



A model for multi-label classification and ranking of learning objects

Vivian F. López^{a,*}, Fernando de la Prieta^a, Mitsunori Ogiwara^b, Ding Ding Wong^c

^a *Departament Informàtica y Automàtica, University of Salamanca, Plaza de la Merced S/N, 37008 Salamanca, Spain*

^b *Department of Computer Science, University of Miami, 1365 Memorial Drive, Coral Gables, FL 33146, USA*

^c *School of Computing and Information Sciences, Florida International University, ECS 251, 11200 SW 8 ST, Miami, FL 33199, USA*

ARTICLE INFO

Keywords:

Multi-label classification
Learning objects
Tagging
Metadata

ABSTRACT

This paper describes an approach that uses multi-label classification methods for search tagged learning objects (LOs) by Learning Object Metadata (LOM), specifically the model offers a methodology that illustrates the task of multi-label mapping of LOs into types queries through an emergent multi-label space, and that can improve the first choice of learners or teachers. In order to build the model, the paper also proposes and preliminarily investigates the use of multi-label classification algorithm using only the LO features. As many LOs include textual material that can be indexed, and such indexes can also be used to filter the objects by matching them against user-provided keywords, we then did experiments using web classification with text features to compare the accuracy with the results from metadata (LO feature).

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

The concept of reusable learning objects has evolved into a central component within the current context of e-learning. Chiappe, Segovia, and Rincon (2007) recently described a learning object (LO) as a digital, self-contained and reusable entity with a clearly instructional content, containing at least three internal and editable components: content, learning activities, and elements of context. Additionally, LOs should have an external information structure, the metadata, which can facilitate its identification, storage and retrieval. Given this definition, it is possible to arrive at a certain consensus regarding LOs: they must be a minimal content unit (self-contained) that intends to teach something (instructional purpose) and can be reused (reusability) on different platforms without any compatibility problems.

A study by Bauer and Stefan (2008) pointed out that for administration and exchange of LOs, meaningful metadata are required. Typically, learning material is not limited to text, but includes multimedia content, such as images, audio and video. Metadata not only describe the content, but also refer to e.g. didactical methods, domain of usage and relationships to other LOs (Motelet, Baloiian, & Pino, 2006). "The feasibility of the LO paradigm strongly depends on having efficient mechanisms for retrieving relevant LOs for each application context. This can be achieved by tagging LOs with metadata, which will allow for cataloging and classifying them" (Sierra & Fernández-Valmayor, 2008).

Currently, web sites have introduced a number of innovative techniques, known as Web 2.0 that allows its users to interact with

others to exchange content, in contrast to non-interactive Web sites where users are limited to passive viewing information. These techniques have changed the way people create, share and organize information on the Web, encouraging the active involvement of end users. The advent of Web technologies allowing for large numbers of users to participate in content production, sometimes termed "collective intelligence" by O'Reilly (2008) emerging from the contribution of many has been discussed as a promising phenomenon that requires further investigation. The gaining, recovering and classification of LOs for each application context can be achieved by tagging LOs with metadata: the annotation of LOs could be moved from few authors to a potentially much larger number of users with what has come to be called "collaborative tagging" (Bauer & Stefan, 2008).

This paper describes an approach that uses multi-label classification methods for searching LOs tagged by Learning Object Metadata (LOM) (IEEE-LTSC, 2002), specifically the model offers a methodology that illustrates the task of multi-label mapping of LOs into types queries through an emergent multi-label space, and that can improve the first choice of learners or teachers. In order to build a model to classify and catalog the LOs in types queries, the paper also proposes and preliminarily investigates the use of multi-label classification algorithm using only the LO features. As many LOs include textual material that can be indexed, and such indexes can also be used to filter the objects by matching them against user-provided keywords, we then did experiments using web classification with text features to compare the accuracy with the results from metadata (LO feature).

This paper is structured as follows: Section 2 explains the main concepts and characteristics that establish LOs as the fundamental base within the current context of web-based e-learning. In

* Corresponding author. Tel.: +34 667702837.

E-mail address: vivian@usal.es (V.F. López).

Section 3, we present the tagging for LOs. Section 4 provides some background information on the problem of multi-label classification, the details of the dataset used in this paper and experimental results comparing the two multi-label classification algorithms. In Sections 5 and 6 we present the results of experiments on two datasets: using web classification with text features and metadata (LO feature), to compare the accuracy of the results. We conclude with Section 7, which explains some of the more relevant aspects and Section 8 the future work.

