Distributional Random Oversampling for Imbalanced Text Classification

Alejandro Moreo, Andrea Esuli
Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, Italy
E-mail: {firstname.lastname}@isti.cnr.it

Fabrizio Sebastiani
Qatar Computing Research Institute
Hamad bin Khalifa University
Doha, Qatar
E-mail: fsebastiani@qf.org.qa

ABSTRACT

The accuracy of many classification algorithms is known to suffer when the data are imbalanced (i.e., when the distribution of the examples across the classes is severely skewed). Many applications of binary text classification are of this type, with the positive examples of the class of interest far outnumbered by the negative examples. Oversampling (i.e., generating synthetic training examples of the minority class) is an often used strategy to counter this problem. We present a new oversampling method specifically designed for classifying data (such as text) for which the *distributional hypothesis* holds, according to which the meaning of a feature is somehow determined by its distribution in large corpora of data. Our *Distributional Random Oversampling* method generates new random minority-class synthetic documents by exploiting the distributional properties of the terms in the collection. We discuss results we have obtained on the Reuters-21578, OHSUMED-S, and RCV1-v2 datasets.

1. INTRODUCTION

Many applications of binary text classification exhibit severe data *imbalance*, i.e., are characterized by sets of data in which the examples of one class are far outnumbered by the examples of the other. Such cases are especially frequent in information retrieval and related tasks, where the binary distinction to be captured is between a class of interest and "the rest", i.e., between the (typically few) documents relevant to a certain concept (e.g., as expressed by a query) and the (typically many) documents unrelated to it. This phenomenon is exacerbated in applications of multi-label multiclass (MLMC) text classification, i.e., applications where, given a set $\mathcal{C} = \{c_1, ..., c_{|\mathcal{C}|}\}$ of classes, each document may be labelled by several classes at the same time[1]. In these applications the average prevalence (i.e., relative frequency) of a class is low, since $\mathcal{C}$ typically exhibits a power-law behaviour, with few classes having high prevalence and very many classes having low or very low prevalence.

Severe imbalance is known to degrade the performance of a number of inductive learning algorithms, such as decision trees, neural networks, or support vector machines [8]. The main approaches previously proposed for solving this problem may be grouped into the following classes: [7, 9]: (i) data-level approaches, which perform a random resampling of the dataset in order to rebalance class prevalences (ii) algorithmic approaches, which focus on adapting traditional classification methods to scenarios where data are imbalanced; and (iii) cost-sensitive learning approaches, that combine the data-level and algorithmic approaches by imposing a higher cost on the misclassification of examples from the minority class. We here focus on approaches of type (i), most of which rely on oversampling the minority class (i.e., adding new minority-class training examples, typically duplicates or quasi-duplicates of the existing ones) and/or undersampling the majority class (i.e., removing some majority-class examples from the training set), with the goal of rebalancing the class distribution in the training set.

We propose a novel method based on oversampling the minority class, and specifically designed to deal with types of data (such as text) where the *distributional hypothesis* (according to which the meaning of a feature is somehow determined by its distribution in large corpora of data – see [6]) may be assumed to hold. Our method, dubbed *Distributional Random Oversampling* (DRO), consists of extending the standard vector representation (based on the bag-of-words model) with random latent dimensions based on distributional properties of the observed features. We assign to each document a discrete probabilistic function that operates in a latent space and is queried as many times as desired in order to oversample a given document (i.e., to produce distributionally similar versions of it). Since this generative function is based on the distributional hypothesis, the expectation is that the variability introduced in the newly generated examples reflects semantic properties of the terms that occur in the document being oversampled. We present the results of experiments conducted on popular text classification benchmarks such as Reuters-21578, OHSUMED-S, and RCV1-v2.

Our method is presented in Section 2; Section 3 discusses our empirical results, while Section 4 concludes.

