E-Mail Processing with Fuzzy SOMs and Association Rules

Laura Lanzarini, Augusto Villa Monte, César Estrebou
III-LIDI (Institute of Research in Computer Science LIDI)
Faculty of Computer Science, National University of La Plata
La Plata, Buenos Aires, Argentina

ABSTRACT

E-mail texts are hard to process due to their short length. In this article, the use of a diffuse neural network that is capable of identifying the most relevant terms in a set of e-mails is proposed. The associations between these terms will be measured through association rules built with the terms identified by the network. The metrics support, confidence and interest of the rules will be used to qualify the corresponding terms. The method proposed has been used to process e-mails of the PACENI Project (Support Project for Improving First-Year Teaching in Courses of Studies in Exact and Natural Sciences, Economic Science and Computer Science). With this type of analysis, the most common topics of student questions have been identified. Even though this new information can have various applications, they all involve, as a first instance, an improvement in student service.

Keywords:Information Retrieval, Data Mining, Text Mining, E-mails Analysis, FSOM, Association Rules.

I. INTRODUCTION

Distance education platforms are a learning environment through which teachers and students interact by performing various types of activities.
In this context, electronic mail is the most commonly used mechanism, and it is therefore of interest for the study of techniques that allow analyzing and modeling the information shared through this medium. For example, it would be relevant knowing the topics most frequently enquired by students. This could have various applications:

- It would allow detecting shortcomings in the information provided, for instance, lack of information regarding exam dates or the need for reinforcement in any given topic because the theoretical material provided has not been clear enough.
- Automatically organizing e-mails to improve student service.
- Automatically identifying core discussion topics in order to improve decision-making.

An e-mail has a date, a set of addresses, a subject, and a body. The latter, even though it may contain various types of information, consists basically of text and can therefore be analyzed by means of text mining techniques.
Text mining is a branch of Data Mining, and its main purpose is the extraction of high-quality information from documents. It has numerous applications in various areas:

- In Biomedicine, it has been used to automate the identification and extraction of information from the numerous papers published each year [1].
- In Molecular Biology, it has been used to automatically extract information about genes, proteins and their functional relations from large collections of texts [10].
- In Education, it has been used to facilitate resource searches by combining the documents from various Web sites from related organizations [2].
- In the commercial context, it has been used to analyze the information generated by a consumer complaint Web site in order to obtain word relations that allow understanding the data [6],
- In the hospitality industry, using information available on Internet about hotels and possible tourists, it has been used to develop competitive strategies by analyzing demographic features and browsing habits [11].

All these works are representative of the diversity of areas in which text mining techniques are applied. However, regardless of the type of problem at hand, in most of the cases the main purpose is determining the relevance of the document based on a previous query. This allows more efficient automatic classification and access.
However, the extraction of information from e-mails is based on some special considerations, since, in general, the texts are short and their wording is quite abbreviated. Thus, some of the metrics used are no longer relevant, such as text length or the frequency of any given word within it.
The method proposed in this paper was applied to the e-mails of the Tutors Program (PACENI). This program is promoted by the Ministry of Education and its purpose is reducing the number of students that drop out from their university courses of study during their first year. This program was implemented at UNLP in the 2009 school year. Through it, first-year students are accompanied by tutors, post-graduate students or advanced students, who help them overcome the initial difficulties of university life.
For the processing stage, a dictionary built automatically from the reduction of each word to its root (stemming) [7] and its

subsequent selection was used. By using this dictionary, each e-mail was represented as a numerical vector and was then used to train a FSOM (Fuzzy Self Organizing Map) neural network. Based on the weights of each neuron of the trained network, and taking into account the degree of belonging of the corresponding examples, the most common combinations of terms were identified. Finally, association-rule metrics are used to establish the relevance of each combination.
This paper is organized as follows: in Section 2, some related works are mentioned; in Section 3, SOM and FSOM networks and their training mechanism are briefly described; in Section 4, the method proposed is detailed; in Section 5, the results obtained are presented; and in Section 6, conclusions are drawn and future lines of work presented.

