A novel topic modeling based weighting framework for class imbalance learning

Sudarsun Santhiappan
Department of Computer Science and Engineering
IIT Madras
Chennai, India
sudarsun@cse.iitm.ac.in

Jeshuren Chelladurai
Department of Computer Science and Engineering
IIT Madras
Chennai, India
jeshuren@cse.iitm.ac.in

Balaraman Ravindran
Department of Computer Science and Engineering
Robert Bosch Centre for Data Science and AI (RBC-DSAI)
IIT Madras
Chennai, India
ravi@cse.iitm.ac.in

ABSTRACT

Classification of data with imbalance characteristics has become an important research problem, as data from most of the real-world applications follow non-uniform class distributions. A simple solution to handle class imbalance is by sampling from the dataset appropriately to compensate for the imbalance in class proportions. When the data distribution is unknown during sampling, making assumptions on the distribution requires domain knowledge and insights on the dataset. We propose a novel unsupervised topic modeling based weighting framework to estimate the latent data distribution of a dataset. We also propose *TODUS*, a topics oriented directed undersampling algorithm that follows the estimated data distribution to draw samples from the dataset. TODUS minimizes the loss of important information that typically gets dropped during random undersampling. We have shown empirically that the performance of TODUS method is better than the other sampling methods compared in our experiments.

CCS CONCEPTS

• **Mathematics of computing** → **Resampling methods**; • **Computing methodologies** → *Topic modeling*; Supervised learning by classification;

KEYWORDS

Class imbalance learning, Data distribution estimation, Directed undersampling, Topic modeling

CoDS-COMAD '18, January 11–13, 2018, Goa, India

ACM ISBN 978-1-4503-6341-9/18/01…$15.00
https://doi.org/10.1145/3152494.3152496

ACM Reference Format:
Sudarsun Santhiappan, Jeshuren Chelladurai, and Balaraman Ravindran. 2018. A novel topic modeling based weighting framework for class imbalance learning. In *CoDS-COMAD '18: The ACM India Joint International Conference on Data Science & Management of Data, January 11–13, 2018, Goa, India.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3152494.3152496

1 INTRODUCTION

Learning from imbalanced datasets has become an important research area as all practical data sets have inherent imbalance characteristics. Credit card fraud classification, classifying cancerous patients from non-cancerous, network anomaly detection, factory production defect classification, conversion of clickable online ads are some of the examples of binary class imbalance problems. Multi-class problems like disease classification using ICD-10[1] codes, job occupation classification using O*Net[2] codes suffer from severe class distribution skew leading to hard multi-class imbalance problems.

Non-uniform class proportions lead to poor classification performance [16], as most of the classifiers in their simplest form assume uniform class distribution. Several methods to address the class imbalance condition are available in the literature [5, 11]. Typically, the methods are categorized into sampling methods, cost-sensitive methods, kernel methods and active learning methods. Sampling based class imbalance methods modify the data set distribution by undersampling, oversampling or synthetic oversampling to induce artificial balance in class proportions. Random oversampling from minority class, suffers from overfitting problem [21]. Synthetic oversampling is non-trivial for the additional effort towards identification and cleansing of synthetic samples that lead to overfitting.

Random undersampling from majority class has been the most popular technique for its simplicity and speed. But, instead of random undersampling, where there is a possibility of losing a good portion of information about the majority class, directed or informed undersampling methods [9] were proposed. They perform smart selection of candidate data

[1] http://www.cdc.gov/nchs/icd/icd10cm.htm
[2] http://www.onetonline.org/

points from the majority class based on data characteristics and domain specific insights.

The rationale for undersampling is typically derived in terms of: a) *data clusters representatives*, where pockets of data points are represented by a single representative point and the others from the same pocket become redundant, b) *data points closer to classifier decision boundary*, which serve as the key ingredient for the construction of decision boundary, while also making the other data points that are away from the decision boundary redundant, c) *misclassified data points*, where an iterative method like boosting, upweights them to force the classifier to bias towards them, d) *noisy data points*, where cleaning methods like OSS identify and prune them from the training dataset [1, 20, 26, 29].

For a typical directed undersampling task, it is assumed that the samples drawn with replacement from the majority class are representative of the original distribution, such that a probabilistic sampler can pick the required number of data points to balance against the size of minority class. Instead of assuming uniform distribution for the majority class data points, we propose to allow the probabilistic sampler to pick the required number of data points based on the estimated data distribution. The estimated data distribution assigns higher probability for important data points, as identified by topic modeling [13], and hence the chance of losing those instances during random undersampling is minimized.

Topic models are statistical models for discovering latent factors that influence the data distributions. Topic modeling was originally proposed to discover latent topics occurring in a text corpus, where a text document is assumed to be a mixture of latent topics and each latent topic generates a vocabulary of terms. Although developed for text processing, the method can be applied to general data [24], where feature values are non-negative and could be described by a mixture of conditionally independent multinomial distributions. For the general data setting, assuming feature values to be non-negative is not a strong limitation as majority of the enterprise data features are based on one of: counting, boolean indicators or quantitative measurements.

We propose a novel unsupervised topic modeling based data weighting framework for imbalanced binary classification tasks, where we compute the data distribution by marginalizing the joint distribution of data points and latent topics estimated by topic modeling. The weighting framework consists of the following steps:

(1) Represent data as a matrix with features as rows and data points as columns
(2) Run topic modeling to estimate the probabilistic factorizations
(3) Estimate data distribution by marginalizing data-topic joint distribution
(4) Compute weights for the majority and minority data points independently as a function of the estimated data distribution
(5) Normalize the majority data weights to make it a probability distribution
(6) Perform undersampling from the majority class following the majority class data distribution to balance against the size of minority class.

The main contributions of this work are summarized as follows:

- A novel unsupervised weighting framework for estimating data distribution based on topic modeling.
- *TODUS*, a novel directed undersampling algorithm, which minimizes information loss that typically occur during random undersampling.
- A novel rationale based on topic modeling, for directed undersampling from the majority class following the estimated data distribution.