2. THE LATENT SPACE OVERSAMPLING FRAMEWORK

We assume a binary classification context, with classes $\mathcal{C} = \{c, \bar{c}\}$. Let $Tr = \{d_1, \ldots, d_{|Tr|}\}$ be a set of training documents and $F = \{t_1, \ldots, t_{|F|}\}$ its vocabulary. We use $W_{|Tr|\times|F|}$

[1]MLMC classification is typically solved by training $|\mathcal{C}|$ independent binary classifiers, one for each class of interest.

SIGIR '16, July 17-21, 2016, Pisa, Italy

DOI: http://dx.doi.org/10.1145/2911451.2914722

to denote the document-term matrix, where $w_{ij} \in \mathbb{R}$ is the weight of term t_j in document d_i as computed by a weighting function. By $\vec{d}_i \in \mathbb{R}^{|F|}$ we denote the vectorial representation of document d_i.

We present a general framework for oversampling, that we dub *Latent Space Oversampling* (LSO); our Distributional Random Oversampling method will be a specific instantiation of it. In LSO we oversample minority-class documents by extending the original feature space F with an additional latent space L. Each new synthetic example o_k for a document d_i will be expressed as $\vec{o}_k = [\vec{d}_i; \vec{v}_k] \in \mathbb{R}^{|F|+|L|}$, where $\vec{d}_i \in \mathbb{R}^{|F|}$ is the (fixed) observed part in the original feature space (i.e., a copy of the i-th row of W), and $\vec{v}_k \in \mathbb{R}^{|L|}$ is the variable part in the latent space L, which is generated by some stochastic function.

The vector expansion involves a two-step process for each document d_i, i.e., (i) the estimation of model parameters Θ_i for d_i via a *parameter estimation criterion* $\Psi(W, d_i)$, such that $\Theta_i \leftarrow \Psi(W, d_i)$ is calculated only once for each example d_i; and (ii) the generation of the variable part $\vec{v}_k \leftarrow \mathcal{G}(\Theta_i)$, obtained by means of a *generation function* $\mathcal{G}$. This function is called several times for each minority-class example until the desired level of balance is reached, and exactly once for each majority-class example, since we neither oversample nor undersample majority-class examples. The oversampled matrix is then re-weighted (e.g., in order to bring to bear updated *idf* values, and in order to perform correct length normalization) before training the classifier. Each test document d_t is also expanded to the enlarged vector space before being fed to the classifier; the only difference with the expansion process we carry out for training documents is that any global knowledge involved in the estimation of parameters Θ_t comes from the training data.

Different oversampling strategies could thus be defined by considering different parameter estimation criteria Ψ and different generation functions $\mathcal{G}$. In the following sections we first illustrate one possible such strategy, based on probabilistic topic models (Section 2.1); we then present our DRO method based on the distributional hypothesis (Section 2.2).

2.1 Latent Dirichlet Oversampling

One possible instantiation of the LSO framework is what we will here call *Latent Dirichlet Oversampling* (LDO). LDO relies on *Latent Dirichlet Allocation* (LDA – [1]), a probabilistic topic model that assumes, in order to define the model parameters and the generative function, that each (observed) document in a collection is generated by a mixture of (unobserved) topics. As the weight w_{ij} we here take the raw number of occurrences of term t_j in document d_i.

As the parameter estimation criterion Ψ_{LDO} we may choose any Bayesian inference method (such as Variational Bayes or Gibbs Sampling). The document-specific model parameters are $\Theta_i = [\theta_i; \varphi]$, where θ_i is the topic distribution of d_i and φ is the per-topic word distribution obtained from Tr.

We will choose a generation function $\mathcal{G}_{LDO}$ that returns a vectorial representation of a bag of n words, each of which is drawn by first choosing a topic $z_k \sim Multinomial(\theta_i)$, and then choosing a term $t_j \sim Multinomial(\varphi_{z_k})$. We set $n = length(d_i)$ (i.e., to the total number of word occurrences in d_i) so that the synthetic bag of words will allocate the same number of term occurrences as the original document (thus preserving sparsity in the new space). Note that, in this case, the latent space is mirroring the original feature space, with a dedicated latent dimension for each term in the vocabulary, i.e., $|L| = |F|$. LDO assumes each minority-class document to be governed by similar topic distributions, causing the variable part of oversampled documents to exhibit topically similar patterns.

