Analyzing the Influence of Cardinality and Density Characteristics on Multi-Label Learning

Flavia Cristina Bernardini[1], **Rodrigo Barbosa da Silva**[1] **and Edwin Mitacc Meza**[1]

[1] Laboratório de Inovação no Desenvolvimento de Sistemas (LabIDeS)
Instituto de Ciência e Tecnologia – Universidade Federal Fluminense (UFF)
Rio das Ostras – RJ – Brazil

fcbernardini@puro.uff.br; rodrigo.silva.rio@gmail.com; emitacc@id.uff.br

Abstract. *Two main characteristics of multi-label dataset are cardinality and density, related to the number of labels of (each instance of) a multi-label dataset. The relation between these characteristics and multi-label learning performance has been observed with different datasets. However, the difference in domain dataset attributes also interfere on multi-label learning performance. In this work, we use a real dataset, named The Million Song Dataset, available in the internet, which presents the property of having too many labels associated to their instances (songs), as well as so many instances. In this work we present the processed datasets used to conduct our experiments, and we also describe our experiments results.*

Keywords: Multi-label Learning; Multi-label Dataset Analysis.

1. Introduction

Some real applications are related to the task of classification, such as diagnosis, fault detection, and so on. These problems are commonly treated by machine learning supervised algorithms, which induces classifiers, or predictors, such as neural networks, SVM, decision trees, and so on. These classifiers usually identify just one class of a new instance, or case, from a set of possible labels. However, there are problems related to the task of predicting more than one class for each case. For example, we can mention images and music labeling, failure diagnosis, and others. These kind of problems are tackled by a special type of machine learning, called multi-label learning algorithms. Many multi-label learning methods have been proposed in literature, such as [Schapire and Singer 2000, Shen et al. 2004, Sebastiani 2002, Bernardini et al. 2009, da Gama et al. 2012]. A survey describing multi-label learning methods can be found in [Tsoumakas et al. 2010a]. Two main characteristics analyzed in a multi-label dataset are cardinality and density, both related to the number of labels of each instance of a dataset and also of the entire dataset. Cardinality of a multi-label dataset is the mean of the number of labels of the instances that belong to this dataset, and density of a multi-label dataset is the mean of the number of labels of the instances that belong to this dataset divided by the number of the labels.

Some papers in literature indicate that these dataset characteristics — cardinality and density — may cause different behaviors in multi-label learning methods. In [Tsoumakas et al. 2010a], the authors affirm that two datasets with approximately the same cardinality, but with great difference in density, may not exhibit the same properties, which causes different behaviors in multi-label learning methods.

In [da Gama et al. 2012] we studied the influence of these two characteristics on the performance of the multi-label learners used in our benchmark. We observed that there was a correlation between these characteristics and the results obtained with some datasets; however, the domain of that datasets are quite different, what leaded us to question how the domain features influenced the analysis. In [Spyromitros et al. 2008], the authors proposed a new method called BRkNN, an adaptation of the kNN algorithm for multi-label classification based on Binary Relevance method, and compared this method with LPkNN, another adaptation method of the kNN algorithm based on Label Powerset multi-label method. The authors observed the influence between the LPkNN method and the influence of low density values, using three different datasets, with different domain features, but they could not safely argue that high density lead to improve performance of the LPkNN. These works analyze the relationship between cardinality (and density) and multi-label learning algorithms results using different datasets, with different cardinality and density values, and different domain dataset attributes. In this way, it is unknown how much the domain difference interferes in cardinality and density analysis. One issue that turns difficult this study is the unavailability of a dataset with the same features but different cardinality and density values.

In [Bertin-Mahieux et al. 2011] the Million Song Dataset is presented, a freely-available collection of audio features and metadata for a million contemporary popular music tracks. The dataset does not include any kind of audio music, only the derived features from them. This collection is available as a relational database. This dataset is labeled by tags that can be seen as musical genres. Each song has more than one of these tags associated to. The main advantage of this dataset on other available multi-label datasets is the high number of labels, which allows to vary the number of labels without loosing the multi-label problems characteristic. One problem with this dataset is the transformation process to allow data mining on it using the available data mining and machine learning tools.

