Advances in Soft Computing

Editor-in-chief
Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw, Poland
E-mail: kacprzyk@ibspan.waw.pl

Further books of this series can be found on our homepage: springeronline.com

Peter Sinčák, Ján Vaščák, Vladimír Kvasnička
and Radko Mesiar (Eds.)
The State of the Art in Computational Intelligence
2000. ISBN 3-7908-1322-2

Bernd Reusch and Karl-Heinz Temme (Eds.)
Computational Intelligence in Theory and Practice
2000. ISBN 3-7908-1357-5

Rainer Hampel, Michael Wagenknecht,
Nasredin Chaker (Eds.)
Fuzzy Control
2000. ISBN 3-7908-1327-3

Henrik Larsen, Janusz Kacprzyk,
Sławomir Zadrozny, Troels Andreasen,
Henning Christiansen (Eds.)
Flexible Query Answering Systems
2000. ISBN 3-7908-1347-8

Robert John and Ralph Birkenhead (Eds.)
Developments in Soft Computing
2001. ISBN 3-7908-1361-3

Mieczysław Kłopotek, Maciej Michalewicz
and Sławomir T. Wierzchoń (Eds.)
Intelligent Information Systems 2001
2001. ISBN 3-7908-1407-5

Antonio Di Nola and Giangiacomo Gerla (Eds.)
Lectures on Soft Computing and Fuzzy Logic
2001. ISBN 3-7908-1396-6

Tadeusz Trzaskalik and Jerzy Michnik (Eds.)
Multiple Objective and Goal Programming
2002. ISBN 3-7908-1409-1

James J. Buckley and Esfandiar Eslami
An Introduction to Fuzzy Logic and Fuzzy Sets
2002. ISBN 3-7908-1447-4

Ajith Abraham and Mario Köppen (Eds.)
Hybrid Information Systems
2002. ISBN 3-7908-1480-6

Przemysław Grzegorzewski, Olgierd Hryniewicz,
Maria A. Gil (Eds.)
*Soft Methods in Probability, Statistics
and Data Analysis*
2002. ISBN 3-7908-1526-8

Lech Polkowski
Rough Sets
2002. ISBN 3-7908-1510-1

Mieczysław Kłopotek, Maciej Michalewicz
and Sławomir T. Wierzchoń (Eds.)
Intelligent Information Systems 2002
2002. ISBN 3-7908-1509-8

Andrea Bonarini, Francesco Masulli
and Gabriella Pasi (Eds.)
Soft Computing Applications
2002. ISBN 3-7908-1544-6

Leszek Rutkowski, Janusz Kacprzyk (Eds.)
Neural Networks and Soft Computing
2003. ISBN 3-7908-0005-8

Jürgen Franke, Gholamreza Nakhaeizadeh,
Ingrid Renz (Eds.)
Text Mining
2003. ISBN 3-7908-0041-4

Tetsuzo Tanino, Tamaki Tanaka,
Masahiro Inuiguchi
*Multi-Objective Programming and Goal
Programming*
2003. ISBN 3-540-00653-2

Mieczysław Kłopotek, Sławomir T. Wierzchoń,
Krzysztof Trojanowski (Eds.)
Intelligent Information Processing and Web Mining
2003. ISBN 3-540-00843-8

Ahmad Lotfi, Jonathan M. Garibaldi (Eds.)
Applications and Science in Soft-Computing
2004. ISBN 3-540-40856-8

Mieczysław Kłopotek, Sławomir T. Wierzchoń,
Krzysztof Trojanowski (Eds.)
*Intellligent Information Processing and
Web Mining*
2004. ISBN 3-540-21331-7

Mieczysław A. Kłopotek
Sławomir T. Wierzchoń
Krzysztof Trojanowski (Eds.)

