Artif Intell Rev (2013) 40:305–377
DOI 10.1007/s10462-011-9285-z

Genetic optimized artificial immune system in spam detection: a review and a model

Raed Abu Zitar · Adel Hamdan

Published online: 23 November 2011

Abstract Spam is a serious universal problem which causes problems for almost all computer users. This issue affects not only normal users of the internet, but also causes a big problem for companies and organizations since it costs a huge amount of money in lost productivity, wasting users' time and network bandwidth. Many studies on spam indicate that spam cost organizations billions of dollars yearly. This work presents a machine learning method inspired by the human immune system called Artificial Immune System (AIS) which is a new emerging method that still needs further exploration. Core modifications were applied on the standard AIS with the aid of the Genetic Algorithm. Also an Artificial Neural Network for spam detection is applied with a new manner. SpamAssassin corpus is used in all our simulations.

Keywords Spam · Artificial immune system · Genetic algorithm

In standard AIS several user defined parameters are used such as culling (rejecting) of old lymphocytes (type of white blood cell). Genetic Optimized AIS is used to present culling time instead of using user defined value. Also, a new idea to check antibodies (type of protein produced white blood cells to attack germs) in AIS is introduced. This would make the system able to accept types of messages that were previously considered as spam. The idea is accomplished by introducing a new issue which we call rebuilds time. Moreover, an adaptive weight of lymphocytes is used to modify selection opportunities for different gene fragments (pieces of gene containing the exons).

In this work, core modifications on ANN in the neurons are applied; these modifications allow neurons to be changed over time and to replace useless layers. This method eventually

R. A. Zitar (✉)
Department of Computer Science, School of Engineering and Computing Sciences,
New York Institute of Technology, Amman, Jordan
e-mail: rzitar@nyit.edu

A. Hamdan
Department of Computer Science, Applied Science University, Amman, Jordan

gave us promising results. This approach is called Continuous Learning Approach Artificial Neural Network CLA_ANN.

The final results are compared and analyzed. Results show that both systems, Optimized Spam Detection using GA and Spam Detection using ANN, achieved promising scores comparable to standard AIS and other known methods.

1 Introduction

E-mails are one of the most important forms of communication; e-mails are simple, effective, and a cheap type of communication for almost all computer users. This simplicity and cheapness are attacked by several threats. One of the most important threats is spam; spam e-mails are a problem that almost every e-mail user suffers from. The word "spam" usually denoted a particular brand of luncheon meat, but in recent times, spam is used to represent a variety of junk, unwanted e-mails. It is now possible to send thousands of unsolicited messages to thousand of users all over the world with approximately no cost. As a result, it is becoming common for all users around world wide to receive hundreds of spam messages daily. Spam messages are annoying to e-mail users as they waste time, money, and bandwidth (Carpinter and Ray 2006; CipherTrust Inc 2004).

There are several approaches which try to stop or reduce the huge amount of spam which are received by individuals. These approaches include legislative measures such as antispam laws and origin-based filters which are based on using network information and IP addresses in order to detect whether a message is a spam or not. The most common techniques are filtering techniques, attempting to identify whether a message is spam or not based on the content and other characteristics of the message.

In spite of the large number of methods and techniques available to combat spam, the volumes of spam on the internet are still rising. Nowadays; spam has the potential ability to become a serious problem for the internet community, antispam vendors offer a wide array of products designed to help us to keep spam out. They are implemented in various ways (software, hardware), several techniques (content, rule-based) and at various levels (server and user). The introduction of new technologies, such as Bayesian filtering, Support Vector Machines (SVM), Artificial Neural Network (ANN), Artificial Immune system (AIS)… etc. can improve the accuracy of filters. The implementation of machine learning algorithms is likely to represent the next step in the continuous fight against spam (Carpinter and Ray 2006). This work presents a new solution for spam inspired by Artificial Immune System Model (AIS). The Human Immune System has the capability to defeat against invaders such as bacteria, viruses… etc. which might attack the body. Artificial Immune System has been used successfully for several applications, our aim is to adapted the method for text classification. With help of Genetic Algorithm (GA) this work applies several modifications on standard Artificial Immune System in order to get more accurate results; it also includes a comparison study between genetic optimized spam detection using AIS and Artificial Neural Network for spam detection.

1.1 Problem statement

AIS is a new paradigm that can be classified as a knowledge based system technique which implements machine learning and develops its own library or knowledge base. In this paper, Artificial Immune System (AIS) is used in spam detection and Genetic Algorithm (GA) in optimizing the antispam Artificial Immune system. Few previous studies have used AIS in spam detection and the subject still needs further investigations. No previous work has tried to optimize the large number of parameters of the AIS, especially with important application such as spam detection. In this paper, several parameters are modified or optimized to get more accurate results, the advantages of the general optimization embedded in the genetic algorithm (GA) will be used in order to achieve an optimum AIS. The contribution of this work is very significant in the subject of artificial intelligence. Several analysis and comparisons will be made. Also, a comparative study will be done between genetic optimized spam detection using AIS and spam detection using Artificial Neural Network (ANN) with respect to spam detection. The research outline is as follows (See Fig. 1)

1.2 Contributions of this work

The contribution of this work can be summarized as follows:

1. Demonstrating the use of AIS on spam detection. This subject is still new and requires more applications, verification and testing.
2. Applying Artificial Neural Network in spam detection using SpamAssassin corpus. This corpus is rarely used with ANN.
3. Using (GA) to optimize AIS spam detection in:

   a. Determining when to perform culling (replacing old lymphocytes with new ones).
   b. Determining when to check if the self (legitimate) is changed (to fit the new interest of users).

4. Developing a new approach for learning in AIS that allow new adaptive immune system (lymphocyte) to take place instead of useless lymphocyte in innate immune system.
5. Applying the different techniques in spam detection and comparing and analyzing results.

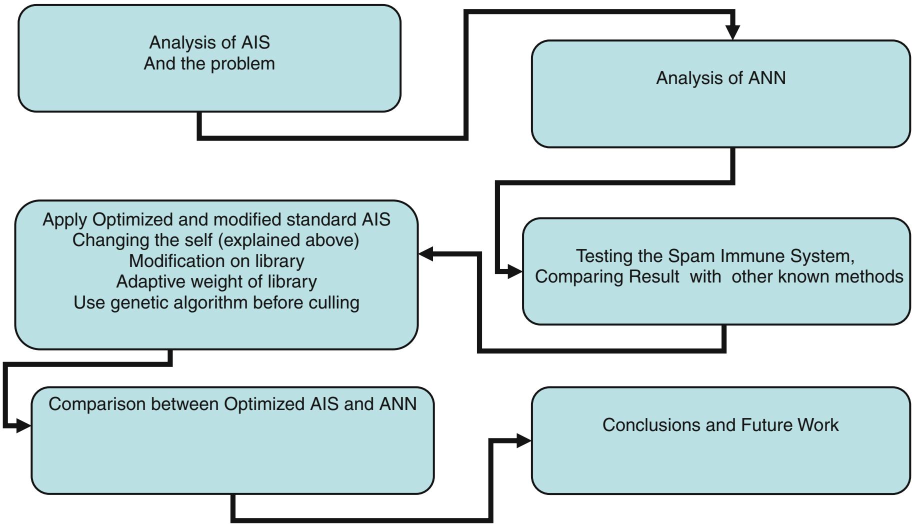

Analysis of AIS
And the problem
Analysis of ANN
Apply Optimized and modified standard AIS
Changing the self (explained above)
Modification on library
Adaptive weight of library
Use genetic algorithm before culling
Testing the Spam Immune System,
Comparing Result with other known methods
Comparison between Optimized AIS and ANN
Conclusions and Future Work

Fig. 1 Research outline in general

1.3 Structure of the paper

The remainder of this work is organized as follows:

Section 2 discusses the problem of spam and several approaches of combating spam, Part 3 discusses the machine learning method which will be used in this paper in details; spam detection using AIS and genetic Optimized spam detection using AIS. Extra details about these methods and how to adapt the artificial immune system and use it in spam detection can also be found.

Part 4 elaborates on spam detection using Artificial Neural Network (ANN). Part 5 presents the results of this thesis. In this chapter, you can find some details about the corpus used for training and testing besides some details about its characteristics. Part 6 presents the conclusion and the future work; this part shows that there is no sufficient approach which can be used alone to perform an excellent accuracy. So a combination of two or more approaches can lead to better results.

