of the ADS database retrieved in March 2018 we selected 137 authorship blocks containing at least three authorships and amounting to 14,054 records. To construct this sample we chose six from the largest ten blocks to account for particularly challenging cases with many homonyms; the remaining were randomly sampled. We were not able to completely disambiguate every record, in particular in four of the six largest blocks corresponding to names of Chinese origin, leaving us with 11,863 labeled authorships (84.4%) that we used for training and testing the classifier.

We set aside 33 from the 137 authorship blocks to evaluate the trained classifiers together with the graph clustering algorithms. The test set comprises 9,545 authorships that we managed to uniquely assign to an author. The test blocks contain between 1 and 181 profiles, with a median number of 2 and an average number of 14.4 profiles per block, for a total of 562 author entities. The skewness in the distribution of the number of author profiles results from the fact that the six larger blocks were entirely used for evaluation.

The training data is built from the remaining 98 blocks that contained 2,487 authorships, almost all being disambiguated. The distributions of authorships and author profiles differ significantly between the training and the test set since four of the six large blocks in the test set show an extremely high number of author profiles. Since the manual disambiguation of such profiles requires enormous effort and a certain level of understanding of astronomy and astrophysics, we aim to train a model using possibly little data that requires a reasonable amount of manual work.

We create a data set to train the classifier by computing all possible pairs of authorships for each of the blocks using the features and functions described in Table 2 and below. This leads to 5.7 million authorship pairs, most of which are part of the test set. The training set

Table 4 BCubed metrics for all blocks in the test set

	Micro-aggregation			Macro-aggregation			# Clusters	
	b3p	b3r	b3f	b3p	b3r	b3f	True	Predicted
Rule-based	0.973	0.855	0.89	0.963	0.638	0.768	14.41	3.615
Decision Tree accuracy, {0: 5, 1:1}	0.936	0.911	0.909	0.749	0.924	0.827	14.41	10.821
Random Forest accuracy, {0: 5, 1:1}	0.947	0.956	0.947	0.811	0.946	0.874	14.41	10.974
Hist-GBDT average precision, {0: 5, 1:1}	0.944	0.96	0.945	0.787	0.96	0.865	14.41	9.333

Bold figures correspond to the setting that achieves the best macro-averaged BCubed F_1 score

comprises 204,299 pairs, with 64.9% belonging to the same author (class 1). The proportion of pairs in class 1 is smaller in the test set, particularly in the four blocks corresponding to names of Chinese origin. Detailed statistics of our data set are given in Table 3.

Results

We have implemented our data processing pipelines and ML algorithms in Python 3.8; classifier training and evaluation were carried out using the scikit-learn library (Pedregosa et al. 2011), for label propagation we use the implementation in NetworkX (Hagberg et al. 2008). The experiments were conducted on an AWS Linux cloud instance (EC2 T2.xlarge) with 4 vCPUs and 16 GiB memory.

We gravitate towards the choice of non-linear, tree-based classification models since these perform better for data with many non-numeric features and expected non-linear interactions. This has also been confirmed by AND experiments on other data sets, see e.g. Treeratpituk and Giles 2009; Wang et al. 2012; Louppe et al. 2016. The following classifiers have been trained: Decision Tree (DT), Random Forest (RF), and Histogram-based Gradient Boosting Decision Tree (Hist-GBDT), whose implementation in scikit-learn is inspired by LightGBM (Ke et al. 2017). We have performed hyperparameter tuning using randomized search on 200 iterations. In addition, we have also trained each algorithm using different proportions of the respective classes: (i) the original ratio, (ii) a balanced ratio {0:1, 1:1}, (iii) a 2-to-1 ratio in favour of class 0 {0:2, 1:1}, (iv) a 2-to-1 ratio in favour of class 1 {0:1, 1:2}, and (v) a 5-to-1 ratio {0:5, 1:1}. Depending on the expected ratio between the two classes in a block, downsampling one of the classes (typically the class of linked authorships) can have a significant effect on performance (Kim and Kim 2018; Louppe et al. 2016), in particular on blocks that contain many author profiles and thus yield more pairwise comparisons in class 0. Furthermore, we have trained models to maximize the accuracy and average precision scores. (We have also experimented with other scoring functions, with lower performance.) This resulted in 30 models, which we use to create weighted authorship graphs.