2.2 Distributional Random Oversampling

We propose *Distributional Random Oversampling* (DRO), a different instantiation of LSO. DRO is based on the hypothesis that related documents (such as, e.g., the minority-class documents) may be expected to contain semantically similar words, and relies on a direct application of the distributional hypothesis, by virtue of which the meaning of feature t_j is embedded in column $\vec{t}_j^T \in \mathbb{R}^{|Tr|}$ of matrix W. Unlike in LDO, we here take weight w_{ij} to be generated by a real-valued weighting function such as, e.g., tfidf or BM25.

As the parameter estimation criterion we take a function Ψ_{DRO} that returns $\Theta_i = (p_1^i, \ldots, p_{|Tr|}^i)$, where p_k^i will be used as parameters of a multinomial distribution for document d_i. Parameter p_k^i is computed as

$$p_k^i = \frac{\sum_{t_j \in d_i} \vec{t}_j^T[k] \cdot w_{ij} \cdot s(t_j)}{\sum_{k=1}^{|Tr|} \sum_{t_j \in d_i} \vec{t}_j^T[k] \cdot w_{ij} \cdot s(t_j)} \quad (1)$$

i.e., by (i) summing together the k-th components $\vec{t}_j^T[k]$ of the (length-normalized) feature vectors $\vec{t}_j^T$ (i.e., the columns of the W matrix) corresponding to all unique terms $t_j \in d_i$, weighted by (a) their relative importance with respect to the document (the w_{ij} component) and by (b) their relative importance with respect to the classification task (the $s(t_j)$ component)[2], and (ii) normalizing to satisfy $\sum_{k=1}^{|Tr|} p_k^i = 1$.

We will choose a generation function $\mathcal{G}_{DRO}$ that returns a vectorial representation of a bag of n (latent) words, each of which is drawn from $l_k \sim Multinomial(\Theta_i)$. Note that in this case $|L| = |Tr|$. Similarly to the case of LDO we set $n = length(d_i)$, so that sparsity is preserved in the enlarged feature space. In contrast to LDO, the multinomial distribution of DRO is deterministically obtained from the training collection, thus avoiding the need for computationally expensive statistical inference methods.

Each test document is also expanded to the enlarged vector space before being fed to the classifier. In this case note that, in Equation 1, $\vec{t}_j^T$ – which encodes the distributional knowledge – and $s(t_j)$ are the supervised components, i.e., they are obtained from Tr. Instead, w_{ij} is computed partly from the document itself (e.g., the tf component) and partly from the training set (e.g., the *idf* component).

In sum, the rationale of our method is to generate synthetic minority-class vectors where the part corresponding to the latent space is the result of a generative process that brings to bear the distributional properties of the words contained in the document being oversampled.

3. EXPERIMENTS

As the datasets for our experiments we use REUTERS-21578, OHSUMED-S, and RCV1-v2. All these collections are multi-label, i.e., each document may be labelled by zero, one, or several classes at the same time, which gives rise to $|\mathcal{C}|$ binary classification problems, with $\mathcal{C}$ the set of classes in the

[2]In this paper we compute s as the mutual information between the feature and $\mathcal{C} = \{c, \bar{c}\}$.

dataset. For REUTERS-21578[3] we use the standard ("ModApté") split, which identifies 9,603 training documents and 3,299 test documents. We restrict our attention to the 115 classes with at least one positive training example. OHSUMED-S [4] consists instead of 12,358 training and 3,652 test MEDLINE textual records from 1987 to 1991, classified according to 97 MeSH index terms. RCV1-v2[4] comprises 804,414 news stories generated by Reuters from Aug 20, 1996, to Aug 19, 1997. In our experiments we use the entire training set, containing all 23,149 news stories written in Aug 1996; for the test set we pick the 60,074 news stories written during Sep 1996. We restrict our attention to the 101 classes with at least one positive training example.