2 Impact of spam

The most dangerous threat on e-mail is spam so e-mails must be protected by individuals and organizations (CipherTrust Inc 2004). The problem of automatically filtering out spam e-mail using a classifier based on machine learning methods is of a great recent interest (Bart and Binargrl 2003). Controlling spam is a critical requirement for enterprises today. Also, spam can be sent through e-mails, newsgroups, mobile phones' Short Message Service (SMS) (Batista 2001).

Spam e-mail, also called unsolicited bulk e-mail or junk mail, is an internet mail that is sent to a group of recipients who have not requested it. These unsolicited mails have already caused many problems such as filling mailboxes, engulfing important personal e-mails, wasting network bandwidth, consuming users' time and crashing mail-servers, pornography adverts sent to children, and so on (Zhang et al. 2004). Junk mail was already an issue in 1975 when Joe Postel wrote a Request for Comments on the junk mail problem (Postel 1975).

Spam filtering in its simple form can be considered as a text categorization problem where the classes to be predicted are spam and legitimate. A variety of supervised machine-learning algorithms have been successfully applied to a mail filtering task (Zhang et al. 2004). A nonexhaustive list includes: naive Bayes classifiers (Androutsopoulos et al. 2000a; Sahami et al. 1998; Schneider 2003) support vector machines (Drucker et al. 1999; Kolcz and Alspector 2001) memory-based learning (Androutsopoulos et al. 2000b) AdaBoost (Carreras and Andm'Arquez 2001), and a maximum entropy model (Zhang and Yao 2003).

Spam filtering problem has been seen as a text classification problem because most e-mail contains some pattern of textual content. To compensate for the ever changing nature of spam it has been proposed that machine learning techniques can be used to rapidly adapt the statistical parameters over time (Medlock 2003).

Due to the effectiveness and relatively low development cost of text categorization techniques, they have become the dominant paradigm in building antispam filters. Most of these research approaches attempt to classify mail depending on interesting and uninteresting ones, on the basis of machine learning techniques (Cohen 2008; Kolcz and Alspector 2001; Brutlag and Meek 2000; Sahami et al. 1998; Drucker et al. 1999; Androutsopoulos et al. 2000c,d; Gee 2003), despite the fact that these techniques suffer from relatively lower accuracy ratings, in other words, they allow categorization of unsolicited mail as legitimate (Wiehes 2005). The challenge in spam detection is to find the appropriate threshold between spam and ham leading to the smallest number of misclassifications, especially of legitimate e-mail (false negatives). However, the problem still exists for one main reason; the effectiveness of any given antispam technique can be seriously compromised by the public revelation of the technique since spammers are aggressive and adaptable (Puniškis et al. 2006; Chhabra 2005; Graham-Cumming 2005).[1]

To get rid of spam there are many methods; social or legislation and technology are the two main tools being used to fight spam. Social or Legislation method tries to solve the solution from its root by discouraging spam senders (spammers) through social methods such as legal defenses. Technology solutions use technical means to make it more difficult for spammers to get their messages through to recipients.

The idea behind any spam filtering technique, (heuristic, probabilistic or keyword based) is fighting spam: usually spam messages seem to be different from legitimate messages, so a good way to identify and stop these spam messages is to detect these differences, the best way to solve this problem would be to use all of these features for a combined and more accurate effect (Goodman et al. 2007; Messaging Anti-Abuse Working Group 2006; Drakeand et al. 2004; Livingston 2001). Ever since the emergence of these technologies, spam-mers have improved their techniques, so that spam still gets to its destina-tion. Antispam solutions have to increase the frequency of the updates and also to develop more heuristics in less time (Saarinen 2003; Cook 2006; Gauthronet and Drouard 2001; Bekker 2003). The need for an automatic process that would quickly learn the characteristics of the new spam without affecting the accuracy of detection on less recent spam has become vital. This research tries to solve this problem (Atkins 2003; Alexandru and Researcher 2009).[2, 3, 4, 5, 6]

Within the technological solutions, there are several ways in which spam is classified:

Content Messages are classified on the bases of their contents (for example, do they contain given words or phrases?). The contents may include plain text, HTML or other enhanced text formats, images, documents, or other attachments. Plain text and HTML parts of the message are being focused upon by most of the systems mentioned here. Also, other attachments within the file could be analyzed. Mere existence of these contents is useful because it helps us to make a classification (Saarinen 2003).

Source Messages are classified on the bases of their source. Also the blacklists and whitelists are examples of classifiers based upon the message source (Medlock 2003; Saarinen 2003).

It must be noted that current electronic e-mail protocols have many problems. One of the problems is that it can be very difficult to ascertain that the stated source is in fact the actual source. It must be noted that some methods of filtering include ways to verify the source of a message to make a decision by checking if all parts of the address are valid. Other filtering techniques use more complex methods such as cryptographic identification. However, these methods are not directly relevant to this work. Automated Text Classification (ATC) is now a major research area within the information systems discipline for many reasons (Sebastiani 2002):

[1] http://en.wikipedia.org/wiki/Spam_(food)#Name_origin. Accessed 1 April 2008.

[2] www.ferris.com/hidden-pages/forms/free-report-information/?showfreeform=1&file_id_carrier=2004/05/611_409SpamCosts.pdf. Accessed 15 Feb 2009.

[3] www.informationweek.com/shared/printableArticle.jhtml?articleID=60403016. Accessed 20 Jan 2009.

[4] www.personneltoday.com/Articles/2005/03/09/28519/Spam+costs+UK+businesses+%C2%A313bn+year.htm. Accessed 1 Jan 2009.

[5] www.techworld.com/security/features/index.cfm?featureid=1372. Accessed 10 Jan 2009.

[6] www.e-mailsystems.com/news.php?itemid=219. Accessed 15 Jan 2009.

1. Its domains of application are various and important, and the proliferation of documents in digital form is increasing dramatically in both number and importance.
2. It is indispensable in many applications in which the sheer number of the documents to be classified and the short response time required by the application make the manual alternative improbable.
3. It can improve the productivity of human classifiers in applications in which no classification decision can be taken without a final human judgment.

This paper talks briefly about Legal measures, content filtering, but it will deal with only two methods of machine learning methods in details.

2.1 Legal measures

Self-protection and technical measures may be employed by users to fight spam, e-mail spamming is almost impossible to prevent. Thus, law is needed to regulate spam (Khong 2001). Unfortunately, regulating spam is not a simple matter. Spam regulation is caught in a web of diverging legal theories.

In the United States the discussion on regulation is more regulated and deep. Many related issues have been discussed in relation to spam. The outcome is that spam is not viewed simply as an intrusion of privacy like in Europe (Khong 2001).

To face the problem in the United States both legislative and judicial actions have been undertaken to tackle the spam problem. Before laws were enacted, common law doctrines of trespass and nuisance were first used against spammers (Khong 2001; Hawley 1997).

Eighteen states in the United States of America, have so far enacted laws to regulate spam, some of these states give rights to spam and others limiting the same. In order to harmonize theses conflicting state laws in the Congress many bills have been suggested. Unfortunately, none has successfully passed into law (Khong 2001).[7]

2.2 Spam detection and prevention techniques

To stop the arrival of spam or junk e-mail there are many techniques available. Generally, filters examine various parts of an e-mail message to determine if it is spam or not. On the bases of the parts of the e-mail messages, filtering systems can be further classified and used for spam detection. Origin or address-based filters typically use network information for spam classification, while content filters examine the actual contents of e-mail messages (Gulyás 2006; Wiehes 2005; Delany 2006; Weinberg 2005; Hershkop 2006).[8]

One of the real-world applications of Automated Text Categorization (ATC) task is spam filtering. Spam filtering is a field that has undergone an intensive research in recent years. The TC early approaches were depended on manually construct document classifiers with rules compiled by domain experts. But recent trends in the TC approaches have converted to build classifiers automatically by applying some machine-learning algorithms to a set of preclassified documents (training dataset). This is also called the statistical approach, in the sense that differences among documents are usually expressed statistically as the likelihood of certain events, rather than some heuristic rules written by human. This trend is reflected in the goal

[7] California Colorado, Connecticut Delaware, Florida Idaho, Iowa Illinois, Louisiana Missouri, Nevada North Carolina, Oklahoma Pennsylvania, Rhode Island Tennessee, Virginia, Washington, Virginia West (2009) An up-to-date summary of the state laws is provided by Professor Sorkin at his website, see http://www.spamlaws.com/state/. Accessed 20 Jan 2009.