In a subsequent step we have computed author profiles from the weighted authorship graphs as communities derived through label propagation. We compare the results of the entire procedure with the rule-based hybrid approach of Milojević 2013 (see “[Base-line](#)” section). Below we report the metrics of our best models in terms of micro- and

Table 5 Classification report of Random Forest trained to maximize accuracy on five times as many pairwise authorship comparisons in class 0 than in class 1

	Precision	Recall	F ₁ -score	Support
0	0.83	0.98	0.90	2,919,297
1	0.97	0.77	0.86	2,584,483
Accuracy			0.88	5,503,780
Macro avg	0.90	0.88	0.88	5,503,780
Weighted avg	0.90	0.88	0.88	5,503,780

macro-aggregated BCubed F₁ scores. Note that the metric names are abbreviated in the tables; e.g., “BCubed precision” is abbreviated as “b3p”.

Evaluation on the entire test set

For each algorithm, the models that perform best in terms of macro-aggregated BCubed F₁ score are listed in Table 4. Random Forest (RF) trained to maximize accuracy on a data set in which authorship pairs of different authors (class 0) were five times more strongly represented than those of the same author (class 1) outperforms all others, with an average micro BCubed F₁ score of 0.947 and an average macro BCubed F₁ score of 0.874. Hyperparameter optimization yielded a model with 30 trees, split using entropy, a maximum tree depth of 25, and at least 5 samples per leaf. The best Hist-GBDT model with a learning rate of 0.15, maximum tree depth of 3, 250 iterations, and at least 50 samples per leaf, is only slightly worse.

The best model yields similar values for both micro-averaged BCubed precision and recall. However, as shown in Table 4, the macro-aggregation yields significantly lower precision values, and this difference stems from the large blocks corresponding to Chinese names. This is also reflected on the difference between the average predicted number of clusters and the average true number of author profiles. An evaluation of the corresponding classifier on the test set already indicates this trend by showing higher precision on authorship comparisons in class 1 than class 0: while two authorships from the same author are only rarely predicted to belong to different authors, this error happens more frequently for authorships belonging to different authors. More precisely, the proportion of authorships from the same author among those predicted as such is 97%, while only 83% for the authorships from different authors. The precise numbers are displayed in Table 5. It is worth noting that other configurations achieve significantly better pairwise metrics. Balancing both classes, for instance, yields a classifier with F₁ score of 0.96 and no value less than 0.94.

As Table 4 shows, each of the models based on the suggested procedure performs better than the name-based baseline algorithm, for both micro- and macro-averaged metrics. All three classifiers perform best on the overall test set when the class proportions are {0:5, 1:1}. This can be explained by the fact that a majority of the comparisons in the test set stem from the 4 large homonymous profiles.

It is worth mentioning that the choice of the scoring function has less effect than the weighting of the two classes. The performance based on classifiers that maximize the average precision score is nearly equal for all three algorithms as when the accuracy score is used. The effect of class weighting is particularly reflected on the ratio of precision and recall: if the proportion of authorship comparisons in class 1 is increased, the BCubed

Table 6 BCubed metrics for large blocks in the test set

	Micro-aggregation			Macro-aggregation			# Clusters	
	b3p	b3r	b3f	b3p	b3r	b3f	True	Pred.
Rule-based	0.967	0.545	0.646	0.968	0.583	0.727	79.167	13.5
Decision Tree accuracy, {0: 5, 1:1}	0.682	0.916	0.766	0.705	0.922	0.799	79.167	52.0
Random Forest average precision, {0: 5, 1:1}	0.761	0.939	0.832	0.778	0.944	0.853	79.167	54.5
Hist-GBDT (average precision, {0: 5, 1:1})	0.732	0.957	0.817	0.751	0.96	0.843	79.167	44.5

Bold figures correspond to the setting that achieves the best macro-averaged BCubed F_1 score

Table 7 Blockwise evaluation of the best performing algorithm based on Random Forest and label propagation on the six largest blocks from the test set

	b3p	b3	b3f	Ttotal authorships	Disambiguated authorships	Author profiles	Predicted clusters
binzel.r	0.999	1.0	0.999	1,599	1,599	2	1
lee.j	0.569	0.918	0.702	1,461	1,295	181	110
zhang.y	0.557	0.881	0.682	1,784	1,211	122	72
wang.j	0.641	0.934	0.761	2,170	1,313	107	76
russell.c	1.000	0.984	0.992	1,522	1,522	9	12
chen.y	0.801	0.920	0.856	1,470	1,056	54	21

recall grows at the expense of BCubed precision. For example, the micro-averaged BCubed recall reaches 0.98 if class 1 is represented twice as strongly as class 0, while BCubed precision drops to 0.89. This is in line with expectations, as explained above, since an over-representation of authorship pairs of the same author will tend to lead to more frequent merging of different author profiles, while in the opposite case the profiles are less complete, but mix-ups occur less frequently. Depending on the application scenario, a corresponding preference of the error type can be useful.

