Automatic Cataloguing of Web Resources on a Personalized Taxonomy

Nuno Escudeiro
Departamento de Engenharia Informática, Instituto Superior de Engenharia do Porto
R. Dr António Bernardino Almeida, 431, 4200-072 PORTO,
Portugal
nuno@dei.isep.ipp.pt
LIAAD-INESC SA - Laboratório de Inteligência Artificial e Apoio à Decisão
Rua de Ceuta, 118 – 6º, 4050-190 PORTO, Portugal
Paula Escudeiro
Departamento de Engenharia Informática, Instituto Superior de Engenharia do Porto
R. Dr António Bernardino Almeida, 431, 4200-072 PORTO,
Portugal
paula@dei.isep.ipp.pt
GILT - Graphics, Interaction and Learning Technologies
R. Dr António Bernardino Almeida, 431, 4200-072 PORTO, Portugal
José Bidarra
bidarra@univ-ab.pt
Departamento de Ciências Exactas e Tecnologia, Universidade Aberta,
Rua Escola Politécnica, 147, 1269-147 Lisboa, Portugal

Abstract: Information overload is a major concerns retrieval systems face. Information is ubiquitous, available from many distinct sources and the main issue is to get just the right piece of information that might satisfy our specific needs. Many of these sources organize their informational resources on a given ontology. However, these ontologies are static and do not allow for personalization. This fact degrades the value of the service if there is no easy mental mapping between user specific needs and the general source ontology. Organizing informational resources according to particular needs might increase users' satisfaction and save their time. In this paper we present a methodology to filter and organize informational resources according to users' interests, thus granting users with a personalized edition of the resource, especially tailored towards their specific needs. We believe that this methodology may be applied in educational scenarios, where we have a repository of educational objects that are organized according to specific objectives, automatically producing specific courseware. Our experimental results confirm that it is possible to automatically personalize document resources with high precision at a reduced editor workload.

Introduction

Nowadays people are overwhelmed by information which is available from many different sources. The web brought to us a new media where everyone can publish regardless of validity, structure, format or language. In this new scenario, the problem is not to get information but to gather the right few pieces of information and to organize them in a way that satisfies our particular needs.

Besides the web, there are many more sources continuously providing fresh information. It is common to have information distributed through RSS feeds freely available. These feeds present information on an orderly fashion, catalogued on a given ontology previously defined. However, this ontology is static and does not allow for personalization. What if users are interested in a distinct taxonomy? A given resource is useful for a given group of users if it is organized in a way that maps users' interests; however it is useless if there is no easy mental mapping between users' interests and the way information is organized. Organizing collections of objects according to particular needs might increase users' satisfaction and save their time. These objects may be general news stories, collected from a newspaper feed, as we have used in this work, or a set of learning objects in an institutional repository or any other kind.

In this paper we describe myExpresso, a prototype system[19], applied to the Expresso newspaper [22], to deploy and test our methodology. myExpresso requests users to specify their interests through a taxonomy and to

identify a few exemplary news stories representing each class in the taxonomy. Based on that exemplary news, the system builds a model of user's interests which is then used to automatically label fresh news stories that are acquired from the newspaper feed. myExpresso filters news stories and presents them organized according to user interests, granting users with a personalized edition of the newspaper, especially tailored towards their specific needs.

We believe that the application of this methodology in an educational setting might contribute to build – or complement – and maintain educational resources and greatly improve their usefulness. Our methodology may be applied to explore and organize a repository of learning objects, according to specific objectives, thus working as a courseware semi-automatic editor. Students themselves might use such a tool to build a course on a specific competence; or to help them collect a set of learning objects that might help in understanding a given concept or how to apply a given technique.

myExpresso may be seen as an automatic resource compiler, a system that seeks and retrieves a list of the most authoritative documents for a given topic [4]. This is a very broad definition, under which many distinct types of systems may be considered, including, for instance, search engines. In this work we are interested in an automatic resource compiler, which has the responsibility of collecting and organizing a collection of relevant objects, in a continuous effort to keep the collection up-to-date and organized according to user specific needs. Many automatic resource compilation systems and methodologies have been proposed in the past, exhibiting many interesting ideas and characteristics.