[8] http://spam.abuse.net. Accessed 20 Jan 2009.

of statistical spam filtering, which aims at building effective spam filters automatically from e-mail corpus (Zhang et al. 2004; Sebastiani 2002).

Most of the techniques applied to the problem of spam are useful, and the key role among them which can reduce spam e-mails is the content-based filtering. But its success has forced spammers to periodically change their practices and behaviors in order to bypass these kinds of filters (María et al. 2006).

2.2.1 Origin-based filters

Origin based filters are methods which based on using network information in order to detect whether a message is a spam or not. IP and the e-mail addresses are the most important pieces of network information used. There are several major types of origin-Based filters such as Blacklists, Whitelists, and Challenge/Response systems (Saarinen 2003).

2.2.2 Blacklists

Blacklists, which known as Realtime Block Lists (RBLs) or Domain Name System Black Lists (DNSBL) are real time of lists of the IP addresses of machines or computers that send or relay spam. This method has the ability to detect spam letters based on its origin rather than its content (Gulyás 2006). The main idea of this method based on getting IP addresses of known spammers or suspected and then entering these addresses into database and made available through the internet. In order to check and see if any of these addresses of the received e-mails are listed. Filters can be used. These lists are accessed directly or provided in periodic to many of the internet ISPs, universities…etc. Blacklists are a popular successful method of blocking e-mail from known spammers but actually there are some disadvantages; there are a lot of opinions that reject the use of RBLs, an IP address of a non spammer can be on these lists and it will take a long time to get it removed. Also, the effectiveness of blacklists depend on the people who manage them, if the blacklists are not updated periodically, it means spam will rise (Cook 2006; Delany 2006). Also, Blacklist approach was quickly found to be inadequate since those spammers who are able to send messages from accounts which set up temporarily and used only once or twice (Medlock 2003).

2.2.3 Whitelists

Whitelists is the other face of blacklists; which means that instead of determining a list of "Un-Trusted" e-mail addresses. Whitelists allow users to specify or define a list of "Trusted" addresses. All messages received from those addresses will be classified as legitimate. The main advantage of this technique is that Whitelists would be smaller than Blacklists and easier to maintain (Cook 2006; Delany 2006).

However, there are many disadvantages with this approach: First, Whitelists limit the list of e-mail addresses, in other words, whitelists only accept mails from those who are authorized while any message from senders who are not on the list will be considered as spam. Second, the spammers who can guess an address on the Whitelists can easily send a spam message to that address. Third, Whitelist approaches were also found to be inadequate and not comprehensive due to their restrictiveness (Medlock 2003).

On the basis of all these drawbacks, Whitelist is successfully used only for classifying letters as legitimate (ham) mails, and has nothing to do when the sender is unknown. In case that blacklist and whitelist methods are used together, further filtering only is required for letters that do not match any of the entries in the two lists (Gulyás 2006).

Content filtering identifies spam on the bases of its content, while Whitelists require identifying users (source of the message). A Whitelist is a collection of trusted contacts. If an e-mail comes from the members of this trusted list, it can be marked automatically as a ham (legitimate) letter. Otherwise, it would be considered as spam. Just as the blacklist, the Whitelists also needs a continuous upgrade and refreshment (Gulyás 2006).

Rejecting all e-mails from unknown senders is too strict. A better method can be achieved by sending an auto-reply for every unknown user with an ask for authentication (challenge/response). This can reduce speed and rely on the sender's cooperation.

2.2.4 Challenge/response systems

Challenge/Response systems are an advance version of whitelists in order to avoid ignoring uncertain messages entirely, which means allowing senders who are not on the Whitelists to have received their messages. The main idea based on that is the incoming messages from addresses which are not on the whitelists required an automatic reply (Challenge) to the senders asking the sender to verify and prove that they are real users and not automated mail sender. If this process is continued and completed, then the e-mail will successfully pass the Challenge/Response System. The main advantage of Challenge/Response process is the ability to protect against the process of automated mail sending program by asking the user to respond to a task that is very simple for humans and too difficult for a program. Another main advantage of a Challenge/Response system is that it can protect against spammers who send e-mail manually (Cook 2006; Delany 2006).

2.2.5 Content filtering

While Origin-based filters such as Blacklists and Whitelists examine network information, e-mail headers are used to determine whether a message is spam or not. Content filters examine the message contents to determine whether it is spam or ham (legitimate). Content based filters try to read the text in order to examine its content. Filters which use this technique are called Keyword-Based Filters. There are several popular content filters such as Bayesian filters, Rule Based Filters, Support Vector Machines (SVM) (Cook 2006; Delany 2006).

Content-based filtering is a new technological method to fight spam and there are numerous learning methods which have been developed to implement content-based filtering. Most learning algorithms for spam classification include three main steps (Scavenger 2003):

1. A mechanism for extracting features from messages.
2. A mechanism for assigning weights to the extracted features.
3. A mechanism for combining weights of extracted features to determine whether the mail is spam or not.

Also, there are many articles and papers suggest ways to do content based spam filtering (Chhabra et al. 2004; Dalvi et al. 2004; Damiani et al. 2004; Deepak and Parameswaran 2005; O'Brien and Vogel 2003; Pelletier et al. 2004; Soonthornphisaj et al. 2002).

The automated categorization (or classification) of texts into predefined categories has witnessed a booming interest in the last 10 years, due to the increased availability of documents in digital form and the urgent need to organize them (Sebastiani 2002).

The two most common approaches to spam filtering are Naive Bayes (Mitchell 1997) and Support Vector Machines (Vapnik 1999). Therefore, a considerable number of evaluations using these approaches have been published.

2.2.6 Bayesian filters

The most well known commercial machine learning approach used in spam filtering is the use of Naive Bayes classifiers, Naive Bayes classifier is a probabilistic classifier. Briefly, it calculates and uses the probability of certain words/phrases occurring in the known examples (messages) in order to categorize new examples (messages). Naive Bayes has been shown to be very successful at categorizing text documents (Delany 2006).

Bayesian Filters (statistical method) filters work by analyzing the words of the message inside an e-mail to calculate the probability that the message is spam or not. The calculation based on words which determine that the message is spam and the words which determine that the message is not spam. Bayesian filters need to be trained on examples to be able to determine in future whether any message that arrives is spam or not. The problem with Bayesian filters, like many other content filters is that they require a complete message to be received to start calculations so as to determine whether it is spam or not. Thus, these calculations are exhaustive and require more processing (Cook 2006; Delany 2006).

Naive Bayes (NB) is a widely used classifier method in text categorization problem task which has enjoyed a blaze of popularity in antispam research (Androutsopoulos et al. 2000a; Sahami et al. 1998; Schneider 2003; Androutsopoulos et al. 2000b; Pantel and Lin 1998) and often serves as a baseline method for comparison with other approaches.

Naive Bayesian classifiers are one of the best classifiers for classifying text (Agrawal et al. 2000) for reasons of simplicity, efficiency, flexibility and updatability. One of the major drawbacks of naïve bayesian classifier is the unrealistic independence assumption among individual terms/phrases which is the main idea behind Naive Bayes classifiers. Regardless of this, they often produce satisfactory results. Bayes classifiers are probabilistic in nature because they not only perform a classification but also predict a degree of confidence in the class assigned. This is important when the decision for accepting or rejecting a predicted class must be made. For example, misclassifying can cause a problem; it may classify a legitimate message into a junk message which is much more expensive than classifying junk mail into a legitimate message. To avoid the more expensive case, the decision can be made to "trust" a classifier only if it assigns class "junk" with a very high probability (Itskevitch 2001).

2.2.7 Support Vector Machine SVM

Support Vector Machines (SVM) have had success in classifying text documents (Delany 2006; Joachims 1998; Dumais et al. 1998; Cardoso-Cachopo and Oliveira 2003). SVM has prompted a significant research into applying them to spam filtering (Delany 2006; Drucker et al. 1999; Kolcz and Alspector 2001). SVMs are kernel methods whose central idea is to embed the data representing the text documents into a vector space where linear algebra and geometry can be performed (Delany 2006; Cristiani and Scholkopf 2002). SVMs attempt to construct a linear separation between two classes in this vector space.