Our division of the data intended to train the models on the small authorship blocks, which are easier to disambiguate, while the evaluation is carried out also on six of the ten largest blocks, four of which contain dozens or more author profiles each. To see the effect of this division, we have retrained the model with the best configuration of scoring function and class ratio including the block ‘chen.y’, which comprises 1,056 disambiguated authorships corresponding to 54 authors, as part of the training set. The resulting models managed to improve BCubed scores rather marginally.

Block-type differentiated evaluation

If we now examine the two block groups separately, we see the previously mentioned effects even more strongly on the large blocks, as shown in Table 6. The rule-based approach achieves a BCubed precision of almost 0.97 with a BCubed recall value of 0.54. The RF, on the other hand, achieves a BCubed recall of 0.94 with a precision of 0.76. Overall, the performance on these blocks is significantly worse than on the total set, especially

Table 8 BCubed metrics for randomly selected blocks in the test set

	Micro-aggregation			Macro-aggregation			# Clusters	
	b3p	b3r	b3f	b3p	b3r	b3f	True	Pred.
Rule-based	0.974	0.911	0.934	0.939	0.927	0.933	2.636	1.818
Decision Tree accuracy, {0: 1, 1:1}	0.961	0.983	0.971	0.958	0.986	0.972	2.636	2.364
Random Forest average precision, {0: 2, 1:1}	0.974	0.978	0.975	0.98	0.978	0.979	2.636	2.727
Hist-GBDT accuracy, {0: 1, 1:1}	0.957	0.976	0.965	0.955	0.984	0.969	2.636	2.455

Bold figures correspond to the setting that achieves the best macro-averaged BCubed F_1 score

on the four blocks of Chinese surnames. The performance on the individual large blocks is shown in Table 7.

As expected, the performance on the 33 randomly selected blocks, shown in Table 8, is significantly better. Here, RF achieves the best macro-averaged BCubed F_1 score of 0.98. Note that the Decision Tree classifier which is considerably simpler and trained a lot faster is almost as good. However, it is worth noting that the rule-based approach, which takes into account only the authors' names, already performs fairly well, with 0.93 as micro- and macro-averaged BCubed F_1 scores. The difference to the best performing method is seen on the BCubed recall value, which reflects cluster completeness. The individual performance on the 33 randomly selected blocks is shown in Table 9.

Explanation of classification results using LIME

For many ML approaches it is nearly impossible to reconstruct why a prediction was made by a model or to understand its inner workings. In order for humans to trust the output of such models, a large degree of interpretability and explainability is highly desirable. Here we make use of local surrogate models, concretely the Local Interpretable Model-agnostic Explanations (LIME) method to explain individual predictions from our best classifier (Ribeiro et al. 2016). LIME works by training an interpretable model such as a ridge regression or a decision tree that locally approximates the original model. The approximation model is then used to generate explanations for the data point of interest.

We have selected three examples from the block '*chen.y*' that show what additional features are triggered by the model when the name parts do not carry sufficient information, and where the baseline algorithm would rather fail. We also show an example where the classifier has made the wrong decision. In contrast to feature summary statistics and other global insights into the inner workings of a classifier, local interpretations offer a possibility to inspect particular decisions and, for example, facilitate the review of the classifier's decision by human experts where the predicted probability is low.

Table 10 shows two authorships with highly ambiguous names, one of which is missing affiliation information. While this example is difficult to handle properly for a name-based algorithm, a human annotator would quickly realize that both publications belong together, one being an erratum of the other. As shown in the figure accompanying Table 10, the classifier correctly finds that the combination of matching co-authors, bigrams, and acronyms is sufficient to assign a large probability to class 1.

The second example, displayed in Table 11, shows two publications by authors with the same first name, '*Yuxi*', which appears infrequently in the '*chen.y*' block. Nevertheless,