The remainder of the paper is organized as follows. In Section 2, we present some of the prior works on class imbalance learning through sampling methods. In Section 3, we describe the proposed topic modeling based weighting framework, where we compute data point weights by estimating the data distribution using topic modeling. Section 4 describes TODUS, a directed undersampling algorithm, which generates a balanced training corpus by undersampling the majority class based on the data distribution estimated by the weighting framework. Section 5 describes the dataset selection, experiment setup and performance comparison of several sampling methods against TODUS. Finally, we present concluding remarks in Section 6.

2 BACKGROUND

The simplest approach to solving class imbalance problems is to handle the imbalance directly by adjusting the sample population through oversampling (sampling with replacement) or undersampling (eliminating samples to reduce population count) the class populations. Random oversampling follows naturally from its description by augmenting the original minority set with replications of selected minority samples. Random undersampling eliminates data from the original data set. Although oversampling and undersampling methods appear to be functionally equivalent, each method introduces its own set of problematic consequences hindering the learning process [7, 21]. In case of undersampling, removing examples from the majority class may cause the classifier to miss important concepts pertaining to the majority class. Whereas in oversampling, multiple instances of certain examples become "tied," leading to overfitting [21]. Although the training accuracy may be higher in this scenario, the classification performance on the unseen testing data is generally far worse [14].

Directed undersampling based on EasyEnsemble and BalanceCascade [20] overcomes the deficiency of information loss introduced in the traditional random undersampling method. Another example of informed undersampling uses the K-nearest neighbor (KNN) classifier to achieve undersampling. Based on the characteristics of the given data distribution, four KNN undersampling methods were proposed [29], namely, NearMiss-1, NearMiss-2, NearMiss-3, and the "most distant" method. The One-Sided Selection (OSS) method

[17] on the other hand selects a representative subset of the majority class and combines it with the set of all minority examples to form a preliminary set, which is further refined by using data cleaning techniques. Cluster Centroids is a cluster based undersampling method [28], where the required k majority points are extracted by choosing the centroids of k clusters estimated by k-means algorithm.

An inverse random under sampling [25] method is proposed for class imbalance learning, where several distinct training sets are constructed by severely undersampling the majority class to sizes smaller than the minority class, to bias the learned decision boundaries towards the minority class.

Synthetic Minority Over-sampling Technique (SMOTE) [6] generates new synthetic examples along the line between the minority examples and their selected nearest neighbors. Although SMOTE makes the decision regions larger and less specific, the overfitting problem of oversampling persists. More grave is the possibility of minority class noise getting synthetic oversampled. To overcome these issues, only selected sub-samples of the minority class are subjected to synthetic sample generation. Borderline-SMOTE [10] uses only the minority samples near the decision boundary to generate new synthetic samples. MWMOTE [2] generates synthetic samples based on hard-to-learn informative minority class samples by assigning them weights according to their euclidean distance from the nearest majority class samples. SCUT [1] over samples minority class examples through the generation of synthetic examples and employs cluster analysis in order to undersample majority classes. In addition, it handles both within-class and between-class imbalance.

Data cleaning techniques, such as Tomek links [27] identify data instances being border points or noise to "cleanup" unwanted overlapping between classes after synthetic sampling. Tomek links are then removed until all minimally distanced nearest neighbor pairs are of the same class. Some representative work in this area includes the Condensed Nearest Neighbor rule and Tomek Links (CNN+Tomek) integration method [3], the Neighborhood Cleaning Rule [18] based on the Edited Nearest Neighbor (ENN) rule—which removes examples that differ from two of its three nearest neighbors.

An adaptive sampling with optimal cost [23] for class imbalance learning is proposed to adaptively oversample the minority positive examples and undersample the majority negative examples, forming different sub-classifiers by different subsets of training data with the best cost ratio adaptively chosen, and combining these sub-classifiers according to their accuracy to create a strong classifier. The sample weights are computed based on the prediction probability of every sample, by a pair of induced SVM classifiers built on two equal sized partitions of the training instances.

Weighted Extreme Learning Machines (ELM) [8, 30] is proposed as a generalized cost sensitive learning method to deal with imbalanced data distributions, where weights are assigned to every training instance based on users' needs. Although, per-sample weights are possible, the authors proposed to use class proportion as the common weight to every sample from a class. They also proposed an alternate weighting scheme that uses golden ratio in computing the common weights for the majority classes. An adaptive semi-unsupervised weighted oversampling (A-SUWO) method [22] is proposed for imbalanced datasets, which clusters the minority instances using a semi-unsupervised hierarchical clustering approach and adaptively determines the size to oversample each sub-cluster using its classification complexity and cross validation. The minority instances are weighted based on their Euclidean distance to the majority class based on which they are oversampled.

3 TOPIC MODELING BASED WEIGHTING FRAMEWORK

3.1 Weighting Model

Aspect model [12] is a latent variable model for co-occurrence data, which associates an unobserved class variable with each observation. Probabilistic Latent Semantic Analysis (PLSA) [13] is an extension of aspect models for NLP and machine learning tasks for text data. Although the technique is developed for text data, it can be applied on general multinomial data distributions as well. The latent topics estimated by the PLSA modeling on multinomial data can be interpreted as some kind of clustering [15] on the dataset. Given a term-document matrix (TDM), PLSA factorizes it using Expectation-Maximization (EM) into: a) topic-conditional density of terms, b) topic-conditional density of documents and c) topic priors.

An alternate approach to topic modeling is to factorize TDM using Latent Dirichlet's Allocation (LDA) [4], which is claimed to not suffer from the overfitting problem that arises with PLSA modeling. LDA is a generative model on $P(\mathcal{F}, \mathcal{Z})$, where it attempts to backtrack from the data points to find a set of topics that are likely to have generated the collection. There are no direct ways to estimate $P(\mathcal{D})$ from this generative model. We chose to proceed with PLSA modeling for its overfitting characteristics, as the objective is to only estimate the data distribution of the given dataset and not generalization to unseen data. PLSA generates soft clusters of data points by estimating the membership of every data point in a cluster, where each cluster is a representation of a latent topic. It was sufficient for us to fit the PLSA model that gave the best clusters for the given training data, as we rank ordered the data points based only on how unambiguously we could place a data point in a cluster. Besides LDA, there may be other generative models that could generate similar effect.