Thesus (2003) [10] allows for the users to search documents in a previously fetched and classified document collection. In this system documents are classified based on document contents and link semantics.

WebLearn (2003) [16] retrieves documents related to a topic, specified through a set of keywords, and then automatically identifies a set of salient topics, by analysing the most relevant documents retrieved in response to the user query that describes the topic. The identification of these salient topics is a fully automatic process not allowing for user interference.

iVia (2003) [18] is an open source virtual library system that collects and manages resources, starting with an expert-created collection that is augmented by a large collection automatically retrieved from the web. iVia identifies relevant internet resources through focused crawling [3] and topic distillation approaches.

Personal View Agent, PVA, (2001) [5] learns user profiles in order to assist them when they search information in the web. This system organizes documents in a hierarchical structure – the personal view, which is user dependent and dynamic, automatically adapting to changes in user's interest.

Metiore (2001) [1] is a search engine that ranks documents according to user preferences, which are learned from user historical feedback depending on the user objective.

Personal WebWatcher (1999) [17] analyses page requests, learns a user model and suggests web pages potentially interesting to the user. The system operates offline, when learning user models, and at query time, when recommending interesting pages to the user.

The ARC system (1998) [4] compiles a list of authoritative web resources on any topic. The algorithm has three phases: search and grow, weighting and iteration and reposting.

Letizia (1995) [15] is a user interface agent that assists a user browsing the Web. Letizia also suggests potentially interesting links for the user to follow. Interest in a document is learned through several heuristics that explore user actions, user history and current context.

In our work users are able to specify their information needs, by specifying the topic ontology they are interested in, through examples. From there on, the system learns the topic, periodically read the sources to acquire fresh objects, and automatically organizes them according to specific users' interests. Specifying interests through examples seems adequate once it is very flexible and does not require any prior training neither any specific skills from the editor.

We have applied semi-supervised learning algorithms to learn user needs that have shown to be effective since significantly reduce editor workload while maintaining accuracy.

In the remainder of this paper we will describe our methodology, presents and discusses the results we have achieved related to editor workload and accuracy. At last we present our conclusions and future work.

The Methodology and myExpresso Prototype

News in a newspaper, and many other systems that organize document collections on a static directory, such as Yahoo! [23], systems based on Dewey Decimal Classification [6] and other similar taxonomies [21], are presented in an orderly fashion; however this structure is rigid and does not allow for personalization.

We propose a methodology that is intended to allow users to freely personalize informational resources – which may be educational resources – according to their specific needs. This methodology is deployed through the myExpresso prototype that uses common open source technologies, including Lucene [9], MySQL, Java and PHP.

For sake of clarity we define a few concepts we will use throughout this paper: User is a person seeking for information and Editor is the person in charge of adding and updating contents on a specific web site. A Topic is a specific information need described by the Editor. Resources are collections of documents describing a Topic for a given period of time.

The Editor is responsible for specifying a Topic which is of relevance for the site users. Users explore Resources to get information on that Topic.

Editor and User are roles that may be impersonated by the same single user. An Editor may also define and maintain resources on behalf of a group of users with common interests.

Architecture

The methodology we propose [7] executes a continuous loop (Fig.1) where each of the iterations consists on the following phases (adapted from [8],[14]):

- Acquisition: aims to find and retrieve, from the sources, as many relevant documents as possible while retrieving as few non-relevant documents as possible (tasks 1 and 2).
- Pre-processing: comprises any transformation process that is applied to the retrieved documents to generate appropriate document models (task 3).
- Learning: intends to find patterns and to learn user needs (task 4).
- Analysis and Presentation: aims to facilitate the exploration of large document collections and to detect and adapt to drift in user interests (tasks 5 and 6).

The first iteration for a given topic goes from topic definition to document archival, pre-processing, learning and classification (tasks 1 through 5). This iteration is conducted by the editor and results in the first version of the resource. From this point on two distinct threads are executed:

- In the first thread, the user explores the resource (task 6).
- In the second thread (automatically scheduled and triggered) the system periodically refreshes the resource by reading the sources and classifying incoming documents (news stories at myExpresso) (tasks 2 and 5).

Explicit editor effort is required exclusively for topic specification. Labeling a set of exemplary document is the most demanding task that is required from editors on this phase. These exemplary documents will be used to build a model for the topic.