The aim of this work is to present an analysis of the influence of the cardinality and density measures to multi-label learning. To allow this study, we pre-processed the Million Song Dataset. In this work, to initiate our studies, we considered only the Naïve Bayes algorithm to induce the base classifiers, because of its low time consumption for induction of the classifiers and its lack of requirement for adjustment parameters. In this work, we also present this dataset and the data pre-processing step of the Million Song Dataset. To induce the multi-label classifiers, we used the Mulan library[1][Tsoumakas et al. 2010b], based on Weka [Witten and Frank 2005].

This work is organized as follows: Section 2 describes Multi-Label Machine Learning concepts and notations. Section 3 describes the Million Song Dataset, as well as our pre-process step of this dataset. Section 4 describes the experiments and results we obtained. Section 5 concludes this work.

2. Multi-label Machine Learning

Multi-label problems appear in different domains, such as image, text, music, proteins and genome classification [Schapire and Singer 2000, Shen et al. 2004, Sebastiani 2002], and failure diagnosis [Bernardini et al. 2009]. In multi-label problems,

[1] Available at http://mulan.sourceforge.net.

the input to the multi-label learning algorithms is a dataset S, with N instances $T_i, i = 1, ..., N$, chosen from a domain X with fixed, arbitrary and unknown distribution $\mathcal{D}$, of the form $(\mathbf{x}_i, Y_i)$, with $i = 1, ..., N$, for some unknown function $f(\mathbf{x}) = Y$. In this work, we call domain attributes datasets the attributes that compose X. L is the set of possible labels of the domain $\mathcal{D}$, and $Y_i \subseteq L$, *i.e.*, Y_i is the set of labels of the ith instance. The output of multi-label learning algorithms is a classifier $\mathbf{h}$ that labels an instance $\mathbf{x}_i$ with a set $Z_i = \mathbf{h}(\mathbf{x}_i)$, *i.e.*, Z_i is the set of labels predicted by $\mathbf{h}$ for $\mathbf{x}_i$[2].

The number of labels $|L|$ is frequently seen as a parameter that influences the performance of different multi-label methods. There are two measures for evaluating the characteristics of a dataset, objects of this study: cardinality $Card$ and density $Dens$ [Tsoumakas et al. 2010a]. The cardinality of S is the mean of the number of labels of the instances that belong to S, defined by Eq. 1, and the density of S is the mean of the number of labels of the instances that belong to S divided by $|L|$, defined by Eq. 2.

$$Card = \frac{1}{N}\sum_{i=1}^{N}|Y_i| \tag{1}$$

$$Dens = \frac{1}{N}\sum_{i=1}^{N}\frac{|Y_i|}{|L|} \tag{2}$$

Multi-label machine learning methods can be divided into two categories [Tsoumakas et al. 2010a]: problem transformation and algorithm adaptation. In the first category, the multi-label problem is transformed to (many) multiclass (or binary) machine learning problems, and each sub-problem is given to a classic (binary or multiclass) supervised machine algorithm. These (binary or multiclass) classifiers are called in this work base classifiers. In the second category, the machine learning algorithm is adapted to deal with multi-label problems. In this work, we use three methods, commonly used in multi-label learning, named BR, LP and RAkEL. The BR method transforms the original multi-label problem, with $|L|$ labels, into $|L|$ binary problems. The LP method transforms the original multi-label problem into a multiclass problem, considering the relation among the labels. The RAkEL method constructs an ensemble of LP classifiers, which are trained using a small random subset of the set of labels constructed by LP. A more complete description of these (and others) multi-label methods can be found in [Tsoumakas et al. 2010a].

Multi-label machine learners need classifiers evaluation. For this task, there are three groups of measures to evaluate induced multi-label classifiers: based on instances, based on labels and based on ranking [Tsoumakas et al. 2010a]. In this work, we use the first two groups of measure, because multi-label ranking is not the aim of this work. In the first group, we use in this work (i) *Hamming Loss* (Ham), (ii) Accuracy (Acc) and (iii) $F1$, defined by Eqs. 3 to 5[3], respectively. In the second group, we use the micro and macro versions of $F1$ measure. Measures based on labels are calculated based on false positives f_p, false negatives f_n, true positives t_p and true negatives t_n, *i.e.*, measures

[2] In this work, we use T_i to refer to an instance with associated label y_i or Y_i, and we use $\mathbf{x}_i$ when we are not considering the associate label, or $\mathbf{x}_i$ does not have an associated label yet.

