QUINTA: A Question Tagging Assistant to Improve the Answering Ratio in Electronic Forums

Francisco Charte
Dept. Artificial Intelligence and Computer Science
University of Granada
Granada, Spain
Email: francisco@fcharte.com

Antonio J. Rivera
Dept. Computer Science
University of Jaén
Jaén, Spain
Email: arivera@ujaen.es

María J. del Jesus
Dept. Computer Science
University of Jaén
Jaén, Spain
E-mail: mjjesus@ujaen.es

Francisco Herrera
Dept. Artificial Intelligence and Computer Science
University of Granada
Granada, Spain
E-mail: herrera@ugr.es

***Abstract*—The web is broadly used nowadays to obtain information about almost any topic, from scientific procedures to cooking recipes. Electronic forums are very popular, with thousands of questions asked and answered every day. Correctly tagging the questions posted by users usually produces quicker and better answers by other users and experts. In this paper a prototype of a system aimed to assist the users while tagging their questions is proposed. To accomplish this task, firstly the text of each post is processed to produce a multilabel dataset, then a lazy nearest neighbor multilabel classification algorithm is used to predict the tags on new posts. The obtained results are promising, opening the door to the developing of a full automated system for this task.**

I. Introduction

The Internet is used for accessing electronic forums by thousands of users every day. Whether they are students, hobbyists or professionals, all of them try to find an answer to their doubts. These could be related to the way a statistical test is applied, how a specific feature of a programming language is implemented, or what are the ingredients of a cooking recipe, to mention just a few examples. Services such as Yahoo! Answers [1], Quora [2], and Stack Exchange [3], provide specific forums almost about everything. One of the most famous is Stack Overflow, a Stack Exchange forum which has substituted programming languages manuals to a great extent, being used by thousands of programmers every single day.

These forums usually reward people for answering questions by giving them some kind of marks, depending on the votes by others users on the quality of each answer. This way a user can get a certain reputation in their field of expertise, increasing the appreciation of their posts by other users interested in the same topic. These experts have to select the posts where they can help, but reviewing hundreds or thousands of them every day is unfeasible. This is the reason for tagging each post with a set of relevant labels, so that other users can effectively filter the posts. Therefore, the probability of getting a quick and good answer lowers as does the relevance of the tags assigned to the post. However, for a novel user (i.e. a user without a deep knowledge about the forum topic) assigning the proper set of tags to a post is not a trivial task. Hence, the usefulness of an automated system for doing so. This system would propose a set of labels once the user has written the post, allowing the user to delete any of them or add others. Such a system has to start from somewhere to elaborate its predictions, the obvious choice would be the existent posts on the same topic, assuming that all of them are correctly tagged.

978-1-4799-8569-2/15/$31.00 2015 IEEE

The approach we are proposing here, called QUINTA (*QUestIoN Tagging Assistant*), is based on a multilabel classification (MLC) algorithm. MLC algorithms [4] are able to predict a set of several labels (classes) for each data instance, instead of only one as traditional classifiers do. In the aforementioned context, these labels would be the tags assigned to each post. Since the text of raw posts cannot be used to build such a classifier, a prior text mining process [5] has to be applied. The goal is to obtain a multilabel classifier trained as fast as possible, and able to incrementally learn from new posts. The system would use this classifier to assist the tag assignment by the user, recommending an initial set of labels to them. Eventually, the new post and its set of tags would be added to the learning process.

In order to assess the usefulness of the proposed system prototype, it has been tested over six different forums with topics as diverse as computer science, chemistry, philosophy, chess, coffee, and cooking. The promising results gathered from the conducted experimentation encourage us to continue our work in this research path.

The rest of this paper is structured as follows. In section II the bases on which QUINTA relies are introduced. The structure of the proposed system is described in Section III. Section IV covers the experimental testing and discusses the obtained results. Lastly, in Section V the final conclusions are provided.

II. Preliminaries

This section provides the needed background to understand how QUINTA, the proposed system, does its work. First, the bases concerning text mining are introduced. Then, the multilabel classification task is briefly presented.