Topic specification

Topics are specified by the editor, at task 1 (Fig.1). The most relevant features of the topic specification include:

- a taxonomy, representing the ontological structure for the topic – for instance (“sports”, “politics”, “technology”) – and
- a partially classified set of exemplary documents that should include labeled documents on every taxonomy categories.

The topic taxonomy is a hierarchy of concepts specifying the ontological structure of the resource. The root is the topic itself. The taxonomy is merely a way of structuring the resource according to user specific needs.

Each document in the resource is supposed to be associated to just one category – the most specific category in the taxonomy adequately representing the document.

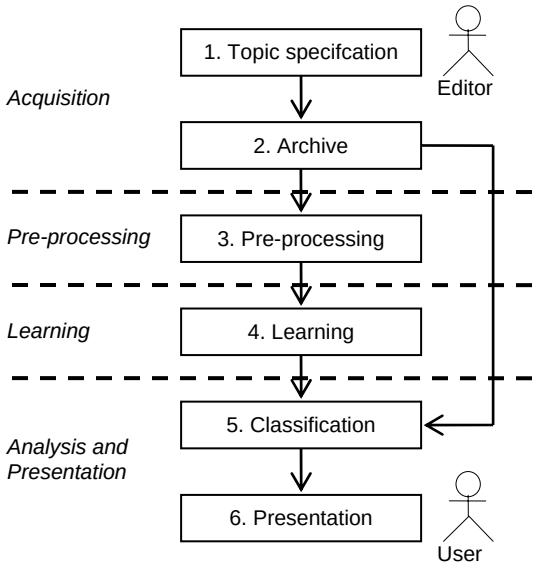

Fig. 1 System architecture

Pre-processing

Each document to be included in the resource is previously transformed by a set of functions that reduce document text, in the title and snippet that is provided by the news feed, to a common standard format adequate for automatic classification. These transformations include lexical analysis, that eliminates punctuation and converts text to lower case, removal of stop words, that eliminates words that do not add semantics, and stemming, that reduces each word to its radical. These two last transformations are language dependant; myExpresso applies a Portuguese list of stop words and a Portuguese stemmer since we are processing Portuguese collections of objects.

After being pre-processed news stories are indexed and standard TF×IDF vectors [20] are generated and stored. Pre-processing and indexing is performed by Lucene classes wrapped in our framework.

Learning

Learning the topic taxonomy in an unsupervised manner, by applying clustering techniques, does not seem appropriate. The user may be interested in an organizational structure different from the one obtained with unsupervised techniques. On the other hand, a supervised learning scheme requires a large number of labeled examples from each category. This is a major drawback since the manual classification of news stories might become costly and highly time-consuming.

We use a semi-supervised solution, requiring the editor to classify a few examples at an initial phase. The system will then learn a classifier for each category in the taxonomy, based on the exemplary pre-labeled documents, using Expectation-Maximization (EM) techniques [2]. In this setting the system learns from few labeled and many unlabeled examples.

News stories content, extracted from the title and snippet, is used to learn the taxonomy applying standard text classifiers to standard TF×IDF vectors.

We apply a Support Vector Machine (SVM) classifier [11] – currently the most accurate classifier for text [2] – wrapped in a simple semi-supervised bootstrapping algorithm [13]. In this method, the classifier is wrapped in a process that iteratively labels unlabeled documents and adds them to the labeled set. This cycle is executed until a certain stopping criterion is met. One of these stopping criteria is based on the concept of classification gradient, which is a way of measuring model improvement between iterations [7]. At the end we store, for each topic, the corresponding SVM model.

Classification

Once a topic model is available we may apply it to classify incoming news stories. For each document to classify, our SVM classifier, SVMLight implementation [12], receives a TF×IDF vector and generates a set of

probabilities, one for each class in the topic taxonomy. These probabilities represent the posterior probability of the document belonging to each class. For each document, the higher probability is compared to a minimum threshold and, when it is higher than that minimum, the corresponding class is attached to the document. When the higher probability is lower then the minimum threshold the document remains unlabeled, meaning that the automatic classifier does not have enough confidence to attach that document to a given class.

Presentation

Users have their resources available and may explore them through a directory that maps their specific interests (Fig. 2).