[3] In Eq. 3, Δ represents the symmetric difference between two datasets.

of the type $B(t_p, t_n, f_p, f_n)$ can be used in this case. Given that t_{p_l}, t_{n_l}, f_{p_l} and f_{n_l} are true positives, true negatives, false positives and false negatives for each label $l \in L$, the micro and macro versions of B measures are given by Eqs. 6 and 7, respectively. $F1(t_p, t_n, f_p, f_n)$ is given by Eq. 8.

$$Hamm(\mathbf{h}, S) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i \Delta Z_i|}{|L|} \tag{3}$$

$$Acc(\mathbf{h}, S) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|} \tag{4}$$

$$F(\mathbf{h}, S) = \frac{1}{N} \sum_{i=1}^{N} \frac{2|Y_i \cap Z_i|}{|Z_i| + |Y_i|} \tag{5}$$

$$B_{micro}(\mathbf{h}, S) = \frac{1}{|L|} \sum_{i=1}^{|L|} B(t_{p_i}, t_{n_i}, f_{p_i}, f_{n_i}) \tag{6}$$

$$B_{macro}(\mathbf{h}, S) = \frac{1}{|L|} B(\sum_{i=1}^{|L|} t_{p_i}, \sum_{i=1}^{|L|} t_{n_i}, \sum_{i=1}^{|L|} f_{p_i}, \sum_{i=1}^{|L|} f_{n_i}) \tag{7}$$

$$F1(t_p, t_n, f_p, f_n) = \frac{2 \times f_p}{2 \times t_p + f_n + f_p} \tag{8}$$

3. The Million Song Dataset

The MSD — The Million Song Dataset[4] [Bertin-Mahieux et al. 2011] — is a freely-available collection of audio features and metadata for a million contemporary popular music tracks. The core of the dataset is composed of features and metadata extracted from one million songs, provided by The Echo Nest[5]. The dataset does not include any kind of audio music, only the derived features from them. Each data music is stored using HDF5 format, which is a data model, library, and file format for storing and managing data. These HDF5 files were constructed using an API provided by The Echo Nest. Each file consists of features extracted from a music, such as version, artist and two types of genres collection associated to each music: (i) Terms, which are tags provided by The Echo Nest, and they can come from a number of places, but mostly from blogs; and (ii) Mbtags, which are tags provided from MusicBrainz specifically applied by humans to a particular artist. Particularly, Mbtags are cleaner than terms for genre recognition.

A HDF5 file has 55 features, and the most important features to use for representing this domain are segment-pitches and segments-timbre. Pitch is the sound property that classifies it as low or high in pitch, or, in other word, bass or sharp sound, respectively. This feature is related to frequency of the signal sound: Higher frequencies, or high pitches, correspond to lower wave length, or sharp sound; Lower frequencies, or low pitches, correspond to higher wave length, or bass sound. Timbre is the sound

[4] `http://labrosa.ee.columbia.edu/millionsong/`

[5] `http://echonest.com/.`

property dependent from the complexity of the signal sound. Perceiving timbre is affected either by frequencies domain aspects, *i.e.* the way the signal can be decomposed in elementary periodical signals, or time domain aspects, *i.e.* the way the signal amplitude varies with time. Timbre is usually defined as the color of the sound, because by timbre we can identify a sound produced by different fonts, such as two musical instruments playing the same accord or two people singing the same melody [Stephanidis 2010]. Other important features are artist name (the singer of the music), title of the music, location (where the music was recorded), year when the music was recorded, time duration, segments-start, bars start, similar artists, terms and mbtags — MusicBrainz tags, provided by MusicBrainz[6]. The last five listed features, jointly to segments-timbres and segments-pitches, are multi-valued. segments-start is a list of V values, where V is variable among songs. Each value of segments-start corresponds to the start, in seconds, of intervals, or segments, of the music. segments-pitches and segments-timbres are arrays of two dimensions, where the first one has 12 positions, and each of these positions has V values.

