Automatic Classification of Teaching and Learning Materials Based on Standard Education Classification Schemes

A. Joorabchi and A. E. Mahdi

Department of Electronic and Computer Engineering, University of Limerick, Limerick, Ireland

Abstract- *With the significant growth in available electronic education materials such as syllabus documents and lecture notes on the Internet and intranets there is a need for developing efficient indexing/categorizing mechanisms to organize such E-documents in institutional repositories. In this paper we describe our approach for automatic classification of syllabus documents in the national Irish syllabus repository. The classifier software component is based on the well-known naïve Bayes classification algorithm. We have also studied the application of a web corpus in training an unsupervised classifier which eliminates the need for manual classification of a training set required in standard classifications systems.*

1. Introduction

Similar to physical libraries, Large-scale digital libraries such as the Irish syllabus repository contain thousands of items and therefore require deploying flexible querying and information retrieval techniques that allow users to easily find the items they are looking for. In order to provide highly precise search results we need to go beyond the traditional keyword-based search techniques which yield a large volume of indiscriminant search results without regard to the content. Classification of materials in a digital library based on a pre-defined scheme improves the accuracy of information retrieval significantly and allows users to browse the collection by subject. However, manual classification is a tedious and expensive job requiring an expert cataloger in each knowledge domain represented in the collection and therefore deemed unfeasible in many cases. Automated Text Classification/Categorization (ATC) - the automatic assignment of natural language text documents to one or more predefined categories or classes according to their contents - has become one of the key techniques to enhance information retrieval and knowledge management of large digital collections. Sebastiani in [1] provides an overview of standard methods for ATC such as Naive Bayes, k-NN, and SVM. Text classification algorithms have been successfully used in a wide variety of practical domains such as spam filtering and cataloging news articles and web pages. However, to the best of our knowledge, ATC methods have not been adapted before to automatically classify a large collection of syllabus document based on a standard education classification scheme such as International Standard Classification of Education (ISCED) [2].

2. Syllabus Classifier

The task of the syllabus classifier is to automatically assign a classification code to each individual course/module based on a predefined education classification scheme. Currently, the Higher Education Authority (HEA) and higher educational institutions in Ireland use the International Standard Classification of Education (ISCED) [2] to provide a framework for describing statistical and administrative data on educational activities and attainment in Ireland. This classification scheme is suitable for subject/discipline based classification of full undergraduate or postgraduate programs however it does not provide the level of detail required for classifying individual modules. The need for a more detailed national education classification standard than that provided by the ISCED has already been recognised by educational authorities within other jurisdictions. This has led some other countries to develop their own national classification of education standards such as JACS [3] in UK and ASCED [4] in Australia. In order to standardise the classification of modules among Irish higher education institutes, HEA is currently considering the development of Irish Standard Classification of Education. The current version of the classifier classifies the syllabus documents based on a draft extended version of ISCED which will be replaced by the Irish Standard Classification of Education in future. The syllabus classifier component is based on the widely used Naïve Bayes algorithm, described in [5]. However, we are also experimenting with the application of a search engine in automatic collection of a training set for creating a fully unsupervised document classification system.

2.1 Multinomial Naïve Bayesian Classifier

The underlying theorem for Naïve Bayesian text classification is the Bayes rule, expressed as:

$$P(A_i \mid B_j) = \frac{P(A_i) \times P(B_j \mid A_i)}{P(B_j)} \quad (1)$$

As indicated, it enables the calculation of the likelihood of event A_i given that B_j has happened. When applied to text classification, eq.1 can be rewritten as:

$$P(Class_i \mid Document_j) = \frac{P(Class_i) \times P(Document_j \mid Class_i)}{P(Document_j)} \quad (2)$$

Such that the Rule is used to calculate the probability of each predefined $Class_i$ given $Document_j$, and the Class with the highest probability is allocated to $Document_j$. In eq.2, $P(Document_j)$ is a constant divider, common to every calculation and therefore can be safely removed from the equation, such that the class of $Document_j$ is estimated as: $Class(Document_j) = \arg\max_i P(Class_i) \times P(Document_j \mid Class_i)$.

