Identification and Acquisition of Domain Dependent Internet Resources *

Emil Gatial[1], Zoltan Balogh[1], Ladislav Hluchy[1], and Peter Vojtek[2]
[1]Institute of Informatics, Slovak Academy of Sciences, Bratislava 845 17, Slovakia,
[2]Faculty of Informatics and Information Technologies, Slovak University of Technology, 842 16 Bratislava, Slovakia
emil.gatial@savba.sk

Abstract. *This article describes methods for identification, domain evaluation and acquisition of Internet documents. The identification of internet resources uses common web search engines API to gather possible candidates for acquisition methods. The acquisition tools implements methods of focused crawler that additionally evaluated Internet document relevancy after downloading. The proposed set of tools partially automates and eases process of acquisition of required Internet resources. The paper briefly introduces basics of used methods, followed by the detailed description of methods implemented in the concrete tools. The closing part poses available results and comparisons of tools. The tools were designed and developed within the NAZOU project.*

1 Introduction

The typical feature of Internet visible resources is the heterogeneity of its content. Even though the web pages are developed using the well formed languages like HTML, XHTML the content is intelligible mostly by human. Recently the initiatives like Semantic Web proposes description languages (RDF, OWL) processable by computers, but these techniques are still rarely used. This fact is one of the reasons to design and develop methods for automated Internet documents acquisition with partially described content. This paper proposes such set of methods used for acquisition of required Internet resources. The identification of Internet resources uses common web search engines API to gather possible candidates for acquisition methods. The acquisition tools implements methods of focused crawler that additionally evaluated Internet document relevancy after downloading. The proposed set of tools partially automates and eases process of acquisition of required Internet resources. The paper introduces basics of used methods, next the process of resource identification and acquisition is described. The closing part poses available results of tools.

2 Description of used methods

The method for identification of Internet resources (implemented in the RIDAR [1] tool) selects set of candidate URLs according to a given specific set of most pertinent keywords specific for required document domain. RIDAR uses Google and Yahoo API to query the search engine to return set of URLs. A crawling process is characterized by periodical requesting and downloading of Internet documents. In many cases only small portion of acquired documents are convenient for further processing. The method of focused crawling is used for on-line document filtering by using method of single neuron (explained later). The selection of features characterizing the set of convenient documents (in the global manner) plays the most challenging role in the process of document separation. Sometimes the features can be obviously identified (for example specific keywords, tags, etc.), but more likely the features can't be exactly identified because of the tremendous heterogeneity of Internet documents. The tasks of feature identification and document categorization fall into the domains of Information Retrieval (IR) and Artificial Intelligence (AI). The methods like decision trees [2], rule learning [3], artificial neural networks (ANN) [4] [5], K-nearest neighbor algorithm [6], support vector machines (SVM) [7], Naive Bayes methods [8]) and Markov processes [15] [16]. The most recently explored methods regards the hierarchical categorization investigated in the work of Koller and Sahami [9].

In this paper the single domain document membership categorization is explored using the NN approach. The paper put accent on the evaluation of categorization that is used during the web crawling process downloading the pages with similar content. Such approach of processing is known as the focused (sometimes known as topical) crawling is introduced by article [10].

The method used in WebCrawler tool comes out of the well defined principles of crawling process like selection polices, re-access methods, moral rules and parallelization of crawling. The most innovative feature of WebCrawler tool is dynamic URL handling which tries to filter out useless parts of dynamically generated URL and so minimize the multiple access of the same resource.

* This work is supported by projects NAZOU SPVV 1025/2004

3 Identification of Internet Resources

Identification of relevant Internet resources represents the starting point of the acquisition process. A tool is required which would acquire links to existing resources from existing search services (such as Google or Yahoo). RIDAR (Relevant Internet Data Resource Identification) tool was designed and implemented for such purposes. Since the NAZOU project focuses on domain specific processing the RIDAR tool enables identification of most appropriate domain specific resources and stores links to them in form of URLs for further processing.

3.1 RIDAR (Relevant Internet Data Resource Identification)

RIDAR utilizes search based on keywords or collocations describing particular domain in existing search engines. RIDAR tool enables exposition of web service APIs of Google, Yahoo or other available search engines. RIDAR provides interfaces for integrating any search or storage engines (MySQL implemented by default). RIDAR also enables support for search schedule definition and management. Usually licenses are required to access certain search engines. RIDAR provides mechanisms to manage multiple licenses to several search engines. Using licenses RIDAR queries relevant search engines and extract URLs which are stored in a database for further processing by the WebCrawler tool.

4 Estimation of Document Relevancy

Within the NAZOU project there are two kinds of tools possibly applicable to evaluate content domain membership. The first one, named ERID, (Estimate Relevance of Internet Document) implements method from the artificial intelligence domain and the second, named NALIT (Markov Processes based Document Categorization) implements method from Markov processes theory. Each tool has its specific pros-and-cons.