II. Related work

Obtaining information from e-mails is a relevant task whose main purpose is classification and interpretation.
In this sense, the identification of spam e-mails is a generalized problem and has therefore received a lot of attention [13], [16], [19], [14], [12].
There are also approached that seek to automatically identify the author of the e-mail or the core subject of the message. For example, [17] tries to identify the person writing the e-mail from features based on number of words, number of lines, and the frequency of significant key words. [15] proposed a method that assesses the words from e-mails based on their age. The age of a word is calculated based on the frequency with which e-mails including it are received. The problem of this approach is the number of different words that can be used to refer to the same concept.
There is a current approach that has become popular with the appearance of various social networks in work environments. Nowadays, the development of collaborative tasks and the use of e-mails as communication mechanism are common. This creates the need of solving some participation-related issues, which implies identifying project members and their categories, as well as central work topics [5].
The general objective of this paper is related to this latter approach—we try to obtain information from a group of e-mails generated by teacher-student relations during a course carried out through a distance-education platform.

III. SOM and FSOM

The SOM (Self Organizing Maps) neural network was defined by Kohonen in 1982 [8]. Its diffuse version was defined in 1994 by Petri Vuorimaa [18]. In both cases, their main application is the clustering of available information. Its ability to preserve input data topology makes them a visualization tool that is widely used in various areas.
Basically, it can be said that both SOM and FSOM networks are a two-layered structured: the input layer, whose function is only to allow information to enter the network, and the competitive layer, which is responsible for the clustering task. The neurons that form this second layer are connected and have the ability of identifying the number of "hops" or connections that separate them from each of the remaining neurons in this level.

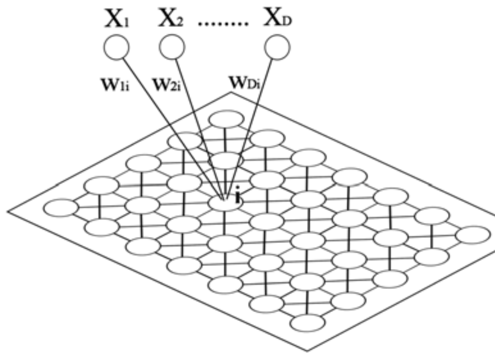

Fig. 1. Classic structure of a SOM network. w_{ji} will be used to denote the weight of the arch going from the j^{th} input neuron to the i^{th} competitive neuron.

Figure 1 shows the structure of a SOM network where the input layer is formed by a D-dimensional vector and the competitive layer has $7x5 = 35$ neurons. Each neuron in this second layer has 8 direct neighbors (immediate connections). This connection pattern can change depending on the problem to solve.
Each competitive neuron is associated to a weight vector represented by the values of the arches that reach this neuron from the input layer. These values, for all the neurons in this layer, are represented in the figure by means of the W matrix. Network weights, W values, are initially random, but they adapt with the successive presentations of input vectors.
Since this is a competitive structure, each input vector is considered to be represented by (or associated with) the competitive neuron that has the most similar weight vector based on a given similarity measure.
The final value of W is obtained by means of an iterative process that is repeated until the weight vectors do not present any significant changes or, in other words, until each input vector is represented by the same competitive neuron than in the previous iteration. During the training process of the SOM network, in each iteration, for each input vector $X = (x_1, x_2, \ldots, x_D)$, the closest winning neuron is identified by means of a distance measure. That is, if the i^{th} competitive neuron has a weight vector $(w_{1i}, w_{2i}, \ldots, w_{Di})$, the SOM identifies the winning neuron as that complies with equation (1)

$$||W_{winner} - X|| = min(||W_i - X||) \qquad i = 1..N \qquad (1)$$

where *winner* is the winning neuron, $||\ .\ ||$ is a distance measure, generally Euclidean distance, and N is the total number of competitive neurons.
Then, the SOM updates only the weight vector for that neuron and its neighborhood following equation (2)

$$W_i = W_i + \alpha * (X - W_i) \qquad (2)$$

where i is the competitive neuron whose vector is being updated and α is a value between 0 and 1 that represents

