A COMPARATIVE STUDY ON ARABIC TEXT CLASSIFICATION

Alaa EL-Halees

Department of Computer Science, Islamic University of Gaza

P.O.Box 108 Gaza, Palestine

alhalees@iugaza.edu.ps

Abstract: *This paper focuses on Automatic Arabic classifications. Arabic language is highly inflectional and derivational language which makes text mining a complex task. In classifying Arabic text, there are many published experimental results. Since these results came from different datasets, authors and evaluation metrics, we cannot compare the performance of the experimented classifiers. In this paper, we compared six well known classifiers, which are: Maximum entropy, Naïve Bayes, Decision Tree, Artificial Neural Networks, Support Vector Machine ,and k-Nearest Neighbor using the same data sets and the same experimental settings. The recall , precision and f-measure for the classifiers are computed and compared. Then, the comparison has been done after applying feature selection on Arabic datasest.*

Keywords: Text Data Mining, Arabic Classification, Machine Language Techniques.

1.0 Introduction

Automatic text classification (which also known as text categorization or topic spotting) is the task of assigning a label to a document based on number of predefined categories. Each category is associated with a topic as Arts, Sports or Science. The goal of classification is to automatically organize and classify documents. Tex classification has been used in many applications such as spam filtering [1], "Yahoo!-like" hierarchical [2], Document indexing [3] word sense disambiguation [4], Web filtering [5] and many other applications. In the past, researchers used to manually defined a set of logical rules to classify documents to a certain category. Nowadays, machine learning methods are used to classify the documents. In the task pre-defined category labels are assigned to documents based on a tainting set of labeled documents [6]. They use many kinds of machine learning methods such as Decision Trees, Neural Nets, k-Nearest Neighbor, Support Vector Machines, Maximum Entropy and others.

There are many researches in classifying English documents (i.e. [7] have a survey). But in Arabic there are few. Classification of Arabic documents is more complex than English, because Arabic language is a highly inflectional and derivational language which makes monophonical analysis a very complex task [8 , 9]. Also, in Arabic scripts some of the vowels are represented by diacritics

which usually left out in the text which creates ambiguity in that text. In addition, Arabic scripts do not use capitalization for proper nouns. Proper nouns is necessary in classification .

In classifying Arabic Documents many algorithms are used such as k-nearest neighbor [9], Naïve Bayes [10], N-Gram Frequency Statistics [11], Maximum Entropy [12] and others. But, it hard to compare these classifiers because each research used different datasets for training and testing. As stated by [13] when he talked about text classifiers, "We have to bear in mind that comparisons are reliable only when based on experiments performed by the same author under carefully controlled conditions". This paper used the same settings to compare six well known classifiers. It compared the performance of the classifiers using the same datasets, in training and testing, the same evaluation metrics and feature selection method. It compared the following classifiers: Maximum Entropy (ME), Support Vector Machine (SVM) , k-Nearest Neighbor (kNN), decision tree (DT), Naive Bayes (NB) and Artificial Neural Nets (ANN).

The rest of the paper is organized as follows: Section 2 summarized related works in comparing text classification methods. Section 3 gave a general theoretical description on the six machine learning classifiers we used in this paper. Section 4 gave a description on the experiments' setting. Section 5 reported our experiments of the classifiers and compared the results. Finally we closed this paper with a summary and an outlook for future work.

2.0 Related Work

Our goal is to compare different methods that classify Arabic Documents. In classifying English text there are many works in this area. For example [14] gave the results of three text 0classification algorithms which described in a multi-class categorization setting. The classifiers they compared are multi-nominal Naive Bayes, partial Decision Trees and Support Vector Machine. Also, [15] compared, with significant test, five methods which are: Support Vector Machine, k-Nearest Neighbors , Neural Nets, Liner Least-sequare Fit mapping and Naïve Bayes. In addition, [16] compared five learning methods wh0ich are: Decision Trees, Naïve Bayes, Bayes Nets Support Vector Machines.

In comparing methods that classify Arabic documents, [9] compared the performance of k-nearest neighbor and Rocchio classifiers. They found that Rocchio classifier has the advantage over k-nearest neighbor. Also, [17] used Vector Space Model and Naïve Bayesian to classify Arabic documents. They compared four techniques (inner product, cosine, Jaccard, Dice) of the vector space model. They found that Naïve Bayesian is the first best classifier, the second is vector space model with Cosine technique.