2. Current context of e-learning

Existing LO standards and specifications focus on facilitating the search, evaluation, acquisition, and reuse of LOs so that they can be shared and exchanged across different learning systems. The most notable standards used for LOs with metadata are: DublinCore (DCMI, 2007) and, most importantly, the IEEE-LOM (IEEE-LTSC, 2002). Since 2002, the Learning Object Metadata (LOM) has been the standard for specifying the syntaxes and semantics of LOM. It uses a hierarchical structure that is commonly coded in XML, and includes element names, definitions, data types, taxonomies, vocabularies, and field lengths. LOM are focused on the minimal set of attributes needed to allow these LOs to be managed. Though the predefined metadata schemas are very valuable for enabling interoperability, they have several shortcomings that hinder their practical application, e.g. creators can be lost in huge and general-purpose catalogs of metadata, particular domain aspects that are not adequately addressed, etc. (Sierra & Fernández-Valmayor, 2008).

LOs are placed inside learning object repositories (LOR), in an attempt to facilitate their reusability so that they can be more easily stored and retrieved. The LOR are highly heterogeneous, each one with a different storage system, query methods, etc. As noted in Brinker, Fürkranz, and Hüllermeier (2006) the search for LOs is a challenging task of e-learning. The continual research in search systems, the ability to create standardized and interoperability processes that can be applied to recovering LOs has made it possible to formalize search and retrieval processes for LOs in different repositories. However, it is difficult to obtain a semantic interpretation of the content of these LOs.

Semantics are gaining importance in the context of e-learning, in which the conceptual structure of LO content is an essential part of learning material. Losing conceptual information content implies the inability to integrate the concepts to learning, which is very important in order to understand any topic in a particular area. The LOR initiative in its current state does not provide such semantics, it only provides descriptions for the resources and says nothing about how to present resources to users in a conceptually clear way. It is necessary to provide semantic information for the user and to use ontologies that highlight the LO structure.

In addition, LOs must also provide educational information that specifies the type of cognitive activities in which students will be involved and the teaching–learning strategies associated with LOs, so that domain concepts can be transferred effectively to student.

Goarany, Kulczycki, and Blake (2010) considers the tag cloud as a lightweight form of ontology that can be used to index services and consequently retrieve services that match a user query, thus we thought these can be used in a pedagogical meaning in retrieving LOs. The tag-based clustering technique is proposed for establishing a similarity between services based on similarities between their tag clouds. To achieve the LO clustering, each object must be tagged with descriptive metadata or information about that resource in order to be easily located and later retrieved from repositories. The problem is that there are no easy or automated

ways of tagging these objects. LOs are tagged according to personal criteria that have been considered most appropriate by the editor or teacher. Furthermore, repositories have a lack of basic characteristics that are expected of any general search engine, such as classification tasks, sorting results, the use of different filtering techniques (such as the collaborative technique), the automated management of repositories and the extraction of statistics that serve to improve the global query process.

3. Using tagging for LOs

A tag is simply a word you use to describe a Web resource; tags are the most popular terms with which the user describes these resources. Therefore tags promise to be a unique source of information about the similarity between resources, a common form of navigation and organization of these resources. The tagging is made from the emergence of social software applications such as Delicious and Flickr. A study by Begelman et al. (2006) pointed out that tagging is a great collaboration tool. Tagging seems to be the natural way for people to classify objects as well as an attractive way to discover new material. A lot of current systems for information discovery, make use of this technology for automatic classification of Web resources. Though the amount of data that are used on the Web is abundant, these usually are not labeled. This is a major obstacle to learning methods. Additionally, most learning systems that are used on the Web are more interested in user preferences than in Web content. According to Hassan-Montero and Herrero-Solana (2006) “tagging is not only an individual process of categorization, but implicitly it is also a social process of indexing, a social process of knowledge construction”.

The tendency towards lightweight, easily accessible mechanism for ontology and metadata creation is best evidenced by the recent appearance of *folksonomies* (Mika, 2005). Users share their resources with their tags, generating an aggregated tag-index so-called folksonomy (also named social tagging mechanism). The Folksonomy term, coined by Thomas Vander Wal in Alfia mailing list, is a one-word neologism that comes from the words taxonomy and folk. Its basic element of information is a (user, resource, [keyword]) triple (Cattuto, Loreto, & Pietronero, 2007). The idea of folksonomy for the LOs, is to allow the users to describe a set of objects with a set of keywords of their own. In a folksonomy, the users invent tags, in the form of keywords, or reuse tags invented by others, to tag these materials (Mathes, 2004). A LO can be associated with tagged keywords selected by many users, which provide enriched semantic features for LO classification. All these tags reflect the semantics of the LO. According to Sierra and Fernández-Valmayor (2008) the cataloging mechanism emerges as a consequence of the collaborative behavior of a community of users. But the folksonomies, from the perspective of knowledge representation, have noise, the set of tags lack of consistency and accuracy. The folksonomies are ambiguous and does not constitute a shared vocabulary, due to the different uses made by users of the tags, which have no explicit semantics.