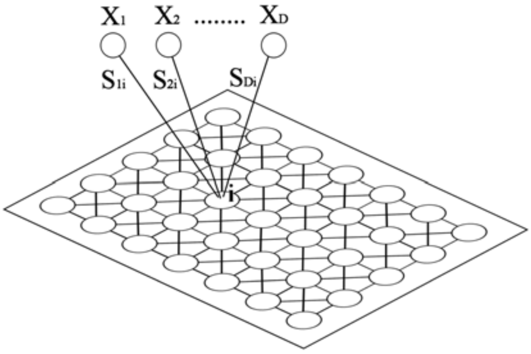

Fig. 2. FSOM network architecture used. The input arches of the i^{th} competitive neuron correspond to one-dimensional Gaussian functions.

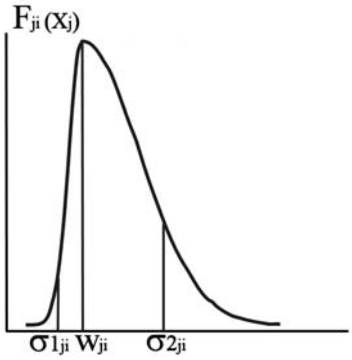

Fig. 3. Gaussian function corresponding to the arch that joins the j^{th} input neuron with the i^{th} competitive neuron

a learning factor.

The weight vectors of the remaining competitive neurons remain unchanged.

Equation (2) has variations that can be consulted in [9].

The concept of neighborhood is used to allow the network to adapt correctly. This implies that neighboring competitive neurons represent similar input patterns. For this reason, the training process (obtaining W values) is started with a wide neighborhood that is then reduced as iterations occur.

On the other hand, the FSOM network, whose architecture is shown in Figure 2, shares the main characteristic of SOM networks in regards to respecting the topology of input data. That is, a two-layered architecture is used in both cases. However, the input arches for each competitive neuron in FSOM represent one-dimensional, non-symmetrical Gaussian functions, as shown in Figure 3. It can be observed that the maximum value of the function is reached at w_{ji} and that the deviations to the left and right, represented by $\sigma 1$ and $\sigma 2$, respectively, are different. This allows having an adequate coverage of the input space. For this reason, the weights of the arches corresponding to the i^{th} competitive neuron in FSOM are represented through the vector indicated in equation (3)

$$S_i = (s_{1i}, s_{2i}, \ldots, s_{Di}) \tag{3}$$

being s_{ji} a tuple represented by the three parameters of the j^{th} Gaussian function, as shown in (4).

$$s_{ji} = (w_{ji}, \sigma 1_{ji}, \sigma 2_{ji}) \tag{4}$$

Additionally, in FSOM, each input vector is not associated to an only winning neuron, but it has a degree of belonging for each of them.

Equation (5) shows how to calculate the degree of belonging for input vector X with respect to the i^{th} competitive neuron.

$$F_i(X) = min(F_{ji}(x_j)) \quad j = 1..D \tag{5}$$

being

$$F_{ji}(x_j) = \begin{cases} exp((-\frac{1}{2})(w_{ji} - x_j)^2/\sigma 1^2_{ji}) & x_j < w_{ji} \\ exp((-\frac{1}{2})(w_{ji} - x_j)^2/\sigma 2^2_{ji}) & x_j \geq w_{ji} \end{cases} \tag{6}$$

Note that in (6), a Gaussian function with mean w_{ji} and deviation $\sigma 1_{ji}$ or $\sigma 2_{ji}$, for values of the j^{th} attribute of the input vector to the left of right of the mean value w_{ji}, is used. Thus, each competitive neuron can be seen as D Gaussian, non-symmetrical functions, one for each attribute of the input vector.

FSOM diffusely clusters the input space from a competitive training that is similar to the one used by SOM. However, unlike SOM that only learns cluster centers, FSOM learns both centers and deviations around them.