3.0 Text Classification

Text classification is defined as follows: Given as set of training Documents $D = \{d_1, \ldots, d_n\}$ and set of pre-defined categories $C = \{c_1, \ldots, c_m\}$, the classifier assigns a Boolean value to each pair $\langle d_j, c_i \rangle \in D \times C$. If the value is true then document d_j is assigned a category c_i [13]. Using this definition, this study compared the following classifiers:

3.1 Decision tree

A decision tree text classifier is a machine learning approach consists of a tree in which internal nodes are labeled by words, branches departing from them are labeled by tests on the weight that the words has in the representation of the test document, and leaf nodes are labeled by categories c_i . Such a classifier categorizes a test document d_j by recursively testing for the weights. That the words labeling the internal nodes have in the representation of d_j , until a leaf node c_i is reached; the label of this leaf node is then assigned to d_j [13, 5]. In this study, we used C4.5 decision tree . C4.5 is a typical and effective decision tree method and it was used in some works to classify documents such as in [18, 19].

3.2 Maximum Entropy

The maximum entropy model estimates probabilities based on the principle of making as few assumption as possible, other than the constrained imposed. The constraints are derived from training process which expresses a relationship between the binary features and the outcome [21,20]. In text classification, maximum entropy is a model which assigns a class *c* of each word *w* based on its document *d* in the training data *D*. Conditional distributed *p(c|d)* is computed as follows [20]:

$$p(c \mid d) = \frac{1}{Z(d)} \exp\left(\sum_i \alpha_i f_i(d, c)\right) \quad (1)$$

Where Z(d) is a normalization function which is computed as:

$$Z(d) = \sum_c \exp\left(\sum_i \alpha_i f_i(d, c)\right) \quad (2)$$

And the parameter α_i must be learned by estimation. It can be estimated by a iterative way. In the equation, f_i *(d,c)* is a binary valued feature which makes a prediction about the outcome. In classification the feature presented by each instance to be classified. The type of feature could be either Boolean, that presents if the word is in the text, or integer which presents frequency of the word in the text. In this study integer type is used because it gives more information than Boolean. More precisely the feature can be formulated as [20]:

$$f_{(w,c')}(d,c)=\begin{cases}0 & \text{if } c\neq c' \\ \dfrac{N(d,w)}{N(d)} & \text{Otherwise}\end{cases} \qquad (3)$$

Where *N(d,w)* is the number of times word *w* occurs in document *d*, and *N(d)* is the number of words in *d*.

3.3 Artificial Neural Networks

Artificial Neural Networks is classification method. Many researches used it for classification such as [22,23,24] . Artificial Neural Networks are network of units. In text classification the input units represent words in the document *D*, the output represent the categories *C*, and the weights *W* on the edges that connect units represent conditional dependence relations. For classifying a test document d_j , its term weights wk_j are assigned to the input units; the activation of these units is propagated forward through the network, and the value that the output c_i takes up as a consequence determines the categorization.

3.4 Support Vector Machine

Support Vector Machineis a learning algorithm proposed by [25]. Using SVM in text classification proposed by my researchers such as [26], and subsequently used in [27 ,28]. In its simplest linear form, an SVM is a hyperplane that separates a set of positive examples from a set of negative examples with maximum margin. The following dot product formula is used for the output of a linear SVM:

$$y=\boldsymbol{w}.\boldsymbol{x}-b \qquad (4)$$

where $\boldsymbol{x}$ is a feature vector of classification documents composed of words. $\boldsymbol{w}$ is the weight of corresponding $\boldsymbol{x}$. *b* is a bias parameter determined by training process**.**

3.5 k-Nearest Neighbor

k-Nearest Neighbor is a very simple method to classify document. In training phase, documents have to be indexed and convert to vector representation. To classify new document *d*, the similarly of its document vector to each document vector in the training set has be computed. Then its *k* nearest neighbor is determent by measuring similarity may be measured by for example the Euclidean distance [29]. k-Nearest Neighbor has been used by [30,31] in classifying text documents.