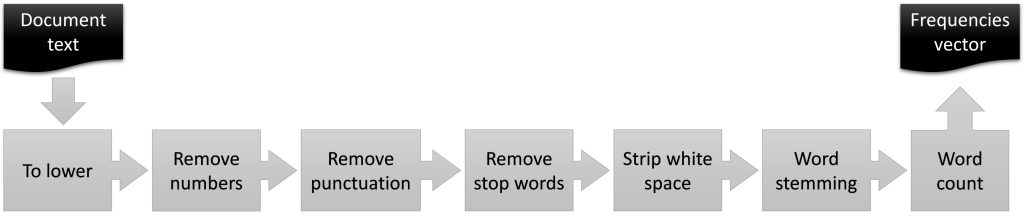

Fig. 1: Typical text mining pipeline

A. Text Mining

Text mining is a common task in fields such as information retrieval (IR) systems [6], aimed to return a set of relevant documents for a given query, and some classification systems, as the ones used for spam filtering [7] in electronic mail services. In this context a *document* can be a web page, an electronic mail, a post in a forum, etc. Depending on the system's goal, the original text for each document is represented in one way or another. The most usual approach is to transform it into a vector of word frequencies, storing each document as a row to generate a standard dataset.

The text in each document has to be normalized, producing a case-insensitive result and avoiding a different treatment for two words that only differ in some suffix or prefix (*word stemming*). Moreover, not all the words appearing in a document deliver useful information for the task at glance. Very common words (*stop words*), such as prepositions and articles, are not usually valuable as predictors. Spaces, numbers and other symbols are not useful either. To get rid of all these elements, each document is introduced in a text mining pipeline such as the one represented in Fig. 1, which applies several operations aimed to produce the final frequencies vector.

In an IR system the vector for each document is compared with the one generated by a query, usually through a common technique known as TF/IDF (*Term Frequency/Inverse Document Frequency*, [8]) and a similarity metric such as the cosine distance [9].

B. Multilabel Classification

Classification [10] is one of the most usual tasks in machine learning. The goal is to learn a model from a set of labeled patterns, thus incoming new ones can be classified as accurately as possible. Traditional classification methods assume that only one category, whether it is binary or multi-valued, will be assigned to each pattern. However, there are many real-world applications where each data instance is associated to a set of categories, classes or labels. Image [11], music [12] and video [13] categorization, protein function identification [14], and document labeling [15] are among them. This relatively recent problem is known as MLC [16].

In the field of text processing MLC has been applied to e-mail classification [15], bibliographic entries categorization [17], labeling of medical symptoms [18], and classification of malfunctions in flights [19], among others. In the best of our knowledge there is not any multilabel system for tagging of user questions.

A multilabel classifier takes as input a set of already labeled patterns, learning from them to be able to predict the set of labels (*labelset*) for new data samples. There are many MLC algorithms based on data transformation. This approach produces several binary datasets, one for each label [20], or one or more multiclass datasets, each one for a subset of labels [21]. The goal is to being able to process them using traditional classification methods. Moreover, some MLC proposals are founded on traditional classification methods, adapted to deal with several labels. There are MLC algorithms based on most classification techniques, such as trees [22], Support Vector Machines [23], neuronal networks [24], kNN [25], etc.

Regarding multilabel datasets (MLDs), most of them share some common characteristics, such as a high number of input features, some imbalance level and usually a large set of labels, albeit only a few of them appear in each data instance. The specific trails of MLDs have demanded the design of new metrics, both to characterize the data and to evaluate classification performance. The best known characterization metric is called *Card* [16], defined as in (1). Let D be an MLD and Y_i the set of labels assigned to the *i-th* instance, *Card* is the average number of labels active in each sample. By dividing this measure by the total number of labels used in D a dimensionless measure, known as label density (*Dens*), is obtained. Another important metric is *variety*, calculated as the number of different labelsets present in D. Additional metrics aimed to appraise imbalance levels [26] and measure label concurrence [27] have been also proposed.

$$Card = \frac{1}{\mid D \mid} \sum_{i=1}^{|D|} \mid Y_i \mid. \tag{1}$$

To assess the performance of an MLC classifier, more than a dozen evaluation metrics have been defined. Definition of most of them can be found in [4]. The metrics used in our experimentation will be explained later, in Section IV.