The training method used to train the FSOM network in this article uses an equation that is similar to (1) to determine the winning neuron, where || . || represents Mahalanobis distance considering that the attributes of vector X are independent, that is,

$$|| W_{winner} - X || = \min_{i=1..N} \left(\sqrt{\sum_{j=1}^{D} \left(\frac{(w_{ji} - x_j)^2}{\sigma^2_{ji}} \right)} \right) \tag{7}$$

In each iteration, for the winning neuron and its neighborhood, the corresponding Gaussian functions are adjusted based on equations (8) and (9).

$$w_{ji} = w_{ji} + \alpha * (x_j - w_{ji}) \quad j = 1..D \tag{8}$$

$$\sigma_{ji} = \sigma_{ji} + 2 * \alpha * \sigma_{ji} * ((x_j - w_{ji})^2 - \sigma^2_{ji}) \quad j = 1..D \tag{9}$$

Once training is completed, a correction process is applied to the deviations obtained to ensure full coverage of the input space.

Let's assume that the k^{th} competitive neuron is an immediate neighbor of the i^{th} neuron, then their respective deviations are adjusted as follows:

$$a = (w_{ji} - cte * \sigma_{ji}) - (w_{jk} + cte * \sigma_{jk}) \quad j = 1..D \tag{10}$$

$$\sigma 1_{ji} = \sigma_{ji} + a * \sigma_{ji}/(\sigma_{jk} + \sigma_{ji}) \quad j = 1..D \tag{11}$$

$$\sigma 2_{jk} = \sigma_{jk} + a * \sigma_{jk}/(\sigma_{jk} + \sigma_{ji}) \quad j = 1..D \tag{12}$$

Note that the result of the training algorithm allows obtaining symmetrical Gaussian functions, which are then adjusted as

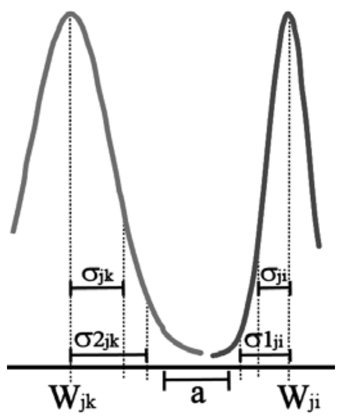

Fig. 4. Gaussian function deviation adjustment after training.

indicated in (11) and (12) to obtain the deviations to the left and right, called $\sigma 1$ and $\sigma 2$, respectively. This final adjustment is illustrated in Figure 4.

IV. Proposed Method

To be able to operate with the network described above, e-mails have to be represented by numerical vectors. To this end, a dictionary of terms will be built by processing an only text formed by the concatenation of the subject and body of each e-mail. Each word in the text is reduced to its root by applying a stemming algorithm [4]. This process is important for processing text in Spanish due to the syntactic changes related to gender, number, and tense. For example, words such as 'trabajo' (work), 'trabajar' (to work), 'trabaja' (he/she works), 'trabajos' (the works), 'trabajoso'(laborious) are reduced to the common root 'trabaj' by applying the stemming algorithm.

Once the root of each word is obtained, its frequency of use in the entire text and its average length are calculated. By means of statistical analysis processes, terms that are less relevant are discarded; the dictionary to represent e-mails is then built with the remaining terms.

Then, each e-mail is represented by a fixed-length binary vector. The number of elements in the vector is determined by the number of words in the dictionary. Each position will have a value of 1 if the word appears in the e-mail or a value of 0 if it does not.

Be D the number of words in the dictionary and M the number of available e-mails, each e-mail will be represented as follows:

$$email_p = (m_{p1}, m_{p2}, ..., m_{pD}) \quad with\ p = 1..M \tag{13}$$

$$m_{ij} = \begin{cases} 1 & the\ word\ j\ is\ in\ e-mail\ i \\ 0 & otherwise \end{cases} \quad j = 1..D \tag{14}$$

Using the vectors defined in (13), the FSOM network is trained by applying the algorithm described in the previous section.