Bauer et al. indicates one alternative solution to reduce the noise in the set of tags by the systematic separation of generic tags. Upon the assumption that erroneous tags are specific tags and hence not part of the generally accepted set of tags, separation can improve the overall quality of tags. By applying taxonomies, ontologies or stemming algorithms, the problems of homonyms, synonyms or plural words can be overcome. Those mechanisms can be applied as part of an automatically triggered cleaning or separation process. But Bauer et al. also pointed that a major concern that has been voiced against this solution is that untrained people cannot achieve the metadata quality that professionally trained staff reach. However, Surowiecki argues for the superiority

of the wisdom of the crowds which is conceptualized as collective intelligence, i.e. the ideas of many people bring in a wider variety compared to one single person (Surowiecki, 2004). Specifically, for generating metadata a broad spectrum of descriptions is valuable, because the community of users of LOs represents the potential LO seekers as well as several application contexts in which the LOs are used. Particularly, associations between LOs, topics and application areas represent the extra value derived from group intelligence (Weiss, 2005). Typically, users can post free-text descriptions to Web resources and additionally they can select from a list of most frequently used descriptions. Thus, more generic tags used by many people can be distinguished from specific tags used by a few people (Golder & Huberman, 2005).

Generally, the first group is more interesting for organizations, because they represent the common agreement within the respective community and can be used to describe resources, here LOs, on an institutional rather than an individual level. In that sense, the number of top-ranked tags can be interpreted as a measure of the semantic breadth of a resource (Cattuto et al., 2007). But in order to be useful, it is necessary that the tags converge to a stable set of generic tags. The idea is to apply a certain threshold as a criterion for separation in order to filter the set of generic tags to be used for annotating LOs. A LO can be associated with tagged keywords selected by many users, which provide enriched semantic features for LO classification. All these tags reflect the semantics of the LO.

Therefore, the tagging constitutes in principle a way of addressing the shortcomings of predefined metadata schemas and a hybrid solution between the semantic and syntactical approaches used in the Web.

To solve the problems described, researchers have begun to seek the underlying structure folksonomies that are statistically analyzed using tag clouds, tag networks or tag clusters (Rollett, Lux, Strohmaier, Dosinger, & Tochtermann, 2007). Clusters tags may suggest how to identify implicit definitions of concepts within a folksonomy, may be a starting point for automating the extraction of unstructured formal vocabulary that could be used to improve recovery of tagged items and help users in the choice of labels to be used. Begelman et al. (2006) pointed out that “if we could automatically and dynamically cluster tags without putting more burden on the user, we could provide a much stronger service. Searching, subscribing and exploring would be much more effective”.

The explosive growth of type different LO, especially non-textual LO, has made the problem of LO classification increasingly challenging, as such objects often suffer from a lack of easy-extractable features with semantic information, because there are no interconnections between them, neither training examples with category labels. In addition, many LOs have a limited text description while other objects are unlabeled or have a lot of label associated. It is therefore desirable to find a way to deal with all these deficiencies of the LOs to improve the recovering and classification of LOs. In this paper we did research on social tags and multi-label classification with the aim of overcoming the above difficulties of LO.

4. Multi-label classification

In this research what is intended to be demonstrated is that multi-label classification can be applied to the organization of LOs to illustrate the idea of using collaborative tagging in finding a LO between learning materials of different heterogeneous LOR. According to Tsoumakas, Katakis, and Vlahavas (2010) the learning from multi-label data has attracted recently significant attention, motivated by an increasing number of new applications, to name

a few typical like: social network (Mika, 2005; Wu, Zhang, & Yu, 2006), text categorization (McCallum, 1999; Schapire & Singer, 2000; Yang, 1999), semantic annotation of images (Boutell, Luo, Shen, & Brown, 2004; Zhang & Zhou, 2007), music categorization into emotions (Li & Ogihara, 2003; Trohidis, Tsoumakas, Kalliris, & Vlahavas, 2008) and bioinformatic (Diplaris, Tsoumakas, Mitkas, & Vlahavas, 2005; Roth & Fischer, 2006; Zhang & Zhou, 2006).

In Katakis, Tsoumakas, and Vlahavas (2008) it is indicated that traditional *single-label* classification is concerned with learning from a set of examples that are associated with a single label λ from a set of disjoint labels L , $|L| \leq 1$. If $|L| = 2$, then the learning task is called *binary classification* (or *filtering* in the case of textual and web data), while if $|L| \geq 2$, then it is called *multi-class classification*. In multi-label classification, the examples are associated with a set of labels $Y \subseteq L$.