III. Structure of QUINTA

This section describes the general structure of the proposed tagging system, QUINTA, and details the essential steps for each one of the main phases it accomplishes, text mining, MLD generation, and tag set prediction.

The proposed system is structured as depicted in Fig. 2. The first three steps could be executed offline, getting a classifier ready for each one of the forums, or they could be completed each time a user enters a specific forum, as long as the training

time is short enough. This is the reason a lazy MLC algorithm is used, as it is able to start classifying new posts after loading the existent posts database without building a model. The other four steps will be repeated for each new post entering the forum. After applying the same text mining pipeline shown in step 2, the obtained data sample is given to the classifier. As a result, a set of proposed labels are provided to the user. This prediction, maybe modified by the user, would be added (step 8) to the classifier aiming to improve its performance as it gets more data.

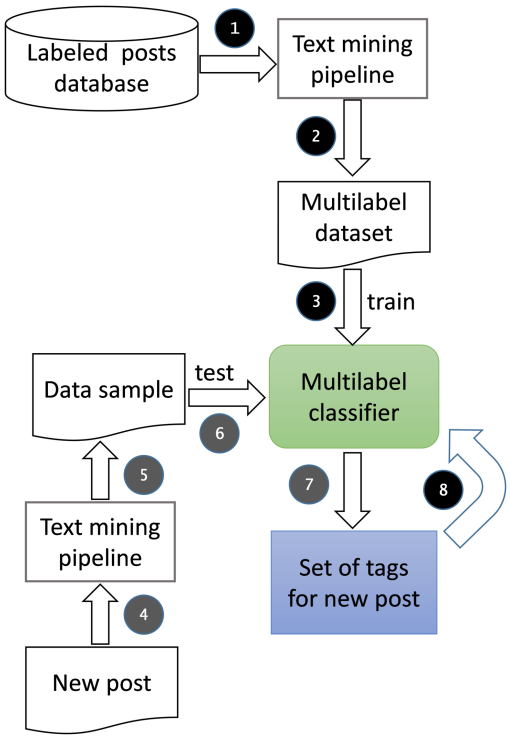

Fig. 2: Structure of QUINTA

Additional details on how each phase is accomplished are given in the following subsections.

A. Text mining of posts

Some forum platforms, such as Stack Exchange [3], make public all the data they work on, including posts' text and assigned tags previously anonymized, under a Creative Commons (CC BY-SA) license. In this case the size of the data is above 20 GB in compressed form. However, it is possible to select only part of these data, for instance filtering by forum.

The mining of these posts has been made with R, specifically using the methods provided by the tm package [28]. Since in some forums it is usual to include example code, the first step is to remove all markup which can make the word extraction process harder. Then, the usual data mining pipeline outlined above (see Fig. 1) is applied. At the end of the process an R data frame is obtained, containing a row for each post and a column for each word with frequencies at each cross. Aside, a second data frame holds for each post the set of tags manually assigned by the users.

B. MLD generation

Aiming to make available the previous data to any MLC classifier implemented in MULAN [29], the next step was to generate a standard MLD for each forum from the R data frames. To do so, the R mldr package [30] was used. Taking as input an R data frame and a list with the indexes of the columns acting as labels, this package is able to generate an object which can be saved in the standard ARFF format.

Besides its ability to generate MLDs from heterogeneous data, the mldr package is also able to provide a large set of metrics for any MLD. These functions have been used to explore the characteristics of the MLDs generated from the selected forums, described in Section IV.

C. New posts classification

Once the MLD is available, an MLC algorithm can be used to classify new incoming posts. The goal is to spend as little time as possible training the classifier, making it ready for the user quickly. Moreover, it would be interesting to be able to add these new posts to the system, gradually increasing its effectiveness. These are the main reasons to select a lazy, instance-based MLC algorithm, MLkNN [25].

The incoming posts are given the same text processing described before, obtaining a non-labeled data sample as a result. This instance is the input for MLkNN, which will predict a set of labels from the labels appearing on the k nearest neighbors of the new sample. These will be the tags proposed by the system to the user, allowing the addition of new tags and removing of existent ones before including the sample into the MLD.