Table 9 Blockwise evaluation of the best performing algorithm based on Random Forest and label propagation on the 33 randomly sampled blocks from the test set, i.e. ignoring the large blocks

	b3p	b3	b3f	Total author-ships	Disambiguated authorships	Author profiles	Predicted clusters
ables.s	1.0	1.0	1.0	11	11	1	1
adam.m	1.0	1.0	1.0	16	16	6	6
annibali.f	1.0	1.0	1.0	39	39	1	1
aramo.c	1.0	1.0	1.0	100	100	1	1
botvina.a	1.0	1.0	1.0	12	12	1	1
campbell-brown.m	1.0	1.0	1.0	51	51	1	1
carpenter.j	0.969	0.936	0.952	196	190	7	10
cotzomi.j	1.0	1.0	1.0	30	30	1	1
desiante.r	1.0	1.0	1.0	54	54	1	1
ermakov.s	0.951	0.954	0.953	35	35	5	5
gretskov.p	1.0	1.0	1.0	23	23	1	1
katgert.p	1.0	1.0	1.0	63	63	1	1
klein.u	0.969	0.992	0.980	229	228	5	4
kuga.k	0.820	1.0	0.901	10	10	2	1
luthcke.s	1.0	1.0	1.0	23	23	1	1
melioli.c	1.0	1.0	1.0	15	15	1	1
mendez.r	0.988	1.0	0.994	165	162	4	3
mercuri.s	1.0	1.0	1.0	12	12	1	1
milller.v	1.0	0.932	0.965	22	22	6	7
moreno.c	1.0	1.0	1.0	21	21	3	3
morlok.a	1.0	1.0	1.0	24	24	1	1
naef.d	1.0	1.0	1.0	70	70	1	1
nestorov.g	1.0	1.0	1.0	34	34	1	1
phan.n	1.0	1.0	1.0	21	21	1	1
ranjan.s	1.0	0.825	0.904	10	10	3	4
ravelo.a	1.0	1.0	1.0	10	10	1	1
ribeiro.a	0.982	0.987	0.984	75	75	10	10
rovira.m	1.0	0.831	0.908	65	65	2	4
roy.b	0.865	0.846	0.856	12	12	4	5
salama.f	1.0	1.0	1.0	45	45	1	1
sarasso.m	1.0	1.0	1.0	16	16	1	1
vargas.c	0.815	1.0	0.898	14	13	5	4
vincent.m	0.791	0.956	0.865	36	36	6	5

they do not belong together, and this is correctly predicted by the classifier with a high probability. LIME shows that in this case the decision for class 0 is supported by the lack of agreement on any content-based features (journal type and rank, bigrams or acronyms match, abstract similarity), co-authors, or city. The name-based features on their own are not sufficient to assign the two authorships to the same author profile, a sensible behaviour in general. Note that this is a typical case where the baseline from Milojević (2013) has few chances of getting the correct prediction.

Table 10 Metadata for two publications from the same author, one being an erratum of the other, and the explanation of the classifier's decision to assign both to the same author, despite the name ambiguity and lack of affiliation in the second record

	Authorship 1	Authorship 2
Name	Chen. Y.	Chen. Y.
Affiliation	Max-Planck-Institut für Gravitationsphysik, Am Mühlenberg 1, 14476 Potsdam, Germany	–
Year	2006	2007
Journal	Physical Review D	Physical Review D
Title	Frequency noise and intensity noise of next-generation gravitational-wave detectors with RF/DC readout schemes	Erratum: Frequency noise and intensity noise of next-generation gravitational-wave detectors with RF/DC readout schemes [Phys. Rev. D 73, 122005 (2006)]
Abstract	The sensitivity of next-generation gravitational-wave detectors [...]	The sensitivity of next-generation gravitational-wave detectors [...]
Co-authors	Kawamura, S.; Mio, N.; Somiya, K.	Kawamura, S.; Mio, N.; Somiya, K.

signature_match=true	0.23
middle_name_init_equal=both-missing	0.12
journal_top=false	-0.08
coauthors_match=true	0.10
journal_astro=false	-0.08
middle_name_equal=both-missing	0.07
top_bigrams_match=true	0.05
acronyms_match > -1000.00	0.03
aff_sim <= -1000.00	-0.03
journal_astro=false	-0.03

Predicted probability for class 1: 0.94

Table 11 Two homonymous authorship instances, both written by a person named ‘Chen, Yuxi’, correctly separated by the classifier by noticing the lack of agreement on the content-based features

Authorship 1		Authorship 2	
Name	Chen, Yuxi	Chen, Yuxi	
Affiliation	Center for Space Environment Modeling, University of Michigan, Ann Arbor, Michigan USA	Department of Astronomy, University of Massachusetts, Amherst, MA 01003, USA	
Year	2016	2012	
Journal	Journal of Geophysical Research (Space Physics)	Monthly Notices of the Royal Astronomical Society	
Title	Extended magnetohydrodynamics with embedded particle-in-cell simulation of Ganymede’s magnetosphere	Deep 1.1 mm-wavelength imaging of the GOODS-S field by AzTEC/ASTE - II. Redshift [...]	
Abstract	We have recently developed a new modeling capability to embed the implicit particle-in-cell (PIC) [...]	We report the results of the counterpart identification and a detailed analysis of the [...]	
Co-authors	Borovikov, Dmitry; Daldorff, Lars K. S.; Dorelli, John C.; [...]	Aretxaga, I.; Austermann, J. E.; Capak, P.; Ezawa, H.; Giavalisco, M.; Guo, Yicheng; [...]	