Intelligent Information Processing and Web Mining

Proceedings
of the International IIS: IIPWM'04 Conference
held in Zakopane, Poland, May 17-20, 2004

With 152 Figures
and 65 Tables

Springer

Prof. Dr. Mieczysław A. Kłopotek
Prof. Dr. Sławomir T. Wierzchoń
Dr. Krzysztof Trojanowski

Polish Academy of Sciences
Institute of Computer Sciences
ul. Ordona 21
01-237 Warszawa
Poland

Library of Congress Control Number: 2004102409

ISSN 16-15-3871
ISBN 3-540-21331-7 Springer-Verlag Berlin Heidelberg NewYork

Springer-Verlag is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Cover design: Erich Kirchner, Heidelberg
Typesetting: Digital data supplied by the editors

Printed on acid-free paper 62/3020Rw-5 4 3 2 1 0

Preface

This volume contains articles accepted for presentation during The Inte
ligent Information Processing and Web Mining Conference IIS:IIPWM'(
which was held in Zakopane, Poland, on May 17-20, 2004. This conference
a continuation of a series of 13 successful symposia and conferrences on Inte
ligent Information Systems, organized by the Institute of Computer Scien
of Polish Academy of Sciences, devoted to new trends in (broadly understoo
Artificial Intelligence.

The idea of organizing such meetings dates back to 1992. Our main i
tention guided the first, rather small-audience, workshop in the series was t
resume the results gained in Polish scientific centers as well as contrast the
with the research performed by Polish scientists working at the universiti
in Europe and USA and their foreign collaborators. This idea proved to b
attractive enough that we decided to continue such meetings. As the yea
went by, the workshops has transformed into regular symposia devoted t
such fields like Machine Learning, Knowledge Discovery, Natural Langua
Processing, Knowledge Based Systems and Reasoning, and Soft Computin
(i.e. Fuzzy and Rough Sets, Bayesian Networks, Neural Networks and Evo
lutionary Algorithms). At present, about 50 papers prepared by researche
from Poland and other countries are usually presented.

This year conference devotes much more attention to the newest devel
opments in the area of Artificial Intelligence, related to broadly understoo
Web mining. In connection with these and related issues, contributions wer
accepted, concerning:

- recommenders and text classifiers
- natural language processing and understanding for search engines and other web applications
- computational linguistics
- information extraction and web mining
- data mining, machine learning and knowledge discovery technologies item knowledge representation
- web services and ontologies
- logics for artificial intelligence
- foundations of data mining
- medical and other applications of data mining

The above-mentioned topics were partially due to invited sessions organized by Erhard W. Hinrichs, Zbigniew W. Raś, Roman Świniarski, Henryk Rybiński, and Ryszard Tadeusiewicz.

Out of an immense flow of submissions, the Program Committee has selected only about 40 full papers for presentation and about a dozen of posters, constituting about 55% of the total number of submitted contributions.

- Alicja Wakulicz-Deja (University of Silesia, Poland)
- Jan Węglarz (Poznan University of Technology, Poland)
- Stefan Węgrzyn (Polish Academy of Sciences, Poland)
- Krzysztof Zieliński (University of Mining and Metallurgy, Poland)
- Djamel A. Zighed (Lumière Lyon 2 University, France)
- Jana Zvarová (EuroMISE Centre, Czech Republic)

We would like also to thank additional reviewers

- Witold Abramowicz (Poznań University of Economics, Poland)
- Jerzy Cytowski (Warsaw University, Poland)
- Andrzej Czyżewski (Gdańsk University of Technology, Poland)
- Jirka Hana (The Ohio State University, USA)
- Bożena Kostek (Gdańsk University of Technology, Poland)
- Geert-Jan Kruijff (Universität des Saarlandes, Germany)
- Anna Kupść (Polish Academy of Sciences, Poland)
- Małgorzata Marciniak (Polish Academy of Sciences, Poland)
- Agnieszka Mykowiecka (Polish Academy of Sciences, Poland)
- Zygmunt Vetulani (Adam Mickiewicz University, Poland)