There is only one adjustable parameter for the MLC algorithm, which is k, the number of nearest neighbors to take into account. Its recommended value, used by default, is 5.

IV. QUINTA Testing

In this section the datasets used to test the QUINTA system are enumerated and their main traits are shown. Then, the metrics used for evaluation of the performance of the system are introduced. Finally, the results produced by the system are analyzed.

A. MLDs and their characteristics

To test the performance of the proposed system, all posts belonging to six different forums in Stack Exchange have been used. The selected topics are quite diverse, so that the vocabulary and tags applicable to each one are dissimilar enough. This is a fact that can be confirmed by observing the word clouds in Fig. 3. Each one shows the words for a forum, with word sizes according to its frequency.

After the text mining phase, six MLDs have been generated with the traits shown in Table I. All of them have been made publicly available [31]. The columns indicate, from left to right, the Stack Exchange forum which the dataset belongs to, the number of instances (posts), the number of attributes (different words), the number of labels (tags), the number of labelsets (tag combinations), the cardinality (mean number of

(a) Chemistry (b) Chess (c) Coffee

(d) Cooking (e) Computer science (f) Philosophy

Fig. 3: Word clouds for each forum after text mining.

tags by post), and the label density. As can be seen, with the exception of the Coffee MLD all of them have *Card* > 2, which means that each instance has two or more labels. In general, the *Dens* measure is quite low in all cases, meaning that a significant level of sparsity exists in the label space, since *Card* is relatively low compared with the total number of labels.

TABLE I: Characteristics of the multilabel datasets

Forum	#Inst.	#Attrib.	#Labels	#Labelsets	Card	Dens
Chemistry	6961	715	175	3032	2.1093	0.0121
Chess	1675	812	227	1078	2.4113	0.0106
Coffee	225	1886	123	174	1.9867	0.0162
Cooking	10491	977	400	6386	2.2248	0.0056
Comp. sc.	9270	909	274	4749	2.5562	0.0093
Philosophy	3971	1075	233	2249	2.2720	0.0098

These six MLDs are the inputs given to the MLkNN MLC algorithm. 10 folds cross validation has been used to test the system. The parameter k, which sets the number of nearest neighbors, has been assigned values between 5 and 10.

B. Performance evaluation metrics

Three different evaluation metrics [4] have been used to assess the performance of the system: *AUC*, Hamming Loss (HL), and Subset Accuracy (*SA*). *AUC* is the usual *Area Under the Curve ROC*. As stated in [32], it is a good metric to evaluate the global performance of a classifier, with interesting properties when compared with the usual global accuracy, such as more sensitivity, being invariant to a priori class probabilities, etc.

HL is maybe the most used metric to evaluate MLC algorithms. Defined as shown in (2), and being Y_i the true labelset for an instance and Z_i the one predicted by the classifier, *HL* counts the number of differences between them. The resulting value will be in the $[0, 1]$ range. Lower values will indicate less errors, and therefore better classification performance.

$$HammingLoss = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \Delta Z_i|}{|L|}. \quad (2)$$

SA (3), also known as 0/1 subset accuracy and classification accuracy, is the most strict evaluation metric, since it only counts as correct cases those in which Y_i and Z_i are strictly equal. The result will be valued in the $[0, 1]$ range and the goal of any classifier is to maximize it, but most MLC algorithms perform quite poor when evaluated with *SA*.

$$SubsetAccuracy = \frac{1}{|D|} \sum_{i=1}^{|D|} [\![Y_i = Z_i]\!]. \quad (3)$$

The joint use of these three metrics will allow us to get a general view of the global system's performance, as will be discussed below.

C. Analysis of results

The results corresponding to the *HL* metric for each MLD and value of k are shown in Fig. 4. The first fact that can be noticed is that all the values are extremely low, below 0.02 in all cases. *HL* counts misclassified labels, whether they are false positives or false negatives, averaging by the total number of labels as was shown in (2). Therefore, it is not strange that Cooking appears as the MLD with best (lowest) *HL*, since it is also the MLD with more labels, while Coffee, which is the MLD with less labels, is also the worst performer. Notwithstanding, the general low values indicate that the results are quite good. It can also be noted that the number of nearest neighbors (k value) does not have an appreciable influence in the classifier behavior.