A Support Vector Machine (SVM) (Delany 2006; Vapnik 1999; Christianini and Shawe-Taylor 2000) is a linear maximal margin binary classifier. It can be interpreted as finding a hyperplane in a linearly separable feature space that separates the two classes with a maximum margin (Fig. 2). The instances closest to the hyperplane are known as the "support vectors" as they support the hyperplane on both sides of the margin.

SVM has been reported significant performance on text categorization problem with many relevant features (Zhang et al. 2004; Joachims 1998). SVM has also been applied to spam filtering task with excellent filtering accuracy and performance (Zhang et al. 2004; Drucker et al. 1999; Kolcz and Alspector 2001).

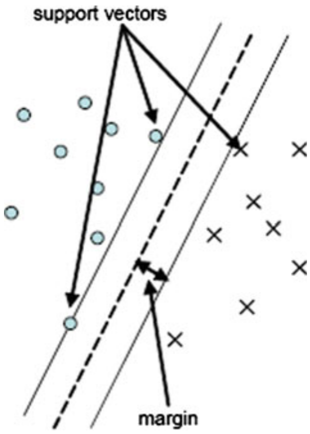


Fig. 2 Support vector machine classifier (*SVM*)

Support Vector Machines (SVM) (Drucker et al. 1999; María et al. 2006; Joachims 2001) is an optimization algorithm that produces (linear) vectors which try to maximally separate the target classes (spam versus legitimate). It is a complex and (relatively) recent algorithm, which has shown excellent results in text classification applications. However, it is difficult to interpret its output.

Rule-Based use a set of rules on the words included in the entire message (Header, Subject, and Body) to determine whether the message is spam or not. Rule based filters were the most common method for spam detection until 2002, when Bayesian filters became more popular (Graham 2003). The limitation of Rule-Based filtering is the rule set which is very large and static and causes less performance and adaptability; the spammers can easily defeat this filters by word obfuscation, for example the word Free could be modified to be F*R*E*E so it will pass the filters (Cook 2006; Delany 2006).

Rule-based systems filter spam based on patterns of keywords within a message's text. Unfortunately, spammers often obscure (obfuscate) these patterns by encoding letters as punctuation or numbers that are visually similar to human readers, but it is an easy method to defeat filters (Dimmock and Maddison 2004).

2.3 Artificial immune system (AIS)

Artificial immune systems (AIS) can be defined as computational systems inspired by theoretical biological immunology, observed immune functions, principles and mechanisms in order to solve problems. Their development and application domains follow those of soft computing paradigms such as artificial neural networks (ANN), Evolutionary Algorithms (EA) and fuzzy systems (FS) (Damiani et al. 2004; Drewes 2002; Graham 2009; Simon 1983; de Castro and Timmis 2002). Despite some isolated efforts, the field of AIS still lacks an adequate framework for design, interpretation and application so it still needs more investigation.

The field of Artificial Immune Systems (AIS) concerns the study and development of computationally interesting abstractions of the biological immune system (Garrett 2005). Artificial immune systems (AIS) (more detail in Sect. 3.2) use the concepts inspired by the theory of how the human Biological immune system works and react to infections. An immune system's main goal is to distinguish between self and potentially dangerous non-self elements. In a spam immune system, the aim is to distinguish between legitimate messages and spam. Like biological pathogens, spam comes in a variety of forms (patterns) and some pathogens will only be little variations (mutations) of others. Self and non-self can be considered as non-spam and spam.

By definition, a "neural network" is a collection of interconnected nodes or neurons. The best-known example of one is the human brain which is the most complex and sophisticated neural network (Timmis et al. 2004; Miller 2009). The term neural network has been moving a round a large class of models and learning methods. The main idea is to extract linear combinations of the inputs and derived features from input and then model the target as a nonlinear function of these features. Neural networks find applications in many different fields (Pascucci 2006).

Artificial Neural Network (ANN) (Sect. 4.2) is a large class of algorithms that have the capability of classification, regression and density estimation (Tretyakov 2004).

Neural network is composed of a complex set of functions that has the ability to be decomposed into smaller parts (neurons, processing unit) and represented graphically as a network of these neurons. There are two classical types of neural networks that are most often used when the term ANN is used; the perceptron and the multilayer perceptron. This work presents the perceptron algorithm since the problem of spam detection is almost linear.

Spam presents a unique challenge for traditional filtering technologies: both in terms of the overwhelming number of messages (millions of messages daily) and in the breadth of content (from pornographic to products and services, to finance). Add to that the fact that today's economic fabric depends on e-mail communications which are equally broad and plentiful and whose subject matter contextually overlaps with that of many spam messages. Consequently you are going to have got a serious challenge (Miller 2009).

The basic principal used in any spam filtering technique, whether heuristic or keyword-based, is identical: spam messages generally look different from good messages and detecting these differences is a good way to identify and stop spam. The difference between these technologies really comes down to the problem of distinguishing between these two classes of e-mail (Miller 2009).

Simply, the neural network approach attempts to simulate the way that humans visually recognize spam from non-spam e-mails. Without being exposed to every spam message created, most users know how to recognize spam from legitimate communications, even from other solicited bulk communications like newsletters. The reason for this is generally because our brains are exposed both consciously and subconsciously to a wide variety of message content, both good and bad, on a daily basis, and the brain learns to make a fast decision, highly accurate guesses as to what spam is and what it is not (Miller 2009).

Today, neural networks are important because they are used to solve a wide set of problems, some of these problems have been solved by existing statistical methods, while the others have not. These applications are categorized into one of the following three categories (Tretyakov 2004; Boulevard and Ramon 2008):

- Forecasting: predicting one or more quantitative outcomes from both quantitative and categorical input data.
- Classification: classifying input data into one of two or more categories.
- Statistical pattern recognition: uncovering patterns, typically spatial or temporal, among a set of variables.

In many cases, simple neural network configurations are like many traditional statistical applications give the same solution. For example, a single-layer, feedforward neural network with linear activation for its output perceptron, is equivalent to a general linear regression fit. Neural networks are different from traditional methods in that neural networks can provide more accurate and consistent solutions for problems while traditional methods do not completely apply (Boulevard and Ramon 2008).

2.4 Biological immune system

The interest in studying the immune system in the last decades is increasing; computer engineering, computer scientists, and researchers are interested in studying the capability of this System. The Immune System is composed of a complex set of cells, molecules and organs that have the capability of performing a lot of complex tasks (de Castro and Von Zuben 1999). The immune system is very important since without the immune system any diseases can affect the body of the humans in a serious manner.

The vertebrate immune system (IS) is one of the most complex bodily systems and its complexity is sometimes compared to that of the brain. There is an increase advance in the biology and molecular genetics, so the knowledge of how the immune system behaves is increasing very quickly (de Castro and Timmis 2002).

The advanced research in IS result in that the knowledge about the IS functioning has unraveled several of its main operative mechanisms. These operative mechanisms are very useful since they have demonstrated not only from a biological standpoint, but also under a computational perspective. The AIS immune system is similar to the way the nervous system inspired the development of artificial neural networks (ANN), the immune system has now led to the emergence of artificial immune systems (AIS) as a novel computational intelligence paradigm (de Castro and Timmis 2002).

To protect the body from invading pathogens (threats) there is a complex adaptive system that has embedded in vertebrates Biological immune system. To cover this task, the Biological immune system has evolved a complex pattern recognition and response mechanisms which follow many differential pathways. The response to this threat depends on the form of threat, how it get into the body and the damage it causes, the immune system uses different response mechanisms either to damage the threats or to neutralize its effects (Dasgupta 2006).

The existence of the biological immune system is to protect organisms from any potentially harmful threat (agents) such as bacteria, viruses, and any other strange life forms and substances (threats). These non-self dangerous agents are often called as pathogens. The immune system accomplishes this goal by carefully distinguishing the self (parts of the organism protected by this immune system) from non-self (anything else) (Oda 2003).

The immune system starts its work in defeating when the organs are infected; which leads Cells and molecules to start maintain surveillance for infected organs.

All living organisms have the capacity of presenting some types of defense against strange attack. The evolution of species that resulted in the emergence of the vertebrates also led to the evolution of the immune system of these species. The vertebrate immune system is particularly interesting due to its several computational capabilities (de Castro and Timmis 2002).