3.6 Naïve Bays

Naïve Bays is a probability-based approach. It is used by [32,33,34] in text classification. Naïve Bays classifiers are widely used because of their simplicity and computational efficiency. Given a vector of words *w*, *c* the target classes can be estimated as [33]:

$$p(w\mid c)=\frac{1+n(w,c)}{\mid W\mid +n(c)} \qquad (5)$$

Where *n(w,c)* is the number of the word positions that are occupied by *w* whose class value is *c*. *n(c)* is the number of word positions whose class value is c. |*W*| is the total number of distinct words.

4.0 Experiments Settings

In our experiments we used Weka [35] machine learning project open-source software to implement all the classifiers except for maximum entropy we used JavaBased opennlp maxent package from (http://maxent.sourceforge.net).

In all of the experiments, 10-fold cross-validation was employed. In each classifier, all documents are divided almost equally to ten parts, nine of them is used for training and one for testing. The results are the average of the ten iterations. This process produce more reliable results and used all the corpus for both training and testing phases [36].

All classifiers we used in this study are supervised learning. Therefore, it needs a training corpus. Our experiments trained the system using Arabic documents collected from the Internet. It mainly collected from Aljazeera Arabic news channel (www.aljazeera.net) which is the largest Arabic site. The documents categorized into six domains: *politics, sports, culture and arts* , *science and technology* , *economy* and *health*.

To evaluate our experiments we computed recall (the percentage of the total documents for the given topic that are correctly classified) and precision (the percentage of predicted document for the given topic that are correctly classified) which are generally accepted ways of measuring system's performance in this field [37]. The following equations are used for recall and precision:

$$\mathrm{Re}\,call = \frac{CC}{TC} \qquad (6)$$

$$precision = \frac{CC}{TCF} \qquad (7)$$

Where CC number of correct classes found, TCF total number of classes found and TC, total number of correct classes.

Recall and precision can be combined into F-measure. F-measure is a standard statistical measure that is used to measure the performance of the classifier system [38] . The f-measure is computed as:

$$F1 = \frac{2 * \mathrm{Pr}\,ecision * \mathrm{Re}\,call}{\mathrm{Pr}\,ecision + \mathrm{Re}\,call} \qquad (8)$$

5. Experiments and Results

Using the previous settings, we conducted two experiments as follows:

5.1 Experiments without feature selection

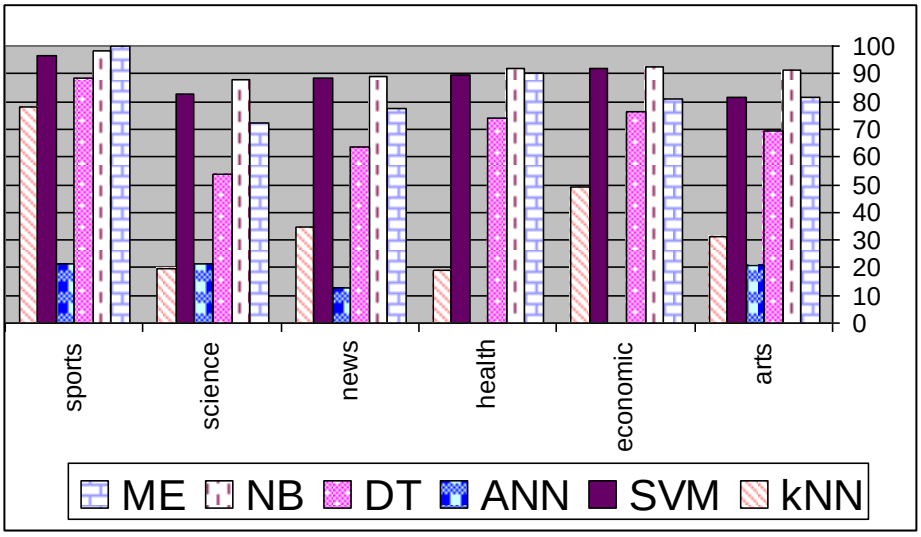
In the first experiment we applied the classifiers to the Arabic corpus. In the preprocessing stage, the text is converted to UTF-8 encoded and punctuations and non- letters are removed. Then text is parsed and all stopwords are removed. Stopwords are terms that are too frequent in the text. These terms are insignificant. So, removing them reduces the space of the items significantly. Then, some Arabic letters are normalized such as أ, إ, آ, is converted to ا, and ى replaced by ي and ة to ه.