Fig. 5 shows results when assessed with the *AUC* metric. Specifically, the average value of the ROC area for all labels in each dataset and each k value has been represented. The values from four of the MLDs are well above the 80% level. Only the Cooking MLD, scraping this level with $k = 10$, and Coffee, which drops under 75%, are below. As can be seen, the performance of the classifier slightly improves as the value of k grows. *AUC* values reflect a trade-off between the sensitivity and specificity of the classifier. In general, values above 75% are considered as a good result.

Finally, in Fig. 6 the results obtained from the *SA* metric are represented. At first sight that the performance also improves as the value of k is incremented can be noticed. This metric only accounts as correct classification cases those in which the whole set of labels predicted by the classifier agrees with the true labelset. Thus, the values are considerably lower than with *AUC*. This metric is heavily affected by the number of distinct label combinations in the MLD. For this reason Cooking, with more than 6000 different labelsets, appears as the MLD with worst outcome. However, there are other factors influencing these results, since the second worst corresponds to the MLD with less labelsets, Coffee.

The MLD generated from the Computer Science forum is the one obtaining best overall classification results. From the characteristics shown in Table I it is not easy to infer a cause for this demeanor. This MLD has the highest *Card* and the second highest number of labelsets, instances and labels. Moreover, the vocabulary used in this forum could be more specific and related to the tags assigned to the posts in comparison with the others. Therefore, classification results change depending on the dataset characteristics, mainly number of labels, number of datasets and specificity of the vocabulary, and the own nature of the evaluation metric.

Overall, it seems that the initial performance of the QUINTA system is quite good, and therefore the tags recommended to the users could be largely accurate to their needs.

V. Conclusions

Electronic forums are used by thousands of people every day to solve their doubts, whether they are related to how to play a game, where to find a piece of information or which are the steps to be followed for cooking a meal. Each question