The task of defeating against foreign attack is accomplished using special detectors called lymphocytes. These lymphocytes are created in a random manner. After that, they are trained to remember infections so that the organism is protected from any future intrusions (threats) as well as past ones (Oda 2003).

The immune system is initially an appealing system for spam detection because of the classification of self and non-self messages. This classification needs to classify between the legitimate messages (the self) and spam (non-self) (Dasgupta 2006; Oda 2003; Secker et al. 2003; Yue et al. 2006).

Living organisms are capable of defeating invasive attackers. The immune system of vertebrate is composed of several of molecules, cells, and organs which are existed every where over the whole body. The main task of the immune system is to give the body the ability to mobilize its defenses. Every element that can be recognized by the immune system is called an antigen (Ag). The cells that belong to our body and are harmless to its function are termed self (self antigens), but any other cells which cause diseases are termed nonself (nonself antigens) (de Castro and Timmis 2002).

Microorganisms such as bacteria, viruses …etc. are classified as pathogens. The immune system is suffering from a main problem which is the ability to recognize (recognition) these pathogens. These pathogens can not be directly recognized by the component of the immune system. Only the small pieces of the pathogens called (Antigen) are recognized by the immune system. After this recognition a disease can be identified (de Castro 2003).

To accomplish the mission of correctly identify and eliminate a disease (such as bacteria, viruses …etc.), the immune system needs to define or recognize the body's own tissues which named as (Self Antigen). Note that the cells and molecules of the body's organisms are also called (Antigen). As a result a disease can be discovered if the immune systems can successed in distinguishing between self / Nonself discrimination (de Castro 2003).

2.4.1 The immune system related subsystem

There are two systems by which the body can identify foreign infections (Fig. 3): The Innate Immune System and the Adaptive Immune System (Janeway 1992, 1993; Fearon and Locksley 1996; Janeway and Travers 1997; Parish and O'Neill 1997; Carol and Prodeus 1998; Colaco 1998; Medzhitov and Janeway 1997a,b, 1998).

2.4.1.1 The innate immune system The Innate immune system (naturally available for combat) has the capability to recognize certain microbes and destroy them. The term innate means that the ability to recognize and respond to most threats (microbes) is born from the

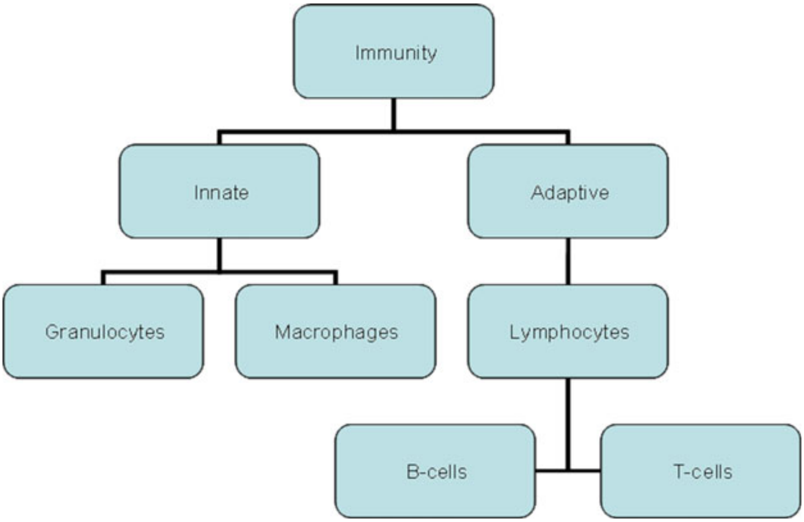


Fig. 3 The immune system de Castro (2001)

first moment of creation. The human immune system can respond to several pathogens of first appearances (de Castro and Von Zuben 1999).

Our innate immune system can destroy many pathogens on first appearance. The response of the innate immune response depends on an important factor which is a class of blood proteins known as complement, which has the ability to assist, or complement, the activity of antibodies (de Castro and Von Zuben 1999; Burnet 1959).

The innate immunity is based on a set of receptors demonstrated in the germinal centers and these receptors are known as Pattern Recognition Receptors (PRRs), the task of the receptors is to recognize molecular patterns associated with microbial pathogens. Microbial pathogens are called Pathogen Associated Molecular Patterns (PAMPs). The PAMPs are only produced by microbes and never by the host organism; this means that the recognition by the PRRs may result in signals indicating the existence of pathogenic agents. Hence, the immune recognition and its related structures must be absolutely distinct from our own cells and molecules, and this is very important to avoid any damage to tissues of the host. The significance of this mechanism is that the innate immunity is also able to distinguish between self and nonself, participate in the self/nonself discrimination issue, and plays an important role in the advance of adaptive immunity (Burnet 1978).

The innate immune recognition main important aspect is the fact that it induces the expression of co-stimulatory signals in Antigen Presenting Cells (APCs) that will lead to T cell activation (Sect. 2.5.6.2), which will result in the promoting of the start of the adaptive immune response. This way, adaptive immune recognition without innate immune recognition may result in the negative selection of lymphocytes that express receptors involved in adaptive recognition (de Castro and Von Zuben 1999; Burnet 1959).

2.4.1.2 The adaptive immune system The adaptive Immune System (production of antibody specified to determined disease) uses generated antigen receptors which are clonally distributed on the two types of lymphocytes: B cells and T cells (de Castro and Von Zuben 1999). (more detail in Sect. 2.5.6.2). The main merit of the adaptive immune system is its ability to enable the body to respond to any new microbe, even if the body was never previously infected this type of microbe.

The adaptive immune system uses somatically generated antigen receptors which are clonally distributed on the two types of lymphocytes: B cells and T cells. These antigen receptors are generated by random processes and, as a result, the general design of the adaptive immune response is based upon the clonal selection of lymphocytes expressing receptors with particular specificities (de Castro and Von Zuben 1999; Burnet 1959, 1978). The antibody molecules (Ab) play a leading role in the adaptive immune system. The receptors used in the adaptive immune response are formed by piecing together gene segments. Each cell uses the available pieces differently to make a unique receptor, enabling the cells to collectively recognize the infectious organisms confronted during a lifetime (Tonegawa 1983). Adaptive immunity enables the body to recognize and respond to any microbe, even if it has never faced the invader before (Tonegawa 1983).

2.4.2 Biological immune system history

Immunology is a rather new science. The scientist Edward Jenner is the first who addressed its origin. He discovered also that the vaccinia, or cowpox, induced protection against human smallpox, a frequently lethal disease, approximately 200 years ago, in 1796 (Tonegawa 1983; de Castro and Von Zuben 2009). Jenner baptized his process vaccination, an expression that

Table 1 Summary of the main ideas and research in the immunology field

Aims	Period	Pioneers	Notions
Application	1796–1870	Jenner Koch	Immunization pathology
	1870–1890	Pasteur Metchininkoff	Immunization phagocytosis
Description	1890–1910	Von Behring & Kitasato Ehrlich	Antibodies cell receptors
	1910–1930	Bordet Landsteiner	Specificity Haptens
Mechanisms (system)	1930–1950	Breinl & Haurowitz Linus Pauling	Antibody synthesis antigen template
	1950–1980	Burnet Niels Jerne	Clonal selection network and cooperation
Molecular	1980–1990	Susumu Tonegawa	Structure and diversity of receptors

still describes the inoculation of healthy individuals with weakened or attenuated samples of agents that cause diseases and aiming at obtaining protection against these diseases.

In the last few years (Table 1), most of the work in immunology is focusing on: apoptosis, antigen presentation, cytokines, immune regulation, memory, autoimmune diseases, DNA vaccines, intracellular and intercellular signaling, and maturation of the immune response (de Castro and Von Zuben 1999).

2.4.3 Immune system fundamentals

The immune system is a great gift from God that all living beings are endowed with this gift. The immune system is very complex and its complexity varies according to its characteristics. For example, some plants have protective spines to provide protection from predators that attack them. Animals have bones (vertebrates) which contain a developed and a highly effective and complex immune system. It is consisted of a vast array of cells, molecules and organs that work together to maintain and keep life. The focus here will be on the immune system of vertebrates, more specifically of humans. This is because of its interesting features, characteristics from a biological and computational perspective. The wide knowledge available about its implementation and its broad applicability in the design of AIS (de Castro and Timmis 2002; de Castro 2003).