As the evaluation measures we use microaveraged F_1 (F_1^μ) and macroaveraged F_1 (F_1^M).

We compare the performance of LDO[5] and DRO with the following baselines: (i) *Random Oversampling* (RO), a method that performs oversampling by simply duplicating random minority-class examples; (ii) *Synthetic Minority Oversampling Technique* (SMOTE – [2]), a method that generates new synthetic minority-class examples as convex linear combinations of the document d_i being sampled and a document randomly picked among the k minority-class nearest neighbours of d_i (typically using $k = 5$); (iii) *Borderline-SMOTE* (BSMOTE – [5]), a more recent version of SMOTE that only oversamples those borderline minority-class examples that would be misclassified as negatives by a k-NN classifier; (iv) DECOM [3], a probabilistic topic model that assumes all documents belonging to the same class to follow the same topic distribution that, once determined, is used to oversample minority-class examples following the LDA generation procedure[6]; (v) a bag-of-words model (BoW) where no oversampling is performed. For LDA-based methods we follow the related literature and set the number of topics to 30; in order to favour convergence we set the number of iterations to 3,000 and perform 10 passes.

As the learner of our experiments we adopt linear-kernel SVMs (in the popular SVM-light implementation[7]); in all our experiments we use the default SVM-light parameters. All methods are fed with the same preprocessed version of the datasets where, for each distinct binary decision problem, the top 10% most informative words have been selected, using mutual information as the selection function and tfidf as the weighting function. We perform oversampling of the minority class until a desired prevalence α for the minority-class is reached; we let α range on $\{0.05, 0.10, 0.15, 0.20\}$. We do not consider undersampling in this paper, i.e., all negative examples are picked exactly once. The results we present are all averages across 5 random trials we have run for each setting. For each dataset we partition the classes into (i) *HighPrevalence* (HP), the classes with a prevalence higher than 0.050; (ii) *LowPrevalence* (LP), the classes with a prevalence in the range $[0.015, 0.050]$; and (iii) *VeryLowPrevalence* (VLP), the classes with a prevalence smaller than 0.015. The reason for partitioning the classes according to prevalence is to allow the results to provide insights as to which classes benefit from oversampling and which do not.

[3]http://bit.ly/1F8AFcO

[4]http://1.usa.gov/1mp7RGr

[5]For LDO we used the Gensim implementation of LDA (see http://bit.ly/1Rl7pFV) which also allows estimating the document-topic distribution of test examples.

[6]For this method, as suggested in [3], we used the MATLAB implementation of Gibbs sampling available at http://bit.ly/1Rl7DNl

[7]http://svmlight.joachims.org/

Dataset	Training	Test	Features	Classes	HP	LP	VLP
Reuters-21578	9,603	3,299	23,563	115	3	50	62
Ohsumed-S	12,358	3,652	26,382	97	9	60	28
RCV1-v2	23,149	60,074	37,211	101	16	73	12

Table 1: Details on the 3 datasets used.

	Prev.	α	BoW	RO	SMOTE	BSMOTE	DECOM	LDO	DRO
F_1^M	HP	.05	.907	.907	.907	.907	.907	.907	.907
		.10	.907	.907	.911	.904	**.912**	.909	.897
		.15	.907	.910	**.911**	.902	**.911**	.908	.905
		.20	.907	.909	**.911**	.899	**.911**	**.911**	.899
	LP	.05	.633	.700	.754	.678	.650	.706	**.761**
		.10	.633	.682	.718	.678	.639	.690	**.766**†
		.15	.633	.662	.684	.678	.629	.679	**.759**†
		.20	.633	.648	.654	.678	.629	.664	**.764**†
	VLP	.05	.426	.485	.478	.426	.441	.484	**.568**†
		.10	.426	.456	.416	.426	.418	.482	**.568**†
		.15	.426	.473	.395	.426	.398	.476	**.567**†
		.20	.426	.473	.387	.426	.398	.474	**.570**†
F_1^μ	HP	.05	.954	.954	.954	.954	.954	.954	.954
		.10	**.954**	.952	.953	.952	**.954**	.953	.950
		.15	**.954**	.953	.953	.951	.953	.952	.951
		.20	.954	.953	.953	.950	**.955**	.952	.947
	LP	.05	.767	.788	.809	.782	.773	.790	**.810**
		.10	.767	.778	.784	.783	.762	.786	**.812**†
		.15	.767	.770	.756	.783	.750	.777	**.807**†
		.20	.767	.764	.731	.782	.738	.774	**.805**†
	VLP	.05	.132	.319	.428	.212	.315	.310	**.509**†
		.10	.132	.272	.357	.212	.280	.308	**.515**†
		.15	.132	.269	.302	.212	.250	.289	**.519**†
		.20	.132	.269	.277	.212	.240	.287	**.507**†