Because MSD contains many multi-valued features, a database-oriented approach to propositionalization is necessary [Krogel et al. 2003]. In [Bertin-Mahieux et al. 2011], they propositionalized only segments-timbre for year prediction task. As described before, segments-timbre has 12 lists, *i.e*, $segT\_list_1, ..., segT\_list_{12}$. In this case, the authors aggregate each list calculating 12 mean values, one for each list, generating the features $mean_{segT\_list_1}$, ..., $mean_{segT\_list_{12}}$. Also, the authors calculate the covariance matrix for the twelve lists. The purpose of this covariance matrix was to verify the variance between each pair of $segT\_list$. The covariance matrix is a matrix whose elements in the (i, j) position is the covariance cov between two random variables x and y; in this case, x is the list $segT\_list_i$, y is the list $segT\_list_j$, $i, j = \{1, ..., 12\}$. The covariance between two random variables x and y, $cov(x, y)$, is defined by the linear correlation coefficient $\rho_{xy} = \frac{\sigma_{xy}}{\sigma_x \sigma_y}$. When $x \neq y$, $cov(x, y) = cov(y, x)$; and when $x = y$, $cov(x, y) = cov(x, x) = \sigma_x^2$. In this case, where there are 12 lists, instead of generating all the $12^2 = 144$ matrix values, only $\sigma^2_{segt\_list_i}, i = \{1, ..., 12\}$ and $\rho_{segt\_list_i segt\_list_j}, i, j = \{1, ..., 12\}, i > j$ are calculated, what means generating 12 variance features and 60 correlation or covariance features, totalizing 78 covariance features. So, in [Bertin-Mahieux et al. 2011], they generated 90 features from the Million Song Dataset.

In this work, we did not only consider these 90 features, but we also considered the segments-pitches multi-valued feature, because we believe that the pitch of the music may influence its genre definition. The same procedure used to generate the features extracted from segments-timbre was used to generate features from segments-pitches. In this way, three features subsets are constructed: (i) means of segments-timbre lists, represented by $\{mean_{segP\_list_1}, ..., mean_{segP\_list_{12}}\}$; (ii) variances of segments-timbre lists, represented by $\{\sigma^2_{segP\_list_1}, ..., \sigma^2_{segP\_list_{12}}\}$; and (iii) correlation coefficients of segments-timbre lists, represented by $\{\rho_{segP\_list_1 segP\_list_2}, \cdots, \rho_{segP\_list_1 segP\_list_{12}}, \rho_{segP\_list_2 segP\_list_3}, \cdots, \rho_{segP\_list_2 segP\_list_{12}}, \cdots, \rho_{segP\_list_{11} segP\_list_{12}}\}$. Considering the aggregations of segments-timbre and segments-pitches, the description features totalize 180 domain features. Each instance was classified by the tags given by MusicBrainz, as

[6] http://musicbrainz.org/

described earlier.

The original dataset contains 1 million songs. The authors also made available a sample of the original dataset containing 10.000 songs, which was used for this work. When analyzing this dataset sample, we observed that (i) there were instances without any label; and (ii) there were labels with too few instances associated to them, as well as there were labels with too many of them. Instances without any label were discarded, resulting 3.710 instances. Labels with too few instances associated to them could be considered noisy labels. Next section describes the experiments realized in this work.

4. Experiments and Results

To evaluate the influence of cardinality and density characteristics to multi-label learning, we considered three multi-label learning methods frequently used in literature, briefly described in Section 2 — BR, LP and RAkEL [Tsoumakas et al. 2010a]. To vary cardinality and density of MSD, we considered that each label should be linked to a minimum of N_0 instances on the dataset. We considered the following values as minimum instances to each label: $N_0 \in \{0, 5, 15, 25, 35, 45, 65, 75, 85, 95, 145, 195\}$, where $N_0 = 0$ means that all the labels were considered; $N_0 = 5$, only labels with 5 or more instances associated with it were considered; $N_0 = 15$, only labels with 15 or more instances associated with it were considered; and so on. Each generated dataset was renamed to MSD-000, MSD-005, MSD-015, MSD-025, MSD-035, MSD-045, MSD-055, MSD-065, MSD-075, MSD-085, MSD-095, MSD-145 and MSD-195[7]. Table 1 describes the main characteristics of each generated datasets, where Min #Inst indicates the minimum number of instances a label has to be associated to be considered; #Inst represents the number of instances resulted after disconsidering labels that do not satisfy the Min #Inst Per Label condition; #Labels represents the number of remaining labels; $Card$ and $Dens$ represent cardinality and density of the resulting dataset. We should remember that each dataset has 180 domain dataset attributes, all numerical ones. Figure 1 shows the relation between cardinality and density over the generated datasets. In this figure, we can observe that cardinality decreasing rate is lower than density increasing rate.