Consider a general dataset $\mathcal{D} = \{d_1, d_2, ..., d_N\}$ sampled from a p-dimensional feature space $\mathcal{F} = \{f_1, f_2, ..., f_p\} \in \mathbb{R}_+^p$. The objective is to estimate the weight w_i for every data point d_i in the dataset. We assume that the feature values are non-negative $\forall f_{ij} \geq 0$, as PLSA modeling assumes a mixture of conditionally independent multinomial distribution on the data represented as a TDM. An aspect model associates an unobserved class variable $z \in \mathcal{Z} = \{z_1, z_2, ..., z_k\}$ with each observation. A class z_k can be regarded as a concept, a data sample refers to. Every data sample can be modeled

as a mixture of multiple concepts to different extents. Using these definitions, a generative model can be defined for the observation pair $\langle d_i, f_j \rangle$ by the following scheme as suggested in PLSA modeling:

(1) Pick a latent class z_k with probability $P(z_k)$
(2) Generate a feature f_j with probability $P(f_j|z_k)$
(3) Select a data point d_i with probability $P(d_i|z_k)$

A joint probability model over $\mathcal{D} \times \mathcal{F}$ is defined by the mixture

$$P(\mathcal{D}, \mathcal{F}) = \sum_{z \in \mathcal{Z}} P(z)P(\mathcal{D}|z)P(\mathcal{F}|z) \tag{1}$$

We can estimate the data distribution $P(\mathcal{D})$ from the data-topic joint distribution $P(\mathcal{D}, \mathcal{Z})$ by marginalizing on $\mathcal{Z}$.

$$P(\mathcal{D}) = \sum_{z \in \mathcal{Z}} P(\mathcal{D}, z) \tag{2}$$

$$P(\mathcal{D}) = \sum_{z \in \mathcal{Z}} P(z)P(\mathcal{D}|z) \tag{3}$$

$$\implies P(d_i) = \sum_{z \in \mathcal{Z}} P(z)P(d_i|z) \tag{4}$$

$P(d_i|z_j)$ is the confidence score of putting the data point d_i in the j^{th} topic and when we sum up all these confidence scores of a data point d_i, we get a measure of how easy or difficult it is to place a data point d_i in a cluster confidently. We have used this idea to rank order the data point. We compute the data point weight w_i by transforming the prior probability through a function W as $W : P(d_i) \mapsto w_i$. The function W can even be an identity function $w_i = P(d_i)$, where the prior probabilities are directly used as sample weights. The estimated data point weights $w_i \in W$ would then be normalized to setup the probability distribution for the data samples, based on which samples could be drawn.

3.2 Characteristics of Estimated Sample Priors

We analyzed the estimated data distribution against different number of topics and sample sizes, while setting the parameters of PLSA to their default values. We used the English version of Europarl[3] corpus to build the PLSA[4] model, followed by estimation of the data distribution. We used the bag-of-words representation for the texts from the Europarl corpus, where stop words were filtered from the unique words list.

A plot of the estimated data distribution against number of topics {5, 25, 50, 75} for a Europarl sample of size 100,000 is shown as the first plot in Figure 1. The second plot in Figure 1 shows the exponential decay characteristics, limited to the top 2500 $P(d_i \in \mathcal{D})$ estimates for three different sample sizes {3K, 50K, 100K}. We observed shape-similarity in the exponential decay characteristics of the estimated data distribution against different corpus sizes and different number of topics. The X-axes of both the plots have the data point indices sorted by $P(d_i)$ in descending order. In the second plot, the Y-axis represents the $P(d_i)$ values in the log scale. The range of the exponential decay is dependent on the size of the corpus, where the range size is found to be inversely proportional to the corpus size.

The data point weights estimated as a function of the prior probabilities are observed to be insensitive to the classes from where the samples are drawn. The first plot in Figure 2 demonstrates the distribution of data points $P(\mathcal{D})$ for one *Abalone*[5] dataset containing 311 minority and 3030 majority samples. It is apparent that the majority samples overshadow the minority samples in the top portion of the response curve. This is due to the majority samples taking precedence over the minority samples due to population difference. In the mid and lower parts of the response curve, we observe the proportional distribution of majority and minority samples. To overcome the overshadowing problem, we considered the ranking order of minority and majority samples individually and scaled them independently to generate a decay response for themselves. The second plot of Figure 2 shows the distribution of minority $P(\mathcal{D}_{min})$ and majority $P(\mathcal{D}_{maj})$ data points as independent exponential decays. The independent computation of minority and majority data distributions help the important samples of the minority class to get more attention while using the data weights aware classifiers. The second plot in Figure 2 shows the data point weights estimated for the minority and majority classes individually by the weighting framework through application of *min-max* normalization as the transformation function. The estimated weights for a majority and minority samples can be normalized to make probability distributions. A random sampler can follow the estimated distributions to sample from majority and minority classes independently.

4 TOPICS ORIENTED DIRECTED UNDER SAMPLING (TODUS) ALGORITHM

TODUS is a directed undersampling method, which undersamples from the majority class dataset to balance against the size of minority class dataset in a corpus. TODUS is different from random undersampling as the under sampling is based on a data distribution estimated through topic modeling based weighting framework instead of assuming uniform sample distribution as in random undersampling. Consider a binary classification dataset $D = D_{maj} \cup D_{min}$, the objective of TODUS is to produce a balanced dataset D_{TODUS}, which has uniform distribution of classes from the original dataset D. This is achieved by running topic modeling with the dataset to estimate the prior probabilities $P(d \in \mathcal{D})$ of every data point d from the dataset $\mathcal{D}$. A random sampler can then follow the estimated probability distribution to draw the required number of majority samples to match the number of samples present in the minority class dataset partition.

Alternately, the data distribution of the majority class data points could be estimated independently, as we don't use the prior probability of minority data points to generate the rebalanced dataset. We did not report our findings as the observed performance improvement was found to be lesser.