There are two major tasks in supervised learning from multi-label data: *multi-label classification* (MLC) and *label ranking* (LR). We would like to implement methods that are able to mine both an ordering and a bipartition of the set of labels from multi-label data. Such a task has been recently called *multi-label ranking* (MLR) (Brinker et al., 2006) and poses a very interesting and useful generalization of MLC and LR.

Multi-label classification methods can be categorized into two different groups: (i) *problem transformation methods*, and (ii) *algorithm adaptation methods* (Tsoumakas & Katakis, 2007). The first group of methods are algorithm independent. They transform the multi-label classification task into one or more single-label classification, regression or label ranking tasks. The second group of methods extend specific learning algorithms in order to handle multi-label data directly.

4.1. Multi-label classification algorithms

We took into account, the resulting view in Trohidis, Tsoumakas, Kalliris, Vlahavas, and Multilabel (2008), to compare classification algorithms: binary relevance (BR), label powerset (LP), random k-labelsets (RAKEL) (Tsoumakas & Vlahavas, 2007) and multi-label k-nearest neighbor (MLkNN) (Zhang & Zhou, 2007; <http://lamda.nju.edu.cn/datacode/MLkNN.htm>; Tsoumakas & Vlahavas, 2007).

We used a machine learning method for performing an empirical evaluation of both algorithms RAKEL and MLkNN based on one multi-label data set of LOs. We also experimented with machine learning, in building a multi-label model using a training data set of LOs and then applying it to a new (unlabeled) data set, in order to obtain predictions and for new instances classification.

Multilabel classifiers such as RAKEL could be used for the automated classification of LO collections in multiple types queries (classes). We have used search for locating LOs by keyword based upon the metadata's contents. Such querying capability would be useful for LO selection in various LOR in the implementation of LO retrieval systems.

The RAKEL algorithm was selected, as a recent method that has been shown to be more effective than the first two (Trohidis et al., 2008). The RAKEL method constructs an ensemble of LP classifiers. This way RAKEL manages to take label correlations into account, while avoiding LP's problems. A ranking of the labels is produced by averaging the zero one predictions of each model per considered label. Thresholding is then used to produce a bipartition as well.

MLkNN was selected, as a recent high-performance representative of problem adaptation methods that is based on k Nearest Neighbors (kNN) lazy learning algorithm.

4.2. Experimental work

The LO dataset was taken after making 60 queries to different repositories because according to Ternier et al. (2008) the access

LOs can take advantage of queries upon metadata for selecting the objects that are most suited to the needs of learners or teachers.

4.2.1. Data set

Here we present the experimental results for a LO dataset, which contains 253 LO examples, annotated with one or more out of the 38 labels corresponding to types queries identified by teachers and pupils as necessary to support their learning discovery activities, such as Programming languages, Algorithm construction, etc. Each LO is described with 1442 features extracted from the Learning Object Metadata (LOM). Here is a description of *LO.arff* data file which was used in the experiments:

Repository: where data was obtained.

Query: used for data obtaining.

Hash: code of learning object.

The remaining fields are counters of the elements of the IEEE LOM (IEEE-LTSC, 2002), which is if you have an element of the General Category. There is a description of these items here (http://man-date.cdlr.strath.ac.uk/graphics/diagrams/IMSMD_Bestv1p3pd2.gif). The data set format (number of labels, number of features, order of attributes, etc.) must conform to the format of the training data set based on the model that was used.

We used the RAKEL classifier from the Java Library for Multi-Label Learning (MULAN) (Tsoumakas, Vilcek, Spyromitros, & Vlahavas, 2011), to obtain the predictions of a trained model for a data set with unlabeled instances. For the experiments we followed the directive that is available on-line in open-source MULAN system, which consists of three parts:

1. We load the multi-label data set that was going to be used for training the model. The training data was provided for the LO, in two text files required by Mulan for the specification of a multi-label data set: an XML file specifying the names of the labels (*LO.xml*), and an *ARFF* file specifying the actual data (*LO.arff*).
2. We created an instance of the learning algorithm that we wanted to train, in order to build a model and obtain predictions. We created an instance of the RAKEL algorithm. RAKEL is actually a meta algorithm and can accept any multi-label learner as a parameter, but is typically used in conjunction with the LP algorithm. It accepts a single-label classifier as a parameter. We used the C4.5 algorithm from Weka for this purpose (J48). RAKEL has three parameters that need to be selected prior to training the algorithm: (a) the subset size, (b) the number of models and (c) the threshold for the final output. We used an internal 10-fold cross-validation on the training set, in order to automatically select these parameters. The subset size was varied from 2 to 5, the number of models from 1 to 100 and the threshold from 0.1 to 0.9 with a 0.1 step. 10 different 10-fold cross-validation experiments were run for evaluation. The results that follow are averages over these 100 runs of the different algorithms. The number of neighbors in MLkNN was set to 10.
3. We trained the classifier using the LO dataset that we loaded with different LOR (Lorinet and Merlot). Two different 10-fold cross-validation experiments were run for evaluation. The results that follow are averages over these 100 runs of both algorithms.