Tabela 1. Datasets Characteristics

	Min #Inst	#Inst	#Labels	$Card$	$Dens$
MSD-000	0	3710	726	3.8919	0.0054
MSD-005	5	3669	483	3.7817	0.0078
MSD-015	15	3587	272	3.4767	0.0128
MSD-025	25	3541	202	3.2937	0.0163
MSD-035	35	3506	161	3.1954	0.0198
MSD-045	45	3466	140	3.1056	0.0222
MSD-055	55	3408	122	2.9759	0.0244
MSD-065	65	3372	107	2.9517	0.0276
MSD-075	75	3345	98	2.8906	0.0295
MSD-085	85	3340	90	2.8189	0.0313
MSD-095	95	3256	84	2.8443	0.0339
MSD-145	145	3080	62	2.6182	0.0422
MSD-195	195	2904	47	2.4938	0.0531

[7] The generated datasets are available at `http://www.professores.uff.br/fcbernardini/papers/compl/MSD_MR/`

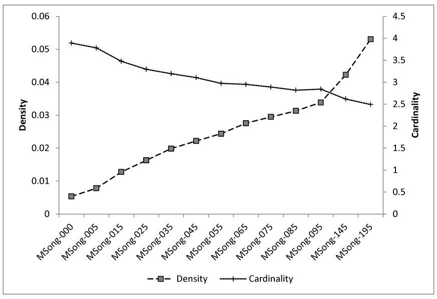


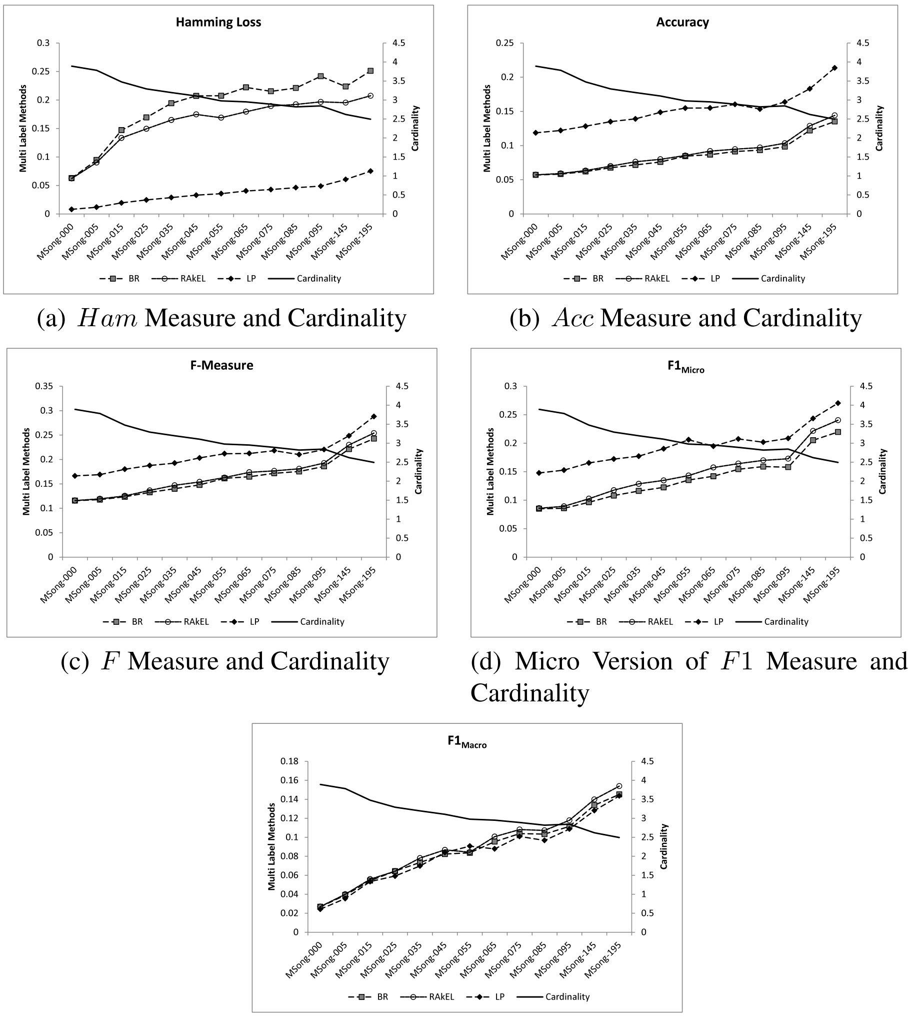
Figura 1. Cardinality and Density Values of Each Dataset