Table 5.1 depicted the results. From these results, we can conclude that naïve bayes classifier has the best performance in general. Although, naïve bayes considered as the simplest classifier, it has the best performance not only in this experiment but with other researches such as work of [39] in English and [17] in Arabic. The other classifiers, support vector machine and maximum entropy, have fair results. On the other hand, the performance of Artificial Neural Networks and k-Nearest Neighbor was inadequate.

	Recall	*Precision*	*F-measure*
Maximum entropy	83.84	86.09	85.96
Naïve Bayes	91.75	92.48	91.81
Decision Tree	71.05	72.68	71.91
Artificial Neural Networks	16.7	10.10	10.81
Support Vector Machine	88.26	88.9	88.33
k-Nearest Neighbor	38.31	65.32	38.6

Table 5.1:Comaparsions between classifiers without using feature selection

Graph 5.1 displays the f-measure of each classifier distributed by the categories. From the category point of view, *sports* category is recognized very well by most classifiers especially maximum entropy which fully recognized all *sports* documents (e.g. 100% f-measure) and Naïve Bayes (e.g. 98.48% f-measure). That reason that *sports* category performed well because *sports* domain (e.g. words used) are limited comparing to other domains. On the other hand, *Science* and *health* is the weakest categories , that may due to the overlap in the words of the two categories.



Graph 5.1: The f-measure of each classifier distributed by the categories

5.2 Experiments with feature selection

In the second experiment we added a new step in the preprocessing phase which is feature selection. In general, the size of the training data sets are very large. To reduce the high dimensionality of the words by removing irrelevant and redundant information, feature selection was performed. In this case the features are the words to be trained in documents. Feature selection usually used to reduce the size of the training corpus to an acceptable level. The benefit of feature selection also includes a small improvement in predication accuracy in some cases [38]. To select the most appropriate words in the document, the Information Gain (IG) is computed for each word in the document. IG has been used by many researches for feature selection in data mining [40] . IG measures the number of bits of information obtained for category prediction by knowing the presence or absence of a feature. It is measured as [41,42]:

$$IG(f_k) = \sum_{c \in (c_i, \bar{c}_i)} \sum_{f \in (f_k, \bar{f}_k)} \Pr(f, c).\log \frac{\Pr(f, c)}{\Pr(f).\Pr(c)} \qquad (9)$$

Where f_k means the presence of the feature k and f_k means the absence of feature k.

In our experiment, the size of dataset is decreased from 1.05MB to 318 KB (by about 68.35%) . No doubt that decrease of the data set will significantly decrease the building time of the model. However, this reduction effect different classifiers in different ways. Table 5.2 present the performance of each classifier after using IG feature selection method.

	Recall	*Precision*	*F-Measure*
Maximum entropy	83.17	84.51	83.83
Naïve Bayes	83.6	85.5	83.9
Decision Tree	74.26	76.46	74.48
Artificial Neural Networks	73.31	74.66	74.33
Support Vector Machine	88.26	88.9	88.33
k-Nearest Neighbor	69.06	80.41	70.07

Table 5.2:Comaparsions between classifiers using Information Gain feature selection

From the table we can noticed that ,feature selection did not effect ,or slightly effected, maximum entropy, naïve bayes, decision trees and support vector machines. However, it significantly improved the performance of Artificial Neural Networks and k-Nearest Neighbor. But, still naïve bayes is the best classifier and support vector machines follows.

6. Conclusion

In this paper, the results of six well known classifiers' methods are compared in Arabic text classification. The classifiers we applied are: Maximum entropy, Naïve Bayes, Decision Trees, Artificial Neural Networks, Support Vector Machine and k-Nearest Neighbor. We first, compared the methods after preprocessing but without using feature selection and we found that the performance of Naïve Bayes is the best, the performance of Maximum Entropy, Support Vector Machine and Decision Tree are acceptable ,but the performance of k-Nearest Neighbor and Artificial Neural Networks was bad. However, after using Information Gain as feature selection, the data was reduced significantly and the performance of k-Nearest Neighbor and Artificial Neural Networks improved significantly. The performance Naïve Bayes of did not change but still the best classifier to Arabic corpus. In addition, The classifiers categorized the Arabic documents into six domains: *politics, sports, culture and arts , science and technology , economy* and *health*. In almost all classifiers, we found that the accuracy of *sports* is the best and *science* is the worst.