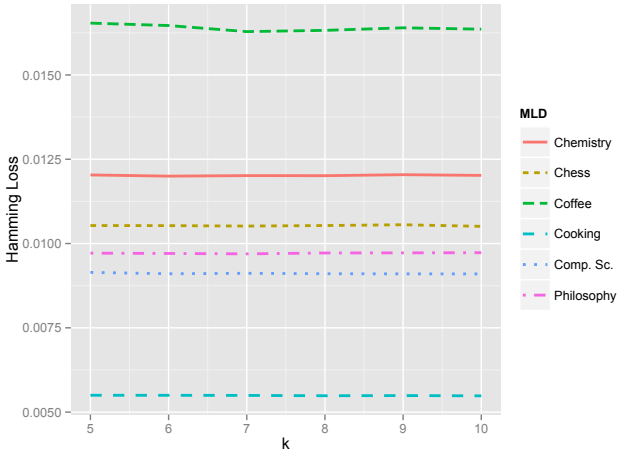

Fig. 4: Classifier performance assessed with Hamming Loss

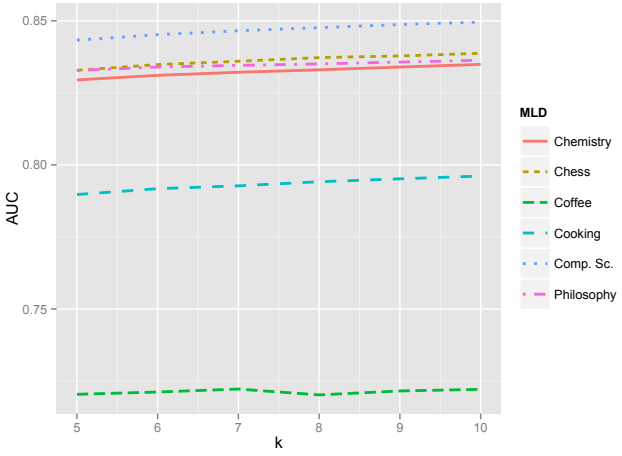

Fig. 5: Classifier performance assessed with AUC

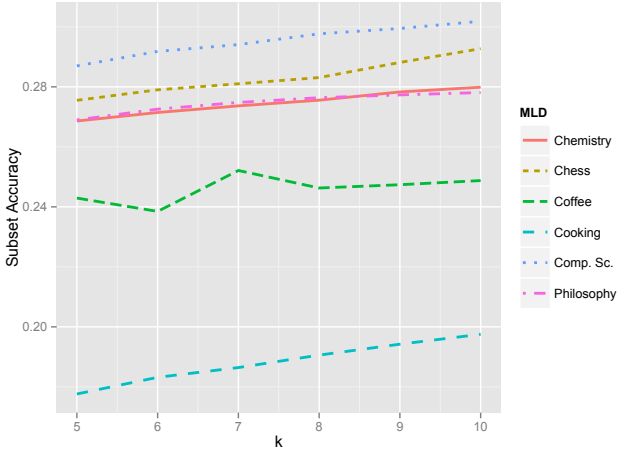

Fig. 6: Classifier performance assessed with Subset Accuracy

is posted being assigned a set of tags in order to ease filtering, so that experts can get only those posts where their knowledge can be useful. Hence the importance of correctly tagging each post to increase the probability of obtaining a quick and correct answer.

In this paper QUINTA, a prototype for an assistant able to tag this kind of posts, has been proposed. The foundations of QUINTA are a text mining pipeline and a multilabel classifier. The former processes the posts contents producing a multilabel dataset, whereas the latter is in charge of predicting the tags for new posts.

The effectiveness of the proposed system prototype has been validated using the posts in six electronic forums with very different topics. These have produced six MLDs with disparate traits, including different number of labels and labelsets. The diversity on the MLDs characteristics influences both the classifier behavior and the evaluation metrics, as has been explained. The results, assessed with three classification performance metrics, are rather positive. Taking as reference the AUC metric, which is a good indicator for the overall classification performance, the results can be considered good (above 75%) for five out of six MLDs. Only the output for the Coffee dataset, with a learning somehow limited by its 225 instances, could be considered as a bad result. This encourages us to continue our efforts in the development of this system, including additional features which could improve its usefulness. These could include imbalanced learning techniques, to alleviate the differences in word frequencies, as well as selection instances capabilities, aimed to choose the best neighbors.

ACKNOWLEDGMENT

F. Charte is supported by the Spanish Ministry of Education under the FPU National Program (Ref. AP2010-0068). This work was partially supported by the Spanish Ministry of Science and Technology under projects TIN2014-57251-P and TIN2012-33856, and the Andalusian regional projects P10-TIC-06858 and P11-TIC-7765.