Table of contents

Part I. Regular Sessions: Machine Learning, Machine Discovery and Data Mining

Part II. Regular Sessions: Biologically Motivated Algorithms and Systems

Part III. Regular Sessions: Natural Language Processing

Part IV. Regular Sessions: Web Mining, Web Technologies and Information Retrieval

Part V. Poster Session

Part VI. Invited Session: Machine Learning Techniques for Computational Linguistics

Part VII. Invited Session: Knowledge Discovery

5. Groenendijk J., Stokhof M. Dynamic Predicate Logic, Linguistics and Philosophy 14, pp. 39-100, 1991
6. Hausser R., Foundations of Computational Linguistics, Univeristaet Erlangen Nuernberg, Germany, 1999
7. Huzar Z., Labuzek M., A tool assisting creation of business models, Foundations of Computing and Decision Science, Vol. 27. - No. 4 - 2002, Institute of Computing Science, Poznan Technical University, 2002
8. Minsky M. A Framework for Representing Knowledge, in The Psychology of Computer Vision, ed. WinstonP. H., Mc Craw-Hill Computer Science Series, 1975
9. Meta Object Facility (MOF) Specification, available at www.omg.org
10. Montague R. The Proper Treatment of Quantification in Ordinary English in: Proc. of the 1970 Stanford Workshop on Grammar and Semantics. ed. J. Hintikka et al., D. Reidel Publishing Company, 1973.
11. Tarski A. The Semantic Concept of Truth in: Philosophy and Phenomenological Research 4: 341-375, 1944.

Text Categorization using the Learning Vector Quantization Algorithm

M.Teresa Martín-Valdivia[1], Manuel García-Vega[2], Miguel A. García-Cumbreras[2], and L. Alfonso Ureña López[2]

[1] Departamento de Informática, Jaén University,
Campus Las Lagunillas S/N, Jaén 23071, Spain
[2] Departamento de Informática, Jaén University,
Av. Madrid, 31, Jaén 23071, Spain

Abstract. Text Categorization (TC) consists of assigning predefined categories based on the content of natural language texts. In this paper, we present a new approach that uses the Learning Vector Quantization (LVQ) algorithm to automatically categorize the Reuters-21578 test collection according to its content. The LVQ algorithm is a competitive neural learning method based on the supervised Kohonen model. We have carried out the experiments on one of most popular Reuters partitions (ModApte Split). The results obtained are very promising and encourage us to continue working in this line.

1 Introduction

Nowadays the amount of publicly available information on the web is increasing rapidly every day and automation of categorization and filtering of documents has become an essential procedure. Text Categorization (TC) is a important task for many Natural Language Processing (NLP) applications. Given a set of documents and a set of categories, the goal of a categorization system is to assign to each document a set (possibly empty) of categories that the document belongs to.

TC requires the use of a collection of documents labelled with categories. TC systems based on learning approaches need the division of the collection in a training collection and a test collection. There are several works that use learning approaches to TC tasks including regression [16], Bayesian models [2], decision tree [7] etc. In recent years, several researchers have used neural network approaches to TC. A neural network is a statistical information model that uses learning to adjust the model.

In fact, neural architectures are successfully used by Wiener et al. [15] and by Ng et al. [10]. Both works train one neural network per category. However, Wiener et al. use a multilayer perceptron (MLP) with one hidden layer and Ng et al. use a simple perceptron (without hidden layer).

Yang and Liu [17] also use a MLP but they train one single neural network on all the categories used. This model uses much less time that the above approaches.

Recently, Kohonen et al. [5] describe the implementation of a system that is able to organize vast document collections according to textual similarities. It is based on the Self-Organizing Map (SOM) algorithm. The developed system is known as WEBSOM and the results obtained are very promising.

Martín et al. [9] train one neural network based on the Kohonen model [4] to categorize a multilingual corpus: the polyglot Bible [11]. They use the Learning Vector Quantization (LVQ) algorithm, which is the supervised version of Kohonen's SOM.