[3] http://www.statmt.org/europarl/
[4] https://github.com/lizhangzhan/plsa
[5] https://archive.ics.uci.edu/ml/datasets/abalone

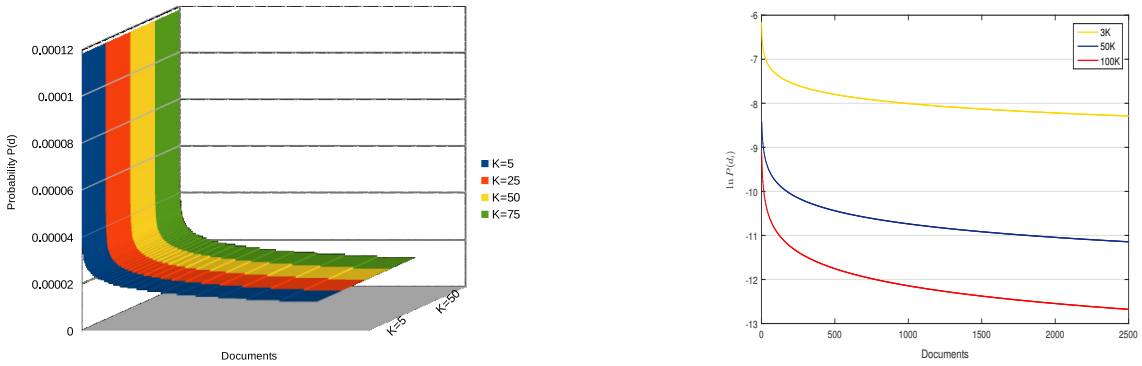

Figure 1: The first plot shows the estimated data distribution $P(\mathcal{D})$ against different number of topics K. The corpus size was 100K, but truncated to 20K items for brevity. The thickness of the lines are just for better visibility. The second plot shows the decay characteristics of estimated data point priors against different corpus sizes. The X-axes of the plots are the data instance indices sorted by $P(d_i)$ in descending order and the Y-axes are the $P(d_i)$ values in the linear and natural log scales respectively.

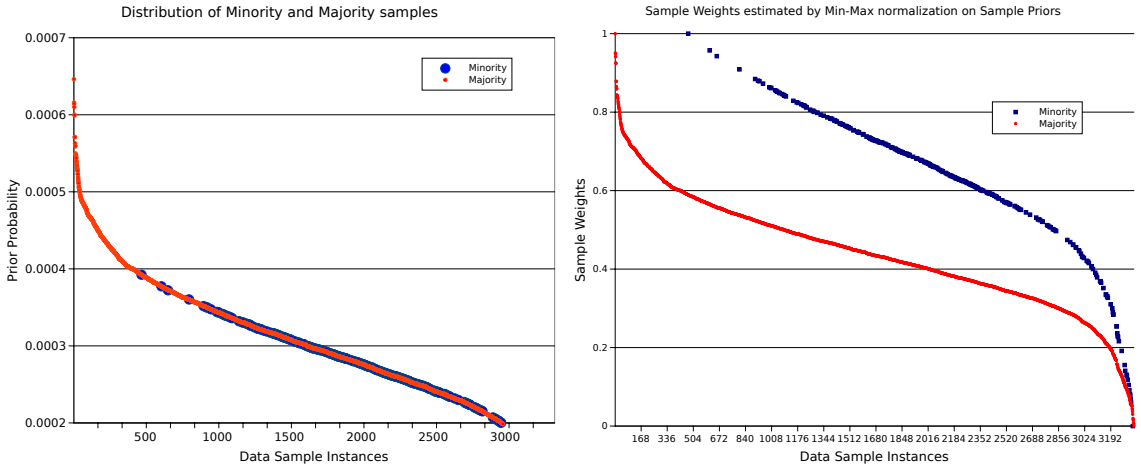

Figure 2: The first plot shows the distribution of minority and majority samples following a smooth a decay response curve. The top portion of the plot is observed to be completely dominated by the majority class samples. The second plot shows the data weights for majority and minority classes estimated independently as a function of data point priors. The X-axis is the data point instances sorted by prior probabilities $P(d_i)$ in descending order.

Algorithm 1 describes the TODUS algorithm for binary classification datasets. Step 1 combines the majority and minority class data points into one dataset $\mathcal{D}$. Step 2 assumes that the feature space of the majority samples from the dataset to be $\mathcal{F}$ and the latent topics as $\mathcal{Z}$. Next, we run PLSA modeling in step 3 on the entire dataset to estimate the factors $P(\mathcal{Z})$, $P(\mathcal{F}|\mathcal{Z})$ and $P(\mathcal{D}|\mathcal{Z})$ based on the symmetric aspect model [13]. We chose PLSA modeling for the overfit characteristics as our objective was to estimate the apparent data distribution of the training data and not generalization for the unseen data points. Step 4 estimates the data point priors by marginalizing the joint distribution of data samples and latent topics, on topics, to provide us with the apparent data distribution. Step 5 splits the data distribution into majority and minority priors. Steps 6 applies min-max normalization on the majority priors to normalize the values to the $[0, 1]$ closed interval. The estimated weights W_{maj} are then normalized to make it a probability distribution again as $P(\mathcal{D}_{maj})$. In step 8, a probabilistic sampler follows the estimated majority class data distribution to draw the required number of data points from the majority class to match the population size of minority samples. Step 9 combines the undersampled majority data points and the actual minority data points to compose the TODUS sample set. This modified dataset is typically twice the size of minority samples. We have evaluated the quality of TODUS generated rebalanced samples using several datasets and reported the results in Section 5.

For plotting the graphs in Figure 2, we computed the minority data weights W_{min} similar to W_{maj} as in step 6 of

Data: $\mathcal{D} = \mathcal{D}_{maj} \cup \mathcal{D}_{min}$
Result: $\mathcal{D}_{TODUS}$