4.1 ERID (Estimate Relevancy of Internet Document)

The cornerstone of NN is artificial neuron that is an abstraction of biological neurons. It usually receives many inputs, called input vector, (analogy of dendrites) and the sum of them forms output (analogy of synapse). The sums (1) of inputs are weighted and the sum is passed to the activation function. The original artificial neuron is the Threshold Logic Unit (TLU) first proposed by McCulloch and Pitts [11]. A perceptron, specific type of artificial neuron (AN), employs a threshold (step function) taking on the values 1 or 0 only. The schema of such AN is depicted on the Fig. 1. The categorization using the ANN approach is based of the evaluation of feature frequencies. The perceptron is a linear threshold classier that separates examples with a hyperplane and it is perhaps the simplest model that can be used standalone.

$$a = \sum_{k_i \in K} N_{k_i}.w_k i \tag{1}$$

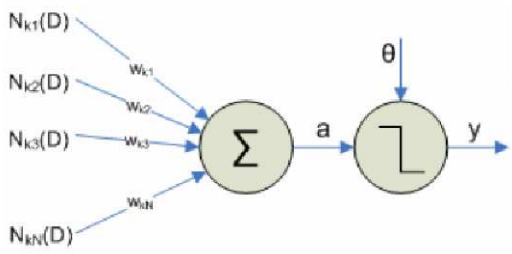

Fig. 1. Neuron abstraction using TLU.

Feature Selection The feature can be possibly represented using the different words contained by document collection. Even the small document collection can contain large amount of features therefore the reduction of feature set is necessary to train artificial neuron. The behavior of artificial neuron is sensitive to the size and quality of reduced feature set.
First steps of reduction are filtering the stop words and use of stemming algorithms. The two approaches can be applied on the resulting words for feature selection: wrapper approach and the filter approach [12]. The filter approach selects the feature set using the preprocessing methods by evaluation of correlation coefficient, mutual information and odds ratio. The wrapper approach selects initial set and measures the performance of the artificial neuron. In the paper the wrapper method is used and partially examined.
For the simplicity the initial set of features for wrapper approach is selected by supervising person. The application of combination of filter and wrapper approaches is planned for the future research.

Learning the TLU The TLU training process covers weights modification according to the preprocessed training set. The proposed algorithm is using the perceptron learning rule [13]. This kind of training is known as supervised training.
The training process repeats until the process exceeds

the specified number of iterations or the TLU is fully trained. The provided experiments showed that the training process depends not only on the size of selected features, but on the quality of features. This fact is used for identification of new features, when features can be combined (created or removed) until the TLU is fully trained.

4.2 NALIT (Markov Processes based Categorization)

The categorization method used in NALIT tool [17] is based on method proposed by Dunning [16]. At first statistical model for each category is created in a learning phase. Each of these category models is constructed from pre-selected training text documents, every document represents certain category in selected categorization.
A training text document (representative of particular category) is processed as a stream of characters. This stream is divided into Markov processes with length k characters (i.e.order of Markov process). Each unique Markov process is stored together with information about its number of occurrences. After processing the whole document, numbers of occurrences for all processes are converted into probabilities (2) (k-th order Markov processes).

$$p(w_1...w_{k+1}) = \frac{T(w_1...w_{k+1}) + 1}{T(w_1...w_k + |A|)} \tag{2}$$

where —A— is size of alphabet, $T(w_1...w_k)$ is number of occurrences of prefix of the Markov process, $T(w_1...w_{k+1})$ is the numer of occurrences of the whole Markov process and $p(w_1...w_{k+1})$ is the computed probability. Then categorization phase can be proceeded, documents to be categorized are passed and categories are assigned to them. The best-fitting category for each document is determined by an evaluation function (3).

$$logp = \sum_{w_1...w_{k+1} \in S} T(w_1...w_{k+1}).logp(w_{k+1}|w_1...w_k) \tag{3}$$

where $T(w_1...w_{k+1})$ are Markov processes present in the text and $p(w_{k+1}|w_1...w_k)$ are probabilities stored in a particular model for each Markov process. The result is probability p, logarithm scaling is used due to avoiding problems of numeric underflow.

4.3 Integration of focused crawler

The crawler periodically passes the downloaded data to the HTML parser (generally, ANN approach can be applied to any document that can be converted to textual document) that extracts text. The frequencies of features are processed from the text. Such frequencies represent the input vector N_k and its weights w_k. According to this vector the output is evaluated and the web crawler decides if the page will be stored or not.