Predicted probability for class 1: 0.1

Feature	Predicted Probability
signature_match=true	0.28
middle_name_init_equal=both-missing	0.12
journal_top=false	-0.08
journal_astro=false	-0.08
middle_name_equal=both-missing	0.10
coauthors_match=false	-0.08
city_equal=false	-0.08
top_bigrams_match=false	-0.05
acronyms_match <= .1000.00	-0.05
abstract_sim <= 0.45	-0.05

Table 12 A pair of authorships from the same author that is correctly assigned to the same author profile due to a combination of name-based features with journal characteristics and matching co-authors

	Authorship 1	Authorship 2
Name	Chen. Y.	Chen. Yang
Affiliation	-	-
Year	1997	1996
Journal	Astronomy and Astrophysics	The Astrophysical Journal
Title	Is RX J1713.7-3946 the remnant of the AD393 guest star?	Approximate Analytical Expressions of the Wind-driven Bow Shock
Abstract	A new supernova remnant RX J1713.7-3946 in the constellation Scorpius is [...]	
Co-authors	Qu, Q.-Y.; Wang, Z. R.	

Predicted probability for class 1: 0.96

Feature	Predicted Probability
signature_match=true	0.22
middle_name_init_equal=both-missing	0.15
journal_top=true	0.12
coauthors_match=true	0.11
journal_astro=true	0.09
middle_name_equal=both-missing	0.07
top_bigrams_match=false	-0.03
aff_sim <= -1000.00	-0.02
acronyms_match <= -1000.00	-0.02
acronyms_bool_match=one-missing	-0.02

Table 13 Incorrectly assigned publications to the same author: The algorithm fails at deciding that these two papers, although similar in topic, do not belong together

Authorship 1		Authorship 2	
Name	Chen. Y.	Chen. Y.	
Affiliation	-	-	
Year	1997	2004	
Journal	Astronomy and Astrophysics	Acta Astronomica Sinica	
Title	Is RX J1713.7-3946 the remnant of the AD393 guest star?	The heating of X-ray gas by radio gas in cluster PKS0745-191	
Abstract	A new supernova remnant RX J1713.7-3946 in the constellation Scorpius is [...]	A calculation about the energy evolution of the relativistic particles in galaxy clusters [...]	
Co-authors	Qu, Q.-Y.; Wang, Z. R.	Jia, S. M. ; Lu, F. J.; Song, L. M.; Wu, M.; Xiang, F.	

Predicted probability for class 1: 0.59

Feature	Value
signature_match=true	0.24
middle_name_init_equal=both-missing	0.14
journal_top=false	-0.08
journal_astro=true	0.10
coauthors_match=false	-0.08
middle_name_equal=both-missing	0.10
acronyms_match <= -1000.00	-0.05
top_bigrams_match=false	-0.05
acronyms_bool_match=true	0.03
0.70 < abstract_sim <= 0.81	0.03

Our third example, in Table 12, shows how the classifier handled the correct attribution to class 1 of an authorship with a strongly ambiguous name to one featuring a first but no middle name (initial). In this case the classifier relied on name-based features such as signature match and similar name structure, but also journal type and rank as well as common co-authors. Nevertheless it is worth mentioning that the fact that the classifier weights the lack of middle name as positive is not always desirable and might induce errors in other cases.

This is precisely the case showcased in Table 13, where the pair of compared authorships are similar to the example above, but in this case the papers are originally from two different authors. The algorithm incorrectly assigns class 1 because it is unable to assign sufficient weight to features that might have been of help, such as the relatively large gap between publication years. The fact that the abstracts are similar and that matching acronyms exist might have contributed to the confusion, which shows that content-based features on their own are also not sufficient to fully disambiguate highly ambiguous clusters.