In this work, we propose the use of the LVQ algorithm to train one neural network that learns the categories of the Reuters-21578 test collection[1]. Reuters is a linguistic resource widely used in TC [6] with the purpose of proving the effectiveness of TC systems.

We need represent the document collection in an appropriate way in order to use the LVQ for TC. There are several models that have been used as information representation models, such as the Vector Space Model (VSM), Probabilistic model and Boolean model. In this work we will use the VSM, which is considered an effective model in the Information Retrieval (IR) community [1].

The paper is organized as follows. First, we describe the LVQ algorithm. Next, we present the information representation model based on the VSM and the evaluation measures used. After this, we show our evaluation environment and results. Finally, we present our conclusion and lines of future work.

2 The LVQ Algorithm

This work proposes the use of competitive neural learning based on the Kohonen model [4] to accomplish the TC task: the Learning Vector Quantization (LVQ) algorithm. The LVQ algorithm has been successfully used in several applications [4], such as pattern recognition, speech analysis, etc. However, there are few studies that use LVQ for TC.

The LVQ algorithm is a classification method, which allows the definition of a group of categories on the space of input data by reinforced learning, either positive (reward) or negative (punishment). LVQ uses supervised learning to define class regions on the input data space. For this, a subset of similarly labelled codebook vectors is placed into each class region.

The basic LVQ algorithm is quite simple. It starts with a set of input vectors x_i and weights vectors w_k which represent the classes to learn. In each iteration, an input vector x_i is selected and the vectors w_k are updated, so that they fit x_i better. The LVQ algorithm works as follows:

For each class, k, a weight vector w_k is associated. In each repetition, the algorithm selects an input vector, x_i, and compares it with every weight

[1] The Reuters-21578 text categorization test collection is available at http://www.daviddlewis.com/resources/testcollections/reuters21578/, thanks to Reuters, Carnegie Group, and David Lewis.

vector, w_k, using the Euclidean distance $\|x_i - w_k\|$, so that the winner will be the codebook vectors w_c closest to x_i in the input space for this distance function. The determination of c is achieved by following decision process:

$$\|x_i - w_c\| = \min_k \|x_i - w_k\| \tag{1}$$

i.e.,

$$c = \arg\min_k \|x_i - w_k\| \tag{2}$$

The classes compete between themselves in order to find which is most similar to the input vector, so that the winner is the one with the smallest Euclidean distance with regard to the input vector. Only the winner class will modify its weights using a reinforced learning algorithm, either positive or negative, depending on the classification being correct or not. Thus, if the winner class and the input vector have the same class (the classification has been correct), it will increase the weights, coming slightly closer to the input vector. On the contrary, if the winner class is different from the input vector class (the classification has not been correct), it will decrease the weights, moving slightly further from the input vector.

Let $x_i(t)$ be an input vector at time t, and $w_k(t)$ represent the weight vector for the class k at time t. The following equations define the basic learning process for the LVQ algorithm:

$$\begin{aligned} w_c(t+1) &= w_c(t) + \alpha(t)[x_i(t) - w_c(t)] \\ &\qquad \text{if } x_i \text{ and } w_c \text{ belong to the same class} \\ w_c(t+1) &= w_c(t) - \alpha(t)[x_i(t) - w_c(t)] \\ &\qquad \text{if } x_i \text{ and } w_c \text{ belong to different class} \\ w_k(t+1) &= w_k(t) \text{ if } k \neq c \end{aligned} \tag{3}$$

where $\alpha(t)$ is the learning rate, which decreases with the number of iterations of training ($0 < \alpha(t) << 1$). It is recommended that $\alpha(t)$ be rather small initially, say, smaller than 0.3, and that it decrease to a given threshold, ν, very close to 0 [4]. In our experiments, we have initialized $\alpha(t)$ to 0.1.