For future work, a way to increase the performance of the classifiers for Arabic corpus should be investigated especially in domain which more open such as news and science. One way could be using other kinds of classifiers or combine more than one method. Also, the experiments in this paper used default parameters for the classifiers methods, changing the parameters could be another way to increase the performance of the classifiers.

References

[1] D. Lewis . "(Naive) Bayesian Text Classification for Spam Filtering". ASA Chicago Chapter Spring Conference., Loyola University, May 7, 2004.

[2] P. Ipeirotis, L. Gravano, and M Sahami. "Automatic Classification of Text Databases Through Query Probing". In Proceedings of WebDB 2000. Pp. 117-122.

[3] M. Ruiz, A. Diekema, and P. Sheridan." CINDOR Conceptual Interlingua Document Retrieval". In Proceedings of TREC-8 Evaluation 1999.

[4] J. Hidalgo, M. Rodríguez, and J. Cortizo. "The Role of Word Sense Disambiguation in Automated Text Categorization". Applications of Natural Language to Data Bases 2005.

[5] R. Du, R. Safavi-Naini ,and W. Susilo. "Web filtering using text classification" . In The 11th IEEE International Conference on Networks, 28 September - 1 October 2003, 325-330.

[6] A. Ozgur, L. Ozgur, and Tunga Gungor, "Text Categorization with Class-Based and Corpus-Based Keyword Selection". Lecture Notes in Computer Science, Vol.3733, pp.607-616, Springer-Verlag, 2005.

[7] L. Aas and L. Eikvil, "Text Categorization: A survey". Norwegian Computing Center, Report NR 941, 1999**.**

[8] B. Hammo, H. Abu-Salem, S. Lytinen, and M. Evens, "QARAB: A Question Answering System to Support the Arabic Language. In the proceedings of Workshop on Computational Approaches to Semitic Languages. ACL 2002, Philadelphia, PA, July. p 55-65 2002.

[9] M. Syiam, Z. T. Fayed , and M. B. Habib. "An Intelligent System For Arabic Text Categorization". The International Journal of Intelligent Computing and Information Volume 6 Number 1 January 2006.

[10] M. El-Kourdi, A. Bensaid, and T. Rachidi. " Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm". In the proceedings of 20th International Conference on Computational Linguistics . August 28th. Geneva 2004.

[11] L. Khreisat. "Arabic Text Classification Using N-Gram Frequency Statistics A Comparative Study". In the proceedings of the 2006 International Conference on Data Mining, DMIN 2006, Las Vegas, Nevada, USA, June 26-29, 2006. Pp.78-82

[12] El-Halees A. M. (2007). "Arabic Text Classification Using Maximum Entropy ". In The Islamic University Journal (Series of Natural Studies and Engineering) .Vol 15, No. 1 Jan. pp. 157-167.

[13] F. Sebastiani. "Machine learning in automated text categorization". ACM Computer Survey. 34(1).Pp 1-47 2002.

[14] H. Berger, and D. Merkl. "A Comparison of Text-Categorization Methods Applied to N-Gram Frequency Statistics". In the proceedings of Australian Conference on Artificial Intelligence 2004.Pp 998-1003.

[15] Y. Yang, and , X. Liu. "A re-examination of text categorization methods". In the proceedings of SIGIR-99, 1999.

[16] S. T. Dumais, J. Platt, D. Heckerman and M. Sahami. "Inductive learning algorithms and representations for text categorization". In Proceedings of ACM-CIKM98, Nov. 1998, pp. 148-155.

[17] M. N. Al-Kabi, and S. I. Al- Sinjilawi. "A Comparative Study of the Efficiency of Different Measures to Classify Arabic Text". The University of Sharjah Journal of Pure and Applied Sciences vol 5 No 2 June 2007.

[18] I. Ilovich, and S. Markovitch. "Feature Generation for Text Categorization Using World Knowledge". In the Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005.pp 1048-1053.