Runtime analysis

The three classification algorithms differ significantly in terms of training time complexity. For the Random Forest classifier with k trees the complexity amounts to $O(b^2 \cdot \log(b) \cdot d \cdot k)$, where b is the block size, d and k the number of features and trees, respectively. Random Forest is comparatively faster than many other algorithms and can be easily parallelized. In our experiments, both ensemble methods exhibit comparable training times. When limiting the size of the training data to 100,000 records, the randomized hyperparameter optimization procedure (see Fig. 2) with 200 parameter combinations runs in 22–36 minutes for Random Forest and in 30–45 minutes for Hist-GBDT with the above-mentioned hardware configuration. These numbers depend on the parameter ranges used for hyperparameter tuning and are thus not fully comparable. The mean fitting time for both algorithms amounts to less than 10 seconds for a fixed set of hyperparameters. When trained on larger training data folds with a maximum number of 500,000 data points, the entire hyperparameter tuning of Hist-GBDT takes up approximately 3 hours; a single fit for both RF and Hist-GBDT amounts to roughly 2 minutes on average.

The run-time complexity of a decision tree equals $O(\text{depth of tree})$ and $O(\text{max. depth of trees} \cdot k)$ for a RF consisting of k trees. Thus, the run time of both ensembles will differ by a constant depending on the tuned hyperparameters for the number of trees and the tree depth. The clustering through Label Propagation exhibits near-linear time complexity in the number of edges. In the worst-case scenario, the number of edges is $b \cdot (b - 1)$, yielding a run-time complexity of $O(b^2)$. The evaluation on the entire test set took approximately 25 minutes.

Conclusions

We present a novel and well-performing approach to the disambiguation of author entities in the dedicated Astronomy and Astrophysics ADS database. It is based on the combination of authorship grouping into blocks, the supervised training of a classifier to decide on the common authorship of publication pairs, and the creation of author profiles via community detection using label propagation. For the choice of the classifier we try and compare

three non-linear, tree-based methods: Decision Tree, Random Forest, and Histogram-based Gradient Boosting Decision Tree.

Our best performing model uses a Random Forest that maximizes accuracy and a class balance ratio of $\{0:5,1:1\}$, i.e. using 5 times more examples of non-matching authorships than of matching ones. Hist-GB performs nearly as well, with the same class balance, which plays a more important role than the choice of optimization function. The subsequent detection of communities via label propagation, a simple and fast algorithm, yields a micro- and macro-averaged BCubed F_1 scores of 0.95 and 0.87, respectively, on a test set comprising a mix of randomly selected blocks and six of the largest ten. The BCubed precision tends to drop on the large blocks with many homonymous names due to false positive ambiguous authorships that have the tendency to cause cluster mix-ups. There are various modifications of label propagation that could be utilized to detect such hubs and ameliorate this problem (Chin and Ratnavelu 2016). However, we would like to stress the advantage of label propagation over alternative clustering approaches such as density-based or hierarchical models, since it does not require setting or training any hyperparameters or any further knowledge about the author network structure.

We have provided explanations of the trained classifier using LIME in order to provide insights into the predictions made by the model. This possibility to inspect particular decisions could facilitate the review of the classifier's output by human experts and thus make the approach easier to integrate into a semi-automated workflow. We believe that our results represent a promising avenue towards the implementation of author entities for ADS in production and have provided our code and manually labeled data to foster the further development of AND for ADS and other databases.

Acknowledgements The authors acknowledge that their work has been informed by their participation in the Project "A Global Approach to the Gender Gap in Mathematical, Computing, and Natural Sciences: How to Measure It, How to Reduce It?" funded by the International Science Council (ISC). This research has made use of NASA's Astrophysics Data System. The authors thank A. Accomazzi for facilitating access to the ADS records. We thank the anonymous reviewers for their critical feedback that helped improve this manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL.