3 Information Representation Model

In order to represent appropriately the document collection, we have decided to use the VSM as information representation model.

3.1 Vector Space Model

The VSM [1] was originally development for IR community, but it can be used in other NLP tasks such as TC or Word Sense Disambiguation (WSD) [8].

The VSM represents a document by a weighted vector of terms. A weight assigned to a term represents the relative importance of that term. One common approach for term weighting uses the frequency of occurrence of a particular word in the document to represent the vector components [12]. In order to calculate the term weights, we have used the standard $tf \times idf$ equation, where tf is the frequency of the term in the document, and idf is the inverse document frequency defined as:

$$idf = \log_2 \left(\frac{M}{df} \right) \tag{4}$$

where df_i (document frequency) is the number of documents in the collection in which the term occurs, and M is the total number of documents.

Thus, the weight w_{ij} is calculated by the following equation:

$$w_{ij} = tf_{ij} \times idf_i \tag{5}$$

where tf_{ij} (term frequency) is the number of occurrences of term i in document j.

Categories are also represented by term weight vectors. The similarity between documents and categories is computed by the cosine of the angle of their vectors. Thus, the similarity between document j and category k is obtained with the following equation:

$$sim(d_j, c_k) = \frac{\sum_{i=1}^{N} w_{ij} \cdot c_{ik}}{\sqrt{\sum_{i=1}^{N} w_{ij}^2 \cdot \sum_{i=1}^{N} c_{ik}^2}} \tag{6}$$

where N is the number of terms in whole collection, w_{ij} is the weight of term i in document j and c_{ik} is the weight of term i in category k.

3.2 Evaluation Measures

The effectiveness of a classifier can be evaluated with several measures [13]. The classical "Precision" and "Recall" for IR are adapted to the case of TC. To that end, contingency table for each category should be generated (Table 1), and then the precision and recall for each category are calculated following equations 7 and 8.

Table 1. Contingency Table for i Category

	YES is correct	NO is correct
YES is assigned	a_i	b_i
NO is assigned	c_i	d_i

$$P_i = \frac{a_i}{a_i + b_i} \tag{7}$$

$$R_i = \frac{a_i}{a_i + c_i} \tag{8}$$

In order to measure globally the average performance of a classifier, two measures can be used: micro-averaging precision P_μ and macro-averaging precision P_{macro}.

$$P_\mu = \frac{\sum_{i=1}^{K} a_i}{\sum_{i=1}^{K} (a_i + c_i)} \tag{9}$$

$$P_{macro} = \frac{\sum_{i=1}^{K} P_i}{K} \tag{10}$$

where K is the number of categories.

4 Experiments

When we evaluate a TC system, it is necessary to divide the data collection into two subsets: a training set and a test set. The collection contains a set of documents and, for each document, a specification of which categories that document belongs to. There are currently several TC linguistic resources, from which a training subset and a test subset can be obtained, such as the huge TREC collection [14], OHSUMED [3] and Reuters-21578 [6]. In our experiments we have selected Reuters because it has been used in many other studies, facilitating the comparison of results [18].

4.1 Reuters-21578 Test Collection

The Reuters-21578 test collection consists of 21,578 newswire stories about financial categories collected during 1987 from Reuters. For each document, a human indexer decided which categories from which sets that document belonged to. There are 135 different categories, which are overlapping and non-exhaustive, and there are relationships among the categories. Figure 1 shows a document from Reuters-21578 with TOPICS categories "crude" and "nat-gas".