[19] D.E. Johnson, F.J. Oles, T. Zhang, and T. Goetz. "A decision-tree-based symbolic rule induction system for text categorization". IBM Systems Journal, Sept 2002 by Volume 41, Number 3, 2002

[20] A. Berger, and D. Pietra. "A Maximum Entropy Approach to Natural Language Processing" . Computational Linguistics, Vol., 22. p. 39-7,1996.

[21] K. Nigam, J. Lafferty, and J. McCallum. "Using Maximum Entropy for Text Classification". In IJCAI-99 Workshop on Machine Learning for Information Filtering, pp. 61-67. 1999.

[22] C. H. Li , and S. C. Park. "Neural Network for Text Classification Based on Singular Value Decomposition". In the Proceedings of the 7th IEEE International Conference on Computer and Information Technology**,** Pages 47-52 , 2007**.**

[23] M. E. Ruiz, and P. Srinivasan . "Hierarchical Text Categorization Using Neural Networks" Information Retrieval . Volume 5, Number 1. Pages 87-118. Jan 2002.

[24] M. E. Ruiz, and P. " Automatic Text Categorization Using Neural Networks"
In the Proceedings of the 8th ASIS SIG/CR Classification Research Workshop. Medford, New Jersey. pp 59-72. 1998.

[25] C. Cortes, and V. Vapnik. "Support-Vector Networks", Machine Learning, 20, 1995.

[26] T. Joachims. "Text Categorization with Support Vector Machines: Learning with Many Relevant Features". In the Proceedings of ECML-98, 10th European Conference on Machine Learning

[27] S. T. Dumais, J. Platt, D. Heckerman and M. Sahami. "Inductive learning algorithms and representations for text categorization". In The Proceedings of ACM-CIKM98, Nov. 1998, pp. 148-155.

[28] H. Taira, and M. Haruno . "Feature selection in SVM text categorization", In the Proceedings of the sixteenth national conference on Artificial intelligence Pages: 480 - 486 , 1999.

[29] B. .Dasarathy, " Nearest Neighbor (NN) Norms : NN Pattern Classification Techniques", IEEE Computer Society Press, 1991.

[30] E.H. Han, G. Karypis, and V. Kumar. " Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification". In the Proceedings of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining**.** Pages: 53 - 65 .2001.

[31] O.W Kwon, and J. H. Lee." Text categorization based on k-nearest neighbor approach for web site classification". International Journal of Information Processing and Management. Volume 39 , Issue 1 . Pages: 25 - 44 . 2003.

[32] M. Kłopotek, and M. Woch. "Very Large Bayesian Networks in Text Classification", Computational Science, Volume 2657 , 2003.

[33] F. Colace, and M. De Santo, "A Bayesian Approach for Text Classification" Information and Communication Technologies,, Vol 1, Pages 1323- 1326 2006.

[34] W. Dai, G. R. Xue, Q. Yang, Y. Yu. "Transferring Naive Bayes Classifiers for Text Classification. Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada. AAAI Press 2007 540-545

[35] I. Witten and E. Frank . "Data Mining: Practical machine learning tools and techniques", 2nd Edition, Morgan Kaufmann, San Francisco, 2005.

[36] P.R. Cohen. "Empirical Methods for Artificial Intelligence", MIT, Press Cambridge, MA. 1995.

[37] Apte , Damerau, and Weiss. "Towards Language Independent Automated Learning of Text Categorization Models" . Research and Development in Information Retrieval P 23-30. 1994.

[38] K. Uchitmoto, Q. Ma, M. Murata, H. Ozaku, , H . Isahara " Named Entity Extraction Based on A Maximum Entropy Model and transformation Rules". Journal of natural Language Processing. Vol. 7. Issue 2 P. 63-90 . (2000)

[39] N. Friedman , D. Geiger, M. Goldszmidt. "Bayesian Network Classifiers". Machine Learning , 29:131 -163. 1997

[40] Y. Yang, and J. Pedersen, " Comparative Study on Feature Selection in Text Categorization" in Proceedings of ICML-97, 14th International Conference on Machine Learning , 1997.

[41] Y. Liu. " A Comparative Study on Feature Selection Methods for Drug Discovery" . J. Chem. Inf. Comput. Sci., 44 (5) pages 1823-1828. 2004.

[42] T. Mitchell, " Machine Learning". The Mc-Graw-Hill Companies, Inc., 1997.