The immune system performs several functions. One of its main functions is that the immune system together with other bodily systems it maintains a constant state of our essential functions, named as homeostasis. One of its most amazing roles however is the protection of the organism against any foreign attack of disease which may cause agents, called pathogens, and the exclusion of malfunctioning cells (de Castro 2003).

2.4.4 Immune system pattern recognition

Biological immune system has a pattern recognition which can fundamentally arise at the molecular level. The surface receptors of T-cells and B-cells present a certain “shape” (or match) that has to be matched with the shape of an antigen (pathogen). There are other features that are involved in this recognition of an antigen by a cell receptor, but this is outside of this thesis.

The T-Cell Receptor is called TCR and the B-Cell Receptor is called BCR or antibody (Ab). Both B-cells and T-cells present surface receptors for antigens. The common feature between B-cells and T-cells is that they present surface receptors for antigens. However, they differ in the basic structures of the receptors (antibodies and TCRs) and the types of antigens that each one is able to recognize. While antibodies can identify and bind with antigens which are free in solution, TCRs can only identify and bind with antigens presented by molecules of our own body, known as Major Histocompatibility Complex (MHC) (de Castro 2003; de Castro and Von Zuben 2009).

It is essential to know that the recognition in the immune system depends on the shape of complementarity. Antigens and cell receptors have to have complementary shapes in order to bind together. It is the binding together of the receptor with the antigens that trigger an *immune response*, the reaction of the immune system against the pathogen that displays the antigen recognition (de Castro 2003).

Similar to the use of artificial neural networks, performing pattern recognition, AIS usually contains three stages (de Castro and Timmis 2002):

1. Defining a representation for the patterns.
2. Adapting (learning or evolving) the system to identify a set of typical data.
3. Applying the system to recognize a set of new patterns (that might contain patterns used in the adaptive phase).

The way of Learning is usually addressed to the processes of acquiring knowledge from experience and abstracting this knowledge to solve new, previously unseen problems. The process of immunizing (through vaccination, for example) is an obvious example of an immune learning mechanism. Similar strategies can be used to solve problems like pattern recognition, concept learning, etc. (de Castro and Von Zuben 2009).

Pattern recognition is an important area in new research directions. Pattern recognition is a research area that studies the function and design of systems that recognize patterns in data. It encloses sub-systems like discriminate analysis, feature extraction, error estimation, cluster analysis, grammatical inference and parsing (sometimes called syntactical pattern recognition). Important application areas are image analysis, character recognition, speech analysis, man and machine diagnosis, person identification and industrial inspection (de Castro and Von Zuben 2009).

2.4.5 The immune system structure

The generation and development of immune cells is the responsibility of two organs;*bone marrow* and the *thymus*. The bone marrow is the place where all blood cells are generated and where some of these cells are developed. The thymus is the organ to which a class of immune cells named T-cells migrates and maturates (de Castro 2003).

There are also many types of immune cells, but in this work this work will focus and concentrate on the Lymphocytes. Lymphocytes are white blood cells whose main goal is the recognition of pathogens. There are two main types of lymphocytes: B-cells and T-cells, both originated in the bone marrow. Those lymphocytes developed within the bone marrow are named B-cells, and those that migrate to and develop within the thymus are named T-cells. Both these types of cells present receptor molecules on their surfaces which are responsible for recognizing the antigenic patterns displayed by pathogens or some of their parts (Tonegawa 1983).

The immune system is composed of tissues and organs that are distributed all over the body. They are known as lymphoid organs, since they are related to the production, growing and development of lymphocytes and the leukocytes that compose the main operative part of the immune system. It must be noted that in the lymphoid organs, the lymphocytes interact

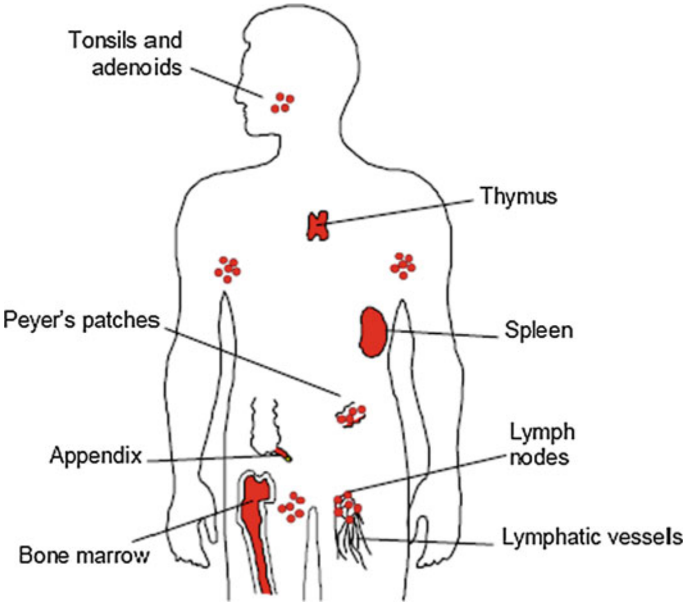


Fig. 4 Immune system structures de Castro and Von Zuben (1999)

with significant non-lymphoid cells, this interaction happens either during their maturation process or during the establish of the immune response. The lymphoid organs are divided into primary (or central) and secondary (or peripheral). The primary is responsible for the production of new lymphocytes, while the secondary is the place where the lymphocyte repertoires meet the antigenic universe (de Castro and Von Zuben 1999).

The lymphoid organs, and their main functions, include (Fig. 4) (de Castro and Von Zuben 1999).

- **Tonsils and adenoids** specialized lymph nodes contain immune cells that protect the body against invaders of the respiratory system;
- **Lymphatic vessels** constitute a network of channels that transport the lymph (fluid that carries lymphatic cells and exogenous antigens) to the immune organs and blood;
- **Bone marrow** a soft tissue contained in the inside part of the longest bones, and is responsible for the generation of the immune cells;
- **Lymph nodes** act as convergence sites of the lymphatic vessels, where each node stores immune cells, including B and T cells (site where the adaptive immune response takes place);
- **Thymus** a few cells migrate into the thymus, from the bone marrow, where they multiply and mature, transforming themselves into T cells, capable of producing an immune response.
- **Spleen** the site where the leukocytes destroy the organisms that invaded the blood stream;
- **Appendix and Peyer's patches** specialized lymph nodes contain the immune cells destined to protect the digestive system.

This paper is not going to focus on all of these lymphoid organs, but what is necessary for this study will be demonstrated here.

2.4.6 The immune system cell structure

The structure of the immune system is composed of a great diversity of cells that are created in the bone marrow, where abundance of them mature. After that, they start migrating to patrolling tissues, circulating in the blood and lymphatic vessels. Some of these cells are responsible for the general defense, whereas other cells are "trained" to combat specific pathogens. For an efficient functioning, it is necessary to have a continuous cooperation among the agents (cells) (de Castro and Von Zuben 1999).

2.4.6.1 The lymphocytes Lymphocytes are small leukocytes that have a major responsibility in the immune system. A lymphocyte consists of two main types: B lymphocyte (or B cells), which once are activated, they are differentiated into plasmocyte (or plasma cells) which are capable of secreting antibodies; and T lymphocyte (or T cell). Small resting cells are responsible for forming most of the lymphocytes. Also, these small resting cells are the ones which only exhibit functional activities after some kind of interaction with the respective antigens, necessary for proliferation a specific activation. The B and T lymphocytes state, on their surfaces, receptors highly specific for a given antigenic determinant. The B cell receptor is a form of the antibody molecule bound to the membrane, and which will be secreted after the cell is appropriately activated (de Castro and Von Zuben 1999).

2.4.6.2 B cells and antibodies The B cells main functions are the production and secretion of antibodies (Ab) as a response to exogenous proteins like bacteria, viruses and tumor cells. Each B cell is planned to produce a specific antibody. These antibodies are specific proteins that identify and bind to another particular proteins. The production and binding of antibodies is usually a way of signaling other cells to kill, swallow or remove the bound substance (de Castro and Von Zuben 1999).

2.5 Literature review for spam detection

This section talks about several previous studies which talk about spam detection and preventing in general, There are several research which talk about combating spam using machine learning method, but there is a few studies which talk about using artificial immune system in spam detection.