Each dataset was given to each multi-label method we considered in this work — BR, RAkEL and LP. In this work, we used only the Naïve Bayes algorithm [Mitchell 1997] to induce the base classifiers, because its low time consuming for induction of the classifiers and its lack of requirement for adjustment parameters. We used the implementation of the multi-label learning algorithms available at Mulan library [Tsoumakas et al. 2010a]. Mulan is based on Weka, a collection of machine learning algorithms for data mining tasks [Witten and Frank 2000]. Figures 2(a) to 2(e) shows respectively the results obtained on each measure used in this work — $Hamm$, Acc, F, $F1_{micro}$ and $F1_{macro}$ — for each dataset and each multi-label algorithm in contrast to cardinality values of the datasets. We should observe that all these measures present the following characteristic: the lower the cardinality value, the better the result, *i.e.*, higher the measures values, independently from the multi-label algorithm considered. In fact, even considering LP algorithm and $Hamm$ measure — Figure 2(a) —, where the values are lower than the BR and RAkEL values, we can observe this relation. Figs. 3(a) to 3(e) also shows respectively the results obtained on each measure used in this work, but in these plots these results are contrasted to density values of the datasets. We also should observe that all these measures present the following characteristic: the higher the cardinality value, the better the result, *i.e.*, higher the measures values, independently from the multi-label algorithm considered. These plots (as well as the ones shown in Figs. 2(a) to 2(e)), show that the multi-label methods improve the results while $Dens$ increases (or $Card$ decreases), and there is not a stationary point. This observation indicates that diminishing the complexity problem improves the multi-label learning results and therefore the threshold for selecting the labels to be learned should have an expert domain control.

To evaluate the correlation between the learning methods and cardinality and the learning methods and density, we measured the correlation between each algorithm results and the cardinality values, and also between the results and the density values. Because Pearson Correlation is a parametric statistic, we first executed the Anderson-Darling's normality test for all algorithms results. In some results we could reject the normality test, what leaded us to measure Spearman's rank correlation[8] [Ekstrøm 2011].

Spearman's rank correlation was calculated between $Card$ and each measure results, whose values are shown in Table 2, and also was calculated between $Dens$ and each measure results, whose values are shown in Table 3. In Table 2, we can observe that correlation values are very close to -1, or, in some cases, $\rho = -1$, which indicates that

[8] Anderson-Darling's normality test and Spearman's rank correlation was calculated using R software, available at `http://www.r-project.org/`