For the empirical evaluation of both algorithms we used the CrossValidate method of the Evaluator class of Mulan. This returns a MultipleEvaluation object, which was printed to see the results in terms of applicable evaluation measures available in Mulan. To

obtain predictions we loaded the unlabeled data instances. The learner returns an instance of MultiLabelOutput class as a result of prediction. The results of the learner output contain bipartition of labels, label confidences and rankings as predicted for given instances.

4.2.2. Experimental result

According to Tsoumakas, Katakis, and Vlahavas (2009) the evaluation of methods that learn from multi-label data requires different measures than those used in the case of single-label data. There are various measures that have been proposed for the evaluation of bipartitions and rankings with respect to the multi-label data.

Given a multi-label test set $Z = \{(z_i, Z_i) \mid 1 \leq i \leq n\}$, based on the study (Zhang, 2009), the following popular multi-label evaluation metrics can be utilized:

1. Hamming loss.
2. One-error.
3. Coverage.
4. Ranking loss.
5. Average precision.

where z is the test instance ($z \in X$), $X = R^d$ is the input space and Z predicted label set for z ($Z \subseteq L$).

According to Zhang (2009) *Hamming loss* evaluates how many times an instance-label pair is misclassified; *One-error* evaluates how many times the top-ranked label is not in the set of proper labels of the instance; *Coverage* evaluates how many steps are needed, on average, to move down the label list in order to cover all the proper labels of the instance; *Ranking loss* evaluates the average fraction of label pairs that are reversely ordered for the instance; *Average precision* evaluates the average fraction of labels ranked above a particular label $l \in Z$ which actually are in Z . Note that for the first four metrics, the smaller the metric value the better the performance. While for *Average precision*, the bigger the metric value the better the performance.

We then performed experiments using these measures to compare both algorithms. Results are displayed in Table 1 which shows the predictive performance of both competing multilabel classification algorithms using these measures. In the case of RAKEL algorithm all metrics significantly outperform the MLkNN algorithm. Experimental results indicate that not only is the RAKEL algorithm more efficient in training and testing than MLkNN, but that it also improves predictive accuracy. Table 1 shows the Classification Accuracy or Subset Accuracy of both algorithms. We notice that RAKEL dominates the MLkNN algorithm in this measure.

The results demonstrate that the RAKEL algorithm can be used to better the classification of LOs in types queries based upon the metadata contents. Fig. 1 shows how RAKEL separates the LOs according to their keywords for labeling the LO in types queries. In addition, Fig. 1 shows the number of examples annotated with each label. We noticed that based on the ease of predictions we can rank the labels in the following descending order: $L_2, L_3, L_5, L_6, L_7, L_8, L_9, L_{11}, \dots, L_{38}$.

Table 1

Performance results of each compared algorithm (mean \pm std. deviation).

Evaluation criterion	RAKEL algorithm	MLkNN algorithm
Classification accuracy	0.1862 \pm 0.1056	0.0948 \pm 0.1056
Hamming loss	0.078 \pm 0.0156	0.0864 \pm 0.0150
One-error	0.5180 \pm 0.0888	0.7029 \pm 0.1094
Coverage	10.4075 \pm 2.4644	22.0614 \pm 2.7271
Ranking loss	0.1766 \pm 0.0639	0.4187 \pm 0.0519
Average precision	0.5581 \pm 0.0738	0.3550 \pm 0.0524

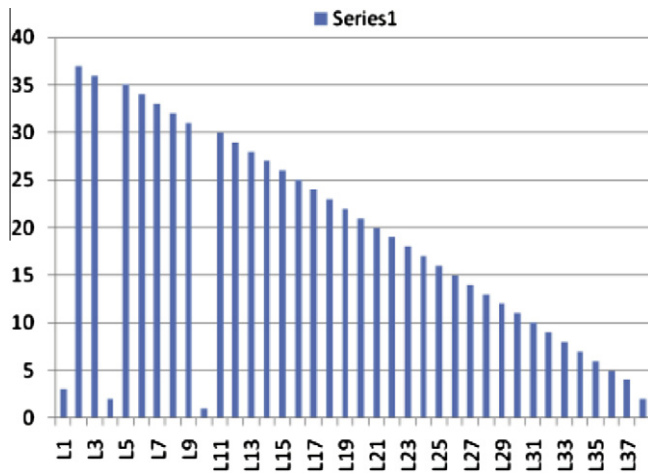


Fig. 1. Number of examples annotated with each label.