Be N the number of competitive neurons that form the FSOM network, as already mentioned, tuple vectors Si described in equation (3) and (4) contain the information required to calculate the degree of belonging for each input vector with respect to each competitive neuron. However, if only w_{ji} values are considered (that is, the first element of each tuple) the centroids of each cluster can be used.

Be $W_i = (w_{1i}, w_{2i}, \ldots, w_{Di})$ the centroid of the i^{th} competitive neuron. Its elements are real values that are equal to or higher than zero, where a high value in the j^{th} element implies a frequent appearance of the j^{th} dictionary word among the input patterns represented by this competitive neuron.

After the training process, taking advantage of the FSOM network structure, the positions of W_i vectors between neurons that are connected directly have been reinforced. Thus, each W_i vector is increased by the average value of its immediate neighbors only at those positions where a minimum threshold is exceeded. As a result, common words are highlighted.

Finally, less significant words, that is, those words whose own weight is not enough to be clearly represented by a limited subset of neurons, have to be discarded. This is the case of words that are combined with many terms or that are infrequently used. In either case, these are terms that provide little information, since in the first case they do not determine the subject matter and in the second case are not sufficiently supported (number of occurrences) to be considered significant. The trained FSOM network is able to detect these words because they do not go beyond a minimum threshold in any of the vectors associated with the competitive neurons (Equation 15). Therefore, irrelevant words will be removed from each W_i vector (equation 16) by means of this threshold.

$$WEnd_i = (wEnd_{1i}, wEnd_{21}, \ldots, wEnd_{Di}) \quad k = 1..N \tag{15}$$

where

$$wEnd_{ji} = \begin{cases} w_{ji} & w_{ji} > threshold \\ 0 & otherwise \end{cases} \quad j = 1..D \tag{16}$$

and irrelevant words are obtained with equation (17).

$$IrrelevantWords = \{word_j \mid wEnd_{ji} = 0\} \tag{17}$$

with $j = 1..D$ and $i = 1..N$

It should be noted that each element in $WEnd_i$ will contain a real value that is greater than zero in the positions that correspond to relevant terms. This allows identifying the frequent terms in the e-mails represented by the k^{th} neuron of the network, which can be used to form various association rules. An association rule is an expression with the following format

IF (antecedent) THEN (consequence)

where both the antecedent and the consequence are logical expressions referring to the words present in the e-mail.

The following are examples of association rules:

- IF ('board' $\wedge$ 'exten' $\wedge$ 'certific') THEN ('transcr' $\wedge$ 'present' $\wedge$ 'academ' $\wedge$ 'approve')
- IF ('pending') THEN ('academic subject' $\wedge$ 'certificate')

The first rule indicates that each time an e-mail has the words 'board', 'exten' and 'certific', the words 'transcr', 'present', 'academ' and 'approve' are also present. This rule refers to the approval by the academic board of extensions to present school transcripts. The second rule shows the relationship between the word 'pending' and the words 'academic subject' and 'certificate'. It also refers to pending academic subject certificates.

Rules are formed by combining in all possible ways the terms that appear in any given weight vector defined as in equation (15).

There are various metrics that can be used to determine the importance of a rule. The most common ones are:

- Support: It is the proportion of examples (e-mails) that fulfill the rule. For example, if the words 'pending', 'academic subject' and 'certificate' are present in 300 e-mails from a total of 3,000 e-mails, the support of the rule

  IF ('pending') THEN ('academic subject' $\wedge$ 'certificate')

  will be 300/3000 = 0.1.
- Confidence: it is the quotient of the number of examples that fulfill the rule and the number of those that only fulfill the antecedent. Let us consider again the rule IF ('pending') THEN ('academic subject' $\wedge$ 'certificate') verified by 300 of the 3,000 available e-mails. Let us assume that after revising the available e-los, it is observed that 350 of those contain the word 'pending,'; the confidence of this rule will be 300/350 = 0.85

The importance of the rules obtained in this paper depends on the average of the three previously mentioned metrics. Therefore, the result that can be obtained is the interpretation of the most relevant rules.