In this model each documents is represented as a vector of words in a multidimensional space, where each dimension corresponds to a distinct word and the distance along that dimension is the number of

times the given word occurs in the document. In a standard supervised setting, a set of manually classified training documents is used to parameterize the class prior probabilities, $P(Class_i)$, and the class-conditioned (word) probabilities, $P(Document_j \mid Class_i)$. The conditional probability of each word appearing in the vocabulary, w_k, being part of a given class, $Class_j$, is estimated by:

$$P(w_k \mid Class_j) = \frac{n_k + 1}{n + \mid Vocabulary \mid}, \tag{3}$$

where n_k is the number of times the word occurs in the training documents which belong to the $Class_j$, n is the total number of distinct words in the training documents which belong to the $Class_j$, and *Vocabulary* is a set of all distinct words which occur in all training documents. Each estimate is primed with a count of one to avoid probabilities of zero (Laplace smoothing). For each class the prior probability is estimated as:

$$P(Class_i) = \frac{\mid Document_i \mid}{\mid Documents \mid} \tag{4}$$

where *Documents* is a set of all training documents and $Document_i$ is a subset of *Documents* which belong to the $Class_i$. If a document, $Document_j$, is to be classified, the most likely class C_{NB} for $Document_j$ would be computed as:

$$C_{NB} = \arg\max_{i \in V} P(Class_i) \prod_{k=1}^{\mid Document_i \mid} P(w_k \mid Class_i) \tag{5}$$

where V is a set of all possible target classes. Multiplying lots of probabilities, which are between 0 and 1 by definition, can result in floating-point underflow. To avoid this problem we perform all computations by summing logs of probabilities rather than multiplying probabilities, such that:

$$C_{NB} = \arg\max_{i \in V} P(Class_i) + \sum_{k=1}^{\mid Document_i \mid} P(w_k \mid Class_i) \tag{6}$$

The Class with the highest final un-normalized log probability score is the most probable. Reducing the size of the vocabulary by selectively choosing the words to provide as input to the learning algorithm can improve both the accuracy and scalability of the classification. For an overview of feature selection methods see [6]. In this work, stop-word removal is used to reduce the size of vocabulary by excluding the words that appear frequently in all the training documents and therefore have no predictive value. We used a generic stop-list of 526 words from Bow toolkit [7], a list of common words such as "the", "of", "is", etc and enhanced it with a syllabus domain-specific stop-list of 50 words such as “student”, “semester”, “lecturer”, etc. We do not use stemming or any other common feature selection/reduction

method. However, during the unsupervised training process a word frequency threshold is used for non-leaf nodes in the classification scheme hierarchy tree (see 3).

3. Unsupervised Training

A major difficulty of supervised approaches for text classification is that they require a great number of training instances in order to construct an accurate classifier. Joachims [8] measured the accuracy of Bayes classifier with a dataset of 20,000 Usenet articles, called 20-Newsgroup collection. She reports that the Bayes classifier achieves the highest accuracy of %89.6 when trained with 13,400 documents (670 documents per class), and the accuracy reduces to %66 when 670 documents (33 documents per class) are used to train the classifier. As this and other experimental results show increasing the size of training corpus improves the accuracy of the classifier substantially. However, manual classification of documents for the purpose of training a classifier is a tedious and expensive job. Motivated by this problem, the semi-supervised and unsupervised training methods are being researched to train a classifier with a limited number of training documents and no training documents, respectively (see [9] for an overview). In this work we experiment with an un-supervised web-based approach to train a Naïve Bayes classifier used for classifying syllabus documents based on a hierarchical education classification scheme.

The classification scheme used here is an extended version of ISCED [2] represented in XML. The ISCED is a hierarchical scheme with three levels of classification: broad field, narrow field, and detailed filed. Accordingly, the scheme uses a 3-digit code in a hierarchical fashion for classifying fields of education and training , such that the first digit represents 'broad field', the second digit represents the 'narrow field' and third digit resents the 'detailed field' of a given document. There are 9 broad fields, 25 narrow fields and about 80 detailed fields. We have extended this by adding a forth level of classification, subject field, which is represented by a letter in the classification coding system. For example a module assigned the classification code "482B" would indicate that module belongs to the broad field of "Science, Mathematics and Computing", the narrow field of "Computing", the detailed field of "Information Systems" and the subject field of "Databases", where the broad fields, narrow fields and detailed fields represent the branches of the upper three levels of the classification hierarchical tree, from top to bottom respectively, and the subject fields represent the leaves of the tree.