(a) Ham Measure and Cardinality

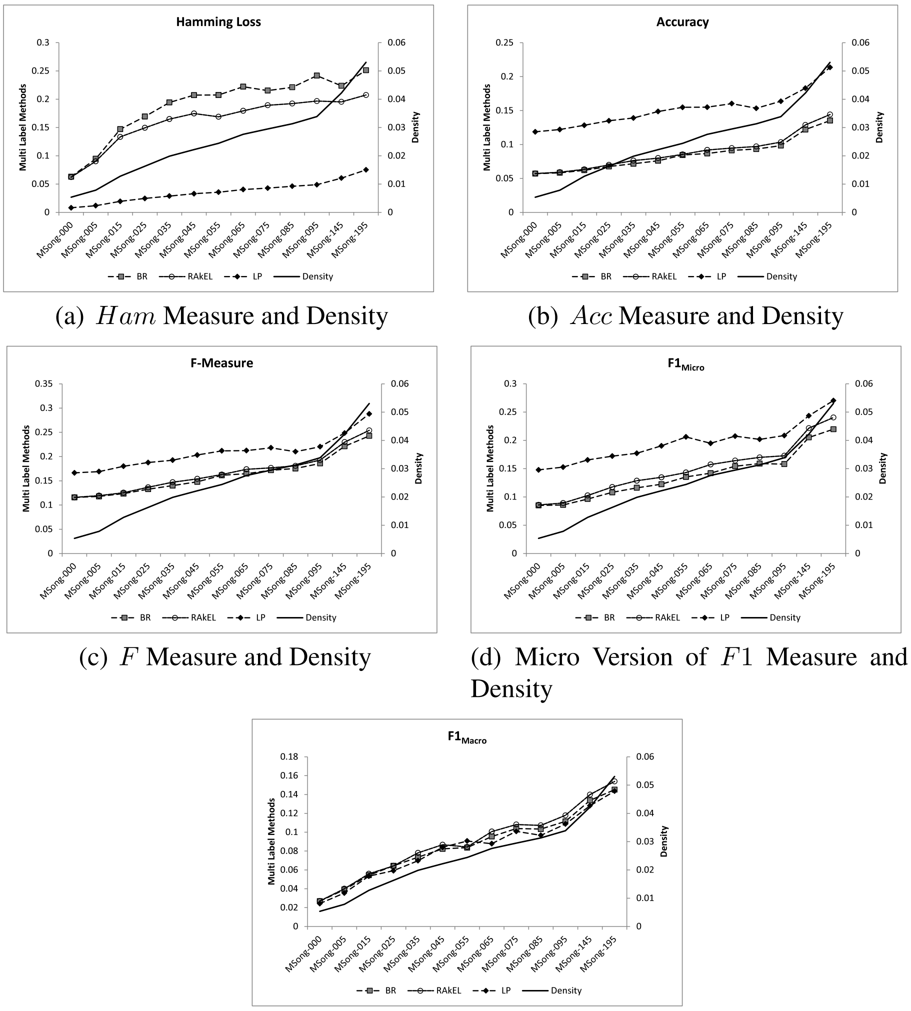
(b) Acc Measure and Cardinality

(c) F Measure and Cardinality

(d) Micro Version of $F1$ Measure and Cardinality

(e) Macro Version of $F1$ Measure and Cardinality

Figura 2. Comparison between Cardinality and Multi-Label Learning Measures.



(a) Ham Measure and Density

(b) Acc Measure and Density

(c) F Measure and Density

(d) Micro Version of $F1$ Measure and Density

(e) Macro Version of $F1$ Measure and Density

Figura 3. Comparison between Density and Multi-Label Learning Measures.

$Card$ inversely impacts each evaluation measure. Similarly, in Table 3, we can observe that correlation values are very close to 1, or, in some cases, $\rho = 1$, which indicates that $Dens$ proportionally impacts each evaluation measure. Correlation between the results and $Card$, as well as between the results and $Dens$, was expected; however, the absolute correlation values are nearest to 1, more times than we were expecting. Even with a high dimensional feature domain, the results improve when $Card$ is lower, or when $Dens$ is higher.

Tabela 2. Correlation ρ between $Card$ and multi-label evaluation measures to MSD-MR datasets

	BR	RAKEL	LP
$\rho(Hamm, Card)$	-0.962	-0.978	-0.995
$\rho(Acc, Card)$	-0.995	-0.995	-0.945
$\rho(F, Card)$	-0.995	-0.995	-0.945
$\rho(F1_{micro}, Card)$	-1.000	-0.995	-0.956
$\rho(F1_{macro}, Card)$	-0.984	-0.978	-0.978

Tabela 3. Correlation ρ between $Dens$ and multi-label evaluation measures to MSD-MR datasets