1 $\mathcal{D} = \mathcal{D}_{maj} + \mathcal{D}_{min}$
2 Let $\mathcal{F}$, $\mathcal{Z}$ be the features and latent topics of $\mathcal{D}$
3 Run PLSA modeling on $\mathcal{D}$ as $P(\mathcal{F}, \mathcal{D}) = \sum_{z \in \mathcal{Z}} P(z)P(d \in \mathcal{D}|z)P(\mathcal{F}|z)$
4 Now $P(d \in \mathcal{D}) = \sum_{z \in \mathcal{Z}} P(z)P(d \in \mathcal{D}|z)$ `/* estimate the sample priors by marginalizing the joint distribution */`
5 Split $P(\mathcal{D}) \Longrightarrow P(\mathcal{D}_{min}) \cup P(\mathcal{D}_{maj})$, where $P(\mathcal{D}_{min}) = P(d \in \mathcal{D}_{min})$ and $P(\mathcal{D}_{maj}) = P(d \in \mathcal{D}_{maj})$ `/* split the data distribution into majority and minority priors */`
6 $W_{maj} = MinMaxNormalize\left(P(\mathcal{D}_{maj})\right)$ `/* min-max normalize the majority data point priors separately to estimate weights` $W \in [0, 1]$ `*/`
7 $P(\mathcal{D}_{maj}) \leftarrow Normalize(W_{maj})$ `/* normalize` W_{maj} `to estimate` $P(\mathcal{D}_{maj})$ `*/`
8 $\mathcal{D}_{maj}^{undersampled} \sim \mathcal{D}_{maj}$ with $P(\mathcal{D}_{maj})$ `/* draw` $\|\mathcal{D}_{min}\|$ `samples from the majority class using` $P(\mathcal{D}_{maj})$ `*/`
9 $\mathcal{D}_{TODUS} = \mathcal{D}_{maj}^{undersampled} + \mathcal{D}_{min}$

Algorithm 1: TODUS Algorithm for Binary Class Datasets

Algorithm 1. We then computed the data distribution of the minority class data points $P(\mathcal{D}_{min})$ by normalizing W_{min} as in step 7. Computing the weights W_{maj} and W_{min} independently avoids the problem of majority points shadowing minority points by its sheer magnitude as shown in Figure 2.

5 EXPERIMENTS

We considered several multi-class datasets from the UCI repository[6], where we converted the multi-class problem into one-vs-rest binary classification. We considered the *one* in one-vs-rest configuration, as the positive (minority) class and the aggregate of the *rest* as negative (majority). This transformation resulted in imbalanced data sets, which is of interest to our problem.

The premise of topic modeling is to represent every data point as a topics-distribution. The general intuition is to expect a similar topics-distribution for every data point belonging to a particular class. Combining data points from multiple classes into one larger class may affect the validity of this intuition. Our method is not affected by this caveat as we don't use the class information of data points when we estimate the data distribution.

In many practical scenarios, the datasets are counts based or discretized independent measurements, which can be modeled as a mixture of multinomial distributions. The assumption holds good for categorical features as well when they are represented in *one-hot encoding*. It is not easy to verify the conditional independence assumption of the PLSA modeling on the dataset ahead of usage, but we can assume this for counts and independent measurements type datasets. The assumption may become invalid with transformed datasets such as *embeddings* as the data dimensions are no longer necessarily independent.

The datasets for experimentation were selected based on the choice of the standard benchmark datasets that class imbalance learning researchers have used in the literature. We could not find any online data repository, exclusively for class imbalance learning research.

Table 1 lists the selected datasets with their meta information. The suffix digit mentioned in the dataset name is the class id that is considered as positive (minority) class. The rest of the classes are aggregated into one class, which becomes the negative (majority) class. As an exception in PageBlocks data set, the classes 3, 4 and 5 were combined to get the positive (minority) and the rest were taken as negative (majority). The last column is the minority to majority sample size ratio, which identifies the class imbalance as marginally to modestly imbalanced. The column "C" declares a compact index code for the datasets, which are referred in the performance evaluation tables. Column "D" lists the dimensionality of the datasets. We have presented the time taken for TODUS preprocessing for all the dataset in column "PP" of Table 1.

We have selected some of the representative directed undersampling, random oversampling and SMOTE methods for our experimentation as listed in Table 2. We chose to limit our focus only to sampling methods as comparing against state-of-art techniques such as cost-sensitive methods, ensemble methods, and kernel methods would make our experimental results less useful.

Classifier: We chose decision trees as the method for learning the classification model on the TODUS-rebalanced dataset. We used the classification performance of the learned model as a surrogate measure for measuring the quality of the samples generated. The assumption we made is that the sample quality correlates positively with classification performance. We preferred using a decision tree, as it does not require special parameter tuning.

Performance Metric: To evaluate the performance of sampling, we used the correctness of classification as the surrogate metric, for a model learned from the majority-undersampled datasets. In most of the practical applications, the minority class performance is more critical than the majority. At the same time, the majority class performance should not be traded off for changing the bias towards minority class. When the imbalance ratio is $R : 1$ and the F_1 scores are F_1^{maj} and F_1^{min}, we computed Weighted Average F_1 (WAF_1) as:

$$WAF_1 = \frac{F_1^{maj} + R * F_1^{min}}{1 + R} \tag{5}$$

[6] http://archive.ics.uci.edu/ml

Dataset	*C*	*D*	*Size*	*Maj*	*Min*	*Ratio*	*PP*
Satimage4	S4	36	4435	4020	415	1:9	2.1
Vehicle1	V1	18	846	634	212	1:3	.26
Ecoli4	E4	8	336	301	35	1:10	.19
Car3	C3	6	1728	1659	69	1:25	.24
Pima1	P1	8	768	500	268	1:2	.32
Haberman2	H2	4	306	225	81	1:3	.07
CMC2	C2	9	1473	1140	333	1:4	.27
Pageblocks345	PB	10	5473	5242	231	1:25	.85
Wisconsin	WS	10	683	444	239	1:2	.23
Yeast4	Y4	8	528	477	51	1:9	.17
Vehicle3	V3	18	846	634	212	1:3	.27
Vehicle2	V2	18	846	628	218	1:3	.28
Vehicle0	V0	18	846	647	199	1:3	.29
Yeast2Vs4	Y2	8	514	463	51	1:9	.23
Yeast1	Y1	8	1484	1054	430	1:3	.29
Ecoli1	E1	8	336	259	77	1:3	.11
Ecoli2	E2	8	336	284	52	1:6	.08
Ecoli3	E3	8	336	301	35	1:9	.11
WineQuality4	W4	13	1599	1546	53	1:30	.32
LetterJ	LJ	16	20000	19253	747	1:25	.33
ConnectDraw	C4	42	67557	61108	6449	1:10	64
Poker Hand	PK	10	1025010	976182	48828	1:20	262

Table 1: UCI Datasets used for performance evaluation along with the respective TODUS preprocessing (column *PP*) time in seconds.