Fig. 2 Resource presentation

Evaluation

It is not yet clear for us whether this proposal is adequate for automatic courseware construction and maintenance or not. Evaluating this requires us to apply the prototype proposed in the current paper in a learning environment, with a repository of learning objects that may be organized by editors (teachers), according to a set of specific objectives, and further explored by end-users (students).

However, we may evaluate whether editor workload can benefit from semi-supervised learning and if so at what expense in classification accuracy. Thus, our evaluation will be focused on the classification task.

Editor Workload Reduction

In our work we have used applied an EM algorithm for semi-supervised learning which wraps an SVM classifier in a process that iteratively labels unlabeled documents and adds them to the labeled set. This cycle is executed until a given stopping criteria is met. With this process we expect to decrease editor workload and achieve similar accuracy – as in a fully supervised setting – with less work.

To evaluate this we have compiled two datasets: one is composed by 200 web pages related to the topic popular events (PE) – that might be used by someone wishing to learn how Portuguese families commemorate these festivities – and the other is composed of 66 documents related to the topic artificial intelligence (AI). Either topic has three classes: Christmas, Easter and Carnival for PE and Research Laboratories, Events and Documents for AI.

We have started by obtaining reference values under a fully supervised setting and have achieved an error rate of approximately 25%, for the AI resource, and an error rate of approximately 8%, for the PE resource. These are the marks we want to achieve at the semi-supervised setting but with a reduced set of labeled documents.

We have observed that the semi-supervised learning algorithm is able of leveraging evidence from labeled and unlabeled examples, significantly reducing the workload required to obtain a certain accuracy level. At the PE resource we have achieved a minimum supervised error of 8% with a workload of 45 labeled documents. Applying

semi-supervised learning, we have an equivalent error rate, of 10%, for a workload of just 33 documents, a reduction of 27% on the workload. A similar reduction on the workload is achieved at the AI resource.

The advantage of semi-supervised techniques becomes clear at this point. These experiments seem to confirm the validity of semi-supervised learning techniques and to prove that it allows reducing the workload without compromising accuracy. These experiments confirm the advantages of semi-supervised learning for problems where instance labeling is very demanding, especially for the workload reduction they allow without significantly compromising accuracy.

Classification Accuracy

To evaluate the classification accuracy of our text classifier we have defined a topic with 11 distinct classes (labeled 1 to 11). A classification model was generated from a training dataset obtained from the Expresso feed, for a three day period. Then we collected 492 news stories from the 9th September 2007 till October, 31st and manually label them on our 11 class taxonomy. Automatic labels have been generated with our previously trained classifier for those news stories and we compute precision for this set. We got a macro-average precision of 87% and a micro-average precision of 96%. Since our dataset in biased towards some categories – categories 2, 6 and 10 have only one document each while category 4 has 64 documents – micro-averaging seems more reliable. These figures confirm high accuracy.

Conclusions and Future Work

Whenever a user, or group of users, wants to be kept informed on some topic, an automatic resource compilation system can be of great value. We may imagine some potential interested parties: organizations, associations and specific interest groups, commercial companies trying to gather information on their market, news wire services, students interested in some subject, to name a few. Nowadays the problem is not to obtain information but to organize it and to extract its intrinsic value in due time.

We believe that an automatic system, which adapts to user information needs, may be very valuable and potentially interesting to private as well as to professional users. In our work we pretend to automate the content retrieval task in order to reduce editorial effort while improving end-user satisfaction.

We have proposed and described a methodology and a prototype that accomplishes some of the aims mentioned above. Although it was not possible to evaluate to what extent user satisfaction is improved with such functionalities, our experimental results show the value of semi-supervised learning techniques, especially when document labeling is an expensive task, and confirm high accuracy on news stories classification.

From this point on, many paths for improvement can be followed. The application of information extraction techniques may add very interesting capabilities. These techniques may generate document summaries, or summarize the content of sets of documents, which may be very valuable at the presentation layer. At a second phase, we may explore information extraction techniques to automatically build a report on the current state of the art of some topic, specifically structured according to the organization preferred by the user.