Delany (2006) in her PhD thesis talked about Using Case-Based Reasoning for spam Filtering. In her thesis she presented E-mail Classification Using Examples (ECUE). Her contribution in the PhD thesis was a content based approach to spam filtering that can handle the concept drift inherent in spam e-mail. She used the machine learning method of case-based reasoning which models the e-mails as cases in a knowledge-base or case-base. Her approach used in ECUE involves two components; a case-base editing stage and a case-base update policy. She presented a new method for case-base editing named Competence-Based Editing which uses the competence properties of the cases in the case-base to determine which cases are harmful to the predictive power of the case-base and should be removed. The update policy allows new examples of spam and legitimate e-mails to be added to the case-base as they are encountered allowing ECUE to track the concept drift.

She made a comparison between a case-based approach and an ensemble approach. The ensemble approach is a more standard technique for handling concept drift and presented a prototype e-mail filtering application that demonstrated how the ECUE approach to spam filtering can handle the concept drift.

The main core of Case-Base Reasoning (CBR) is a problem solving method that solved new problems by re-using or adapting solutions that were used to solve similar previous problems in the past (Brutlag and Meek 2000). The previous problems or past experience are determined as cases, each case contains several feature characteristics of the problem and its solution. A collection of these cases and their characteristic, known as the case-base, is the knowledge base of experience that will be used to solve new emergent problems.

CBR strong point is that it is an alternative problem solving approach that can work well in domains that are not well understood. CBR helps decision making based on what worked in the past without modeling past decisions in detail.

CBR can be represented as a cyclical process that is divided into the four following sub processes (Sahami et al. 1998).

(i) Retrieve the most similar case or cases from the case base.
(ii) Reuse the case to solve the problem.
(iii) Revise the proposed solution, if necessary.
(iv) Retain the solution for future problem solving.

Hershkop (2006) in his PhD thesis presented an implemented framework for data mining behavior models from e-mail data. The E-mail Mining Toolkit (EMT) was a data mining toolkit designed to analyze offline e-mail corpora, including the entire set of e-mail sent and received by an individual user, revealing much information about individual users as well as the behavior of groups of users in an organization. The EMT contained a number of machine learning and difference detection algorithms. These are embedded in the EMT to model the user's e-mail behavior in order to classify e-mail for a variety of tasks. The work has been successfully applied to the tasks of clustering and classification of similar e-mails, spam detection, and forensic analysis to reveal information about user's behavior.

Hershkop, in his thesis, showed that advanced data mining and machine learning techniques implemented for e-mail analysis can supply the infrastructure enabling a new generation of applications which help in solving many of these problems.

Hershkop showed that a user's e-mail archive can be used to model the behavior of a user to protect their e-mail account from misuse and abuse. He talked about an implemented system for mining e-mail data to build a various pool of models based on behavior. He presented his results in the domain of spam detection. To show the general utility of the approach, he also described in his thesis the use of the same methods in the framework of a forensic investigator dealing with large amounts of unclassified and or unknown e-mails.

Hershkop thesis showed that unwanted messages can be detected by taking in consideration the behavior of the e-mail user. He has investigated several particular model combination algorithms and plotted their performance using a number of supervised machine learning classifiers.

In summary the results of Hershkop experiments have shown how combination algorithms can be directly applied to spam classifiers, and used to improve both false positive and detection rates on either strong, weak, or a combination of these classifiers.

Weinberg (2005) in his Master thesis showed that the spam problem is a complex problem, and to deal with this complex problem different strategies should be developed. Such strategies must contain both technical and legal realities simultaneously, in order to be successful. In his thesis, he built a model of the system surrounding the spam problem in the form of a casual loop diagram. His diagram shows the casual interactions between the various technical, legal, social, and economic forces presented in the spam problem system. Based on his diagram, he identified a number of places that solutions could interact with this system. These places comprise a set of possible levers that could be pulled to lighten the spam problem.

This set of levers developed in his thesis used to make sense of the attempted and suggested solutions to date. Various solutions are grouped by how they interact with the system. These solutions categories are then presented in detail by showing, diagrammatically, how they positively and negatively affect the spam system through their interactions with it.

Gulyás (2006) in his Master thesis constructed a meta spam filter utilizing several spam filters at the same time, the meta filter is constructed as a Bayesian network. His method is considered content spam filtering; the bayesian network-based spam filter uses the content of the e-mail. The main phases of the Bayesian network-based solution are the following: First of all there is a need to tokenize the e-mail, which means to separate it into small parts that are used further in the process. These tokens can be sentences, or word-pairs, but usually single words are used to define tokens. After that step the value of every token is determined by looking up in an updated table, what we call the token-dictionary. In this table one or more values are stored for each token. After getting the value of each token, there is a way to calculate a probability of the e-mail being spam or legitimate.

Most implementations do not deal with all the values, usually in order to save processing time they calculate only with the most relevant values. The values of the relevant tokens have the largest distance from the neutral value and so they are close to be an obvious mark of spam or legitimate e-mails. These values are used to establish the so called decision matrix. Most Bayesian filters limit the number of tokens in the decision matrix, usually to 15 or 27 of the most interesting tokens. The final step is to modify the values of the tokens in the dictionary, this gives the continuous learning capability with the feedback; and the final binary result is produced.

Oda (2003), Oda and White (2003, 2005) in their research they used Artificial Immune System (AIS) model to defend e-mail users efficiently from spam messages. They tested their spam immune system using the publicly available SpamAssassin corpus of spam and ham, and enhanced the original spam system by looking at several methods of classifying e-mail messages based on the detectors produced by the biological immune system. The resulting spam immune system classifies the messages with similar accuracy to other machine learning spam filters.

The main idea in spam immune system is that it creates digital antibodies which are used as e-mail classifiers. Then, as the system is expected to read messages, it learns about the user's classification of spam and ham. Once the system is trained, the resulting can achieve classification result as good as to those of existing commercial antispam products.

In a simple example, a spam message could use the string "enl4rge". instead of "enlarge" to avoid filters checking (anti spam) for that word but not for variants upon it. By using regular expression antibodies, the system can assign an identical weight to many possible strings. (These weights are used later to determine a spam score for each message.) "Enlarge" and "enl4rge" and "enlarg3" are all read the same way by the human recipient of a message, so it makes sense to allow the immune system to treat them as the same string.

The weighted average seems to provide a better balance. With the threshold set at 0.7, the immune system correctly classifies 90% of the messages correctly. (More specifically, it classifies 84% of spam and 98% of non-spam).

Secker et al. (2003) presented an immune-inspired algorithm called AISEC (Artificial Immune System for E-mail Classification), this system is able to continuously classify electronic mail as interesting and non-interesting without the need for re-training. In his research he made comparisons with a naïve Bayesian classifier and it shows that the proposed system performs as well as the naïve Bayesian system and has a great potential for augmentation.

AISEC seeks to categorize unknown e-mail into one of two classes (spam and ham) depending on previous experience. It does this by manipulating the populations of two sets of artificial immune cells. Each immune cell captures a number of features and behaviors from natural B-cells and T-cells but for simplicity we refer to these as B-cells throughout.

The natural biological immune system is based on a set of immune cells called lymphocytes consisting of B and T-cells. It is the manipulation of populations of these by various processes which give the system its dynamic nature.

In his method once the algorithm has been trained each B-cell will represents an example of an uninteresting (spam) e-mail by containing words from that e-mail's subject and sender fields in its feature vector. Any new e-mails so as to be classified they are considered to be antigens and so to classify an e-mail, it is first processed into the same format of feature vector as a B-cell and then presented to all B-cells in the algorithm. If the affinity between the antigen and any B-cell is higher than a threshold, the B-cell is said to recognize the antigen and thus classified as uninteresting (spam).

The ultimate goal of his work is to develop a web mining system to return web pages based on a measure of interestingness. The representation used by Secker, is also word based. It is not considered as a resistance to the letter-level obfuscation because the exact matching is used.

Yue et al. (2006) in his research presented a novel behavior-based antispam technology for e-mail service based on an artificial biological immune system-inspired clustering algorithm. His method can constantly deliver the most relevant spam e-mails from the collection of all spam e-mails. In his paper two main concepts were introduced, the first concept defines the behavior-based characteristics of spam while the second is continuously identifying the similar groups of spam. His method based on giving a "score" that can be used as an input to the developed clusters and to use theses scores to identify how likely they are. In his paper, he presents how to define the behavior-based characteristics of spam and how to dynamically identify the similar groups of spam while processing the data streams. Comparing to other known approaches, he proposes a new incremental immune-inspired clustering approach. Data set used in April 2003 standard testing corpus (Mason 2004) as the incoming data streams for his experiments, the set of 1,400 message.