Methods	*Code*
Random Undersampling	RUS
Cluster Centroids	CC
Near Miss 1	N1
Near Miss 2	N2
Near Miss 3	N3
Condensed Nearest Neighbors	CNN
One-sided Sampling	OSS
TODUS	TOD
Without Sampling	WS
Random Oversampling	ROS
SMOTE	SM

Table 2: Directed Undersampling and Oversampling methods

to assign more importance to F_1^{min} score, while computing the performance summary. We used two-sample $t-$test to measure the statistical significance of weighted average F_1-score measured for TODUS against the other listed methods.

Evaluation: We repeated 5-fold cross validation for four times to get the performance measures for 20 runs in total. During every fold, we used TODUS method to sample from the training split to generate a balanced training sample. We trained a J48 classifier on the balanced dataset and evaluated the performance of the classifier model against the testing split of the same fold, using weighted average F_1 score as the performance measure. Likewise, the samples generated by the other methods listed in Table 2 for every cross validation fold were used for training the respective J48 classifiers. The classification models thus built were tested against the respective testing splits to record the classification performance. We used the Python implementation[19] of the methods in Table 2 to run the experiments. Table 3 tabulates the weighted average F_1-scores for each dataset across all the methods. We ran two-sample $t-$test on the weighted average F_1-scores to study the significance of the TODUS performance against the listed methods using MATLAB's *ttest2*[7] API. Table 3 tabulates the performance of TODUS by setting a null-hypothesis for similar performance and the alternate hypothesis for TODUS being better than the other method in comparison. We tabulated the result of the tests as Win, Tie, Loss, where TODUS has outperformed, performed at-par and underperformed respectively. All the experiments discussed in this paper were performed in MATLAB and Python on an Intel Core i5 CPU with 8 GB of RAM.

Interpretation: is observed in Table 3 that TODUS performs better against other balancing directed sampling methods. TODUS has outperformed or performed at-par with all the methods compared based on the number of wins. TODUS was observed to be better than both random oversampling and SMOTE methods in terms of top-3 positions as well besides number of wins. The performance of TODUS in terms of top-3 positions, is lower compared to the cleansing method OSS, where training set balancing is not a requirement and the entire dataset is available for building the models. It is interesting to notice from the $t-$test in Table 3 that TODUS outperformed every other sampling method, if AUC (Area under ROC) was used as the evaluation metric. When the corpus size gets larger, TODUS is observed to scale well but the cleansing methods could not scale. From the experimental results, it is apparent that TODUS is a better sampling strategy to deal with class imbalance learning.

The classification performance of the datasets without any kind of rebalancing (*WS* column in Table 3) is observed to be performing better than any sampling method. But, with a larger imbalanced dataset, it may not be feasible to learn a classifier without undersampling. This fact is apparent for the *Poker* dataset, where the original dataset, oversampling and cleansing methods have all failed.

6 CONCLUSION

We presented a novel weighting framework based on topic modeling, for assigning weights to every sample in a training corpus in an unsupervised fashion. Although, topic modeling was developed for text application, we have successfully demonstrated its use with generic non-negative p-dimensional multinomial datasets $\mathcal{D} \in \mathbb{R}_+^p$. We capitalized on the overfitting characteristics of PLSA modeling in our weighting framework to generate the sample weights, as our objective is

[7] http://in.mathworks.com/help/stats/ttest2.html

DataSet	RUS	CC	N1	N2	N3	CNN	OSS	TOD	WS	ROS	SM
P1	.633	.630	.595	.534	.611	.622	**.639**	**.659**	**.676**	.617	.620
C2	.466	.437	.376	.344	**.481**	.443	.452	**.509**	**.475**	.449	.452
V0	.861	.858	.586	.714	.849	.870	**.883**	.834	**.894**	.881	**.877**
V1	**.617**	.575	.528	.404	**.619**	.578	**.596**	.581	.592	.573	.609
V2	.893	.877	.882	.575	.890	.922	**.926**	.897	**.931**	**.936**	.921
V3	.600	.575	.528	.402	**.621**	**.608**	.589	.586	.592	.579	**.603**
PB	.549	.244	.179	.108	.401	**.757**	**.770**	.344	**.787**	.737	.740
Y1	.541	**.559**	.431	.450	.547	.537	.537	**.610**	**.625**	.538	.525
Y2	.687	.643	.635	.379	.642	.699	.694	.657	**.723**	**.731**	**.764**
Y4	.444	.376	.306	.258	.341	.469	**.489**	.402	.476	**.481**	**.520**
S4	.481	.445	.216	.264	.360	**.567**	**.570**	.488	**.580**	.564	.562
H2	.408	**.439**	.414	.405	**.437**	.428	.424	**.473**	.411	.416	.392
E1	.759	.792	.724	.730	.723	.783	**.804**	**.800**	**.813**	.789	.778
E2	.650	.696	.571	.503	.471	.775	**.770**	.726	**.816**	.764	**.780**
E3	.536	.544	.371	.346	.530	**.619**	**.625**	.566	**.590**	.571	.577
E4	.512	.556	.354	.375	.455	**.601**	**.618**	.547	**.602**	.547	.555
W4	.119	.092	.060	.064	.101	**.155**	.128	.122	.041	**.132**	**.131**
WS	.936	**.938**	**.938**	.935	.906	.915	.924	.935	**.943**	.930	.929
C3	**.421**	.323	.230	.088	.165	.044	.039	**.400**	**.836**	.054	.039
PK	.163	-	.123	-	.160	-	-	**.249**	-	-	-
C4	.322	.182	.225	.180	.304	-	**.358**	**.345**	.194	**.354**	.341
LJ	.509	.258	.257	.191	.286	-	.870	.560	**.895**	**.897**	**.874**
WINS	0	0	0	0	2	1	3	3	9	1	2
TOP3	2	3	1	0	4	6	12	8	15	6	7

Table 3: The table shows the weighted average F_1-scores performance comparison of TODUS against other methods. The best score is underlined and next two scores are highlighted as bold.