5 Testing and Validation

Following tests are performed on the web pages written in English and Slovak language. The aim of provided tests has to prove that ANN approach is suitable method for focused web crawling.
The two tests were realized. The test No. 1 provides reference measurement that evaluates the performance of TLU trained on the training set that is a subset of evaluated HTML pages. The test No. 2 provides evaluation of TLU performance trained on the training set that is subset only of the part of HTML pages. The distinction between the two tests lies in the different training set. While test No. 1 provides only reference evaluation according to the subset of all categorized document, the test No. 2 is made with the TLU trained with the training set that is disjunctive with to evaluating training set. The difference of the tests should show the correlation of behavior of TLU trained on the conjunctive and disjunctive training sets. In the practice this means the degradation rate of TLU performance when the categorization is used on the set that originates from different web site.
In order to evaluate a decision results we first define a contingency matrix representing the possible outcomes of the classification as shown in Table 1. Several measures and the corresponding names used in the IR community are specified in the Table 2.
Many applications require the recall parameter closing to the value of 1 that means the algorithm refuses the minimal number of web pages that should be accepted (False Negatives).

	Class Positive (C+)	Class Negative (C-)
Assigned Positive (A+)	True Positives a	False Positives b
Assigned Negative (A-)	False Negatives c	True Negatives d

Table 1. Contingency table for binary classification

AI	formula
recall	$\frac{a}{a+c}$
precision	$\frac{a}{a+b}$
fallout	$\frac{a}{a+d}$
error rate	$\frac{b+c}{a+b+c+d}$
F1 measure	$\frac{2a}{2a+b+c}$

Table 2. Efficiency measures for binary classification defined in the Artificial Intelligence community

5.1 Evaluation of Training Set in Slovak Language

Training set for the test No. 1 consists of the web pages from the following web sites: *www.ejobs.sk*, *www.praca.sk*, *www.ponuky.sk*, *www.profesia.sk*. The test No. 2 is trained on the training set consists of first three sites. Evaluation is processed on the site *www.profesia.sk*.

	Class Positive (C+)	Class Negative (C-)
Assigned Positive (A+)	124	4
Assigned Negative (A-)	1	121

Table 3. Contingency table for test No.1

Ef. variable	Value
recall	0.99
precision	0.96
fallout	0.50
error rate	0.02
F1 measure	0.98

Table 4. Efficiency measures for test No.1

	Class Positive (C+)	Class Negative (C-)
Assigned Positive (A+)	120	4
Assigned Negative (A-)	5	123

Table 5. Contingency table for test No.2

Ef. variable	Value
recall	0.96
precision	0.98
fallout	0.49
error rate	0.03
F1 measure	0.97

Table 6. Efficiency measures for test No.2

	Class Positive (C+)	Class Negative (C-)
Assigned Positive (A+)	124	12
Assigned Negative (A-)	1	113

Table 7. Contingency table for test No.1

Ef. variable	Value
recall	0.99
precision	0.91
fallout	0.52
error rate	0.05
F1 measure	0.95

Table 8. Efficiency measures for test No.1

	Class Positive (C+)	Class Negative (C-)
Assigned Positive (A+)	119	17
Assigned Negative (A-)	6	108

Table 9. Contingency table for test No.2

5.2 Evaluation of Training Set in English Language

Training set for the test No. 1 consists of the web pages from the following web sites: *www.jobbankusa.com*, *www.jobserve.com*, *www.myjobsearch.com*. The test No.2 is trained on the training set consists of first three sites. Evaluation is processed on the site *www.jobbankusa.com*. Performance of NALIT tool was measured using one profile representing job offers. Training set comprises of job offers in Slovak language from portals *www.profesia.sk*

Ef. variable	Value
recall	0.95
precision	0.87
fallout	0.52
error rate	0.09
F1 measure	0.91

Table 10. Efficiency measures for test No.2

and *www.praca.sk* in the ratio 15:85. Testing was performed on the documents from web sites *www.ejobs.sk*, *www.profesia.sk* and *www.praca.sk*. All pages were manually categorized into groups containing job offer and not containing job offer. In the next step the plain text was extracted to perform tests. The highest success-

Markov process class			I	II	III	IV
Portal	Contains job offer	No. documents	Recall [%]			
ejobs.sk	yes	36	100.00	97.20	94.40	94.40
	no	35	94.20	94.20	100.00	88.50
praca.sk	yes	129	42.60	76.70	82.90	86.00
	no	32	71.80	62.50	78.10	81.20
profesia.sk	yes	249	47.30	84.30	84.70	83.90
	no	250	52.60	75.60	70.40	71.60
			F1 [%]			
			55.30	82.50	83.10	83.40

Table 11. Categorization results

fulness of NALIT tool to identify category of page (if page contains job offer or not) can be achieved using the profile with fourth class Markov model, where it is possible to achieve combined rate of F1 = 83.40%. This rate was growing along with the Markov model class, but experimenting with higher Markov model classes wasnt possible due to computing limitations. The crawling data (presented in Table 11) was collected by running single process of WebCrawler downloading 5 thousands pages from site *www.profesia.sk* (containing all pages regardless of job offer presence). The following table shows basic statistical results of crawling process. Note that the results provides only general performance overview because the crawling process had run in the multiprocessing environment (other running processes can interfere the measured process) where some values can easily exceed over the real time (this is the reason of the big deviation values mostly recognizable in evaluation and processing times).