Table 2: Results on Reuters-21578.

Table 1 shows some details of the document collections used in the experiments. Tables 2 to 4 report the results of our experiments in terms of F_1^M and F_1^μ, for REUTERS-21578, OHSUMED-S, and RCV1-v2, respectively. Results are reported at different levels α of oversampling; we use boldface to highlight the best performing method, while symbol "†" indicates that the method outperforms all others in a statistically significant sense[8]. Note that, for each block of 4 rows identifying a certain set of classes (HP, LP, VLP), the results for BoW are always the same; this is obvious since there is no oversampling in BoW, which thus does not depend on the value of α. Note also that, in all three tables, the first row of the HP results for $\alpha = 0.05$ always contains identical values, since the HP classes have a prevalence ≥ 0.05.

Overall, the results of these experiments indicate that DRO is superior to the other six methods presented (including LDO). In the low-prevalence groups (LP and VLP) DRO is superior in most cases, across the different datasets and the different degrees α of oversampling, and especially so in terms of F_1^M; when DRO is not superior, the differences in performance with the top-performing method are fairly small. This superiority is more pronounced for the VLP classes, where DRO obtained 23 out of 24 best results, almost always with very large margins. In the HP classes, instead, our results do not reveal any clear winner, since the best results are haphazardly distributed among all of the

[8]Two-tailed t-test on paired examples at 0.05 confidence level.

	Prev.	α	BoW	RO	SMOTE	BSMOTE	DECOM	LDO	DRO
F_1^M	HP	.05	.753	.753	.753	.753	.753	.753	.753
		.10	.753	**.758**	.756	.754	.755	.757	.752
		.15	.753	.764	**.767**	.763	.760	.765	.753
		.20	.753	.769	**.771**	.767	.763	.769	.756
	LP	.05	.479	.557	**.603**	.571	.538	.569	.588
		.10	.479	.552	.578	.570	.532	.565	**.588**†
		.15	.479	.526	.550	.569	.525	.555	**.578**
		.20	.479	.514	.524	.568	.523	.542	**.576**
	VLP	.05	.354	.385	.455	**.458**	.354	.396	.451
		.10	.354	.372	.433	.448	.352	.378	**.469**
		.15	.354	.363	.440	.448	.330	.373	**.455**
		.20	.354	.364	.427	.448	.314	.376	**.476**
F_1^μ	HP	.05	.801	.801	.801	.801	.801	.801	.801
		.10	.801	**.803**	**.803**	.802	.802	**.803**	.798
		.15	.801	.804	**.806**	.805	.804	.804	.795
		.20	.801	.805	**.807**	.806	.805	.805	.795
	LP	.05	.616	.662	.672	.666	.647	**.668**	.647
		.10	.616	.657	.654	**.669**	.644	.665	.640
		.15	.616	.645	.625	**.666**†	.640	.658	.626
		.20	.616	.642	.595	**.666**†	.633	.652	.620
	VLP	.05	.282	.299	.518	.437	.365	.313	**.552**†
		.10	.282	.241	.484	.416	.328	.262	**.570**†
		.15	.282	.198	.446	.416	.311	.243	**.553**†
		.20	.282	.200	.415	.416	.291	.251	**.553**†

Table 3: Results on OHSUMED-S.