	t-test on Weighted Average F1										t-test on AUROC									
DataSet	RUS	CC	N1	N2	N3	CNN	OSS	WS	ROS	SM	RUS	CC	N1	N2	N3	CNN	OSS	WS	ROS	SM
P1	W	W	W	W	W	W	T	T	W	W	W	W	W	W	W	W	W	L	W	W
C2	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
V0	L	T	W	W	T	L	L	L	L	L	T	T	W	W	W	W	W	T	W	W
V1	L	T	W	W	L	T	T	T	T	L	T	W	W	W	T	W	W	L	W	T
V2	T	W	T	W	T	L	L	L	L	L	T	W	W	W	W	T	T	T	T	T
V3	T	T	W	W	L	T	T	T	T	T	T	W	W	W	T	W	W	L	W	W
PB	L	W	W	W	L	L	L	L	L	L	L	W	W	W	T	T	T	T	W	T
Y1	W	W	W	W	W	W	W	T	W	W	W	W	W	W	W	W	W	L	W	W
Y2	L	T	T	W	T	L	L	L	L	L	T	T	W	W	T	W	W	W	W	T
Y4	L	T	W	W	W	L	L	L	L	L	T	W	W	W	W	W	W	T	W	T
S4	T	W	W	W	W	L	L	L	L	L	T	W	W	W	W	W	W	W	W	W
H2	W	T	W	W	T	T	T	T	W	W	W	T	W	W	T	T	T	T	T	T
E1	W	T	W	W	W	T	T	T	T	T	W	W	W	W	W	W	W	T	W	W
E2	W	T	W	W	W	T	T	L	T	L	T	T	W	W	W	T	T	T	T	T
E3	W	T	W	W	W	T	T	T	T	T	T	T	W	W	T	W	W	W	W	W
E4	T	T	W	W	W	T	T	T	T	T	T	T	W	W	T	W	T	W	W	W
W4	T	W	W	W	W	T	T	W	T	T	T	W	W	W	W	W	W	W	W	W
WS	T	T	T	T	W	W	T	T	T	T	T	T	T	T	T	W	T	T	T	T
C3	T	W	W	W	W	W	W	L	W	W	W	W	W	W	W	W	W	L	W	W
PK	W	-	W	-	W	-	-	-	-	-	W	-	W	-	W	-	-	-	-	-
C4	W	W	W	W	W	-	L	W	L	T	W	W	W	W	W	-	W	L	W	W
LJ	W	W	W	W	W	-	L	L	L	L	W	W	W	W	W	-	W	T	W	W
Win	10	10	19	20	15	5	3	3	5	5	9	14	21	20	14	15	15	6	17	13
Tie	7	11	3	1	4	8	10	9	8	7	12	7	1	1	8	4	6	9	4	8
Loss	5	0	0	0	3	6	8	9	8	9	1	0	0	0	0	0	0	6	0	0

Table 4: The tables show the summary of two sample t-test on weighted average F_1-score and AUC respectively of TODUS against other methods with significance level at 0.05.

limited to the given dataset and not generalization to unseen data points. We proposed TODUS, a novel directed undersampling algorithm built on top of the weighting framework and established its performance against other undersampling methods considered. Although the proposed method is not as simple as random under sampling, the extra computation time leads to better selection of data points from the majority class with the topic modeling rationale.

We want to extend the idea of undersampling majority class samples based on the estimated data distribution to also consider resampling of minority class samples. We believe that resampling dataset would be a powerful tool for improving classification performance. Besides resampling, we would also be extending the framework to multi-class imbalanced datasets as a future endeavor. The source codes, curated datasets, results and reports are made available at GitHub[8].

REFERENCES

[1] Astha Agrawal, Herna L. Viktor, and Eric Paquet. 2015. SCUT: Multi-Class Imbalanced Data Classification using SMOTE and Cluster-based Undersampling.. In *KDIR*, Ana L. N. Fred, Jan L. G. Dietz, David Aveiro, Kecheng Liu, and Joaquim Filipe (Eds.). SciTePress, 226–234. http://dblp.uni-trier.de/db/conf/ic3k/kdir2015.html#AgrawalVP15

[2] Sukarna Barua, Md. Monirul Islam, Xin Yao, and Kazuyuki Murase. 2014. MWMOTE-Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning. *IEEE Trans. Knowl. Data Eng.* 26, 2 (2014), 405–425. http://dblp.uni-trier.de/db/journals/tkde/tkde26.html#BaruaIYM14

[3] Gustavo E. A. P. A. Batista, Ronaldo C. Prati, and Maria Carolina Monard. 2004. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *ACM SIGKDD Explorations Newsletter - Special issue on learning from imbalanced datasets* 6, 1 (2004), 20–29. https://doi.org/10.1145/1007730.1007735

[4] D. Blei, A. Ng, and M. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3 (January 2003), 993–1022. http://www.cs.berkeley.edu/~blei/papers/blei03a.ps.gz; http://www.bibsonomy.org/bibtex/21d86d39e0f44b3fa45ff97800b5fa9e8/megmed

[5] Paula Branco, Luís Torgo, and Rita P. Ribeiro. 2016. A Survey of Predictive Modeling on Imbalanced Domains. *ACM Comput. Surv.* 49, 2 (2016), 31:1–31:50. http://dblp.uni-trier.de/db/journals/csur/csur49.html#BrancoTR16

[6] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research* 16 (2002), 321–357. http://www.cs.cmu.edu/afs/cs/project/jair/pub/volume16/chawla02a.pdf

[7] Chris Drummond and R.C. Holte. 2003. C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. *Workshop on Learning from Imbalanced Datasets II* (2003), 1–8. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.68.6858

[8] Xingyu Gao, Zhenyu Chen, Sheng Tang, Yongdong Zhang, and Jintao Li. 2016. Adaptive weighted imbalance learning with application to abnormal activity recognition. *Neurocomputing* 173 (2016), 1927–1935. http://dblp.uni-trier.de/db/journals/ijon/ijon173.html#GaoCTZL16

[9] Haixiang Guo, Yijing Li, Jennifer Shang, Gu Mingyun, Huang Yuanyue, and Gong Bing. 2017. Learning from class-imbalanced data: Review of methods and applications. *Expert Syst. Appl.* 73 (2017), 220–239. http://dblp.uni-trier.de/db/journals/eswa/eswa73.html#GuoLSMYB17

[10] Hui Han, Wenyuan Wang, and Binghuan Mao. 2005. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning.. In *ICIC (1)* (2009-04-01) *(Lecture Notes in Computer Science)*, De-Shuang Huang, Xiao-Ping Zhang, and Guang-Bin Huang (Eds.), Vol. 3644. Springer, 878–887. http://dblp.uni-trier.de/db/conf/icic/icic2005-1.html#HanWM05

[11] Haibo He and Yunqian Ma. 2013. *Imbalanced Learning: Foundations, Algorithms, and Applications* (1st ed.). Wiley-IEEE Press.