Compliance with ethical standards

Conflict of interest LS has been employed by the Amazon Development Center Germany since 2016. This work has been done during a leave of absence from Amazon.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Accomazzi, A. et al. (July 2018). New ADS Functionality for the Curator. In *European physical journal web of conferences* (Vol. 186, p. 08001). <https://doi.org/10.1051/epjconf/201818608001>. arXiv: 1710.08505 [astro-ph.IM].
- Ackermann, M. R., & Reitz, F. (June 15, 2018). Homonym detection in curated bibliographies: Learning from dblp's Experience (full version). In: arXiv:1806.06017 [cs]. (visited on 10/10/2020).
- Amigó, E. et al. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. In *Information retrieval* (Vol. 12, No. 4, pp. 461–486). ISSN: 1573-7659. <https://doi.org/10.1007/s10791-008-9066-8>.
- Backes, T. (2018). The impact of name-matching and blocking on author disambiguation. In *Proceedings of the 27th ACM international conference on information and knowledge management. CIKM '18* (pp. 803–812). Torino, Italy: Association for Computing Machinery. ISBN: 9781450360142. <https://doi.org/10.1145/3269206.3271699>.
- Bastrakova, E. et al. (Nov. 2016). Relational machine learning author disambiguation. In *2016 IEEE artificial intelligence and natural language conference (AINL)* (pp. 1–7).
- Caplar, Neven, Tacchella, Sandro, & Birrer, Simon (June 2017). Quantitative evaluation of gender bias in astronomical publications from citation counts. In *Nature astronomy* (Vol. 1, No. 0141, p. 0141). <https://doi.org/10.1038/s41550-017-0141>. arXiv: 1610.08984 [astro-ph.IM].
- Caron, E. & van Eck, N. J. (Sept. 2014). Large scale author name disambiguation using rule-based scoring and clustering. In *Context counts: Pathways to master big and little data. Science and technology indicators conference 2014 Leiden* (pp. 79–86). Universiteit Leiden.
- Cen, L. et al. (2013). Author disambiguation by hierarchical agglomerative clustering with adaptive stopping criterion. In *Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval. SIGIR '13* (pp. 741–744). Dublin, Ireland: ACM. ISBN: 978-1-4503-2034-4. <https://doi.org/10.1145/2484028.2484157>.
- Chin, J. H., & Ratnavelu, K. (2016). Detecting community structure by using a constrained label propagation algorithm. *PLoS One*, 11(5), 1–21. <https://doi.org/10.1371/journal.pone.0155320>.
- Fan, X. et al. (Feb. 2011). On graph-based name disambiguation. *Journal of Data and Information Quality*, 2(2), 1–23. ISSN: 1936-1955, 1936-1963. <https://doi.org/10.1145/1891879.1891883>. (visited on 10/10/2020).
- Ferreira, A. A., Gonçalves, M. A., & Laender, A. H. F. (Aug. 2012). A brief survey of automatic methods for author name disambiguation. In: *SIGMOD Rec* (Vol. 41, No. 2, pp. 15–26). ISSN: 0163-5808. <https://doi.org/10.1145/2350036.2350040>.
- Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), *Proceedings of the 7th Python in science conference* (pp. 11–15). Pasadena, CA USA.
- Huang, J., Ertekin, S., & Giles, C. L. (2006). Efficient name disambiguation for large-scale databases. In J. Fürnkranz, T. Scheffer, & M. Spiliopoulou (Eds.), *Knowledge discovery in databases: PKDD 2006* (pp. 536–544). Berlin: Springer. ISBN: 978-3- 540-46048-0.
- Hussain, I., & Asghar, S. (2017). A survey of author name disambiguation techniques: 2010–2016. *The Knowledge Engineering Review*, 32, e22. <https://doi.org/10.1017/S0269888917000182>.
- Hussain, I., & Asghar, S. (Dec. 1, 2018). DISC: Disambiguating homonyms using graph structural clustering. *Journal of Information Science*, 44(6), 830–847. ISSN: 0165-5515. <https://doi.org/10.1177/016551518761011>. (visited on 10/10/2020).
- Jadidi, M. et al. (2018). Gender disparities in science? Dropout, productivity, collaborations and success of male and female computer scientists. *Advances in Complex Systems*, 21(3), 1750011. <https://doi.org/10.1142/S0219525917500114>. arXiv: 1704.05801.
- Ke, G. et al. (2017). LightGBM: A highly efficient gradient boosting decision tree. In I. Guyon et al. (Eds.), *Advances in neural information processing systems* (Vol. 30, pp. 3146–3154). Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf>.
- Kim, J., & Kim, J. (2018). The impact of imbalanced training data on machine learning for author name disambiguation. *Scientometrics*, 117, 511–526. <https://doi.org/10.1007/s11192-018-2865-9>.
- Kim, J., & Kim, J. (2020). Effect of forename string on author name disambiguation. *Journal of the Association for Information Science and Technology*, 71(7), 839–855. <https://doi.org/10.1002/asi.24298>. eprint: <https://asistdl.onlinelibrary.wiley.com/doi/p>.
- Louppe, G. et al. (2016). Ethnicity sensitive author disambiguation using semi-supervised learning. In A. C. Ngonga Ngomo & P. Křemen (Eds.), *Knowledge engineering and semantic web. KESW 2016* (Vol. 649). Communications in Computer and Information Science. Springer, Cham. https://doi.org/10.1007/978-3-319-45880-9_21.