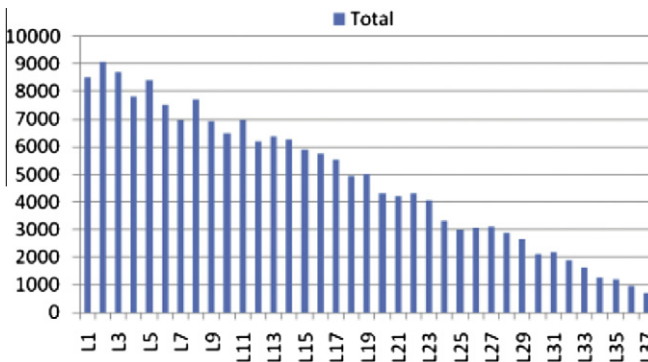


Fig. 2. The ranking for Grammar LO.

Using an instance of the RAKEL algorithm it has been possible to make both an ordering and a bipartition of the set of labels from multi-label data to the MLR task. Fig. 2 shows an example of the ranking for Grammar LO.

5. Initial experiments of web classification using text features

5.1. Data set

The data set for the initial experimental study contained 1000 examples from two repositories: Lornet and Merlot. For each example, the title and description were used to extract the text features for the classification, and the class labels came from the keyword field.

5.2. Text feature extraction

From the title and description of the 1000 examples, we extracted 997 terms (features) after removing stop words and stemming, which are two widely used text preprocessing methods in text mining and information retrieval. We used the toolkit of Mallet (<http://mallet.cs.umass.edu/download.php>) to process the data. After the preprocessing, we obtained an example-term matrix, and each element of the matrix represents the term count which means how many times a term appears in an example. In order to prevent a bias towards longer documents, we normalized the count to term frequency to measure the importance of a term.

5.3. Classification

We used the support vector machine (SVM) with linear kernel as the classifier to conduct the classification. In order to fairly measure the performance, we used 10-fold cross validation to examine the results. In 10-fold cross validation, the 1000 examples were randomly partitioned into 10 subsamples. Of the K subsamples, a single subsample was retained as the validation data for testing the model, and the remaining 9 subsamples were used as training data. The cross-validation process was then repeated 10 times, with each of the 10 subsamples used exactly once as the validation data. The 10 results from the folds then can be averaged (or otherwise combined) to produce a single estimation. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. The average accuracy for the classification using the 1000 examples is 36%. From the 1000 examples, the following examples are correctly classified: 10, 11, 12, 13, 14, 15, 19, 21, 26, 27, 30, 31, 37, 40, 41, 42, 43, 45, 46, 47, 48, 49, 50, 51, 52, 53, 61, 62, 63, 65, 68, 70, 78, 81, 82, and 84. (The examples are ordered based on the same order in the *DatasetLOEngR.arff* data file.)

6. Initial experiments of LO classification using LO feature extraction

6.1. Data set

For the experiment, we used the same metadata (LO features) like in Section 5.1 on the same 1000 examples to perform classification using MULAN to compare the accuracy with the results from pure text features.

6.2. Classification

Table 2 shows the experiment's classification results using the LO features, which indicate that the RAKEL algorithm is more efficient than MLkNN in four measures (Accuracy, Hamming Loss, One-Error, Average Precision). Furthermore, the MLkNN algorithm significantly outperforms RAKEL in terms of Coverage and Ranking Loss. The RAKEL algorithm is substantially superior to MLkNN in Classification Accuracy.

6.3. Observations and discussion

When we looked at the data, we found that the nature of the data made the prediction task very challenging. We do see some potential hierarchies in the examples. For instance, we can use the keywords of the queries as the first layer labels, and the keywords of these examples can be constructed as the sub-class labels of the first layer. However, the hierarchical structure of current examples is not very clear. If we can select some examples whose keywords contain clear hierarchy information (hopefully more than 3 layers), then we can perform hierarchical classification/clustering on the data.

Table 2
Performance results.

Class	RAKEL	MLkNN
Classification accuracy	0.9000 ± 0.1000	0.4000 ± 0.1183
Hamming loss	0.0056 ± 0.0056	0.0333 ± 0.0066
One-error	0.0800 ± 0.0872	0.1700 ± 0.1345
Coverage	0.9300 ± 1.1261	0.7100 ± 0.7778
Ranking loss	0.0547 ± 0.0662	0.0418 ± 0.0458
Average precision	0.9273 ± 0.0793	0.8822 ± 0.0992

7. Conclusions

The search and classification of services for educational content, and specifically LOs, presented in this report constitute the core of the development of distributed, open computer-based educational systems. For this reason, the research in this area has been so active in recent years.