[12] Thomas Hofmann. 1998. Unsupervised Learning from Dyadic Data. MIT Press, 466–472.

[13] Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 289–296.

[14] Robert C. Holte, Liane Acker, and Bruce W. Porter. 1989. Concept Learning and the Problem of Small Disjuncts.. In *IJCAI*, N. S. Sridharan (Ed.). Morgan Kaufmann, 813–818. http://dblp.uni-trier.de/db/conf/ijcai/ijcai89.html#HolteAP89

[15] Young-Min Kim, Jean-François Pessiot, Massih-Reza Amini, and Patrick Gallinari. 2008. An extension of PLSA for document clustering.. In *CIKM* (2008-11-10), James G. Shanahan, Sihem Amer-Yahia, Ioana Manolescu, Yi Zhang, David A. Evans, Aleksander Kolcz, Key-Sun Choi, and Abdur Chowdhury (Eds.). ACM, 1345–1346. http://dblp.uni-trier.de/db/conf/cikm/cikm2008.html#KimPAG08

[16] Miroslav Kubat, Robert C. Holte, and Stan Matwin. 1998. Machine Learning for the Detection of Oil Spills in Satellite Radar Images. *Machine Learning* 30, 2-3 (1998), 195–215. http://dblp.uni-trier.de/db/journals/ml/ml30.html#KubatHM98

[17] Miroslav Kubat and Stan Matwin. 1997. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. In *In Proceedings of the Fourteenth International Conference on Machine Learning*. Morgan Kaufmann, 179–186. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.43.4487

[18] Jorma Laurikkala. 2001. Improving Identification of Difficult Small Classes by Balancing Class Distribution.. In *AIME (Lecture Notes in Computer Science)*, Silvana Quaglini, Pedro Barahona, and Steen Andreassen (Eds.), Vol. 2101. Springer, 63–66. http://dblp.uni-trier.de/db/conf/aime/aime2001.html#Laurikkala01; http://dx.doi.org/10.1007/3-540-48229-6_9; http://www.bibsonomy.org/bibtex/299ad2efa02d1ffb29dced2ee0d3a23b4/dblp

[19] Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. 2017. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research* 18, 17 (2017), 1–5. http://jmlr.org/papers/v18/16-365

[20] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. 2006. Exploratory Under-Sampling for Class-Imbalance Learning.. In *ICDM*. IEEE Computer Society, 965–969. http://dblp.uni-trier.de/db/conf/icdm/icdm2006.html#LiuWZ06

[21] David Mease, Aj Wyner, and a Buja. 2007. Boosted classification trees and class probability/quantile estimation. *The Journal of Machine Learning Research* 8 (2007), 409–439. http://dl.acm.org/citation.cfm?id=1248675

[22] Iman Nekooeimehr and Susana K. Lai-Yuen. 2016. Adaptive semi-unsupervised weighted oversampling (A-SUWO) for imbalanced datasets. *Expert Syst. Appl.* 46 (2016), 405–416. http://dblp.uni-trier.de/db/journals/eswa/eswa46.html#NekooeimehrL16

[23] Yuxin Peng. 2015. Adaptive Sampling with Optimal Cost for Class-Imbalance Learning.. In *AAAI*, Blai Bonet and Sven Koenig (Eds.). AAAI Press, 2921–2927. http://dblp.uni-trier.de/db/conf/aaai/aaai2015.html#Peng15

[24] Jonathan K. Pritchard, Matthew Stephens, and Peter Donnelly. 2000. Inference of Population Structure Using Multilocus Genotype Data. *Genetics* 155 (June 2000), 945–959. http://pritch.bsd.uchicago.edu/publications/structure.pdf

[25] Muhammad Atif Tahir, Josef Kittler, and Fei Yan. 2012. Inverse random under sampling for class imbalance problem and its application to multi-label classification. *Pattern Recognition* 45, 10 (2012), 3738–3750. http://dblp.uni-trier.de/db/journals/pr/pr45.html#TahirKY12

[26] Yuchun Tang and Yan-Qing Zhang. 2006. Granular SVM with Repetitive Undersampling for Highly Imbalanced Protein Homology Prediction.. In *GrC*. IEEE, 457–460. http://dblp.uni-trier.de/db/conf/grc/grc2006.html#TangZ06

[27] I. Tomek. 1976. Two Modifications of CNN. *IEEE Transactions on Systems, Man, and Cybernetics* 7(2) (1976), 679–772.

[28] Show-Jane Yen and Yue-Shi Lee. 2009. Cluster-based undersampling approaches for imbalanced data distributions. *Expert Syst. Appl.* 36, 3 (2009), 5718–5727. http://dblp.uni-trier.de/db/journals/eswa/eswa36.html#YenL09; http://dx.doi.org/10.1016/

[8] https://github.com/rise-iil/a-novel-topic-modeling-based-weighting-framework-for-class-imbalance-learning

j.eswa.2008.06.108

[29] J. Zhang and I. Mani. 2003. KNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction. In *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Datasets*. http://www.bibsonomy.org/bibtex/2cf4d2ac8bdac874b3d4841b4645a5a90/diana

[30] Weiwei Zong, Guang-Bin Huang, and Yiqiang Chen. 2013. Weighted extreme learning machine for imbalance learning. *Neurocomputing* 101 (2013), 229–242. http://dblp.uni-trier.de/db/journals/ijon/ijon101.html#ZongHC13; http://dx.doi.org/10.1016/j.neucom.2012.08.010; http://www.bibsonomy.org/bibtex/28207a6ccea04eab1f69459b673524f93/dblp