The Reuters collection can be divided into various training and test subsets. One of the most popular partitions is the ModApte Split. Our experiments have been carried out with this division.

```
<REUTERS TOPICS="YES" LEWISSPLIT="TRAIN" CGISPLIT="TRAINING-SET"
OLDID="18425" NEWID="2007"> <DATE> 5-MAR-1987 09:24:40.64</DATE>
<TOPICS><D>crude</D><D>nat-gas</D></TOPICS>
<PLACES><D>canada</D></PLACES> <PEOPLE></PEOPLE> <ORGS></ORGS>
<EXCHANGES></EXCHANGES> <COMPANIES></COMPANIES> <UNKNOWN>
&#5;&#5;&#5;E F Y &#22;&#22;&#1;f0025&#31;reuter f
BC-orbit-oil-increases   03-05 0094</UNKNOWN> <TEXT>&#2;
<TITLE>ORBIT INCREASES OIL AND GAS RESERVE VALUES</TITLE>
<DATELINE>    CALGARY, Alberta, March 5 - </DATELINE>
<BODY>&lt;Orbit Oil and Gas Ltd> said the value of its oil and gas
reserves increased by 19 pct to 52.6 mln dlrs from 44.2 mln dlrs
reported at year-end 1985, according to an independent appraisal.
Orbit said it has reserves of 2.4 mln barrels of oil and natural
gas liquids and 67.2 billion cubic feet of natural gas. In
addition, 75 pct owned &lt;Sienna Resources Ltd> has Canadian
reserves of 173,000 barrels of oil and 1.6 bcf of natural gas with
a current value of 2.2 mln dlrs, Orbit said.
Reuter&#3;</BODY></TEXT> </REUTERS>
```

Fig. 1. Document number 2,007 from Reuters-21578

4.2 Results

The Reuters-21578 collection has been pre-processed as usual, removing common words with the SMART[2] stoplist and extracting the word stems using the Porter algorithm [1].

Table 2. Contingency Table for "earn" Category

	YES is correct	NO is correct
YES is assigned	1,039	93
NO is assigned	44	1,398

Table 3. Contingency Table for "coffee" Category

	YES is correct	NO is correct
YES is assigned	10	92
NO is assigned	0	2,472

[2] SMART is one of the most well-known IR experimental systems of public domain at ftp://ftp.cs.cornell.edu/pub/smart

In our experiments, The P_{macro} and the P_{μ} obtained is 0.48 and 0.73, respectively. The best result is obtained for the "earn" category and the worse result is obtained for the "coffee" category. Table 2 and Table 3 show the contingency table for these two case. The precision for the "earn" category is 0.92 with recall 0.96. However, we obtain a precision of 0.10 and recall 1 for the worse case ("coffee" category).

5 Conclusions

This paper uses the LVQ algorithm to train a neural network that learns to categorize text documents. We prove the effectiveness of the classifier on the well-known Reuters-21578 test collection.

The results obtained are very promising and show that our new approach based on supervised Kohonen model is very successful in automatic TC. The LVQ algorithm can be a good alternative to other TC methods although is necessary to continue working in this line in order to improve the results.

The integration of lexical resources (such as Machine Readable Dictionaries or Lexical Databases) with the LVQ algorithm will focus our future work.

6 Acknowledgements

This work has been supported by Spanish Government (MCYT) with grant FIT-150500-2003-412.

References

1. Baeza-Yates, Ribiero-Neto (1999) Modern Information Retrieval. Addison-Wesley.
2. Gómez, J.M., Buenaga, M., Ureña, L.A., Martín, M.T., García, M. (2002) Integrating Lexical Knowledge in Learning Based Text Categorization. In Proceedings of the 6th International Conference on the Statistical Analysis of Textual Data, France
3. Hersh, W., Buckley, C., Leone, T.J., Hickman, D. (1994) OHSUMED: an Interactive Retrieval Evaluation and New Large Test Collection for Research. In Proceedings of the ACM SIGIR
4. Kohonen, T. (1995) Self-organization and associative memory, 2nd edition, Springer-Verlag, Berlin
5. Kohonen, T., Kaski, S. et al. (2000) Self Organization of Massive Document Collection, IEEE Trans. on Neural Networks, **11**
6. Lewis, D.D. (1992) Representation and Learning in Information Retrieval. PhD thesis, Department of Computer and Information Science, University of Massachusetts