- Ma, Y., Wu, Y., & Lu, C. (2020). A graph based author name disambiguation method and analysis via information theory. *Entropy*, 22(4), 416. arXiv: 1710.085050.
- Mihaljević, H., & Santamaría, L. (2020). Measuring and analyzing the gender gap in science through the joint data-backed study on publication patterns. In M. F. Roy & C. Guillopé (Eds.), *A global approach to the gender gap in mathematical, computing, and natural sciences. How to measure it, how to reduce it?*. <https://doi.org/10.5281/zenodo.3697223>.
- Mihaljević-Brandt, H., Santamaría, L., & Tullney, M. (2016). The effect of gender in the publication patterns in mathematics. *PLoS One*, 11(10), 1–23. arXiv: 1710.085051.
- Milojević, S. (2013). Accuracy of simple, initials-based methods for author name disambiguation. *Journal of Informetrics*, 7(4), 767–773. ISSN: 1751-1577. <https://doi.org/10.1016/j.joi.2013.06.006>.
- Moed, H. F. & Halevi, G. (Dec. 1, 2014). A bibliometric approach to tracking international scientific migration. *Scientometrics*, 101(3), 1987–2001. ISSN: 1588-2861. <https://doi.org/10.1007/s11192-014-1307-6>.
- Müller, M. -C., Reitz, F., & Roy, N. (June 2017). Data sets for author name disambiguation: An empirical analysis and a new resource. *Scientometrics*, 111(3), 1467–1500. ISSN: 1588-2861. <https://doi.org/10.1007/s11192-017-2363-5>.
- Newman, M. E. J. (2004). Coauthorship networks and patterns of scientific collaboration. In *Proceedings of the national academy of sciences* (Vol. 101, No. suppl 1, pp. 5200–5205). ISSN: 0027-8424, 1091-6490. <https://doi.org/10.1073/pnas.0307545100>. arXiv: 1710.085052.
- Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3), 036106. arXiv: 1710.085053.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144). San Francisco, CA, USA, August 13–17, 2016.
- Shin, D. et al. (July 1, 2014). Author name disambiguation using a graph model with node splitting and merging based on bibliographic information. *Scientometrics*, 100(1), 15–50. ISSN: 1588-2861. <https://doi.org/10.1007/s11192-014-1289-4>. (visited on 10/10/2020).
- Smalheiser, N. R., & Torvik, V. I. (2009). Author name disambiguation. *Annual Review of Information Science and Technology*, 43(1), 1–43. ISSN: 0066-4200. <https://doi.org/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. arXiv: 1710.085054.
- Šubelj, L. (2019). Label propagation for clustering. In *Advances in network clustering and blockmodeling* (pp. 121–150). Wiley, Chap. 5. ISBN: 9781119483298. <https://doi.org/10.1002/9781119483298.ch5>.
- Sugimoto, C. R., Robinson-García, N., & Costas, R. (2016). Towards a global scientific brain: Indicators of researcher mobility using co-affiliation data. Paper presented at the OECD Blue Sky III Forum on Science and Innovation Indicators, Ghent, September 19–21. arXiv preprint arXiv:1609.06499.
- Tkaczyk, D. et al. (Apr. 2017). CeON/CERMINE: CERMINE 1.13. Version cermine-parent-1.13. <https://doi.org/10.5281/zenodo.569829>.
- Treeratpituk, P. & Giles, C. L. (2009). Disambiguating authors in academic publications using random forests. In *Proceedings of the 9th ACM/IEEE-CS joint conference on digital libraries. JCDL '09* (pp. 39–48). Austin, TX, USA: Association for Computing Machinery. ISBN: 9781605583228. <https://doi.org/10.1145/1555400.1555408>.
- Wang, J., et al. (2012). A boosted-trees method for name disambiguation. *Scientometrics*, 93(2), 391–411. arXiv: 1710.085055.
- Wang, X. et al. (Dec. 2011). ADANA: Active name disambiguation. In *2011 IEEE 11th international conference on data mining (ICDM)* (pp. 794–803). Vancouver, BC, Canada: IEEE. ISBN: 978-1-4577-2075-8 978-0-7695-4408-3. <https://doi.org/10.1109/ICDM.2011.19>. arXiv: 1710.085056 (visited on 10/11/2020).
- Xie, I. & Matusiak, K. K. (2016). Chapter 8—User needs and search behaviors. In I. Xie & K. K. Matusiak (Eds.), *Discover digital libraries* (pp. 231–253). Oxford: Elsevier. ISBN: 978-0-12-417112-1. <https://doi.org/10.1016/B978-0-12-417112-1.00008-9>. arXiv: 1710.085057.