REFERENCES

[1] "Yahoo! Answers." [Online]. Available: https://answers.yahoo.com/
[2] "Quora." [Online]. Available: https://www.quora.com/
[3] "Stack Exchange." [Online]. Available: http://stackexchange.com/
[4] E. Gibaja and S. Ventura, "Multi-label learning: a review of the state of the art and ongoing research," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 4, no. 6.
[5] C. C. Aggarwal and C. Zhai, *Mining text data*. Springer Science & Business Media, 2012.
[6] C. J. Van Rijsbergen, *Information Retrieval*, 2nd ed. Newton, MA, USA: Butterworth-Heinemann.
[7] A. N. Srivastava and M. Sahami, *Text mining: Classification, clustering, and applications*. CRC Press, 2009.
[8] G. Salton, E. A. Fox, and H. Wu, "Extended Boolean information retrieval," *Communications of the ACM*, vol. 26, no. 11, pp. 1022–1036, 1983.
[9] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Eng. Bull.*, vol. 24, no. 4, pp. 35–43, 2001.
[10] S. B. Kotsiantis, "Supervised machine learning: A review of classification techniques," 2007.
[11] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. M. Blei, and M. I. Jordan, "Matching words and pictures," *J. Mach. Learn. Res.*, vol. 3, pp. 1107–1135, 2003.
[12] A. Wieczorkowska, P. Synak, and Z. Raś, "Multi-Label Classification of Emotions in Music," in *Intelligent Information Processing and Web Mining*, ser. AISC, 2006, vol. 35, ch. 30, pp. 307–315.
[13] C. G. M. Snoek, M. Worring, J. C. van Gemert, J. M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proc. 14th Annu. ACM Int. Conf. on Multimedia, Santa Barbara, CA, USA, MULTIMEDIA'06*, 2006, pp. 421–430.
[14] S. Diplaris, G. Tsoumakas, P. Mitkas, and I. Vlahavas, "Protein Classification with Multiple Algorithms," in *Proc. 10th Panhellenic Conference on Informatics, Volos, Greece, PCI'05*, 2005, pp. 448–456.
[15] B. Klimt and Y. Yang, "The Enron Corpus: A New Dataset for Email Classification Research," in *Proc. ECML'04, Pisa, Italy*, 2004, pp. 217–226.
[16] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining Multi-label Data," in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds. Boston, MA: Springer US, 2010, ch. 34, pp. 667–685.
[17] I. Katakis, G. Tsoumakas, and I. Vlahavas, "Multilabel Text Classification for Automated Tag Suggestion," in *Proc. ECML PKDD'08 Discovery Challenge, Antwerp, Belgium*, 2008, pp. 75–83.
[18] K. Crammer, M. Dredze, K. Ganchev, P. P. Talukdar, and S. Carroll, "Automatic Code Assignment to Medical Text," in *Proc. Workshop on Biological, Translational, and Clinical Language Processing, Prague, Czech Republic, BioNLP'07*, 2007, pp. 129–136.
[19] A. N. Srivastava and B. Zane-Ulman, "Discovering recurring anomalies in text reports regarding complex space systems," in *Aerospace Conference*. IEEE, 2005, pp. 3853–3862.
[20] S. Godbole and S. Sarawagi, "Discriminative Methods for Multi-Labeled Classification," in *Advances in Knowl. Discovery and Data Mining*, vol. 3056, 2004, pp. 22–30.
[21] M. Boutell, J. Luo, X. Shen, and C. Brown, "Learning multi-label scene classification," *Pattern Recognit.*, vol. 37, no. 9, pp. 1757–1771, 2004.
[22] Q. Wu, Y. Ye, H. Zhang, T. Chow, and S.-S. Ho, "ML-TREE: A tree-structure-based approach to multilabel learning," *IEEE Trans. Neural Netw. Learn. Syst*, vol. to be published, no. Available online, p. doi: 10.1109/TNNLS.2014.2315296, 2014.
[23] A. Elisseeff and J. Weston, "A Kernel Method for Multi-Labelled Classification," in *Advances in Neural Information Processing Systems 14*, vol. 14. MIT Press, 2001, pp. 681–687.
[24] M. Zhang, "Ml-rbf : RBF Neural Networks for Multi-label Learning," *Neural Process. Lett.*, vol. 29, pp. 61–74, 2009.
[25] M. Zhang and Z. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, 2007.
[26] F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera, "Addressing Imbalance in Multilabel Classification: Measures and Random Resampling Algorithms," *Neurocomputing*, vol. 163, no. 0, pp. 3–16, 2015. doi: 10.1016/j.neucom.2014.08.091.
[27] ——, "Concurrence among Imbalanced Labels and its Influence on Multilabel Resampling Algorithms," in *Proc. 9th Int. Conf. Hybrid Artificial Intelligent Systems, Salamanca, Spain, HAIS'14*, ser. LNCS, vol. 8480, 2014.
[28] I. Feinerer, K. Hornik, and D. Meyer, "Text Mining Infrastructure in R," *Journal of Statistical Software*, vol. 25, no. 5, pp. 1–54, March 2008. [Online]. Available: http://www.jstatsoft.org/v25/i05/
[29] G. Tsoumakas, E. S. Xioufis, J. Vilcek, and I. Vlahavas, "MULAN: A Java Library for Multi-Label Learning," *J. Mach. Learn. Res.*, vol. 12, pp. 2411–2414, 2011.
[30] F. Charte and F. D. Charte, "How to work with multilabel datasets in R using the mldr package ," doi: 10.6084/m9.figshare.1356035 2015.
[31] F. Charte, "Multilabel datasets from Stack Exchange forums," 04 2015. [Online]. Available: http://dx.doi.org/10.6084/m9.figshare.1385315
[32] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.