V. Results

The method described in Section 4 was applied to the 7,624 e-mails from the Tutors Program (PACENI) between April 2009 and December 2010.

The initial dictionary was formed by 6,446 term roots; 486 of these were selected by statistical analysis. The selection criterion used had four stages:

a. First, stop words and user names and last names were removed. Usual courtesy terms were also removed (thank you, dear, regards, favor, etc.), which are identified by means of a taboo list.
b. Next, words of atypical lengths were suppressed, considering as such all words whose value was more than 1.5 times the distance between the first and the third quartile (fourth dispersion). In the case of the PACENI e-mails, words that were longer than 18 characters or shorter than 3.5 characters were discarded. The average length was assessed based on all words that corresponded to the same root term.
c. When analyzing the boxplot corresponding to the occurrence frequency for each root term, it was observed that a large part of the population had a low value. That is, the most commonly used terms were the minority. Therefore, we decided to use those terms with extreme frequency. For the measured population, these were the terms with more than 52 occurrences.
d. Finally, in the boxplot of the reduced population, extreme values corresponding to the terms that are very frequently used in all e-mails are still observed. This reduces their importance. For this reason, those terms whose frequency was higher than 954 —extreme value—were removed.

A FSOM network with 13x13 competitive neurons with 4 neighbors per neuron was used. The initial size of the neighborhood was set as a third of the number of rows in the network, that is, 4 neurons. This value is high, since it is the radius (number of "hops") that determines the area around the winning neuron where weight vectors are modified. The reduction was carried out every 30 iterations, with a maximum of 180 iterations. This value ensures that successive reductions will be carried out until the adaptation only affects the winning neuron.

After training the network, all weights that were not significant were removed from the matrix W; to this end, a threshold equivalent to the mean value plus one time the standard deviation was used.

With the weight vectors of each competitive neuron, the combinations of terms that allow clustering the e-mails were determined.

To measure their relevance, they were used to form the corresponding association rules, considering all possible combinations.

To calculate the three metrics, the degree of belonging of each input vector with respect to the weight vector modified with (15) and (16) was used. Thus, it can be considered that the input vector corresponds to the weight vector if the degree of belonging is higher than 0.5 (estimated mean value).

Finally, the combination was associated with the maximum value obtained by averaging its support, confidence, and interest values.

After 50 independent training sessions of the neural network, the most commonly occurring terms are the following:

- ('windows', 'microsoft','realiz', 'facult')
- ('tramit' , 'entreg', 'analit' , 'alumn' , 'certific')
- ('inic', 'horari', 'cuatrimestr', 'teor', 'clas')
- ('guaran', 'inscripci', 'anot')
- ('matemat', 'pregunt', 'final')

These combinations appear in various orders, but always within the 10 first best positioned ones. This determines their importance within the set of e-mails. Another characteristic that was observed after the various tests is that the neural network allows discarding between 60 and 80 terms by means of the threshold function indicated in equation (16). This reduces considerably the time required to obtain the association rules to be measured.

VI. Conclusions and future lines of work

An e-mail analysis mechanism based on data mining techniques has been presented. Even though the results obtained only refer to the 2009-2010 Tutors Program of the PACENI, this analysis can be applied to other courses with no considerable changes. We have already solved this problem by applying a SOM network [3]. Even though the results obtained back then were satisfactory, a much more significant reduction in term selection is observed with the method proposed in this paper. This allows obtaining rules that are more compact and easier to interpret. Additionally, the assessment of the various metrics by means of the degree of belonging allows better identifying the most relevant groups of words. Building the initial dictionary is essential to obtain good combinations of terms. The proposal presented in this paper included a statistical pre-processing so as to generate the dictionary as automatically as possible. This stage was improved in comparison to that in [3] by the use of a taboo list for usual courtesy words. We are currently working in the resolution of this problem using a dynamic FSOM network in order to avoid limiting the adaptation ability of the network.