We may think of a course as a specific collection of learning objects organised on a given ontology, defined according to the course objectives. In this scenario, we may apply our methodology to automatically compile, from a repository of learning objects, the material that is required for a specific course. We may even apply our methodology to retrieve fresh content from the web and other sources to continuously update courseware based on a previously defined ontology. Students themselves might use such a tool to build a course on a specific competence; or to help them collect a set of learning objects that might help in understanding a given concept or how to apply a given technique.

Acknowledgment

This work is supported by the POSC/EIA/58367/2004/Site-o-Matic Project (Fundação Ciência e Tecnologia), FEDER e Programa de Financiamento Plurianual de Unidades de I & D.

We would like to thank the Expresso newspaper for their support throughout this work.

References

[1] Bueno, D., David, A.A. (2001), "METIORE: A Personalized Information Retrieval System", Proceedings of the 8th International Conference on User Modeling, Springer-Verlag.

[2] Chakrabarti, S. (2003), Mining the web, Discovering Knowledge from Hypertext Data, Morgan Kaufmann Publishers.

[3] Chakrabarti, S., Berg, M., Dom, B. (1999), "Focused crawling: a new approach to topic-specific resource discovery", Proceedings of the 8th World Wide Web Conference.

[4] Chakrabarti, S., Dom, B., Raghavan, P., Rajagopalan, S., Gibson, D., Kleinberg, J. (1998), "Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text", Proceedings of the 7th International World Wide Web Conference.

[5] Chen, C.C., Chen, M.C., Sun, Y. (2001), "PVA: A Self-Adaptive Personal View Agent System", Proceedings of the ACM SIGKDD 2001 Conference.

[6] Dewey, Melvil (1876), A Classification and Subject Index for Cataloguing and Arranging the Books and Pamphlets of a Library (Dewey Decimal Classification).

[7] Escudeiro, N., Jorge, A. (2006), Semi-automatic Creation and Maintenance of Web Resources with webTopic, vol. 4289 of LNCS, chapter VI, pages 82-102, Springer.

[8] Etzioni, O. (1996), "The World-Wide-Web: quagmire or gold mine?", Communications of the ACM, Vol. 39, No. 11, pp 65-68.

[9] Gospodnetic, O., Hatcher. E. (2005), Lucene in action, Manning Publications Co.

[10] Halkidi, M., Nguyen, B., Varlamis, I., Vazirgiannis, M. (2003), "Thesus: Organizing Web document collections based on link semantics", The VLDB Journal, 12, pp 320-332.

[11] Joachims, T. (1998), Text Categorization with Support Vector Machines: Learning with Many Relevant Features, Research Report of the unit no. VIII(AI), Computer Science Department of the University of Dortmund.

[12] Joachims, T. (1999), Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press.

[13] Jones, R., McCallum, A., Nigam, K., Riloff, E. (1999), "Bootstrapping for Text Learning Tasks", IJCAI-99 Workshop on Text Mining: Foundation, Techniques and Applications, pp. 52-63.

[14] Kosala, R., Blockeel, H. (2000), "Web Mining Research: A Survey", SIGKDD Explorations, Vol. 2, No. 1, pp 1-13.

[15] Lieberman, H. (1995), "Letizia: an Agent That Assists Web Browsing", Proceedings of the International Joint Conference on AI.

[16] Liu, B., Chin, C.W., Ng, H. T. (2003), "Mining Topic-Specific Concepts and Definitions on the Web", Proceedings of the World Wide Web 2003 Conference.

[17] Mladenic, D. (1999), Personal WebWatcher: design and implementation, Technical Report IJS-DP-7472, SI.

[18] Mitchell, S., Mooney, M., Mason, J., Paynter, G.W., Ruscheinski, J., Kedzierski, A., Humphreys, K. (2003), "iVia Open Source Virtual Library System", D-Lib Magazine, Vol. 9, No. 1.

[19] Ribeiro, P., Escudeiro, N. (2008), On-line News "à la carte", European Conference on the Use of Modern Information and Communication Technologies

[20] Salton, Wong, Yang (1975), A vector space model for automatic indexing. Communications of the ACM. 18:11, p. 613-620.

[21] http://en.wikipedia.org/wiki/Category:Library_cataloging_and_classification, accessed on February 2008

[22] http://expresso.clix.pt/, accessed on November 2007

[23] http://www.yahoo.com/, accessed on February 2008