We also tried to utilize a multi-label classification algorithm in order to build a model to classify and catalog the LOs in types queries.

The RAKEL algorithm was very effective; it significantly outperforms all the other compared algorithms in terms of Accuracy, Hamming Loss, One-Error, Ranking Loss, Average Precision. It was also proposed for LOs ranking.

When we compared the use classification with text features with the results from metadata (LO feature), based on classification accuracy, the multi-label classification algorithms with LO features, on the same 1000 examples are substantially superior to web classification with text features. The average accuracy for the classification is 36% using text features with respect to the results using metadata (LO feature). The RAKEL algorithm has a classification accuracy of approximately 90%.

The web classification with text features will be a good alternative because many LOs include textual material that can be indexed, and such indexes can also be used to filter the objects by matching them against user-provided keywords or tagging. The sorting system proposed is also very convenient, given that the LOM standard does not define a minimal set of fields that a LO must have; this makes it difficult to evaluate if a LO has a sufficient quality. Using the feedback provided by the users, from the daily use of the application, the multi-classifier goes through a learning process, which allows it to continually improve its results. Multi-label classifiers such as RAKEL could be used for the automated annotation of large LOR collections with multiple LO. This in turn would support the implementation of LO information retrieval systems that query LO collections by tags. Such a querying capability would be useful for LO selection in various applications. These LOs will be processed according to certain classification criteria that have been personalized and are considered most appropriate for the user.

Therefore this model offers a methodology that illustrates the task of multi-label mapping of LOs into types queries through an emergent multi-label space, and that can learn objects ranking tasks to select learning materials establishing a ranking system for the LOs.

8. Future work

For future work we want to experiment with:

1. From a data collection point of view:
 - Obtain more examples with, relatively, a fewer number of labels. And try to see if we can obtain data with multi-labels if applicable.
 - Obtain more external text information, such as tags and comments.
 - If we can find some examples whose keywords are of hierarchical structures, that would be better.
2. Combining these two kinds of features in various ways, for example, combining features, optimization-based integration. If hierarchical structures can be constructed, there are also some methods to ensemble hierarchical clusters.