References

[1] Sophia Ananiadou, Douglas B. Kell, and Junichi Tsujii. Text mining and its potential applications in systems biology. *Trends in Biotechnology*, 24(12):571–579, 2006.

[2] Sophia Ananiadou, Paul Thompson, James Thomas, Tingting Mu, Sandy Oliver, Mark Rickinson, Yutaka Sasaki, Davy Weissenbacher, and John McNaught. Supporting the education evidence portal via text mining. *Philosophical Transactions of the Royal Society A: Mathematical,Physical and Engineering Sciences*, 368(1925):3829–3844, 2010.

[3] Laura Lanzarini Augusto Villa Monte, César Estrebou. E-mail processing using data mining techniques. In *XVI Congreso Argentino de Ciencias de la Computación CACIC 2010*, pages 987–992, 2010.

[4] Dennis D. Perez Barrenechea. A spanish stemming algorithm implementation in prolog and c#. 2006.

[5] Christian Bird, Alex Gourley, Prem Devanbu, Michael Gertz, and Anand Swaminathan. Mining email social networks. In *Proceedings of the 2006 international workshop on Mining software repositories*, MSR '06, pages 137–143, New York, NY, USA, 2006. ACM.

[6] Kuan C. Chen. Text mining e-complaints data from e-auction store with implications for internet marketing research. *Journal of Business and Economics Research (JBER)*, 7(5):15–24, 2009.

[7] Carlos G. Figuerola, Raquel Gómez Díaz, Angel F. Zazo Rodríguez, and José Luis Alonso Berrocal. Spanish monolingual track: The impact of stemming on retrieval. In *Revised Papers from the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, CLEF '01, pages 253–261, London, UK, 2002. Springer-Verlag.

[8] Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.

[9] Teuvo Kohonen. *Self-Organizing Maps*. Springer, 1997.

[10] Martin Krallinger and Alfonso Valencia. Text-mining and information-retrieval services for molecular biology. *Genome Biology*, 6(7):224, 2005.

[11] Kin-Nam Lau, Kam-Hon Lee, and Ying Ho. Text mining for the hotel industry. *Cornell Hotel and Restaurant Administration Quarterly*, 46(3):344–362, 2005.

[12] Xiao Li, Junyong Luo, and Meijuan Yin. E-mail filtering based on analysis of structural features and text classification. In *Intelligent Systems and Applications (ISA), 2010 2nd International Workshop on*, pages 1–4, 2010.

[13] Carlos M. Lorenzetti, Rocío L. Cecchini, Ana Gabriela Maguitman, and András A. Benczúr. Métodos para la selección y el ajuste de características en el problema de la detección de spam. *XII Workshop de Investigadores en Ciencias de la Computación, Area de Agentes y Sistemas Inteligentes*, 2010.

[14] Wanli Ma, D. Tran, and D. Sharma. A novel spam email detection system based on negative selection. In *Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on*, pages 987–992, 2009.

[15] Jason D. M. Rennie. ifile: An application of machine learning to e-mail filtering. In *Proc. KDD Workshop on Text Mining*, 2000.

[16] M.F. Saeedian and H. Beigy. Dynamic classifier selection using clustering for spam detection. In *Computational Intelligence and Data Mining, 2009. CIDM '09. IEEE Symposium on*, pages 84–88, 2009.

[17] O. De Vel. Mining e-mail authorship. *KDD2000 Workshop on Text Mining*, 30(4):55, 2000.

[18] Petri Vuorimaa. Fuzzy self-organizing map. *Fuzzy Sets Syst.*, 66:223–231, September 1994.

[19] Tsan-Ying Yu and Wei-Chih Hsu. E-mail spam filtering using support vector machines with selection of kernel function parameters. In *Proceedings of the 2009 Fourth International Conference on Innovative Computing, Information and Control*, ICICIC '09, pages 764–767, Washington, DC, USA, 2009. IEEE Computer Society.

Received: October 2010. Accepted: February 2011.