	download	evaluation	processing
average [ms]	436	82	17
deviation [ms]	330	110	70
overall [ms]	698598	132412	27291

Table 12. Statistical evaluation of WebCrawler process performance

6 Conclusion

As focused crawling method applied during the acquisition of Internet resources containing domain specific information couples several methods that analyzes processed Internet document, the performance of each component affects overall system performance.
The artificial neural networks and Markov processes can be used in broad field of applications. One of the usages is shown on the web page categorization application for the focused crawling process. The ANN offers two very important properties for categorization; the fast input vector processing and learning possibility that allows dynamic categorization behavior. The Markov model provides good performance as well and therefore described two categorization methods are usable in proposed system. Realized experiments showed that the quality of the selected feature is very important for learning and categorization process. Quality of feature usually reflects the correlation to specific domain of categorization.
The performance of crawling process can be enhanced by parallelization of downloading process. The evaluation results of document relevance should be improved by using combination of both categorization methods, although the testing and tuning of such system should be much more complicated. First categorization method employs only single TLU for categorization, but ANN can be formed into complex networks that can provide much better results by creating the hierarchical categorization [14] of Internet documents.

References

1. Balogh, Z.: RIDAR - relevant Internet data resource identification. Laclavik M. et al.: WIKT 2006 Proceedings, 1st Workshop on Intelligent and Knowledge-oriented Technologies, ISBN 978-80-969202-5-9, Bratislava, (2007) pp. 122.
2. Moulinier, I., Ganascia, J., G.: Applying an existing machine learning algorithm to text categorization. In: S Wermer, E Riloff and G Scheler Eds., Connectionist, Statistical, and Symbolic Approaches to Learning for Natural Language Processing, Springer Verlag, Heidelberg, Germany, (1996) pp. 343-354.
3. Apte, C., Damerau, F., Weiss, S.,M.: Automated learning of decision rules for text categorization. ACM Transactions on Information Systems. (1994) pp. 3(12):233-251.
4. Ng, H.,T., Goh, W.,B., Low, K.,L.: Feature selection, perceptron learning, and a usability case study for text categorization. In: Belkin Nicholas, Desai NarasimhaluA and Willett Peter Eds., Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Philadelphia, PA, (1997) pp. 67-73.

5. Wiener, E., Pedersen, J.,O., Weigend, A.,S.: A neural network approach to topic spotting. In: Proceedings of SDAIR'95, (1995) pp. 317-332.
6. Yang, Y., Pedersen, J.,O.: A comparative study on feature selection in text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997). Morgan Kaufmann Publishers, San Francisco, CA, (1997).
7. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. Technical Report LS-8 Report 23, University of Dortmund, (1997).
8. McCallum, A., Nigam, K.: A comparison of event models for naive Bayes text classification. In: Learning for Text Categorization: Papers from the 1998 Workshop. AAAI Technical Report WS-98-05. AAAI Press, San Francisco, CA, (1998) pp. 41-48.
9. Koller, D., Sahami, M.: Hierarchically classifying documents using very few words. In: ICML 1997: Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco, CA, (1997) pp. 170-178.
10. Chakrabarti S., M. Van Den Berg, Dom, B.: Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery. In: Proceedings of the 8th International WWW Conference, Toronto, Canada, (1999) http://www8.org/w8-papers/5a-search-query/crawling/index.html
11. McCulloch, W., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics, (1943) pp. 7:115 - 133.
12. John, G., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: Machine Learning: Proceedings of the Eleventh International Conference, Morgan Kaufman Publishers, San Francisco, CA, (1994) pp. 121-129.
13. Gallant, S., I.: Perceptron-based learning algorithms. IEEE Transactions on Neural Networks, 1, (1990) pp. 179-191.
14. Ruiz, M., E., Srinivasan, P.: Hierarchical Text Categorization Using Neural Networks. In: Information Retrieval 5, Kluwer Academic Publishers, (2002) pp. 87-118.
15. Teahan, W., J.: Text classification and segmentation using minimum cross entropy. In: RIAO-00, 6th International Conference Recherche d'Information Assistee par Ordinateur, Paris, (2000).
16. Dunning, T.: Statistical identification of language. Techical Report MCCS-94-273, Computing Research Lab (CRL), New Mexico State University, (1994).
17. Vojtek, P., Bielikov, M.: Comparing Natural Language Identification Methods based on Markov Processes. In: Slovko - International Seminar on Computer Treatment of Slavic and East European Languages, Bratislava, (2007).