The classifier starts the training process by reading the XML version of classification scheme and collecting a list of subject fields (leaf nodes). Then a search query created from the name of the first

subject field in the list combined with the keyword “syllabus” is submitted to the Yahoo search engine using the Yahoo SDK [10]. For example, the query created for the subject filed 482B, databases, is “databases syllabus”. The first hundred URL’s in the returned results are passed to the Gate toolkit [9], where the files (HTML, Text, PDF, and MS-Word) that they are pointing to are downloaded and their text content is extracted and tokenized. This process is repeated for all the subject fields in the hierarchy. The tokenized text documents resulting from this process are converted to word vectors which are used to train the classifier described in Section 2.1 for classifying syllabus documents in subject-field level and to create word vectors for the fields which belong to the upper three levels of the classification hierarchical tree. The words used in the name of subject fields have direct effect on the quality of search results and using words that have a high information gain value improves the quality of search results however we have not changed the subject field names in this experiment as we wanted to measure the accuracy of the system with a standard classification scheme in its original form. The other factor affecting the quality of search results is the number of learning-teaching documents in each field that are available online. For example the quality of search results for computer related fields such as databases, programming languages, and artificial intelligence is substantially higher than fields such as veterinary nursing or cereal science which are less populated. Our experiment shows that the number of relevant syllabus documents retrieved from the first hundred URL’s of search result can vary between 20 and 40 depending on these two factors. Although the remaining documents are not syllabus documents, the majority of them can be classified to the subject field or its parent (i.e., a detailed field), making them useful for training the classifier. For example looking for syllabus documents in the subject field of databases a lot of the retrieved documents might not be database-related syllabus documents but they still can be classified to the subject field of databases as their main content discusses some aspect of the database systems. Also in majority of cases that the retrieved document can not be classified to the subject field of databases it still can be classified to the detailed field of computer science which is the parent of databases subject field .

The subject-field word vectors created by leveraging a search engine are used in a button-up fashion to construct word vectors for the fields which belong to the higher levels of hierarchy. We illustrate this method with help of the following example. Assume we want to create a vector of words for the detailed field of information systems. There are four subject fields that descend from it: Systems Analysis and Design, Databases, Decision Support Systems, and information systems management. We build a master vector by combining the vectors of these four subject fields and then normalize the word frequencies by

dividing the frequency of each word in the master vector by the total number of subject field vectors used to create it (4 in this case) and then round the quotient to a positive integer number, as illustrated in Fig.1. During the normalization process, if the frequency of a word is less than total number of vectors it will be removed from the vocabulary. This feature reduction technique reduces the size of vocabulary by removing the words that their frequency is too small to affect the classification results and also avoids decimal frequency values below 1, which both improve the speed of classification process.

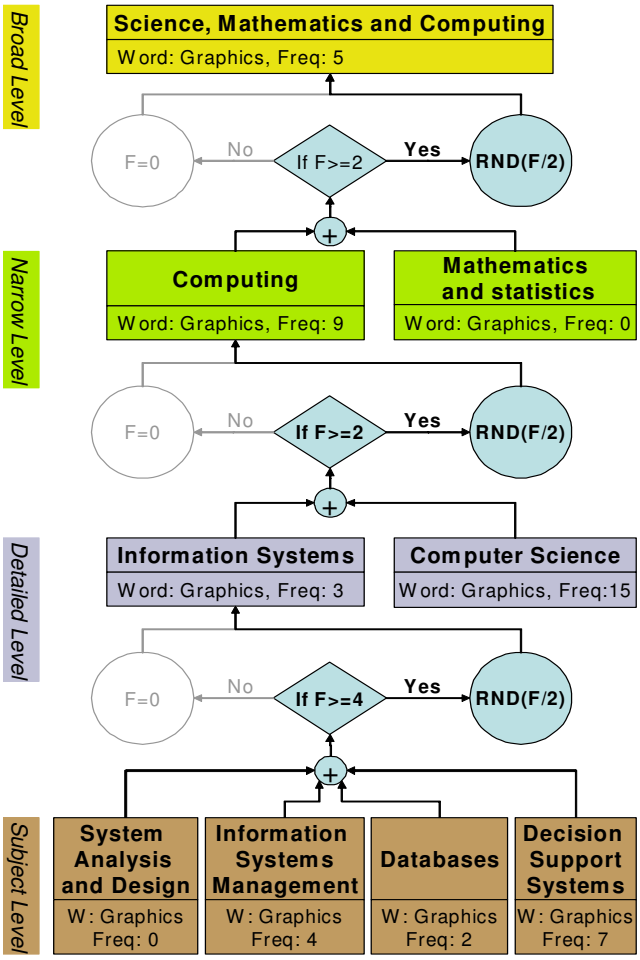

Fig. 1. *The process of word vector creation for broad, narrow and detailed fields of the classification hierarchy.*