	Prev.	α	BoW	RO	SMOTE	BSMOTE	DECOM	LDO	DRO
F_1^M	HP	.05	.843	.843	.843	.843	.843	.843	.843
		.10	.843	**.848**	**.848**	.846	.845	.847	.839
		.15	.843	**.848**	**.848**	**.848**	.845	.847	.838
		.20	.843	.846	.845	**.848**†	.844	.846	.838
	LP	.05	.489	.600	.616	.573	.577	.613	**.617**
		.10	.489	.602	.603	.582	.587	.619	**.631**†
		.15	.489	.597	.584	.583	.591	.617	**.632**†
		.20	.489	.594	.563	.584	.593	.614	**.629**†
	VLP	.05	.048	.249	.257	.148	.263	.269	**.295**
		.10	.048	.237	.210	.148	.271	.261	**.294**†
		.15	.048	.228	.186	.148	.265	.252	**.297**†
		.20	.048	.220	.172	.148	.267	.245	**.295**†
F_1^μ	HP	.05	.877	.877	.877	.877	.877	.877	.877
		.10	.877	**.878**	**.878**	**.878**	.877	.877	.873
		.15	.877	.877	.877	**.878**	.877	.876	.871
		.20	.877	.876	.875	**.878**	.876	.875	.869
	LP	.05	.638	.676	.674	.666	.663	**.677**	.664
		.10	.638	.685	.672	.678	.673	**.691**†	.683
		.15	.638	.680	.656	.680	.675	**.688**†	.680
		.20	.638	.676	.637	.680	.675	**.683**†	.674
	VLP	.05	.106	.408	.408	.268	.426	.446	**.489**†
		.10	.106	.391	.343	.268	.431	.437	**.489**†
		.15	.106	.379	.303	.268	.429	.424	**.497**†
		.20	.106	.367	.283	.268	.430	.415	**.493**†

Table 4: Results on RCV1-v2.

baselines. Moreover, the best system is not substantially better to BoW in the vast majority of cases, which makes the idea of oversampling such classes questionable.

In sum, the results seem to indicate that the smaller the prevalence of the minority class is, the higher is the gain that can be obtained due to the use of DRO. This is an appealing feature for an oversampling method. We attribute this behaviour to DRO's distributional nature, which enables the information of the entire collection to contribute in the generation of each synthetic example (whereas RO and SMOTE-based methods are limited to local information provided by one or two examples, respectively). This could be advantageous for ill-defined classes (as those belonging to LP and VLP). It may instead introduce noise, or even some redundancy, for well-defined ones (i.e., those in HP); this suggests that the best policy may be that of applying DRO to low- or very-low prevalence classes only, while leaving high-prevalence classes untouched.

4. CONCLUSIONS

We have presented a new oversampling method for imbalanced text classification, based on the idea of assigning a probabilistic generative function to each minority-class document in the training set, a function that can be iteratively queried until the desired level of balance is reached. This probabilistic function is built upon distributional representations of the words contained in the document being oversampled, which allows the model to introduce some random variability in the new examples while preserving the underlying semantic properties motivated by the distributional hypothesis.

5. REFERENCES

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, 2002.

[3] Enhong Chen, Yanggang Lin, Hui Xiong, Qiming Luo, and Haiping Ma. Exploiting probabilistic topic models to improve text categorization under class imbalance. *Information Processing & Management*, 47(2):202–214, 2011.

[4] Andrea Esuli and Fabrizio Sebastiani. Improving text classification accuracy by training label cleaning. *ACM Transactions on Information Systems*, 31(4):Article 19, 2013.

[5] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Proceedings of the 1st International Conference on Intelligent Computing (ICIC 2005)*, pages 878–887, Hefei, CN, 2005.

[6] Zellig S. Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

[7] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[8] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449, 2002.

[9] Yanmin Sun, Andrew K. Wong, and Mohamed S. Kamel. Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(4):687–719, 2009.