	BR	RAKEL	LP
$\rho(Hamm, Dens)$	0.978	0.989	1.000
$\rho(Acc, Dens)$	1.000	1.000	0.967
$\rho(F, Dens)$	1.000	1.000	0.967
$\rho(F1_{micro}, Dens)$	0.995	1.000	0.973
$\rho(F1_{macro}, Dens)$	0.995	0.989	0.989

5. Conclusions and Future Work

In [da Gama et al. 2012] we studied the influence of cardinality and density on the performance of the multi-label learners using six different datasets, with different domains. We observed in that work that there was a correlation between these characteristics and the results obtained with these datasets; however, the domain of that datasets are quite different, what leaded us to question how the domain features influenced the analysis. In this work, we present a study of the influence of two characteristics of multi-label datasets — cardinality and density. A second contribution of this work is to present multi-label datasets processed from a real dataset, named The Million Song Dataset. The main advantage of this dataset on other available multi-label datasets is the high number of labels, which allows to vary the number of labels without loosing the multi-label problems characteristic. To analyze how correlated are cardinality and density to multi-label learning algorithms performance, we used three multi-label machine learning methods and five evaluation measures. We can observe in our results that cardinality and density are highly correlated to these evaluation measures considering the chosen three methods and considering the dataset used in this study, which indicates that these characteristics have a great influence on multi-label learners.

In future work, we intend to induce base classifiers using different learning algorithms and other multi-label learning methods to expand our analysis. Also, this work indicates that multi-label learners that take these dataset characteristics into account should exhibit better performance to multi-label problems, which we intend to explore in the future.

Acknowledgment

We would like to thank to PIBIC/CNPq/UFF for the scientific initiation scholarship granted to the research; FAPERJ for the financial support (E-26/110.552/2012); Prof. Alexandre Plastino (UFF) for his valuable observations; Prof. Ana Paula Sobral (UFF) for helping us on the evaluation process of this paper; and the anonymous referees for their helpful comments that allowed to improve this work.

Referências

Bernardini, F., Garcia, A., and Ferraz, I. (2009). Artificial intelligence based methods to support motor pump multi-failure diagnostic. *Engineering Intelligent Systems*, 17(2).

Bertin-Mahieux, T., Ellis, D. P., Whitman, B., and Lamere, P. (2011). The million song dataset. In *Proc. 12th International Conference on Music Information Retrieval (ISMIR 2011)*.

da Gama, P. P., Bernardini, F. C., and Zadrozny, B. (2012). Proposal of a new method for multilabel classification based on random selection and bagging (in portuguese). In *Proc. IX Encontro Nacional de Inteligência Artificial – ENIA 2012*.

Ekstrøm, C. T. (2011). *The R Primer*. CRC Press.

Krogel, M.-A., Rawles, S., Železný, F., Flach, P. A., Lavrač, N., and Wrobel, S. (2003). Comparative evaluation of approaches to propositionalization. In *Proc. 13th Intern. Conf. on ILP, LNCS*, volume 2835, pages 197–214. Springer.

Mitchell, T. (1997). *Machine Learning*. McGraw Hill.

Schapire, R. E. and Singer, Y. (2000). Boostexter: a boosting-based system for text categorization. *Machine Learning*, 39(2–3):135–168.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.

Shen, X., Boutell, M., Luo, J., and Brown, C. (2004). Multi-label machine learning and its application to semantic scene classification. In *Proc. 2004 Int. Symposium on Electronic Imaging – EI 2004*, pages 18–22.

Spyromitros, E., Tsoumakas, G., and Vlahavas, I. (2008). An empirical study of lazy multilabel classification algorithms. In *Proc. 5th Hellenic Conf. on Artificial Intelligence: Theories, Models and Applications – SETN'08*, pages 401–406.

Stephanidis, C. (2010). *The Universal Access Handbook*. CRC Press.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010a). *Data Mining and Knowledge Discovery Handbook*, chapter Mining Multi-label Data. Springer, 2nd edition.

Tsoumakas, G., Vilcek, J., Spyromitros, E., and Vlahavas, I. (2010b). Mulan: A java library for multi-label learning. *Journal of Machine Learning Research*, 12:2411–2414.

Witten, I. and Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java*. Academic Press.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2nd edition.