This process described above can be expressed as follows:

$$F(w_i) = \begin{cases} 0 & \text{if } \mathrm{Sum}\left(F(w_i)\right) < \left|Fields\right| \\ \mathrm{RND}\left(\dfrac{\mathrm{Sum}(F(w_i))}{\left|Fields\right|}\right) & \text{if } \mathrm{Sum}\left(F(w_i)\right) \geq \left|Fields\right| \end{cases}, \quad \mathrm{Sum}\left(F(w_i)\right) = \sum_{n=1}^{\left|Fields\right|} F(w_{n,i}) \qquad (7)$$

Eq. 7 is then repeated in a similar fashion to create word vectors for all the detailed, narrow and broad fields of the classification hierarchy in a bottom-up manner. In rare cases where a detailed or narrow field does not have any descendent, the web-based approach is used to create a word vector for higher level fields.

4. Evaluation & Results

Micro-average precision measure is used to report the performance of the classifier component. The measure is computed as follows:

$$P_m = \frac{\text{Number of correctly classified documents in all classes}}{\text{Total classified in all classes}} \quad (8)$$

The performance of the classifier was measured using a hundred undergraduate and a hundred postgraduate syllabus documents. The micro-average precision achieved for undergraduate syllabi was 0.75 and it decreased to 0.60 for postgraduate syllabi. Examining syllabus documents from both groups indicates that some syllabi are describing courses which contain components belonging to different fields of study. For example a syllabus document could be describing a course which contains both database design and web design components. Classifying such documents which belong to more than one class is more error-prone and requires the classifier to recognize the core component of the course. Since the number of multi-component courses is substantially higher among the group of postgraduate courses, therefore the classification accuracy achieved for this group of syllabus documents is about 15% lower than undergraduate syllabi. Also it should be noted that this level of accuracy is achieved without using any manually classified training document to train the classifier.

5. Conclusions

In this paper, we highlighted the need for semantic indexing of learning and teaching materials in institutional repositories and described our unsupervised machine learning-based approach. In future, we plan to improve the accuracy of the classifier by automatic filtration of training documents obtained by the search engine to increase the percentage of valid training documents. Also, we plan to investigate the impact of multi-class categorization on the accuracy of the system.

References

[1] Sebastiani, F. Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)*, 34, 1 (2002), 1-47.

[2] *International Standard Classification of Education - 1997 version (ISCED97)*. (UNESCO, 2006) [cited 2007 December]; [Online]. Available: http://www.uis.unesco.org/ev.php?ID=3813_201&ID2=DO_TOPIC

[3] *Joint Academic Coding System (JACS) v 1.7*. (HESA - Higher Education Statistics Agency, UK) [cited 2007 December]; [Online]. Available: http://www.hesa.ac.uk/index.php?option=com_content&task=view&id=158&Itemid=233

[4] Trewin, D. *Australian Standard Classification of Education (ASCED)*. (Australian Bureau of Statistics, 2001) [cited 2007 December]; [Online]. Available: http://www.abs.gov.au/AUSSTATS/abs@.nsf/DetailsPage/1272.02001?OpenDocument

[5] T.Mitchell, *Machine Learning*. 1997, McGraw-Hill. p. 180-184.

[6] Forman, G., *Feature Selection for Text Classification*, in *Computational Methods of Feature Selection*. 2007, Chapman

[7] McCallum, A. *Bow: A Toolkit for Statistical Language Modeling, Text Retrieval, Classification and Clustering*. (Released under the open source LGPL licence, 1998) [Online]. Available: http://www.cs.cmu.edu/~mccallum/bow/

[8] Joachims, T. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning* (Nashville, TN, USA, 1997). Morgan Kaufmann Publishers

[9] Seeger, M. *Learning with labeled and unlabeled data*. Technical report, Institute for Adaptive and Neural Computation, University of Edinburgh, 2000. [Online]. Available: http://www.kyb.tuebingen.mpg.de/bs/people/seeger/papers/review.pdf

[10] *Yahoo! Search Web Services Software Development Kit*. (Yahoo! Inc, 2007) [Online]. Available: http://developer.yahoo.com/search/