References

- Bauer, M., & Stefan, R. M. (2008). *Thalman metadata generation for learning objects an experimental comparison of automatic and collaborative solutions. MKWI 2008*. Munchen: Springer.
- Begelman, G., Keller P., & Smadja, F. (2006). Automated tag clustering: Improving search and exploration in the tag space. In *Proceedings of the collaborative web tagging workshop at WWW*.
- Boutell, M., Luo, J., Shen, X., & Brown, C. (2004). Learning multi-label scene classification. *Pattern Recognition*, 37, 1757–1771.
- Brinker, K., Fürkranz, J., & Hüllermeier, E. (2006). A unified model for multi-label classification and ranking. In *Proceedings of the 17th European conference on artificial intelligence (ECAI 06)*, Riva del Garda, Italy (pp. 489–493).
- Cattuto, C., Loreto, V., & Pietronero, L. (2007). Semiotic dynamics and collaborative tagging. *PNAS*, 104(5), 1461–1464.
- Chiappe, A., Segovia, Y., & Rincon, H. Y. (2007). Toward an instructional design model based on learning objects. *Educational Technology Research and Development*, 55, 671–681.
- DublinCore Metadata Initiative (DCMI) (2007). <<http://dublincore.org>>.
- Del.icio.us. <<http://del.icio.us/>>.
- Diplaris, S., Tsoumakas, G., Mitkas, P., & Vlahavas, I. (2005). Protein classification with multiple algorithms. In *Proceedings of the 10th panhellenic conference on informatics (PCI 2005)*, Volos, Greece (pp. 448–456).
- Flickr. <<http://www.flickr.com>>.
- Goarany, K., Kulczycki, G., & Blake, M.B. (2010). Mining social tags to predict mashup patterns, SMUC10, ACM 978-1-4503-0386-6/10/10.
- Golder, S. A., & Huberman, B. A. (2005). *The structure of collaborative tagging systems*. Palo Alto, USA: Information Dynamics Lab: HP Labs.
- Hassan-Montero, Y., & Herrero-Solana, V. (2006). Improving tag-clouds as visual information retrieval interfaces. In *Proceedings of InfoSci2006*.
- IEEE Learning Technology Standard Committee (IEEE-LTSC) (2002). WG12 Learning Object Metadata. <<http://ieeeltsc.org/wg12LOM/>>.
- Katakis, I., Tsoumakas, G., & Vlahavas, I. (2008). Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD 2008 discovery challenge*, Antwerp, Belgium.
- Li, L., & Ogihara, M. (2003). Detecting emotion in music. In *Proceedings of the international symposium on music information retrieval* (pp. 239–240), Washington, DC, USA.
- Mathes, A. (2004). Folksonomies cooperative classification and communication through shared metadata. *Computer Mediated Communication – LIS590CMC*.
- McCallum, A. (1999). Multi-label text classification with a mixture model trained by EM. In *Proceedings of the AAAI 99 workshop on text learning*.
- Mika, P. (2005). Ontologies are us: A unified model of social networks and semantics. In *Proceedings of ISWC05*.
- Motelet, O., Baloian, N., & Pino, J. A. (2006). Learning object metadata and automatic processes: Issues and perspectives. In K. Harman & A. Koohang (Eds.), *Learning objects: Standards, metadata, repositories, and LCMS*. Informing Science Press.
- O'Reilly, T. (2008). What Is Web 2.0. Design patterns and business models for the next generation of software. <<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>> Last access 17.02.08.
- Rollett, H., Lux, M., Strohmaier, M., Dosinger, G., & Tochtermann, K. (2007). The Web 2.0 way of learning with technologies. *International Journal of Learning Technology*, 3(1), 87–107.
- Roth, V., & Fischer, B. (2006). Improved functional prediction of proteins by learning kernel combinations in multilabel settings. In *Proceeding of 2006 workshop on probabilistic modeling and machine learning in structural and systems biology (PMSB 2006)*, Tusula, Finland.
- Schapiro, R. E., & Singer, Y. (2000). Boostexter, A boosting-based system for text categorization. *Machine Learning*, 39(2/3), 135–168.
- Sierra, J.L., & Fernández-Valmayor, A. (2008). Tagging learning objects with evolving metadata schemas. In *Eighth IEEE international conference on advanced learning technologies*. 978-0-7695-3167-0/08. 2008 IEEE Computer Society. p. 829. doi:10.1109/ICALT.2008.129.
- Surowiecki, J. (2004). *The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies and nations*. Little, B& T Press.
- Ternier, S., Massart, D., Campi, A., Guinea, S., Ceri, S., & Duval, E. (2008). Interoperability for searching learning object repositories. *The ProLearn Query Language. D-Lib Magazine*, 14. ISSN 1082-9873.
- Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I., & Multilabel, I. (2008). Classification of music into emotions. In *Proceedings of the 9th international conference on music information retrieval (ISMIR)*.
- Trohidis, K., Tsoumakas, G., Kalliris, G., & Vlahavas, I. (2008). Multilabel classification of music into emotions. In *Proceedings of the 9th international conference on music information retrieval (ISMIR)*.
- Tsoumakas, G., Vlahavas, I. (2007). Random k-labelsets: An ensemble method for multilabel classification. In *Proceedings of the 18th European conference on machine learning (ECML 2007)* (pp. 406–417) Warsaw, Poland.
- Tsoumakas, G., Vilcek, J., Spyromitros, E., & Vlahavas, I. (2011). Mulan: A Java library for multi-label learning. *Journal of Machine Learning Research*, 12, 2411–2414.
- Tsoumakas, G., Katakis, I., & Vlahavas, I. (2009). Mining multi-label data. In O. Maimon & L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (2nd ed.). Springer.
- Tsoumakas, G., & Katakis, I. (2007). Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3, 1–13.
- Tsoumakas, G., Katakis, I., & Vlahavas, I. (2010). Mining multi-label data. In O. Maimon & L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (2nd ed.). Springer.
- O'Reilly, T., What is web 2.0. <<http://www.oreillynet.com/lpt/a/6228>>.

- Weiss, A. (2005). The power of collective intelligence. *netWorker*, 9(3), 16–25.
- Wu, X., Zhang, L., & Yu, Y. (2006). Exploring social annotations for the semantic web. In *Proceedings of WWW06*.
- Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, 1, 67–88.
- Zhang, M. L. (2009). *ML-rbf: RBF neural networks for multi-label learning* Min-Ling. *Journal Neural Processing Letters Archive* (vol. 29). Hingham, MA, USA: Kluwer Academic Publishers (2), doi:10.1007/s11063-009-9095-3.
- Zhang, M. L., & Zhou, Z. H. (2006). Multi-label neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10), 1338–1351.
- Zhang, M. L., & Zhou, Z. H. (2007). Mlkn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40, 2038–2048.