Slavisa Sarafijanovic's research (2007), she designed an antispam system using analogies to the working of the human immune system. The system consists of "Adaptive" part, this part used for collaborative content processing to discover spam e-mail patterns. The main advantage of collaborative spam filtering enables is that the detection of not-previously seen spam content. The system enables local processing of the signature created from the e-mails prior to the deciding whether and which of the generated signatures will be exchanged with other collaborating anti spam systems. The main idea here is to enable only good quality and effective information to be exchanged among the collaborative anti spam system. This system also enables the demonstration of the e-mail content, based on a sampling of text strings of a predefined length and at random positions within the e-mails, and the use of a custom similarity hashing of these strings.

Abi-Haidar and Rocha (2008) in his paper presented a new solution to spam problem detection inspired by the model of the adaptive immune system known as the cross-regulation model. In his paper he showed that the cross-regulation model is hopeful as a bio-inspired algorithm for spam detection in particular and binary classification in general. The cross-regulation model, presented by Carneiro et al. (2007), aims to model the process of distinguishing between harmless and harmful antigen—typically harmless self/non self and harmful non self. He adopted the cross-regulation algorithm for spam detection, which he named the Immune Cross-Regulation Model (ICRM); also he presented an evidence for that his bio-inspired model is relevant for understanding immune regulation itself.

Liu and Zhang (2006) demonstrated a new behavior-based antispam method based on incremental immune-inspired clustering algorithm. He used an "internal image" network to represent the input data set in order to reduce data redundancy. In his research, he presented an incremental clustering algorithm based on artificial immune network which has the capability of constantly identifying similar groups of spam. Experiments on data used for evaluation show that the novel approach provides significantly faster data summarization than completely re-clustering. Also, the technology is reliable, efficient and scalable. Since it's known that no single technology can achieve one hundred percent spam detection with zero false positives, which means any effective method should be used in conjunction with other filtering systems to minimize errors.

Khorsi (2007) summarized most of the techniques used to filter spam e-mails by analyzing the content of the messages. He talked about (Bayesian classifier, K-nearest neighbors, technique of Support Vector Machine SVM, Neural Network, Maximum Entropy, Technique of Search Engines, Genetic Programming, and Artificial Immune System). As a result, he said that there is no technique can be claimed alone to be an ideal solution with 0% False Positive and 0% False Negative. Most of current anti spam systems couple several machine learning techniques for content classification.

Scavenger (2003) in his master thesis had devised a machine learning algorithm. In his machine learning algorithm features are formed from individual sentences in the subject and body of a message by forming all possible word-pairings from a sentence. Then, Weights are assigned to the features based on the strength of their predictive capabilities for spam/legitimate determination. In his method; the predictive capabilities are estimated by the frequency of occurrence of the feature in spam/legitimate collections and by the application of heuristic rules. During classification, total spam and legitimate indication in the message are calculated by summing up the weights of the extracted features of each class, and then the message is classified into spam or ham depending on the result. He compared the algorithm against the popular naïve-bayes algorithm and found its performance exceeded that of naïve-bayes algorithm both in terms of catching spam and for reducing false positives.

Itskevitch (2001) in her master thesis she demonstrated the problem of term dependence by building an associative classifier called Classification using Cohesion and Multiple Association Rules (COMAR). The main advantages of the COMAR classifier is using multiple association rules to classify each new case and employing deep rule pruning that results in much lower running time. The studies show that the hierarchical associative classifier that utilizes phrases, multiple rules and deep rule pruning and uses biased confidence or rule cohesion for rule ranking achieve higher accuracy and is more efficient than other associative classifiers and is also more accurate than Naive Bayes.

Medlock (2003) in his master thesis presented a generative classification model for structured documents based on statistical language modeling theory, as well as introducing two variants of a new discounting technique for higher-order N-gram Language Model (LM's). He applies the model to the spam filtering domain using a new e-mail corpus assembled for his work and report promising results. He also presents the best results achieved to date on the LingSpam e-mail corpus.

Çiltik (2006) in his master thesis proposed spam e-mail filtering methods having high accuracies and low time complexities. The methods are based on the n-gram approach and a heuristics referred to as the first n-words heuristics. the main concern of the research is studying the applicability of these methods on Turkish and English e-mails. A data set for both languages was compiled. Tests were performed with different parameters. Success rates above 95% for Turkish e-mails and around 98% for English e-mails were obtained. In addition, it has been shown that the time complexities can be reduced significantly without sacrificing from success.

Zhou et al. (2007) in his paper "Transductive Link Spam Detection" wrote about linkage information in web search. Linkage information is widely used in web search, link-based spamming has also developed. So far, many techniques have been proposed to detect link spam. Those approaches are basically built on link-based web ranking methods. In contrast, they cast the link spam detection problem into a machine learning problem of classification on directed graphs. They developed discrete analysis on directed graphs, and construct a discrete analogue of classical regularization theory via discrete analysis. A classification algorithm for directed graphs is then derived from the discrete regularization. They had applied the approach to real-world link spam detection problems, and encouraging results have been obtained.

Clark et al. (2003) in his paper presented a neural network based system for automated e-mail classification. Also, He presented LINGER. Linger is a NN-based system used for automatic e-mail categorization problem. Although LINGER was tested in the domain of e-mail classification, Linger is a generic architecture for all kinds ofText Categorization (TC). It is flexible, adaptable and uses configurable options for most of its operation. It consists of two main modules: preprocessing and classification. Preprocessing**:** In this step, the first step, the e-mail message received by the system is converted into a vector that stores 'keywords' or 'stop words' after they are extracted from the main document. Then, the system apply words weight, each word of the list is then weighted by incorporating several weighting schemes such as binary, term frequency, term frequency inverse document frequency. Finally, weight normalization is performed on these keywords. Classification**:** The second step, the classifier is trained using the back propagation algorithm and is a fully connected multilayer perceptron. This classifier is trained for every inbox depending on the preference of the user. Customized spam prevention is thus achieved. We have shown that NNs can be successfully used for automated e-mail filing into mailboxes and spam mail filtering.

Vinther (2002) in his paper presented his look at the actual words in the text, furthermore, to some extent, the structure of the mail. In this method, the final decision of the message to be considered as spam or ham is based on a probability calculated during message processing. Training and updating of the rules can be done automatically with a little help from the user. His theory takes in consideration which words are used in the message "body", "from:", "to:" and "subject:" fields, so it would be possible to distinguish the "junk" mail from the "Legitimate" mail. The input in this method of the neural network is a list of words presented in the e-mail. Furthermore, to limit the huge size of the network, a vocabulary of just a few of hundreds of words is built based on statistics from training both set of good and junk mails. Taking in consideration that we must build a group of lists where each list must contain the words of a category and its probability. The first step to do that is to build a list for each category containing all words found and the probability of finding that word in a mail from that category. The next step is to select the words from the two lists, which should make up the vocabulary of the network. He did that by creating a combined list with absolute differences in probability: For all words, w, in both lists, the difference was computed. If a word is only found in one category, the probability of that word being in a mail from the other category was set to zero.

$$dw = |pw_{junk} - pw_{good}| \tag{1}$$

The new created list is sorted by dw, and e.g. 400 words with the largest values are selected as the vocabulary. The advantage of this approach is that the vocabulary will consist of words, which are both very common in at least one of the categories, and which imply what category the message belongs to. One could also just have selected a number of the most common words from both lists, but that would include a lot words which are just as common in good mails as in junk mails, and hence does not tell anything about what category the mail belongs to.

The neural network used is a standard non-linear feed-forward network with the sigmoid activation function at both the hidden and output neurons.

Ozgur et al. (2004) proposed antispam filtering methods for agglutinative languages in general and for Turkish in particular. His methods are dynamic and based on Artificial Neural Networks (ANN) and Bayesian Networks. The developed algorithms used by Levent are user-specific and adjust themselves with the characteristics of the incoming e-mails. The algorithms have two main components. The first component deals with the morphology of the words while the second component classifies the e-mails by using the roots of the words extracted by the morphological analysis. There are two ANN structures, the single layer perceptron and the multi-layer perceptron, and the inputs to the networks are determined by using the binary model and probabilistic model. Similarly, for Bayesian classification, three different approaches are employed: the binary model, the probabilistic model, and the advanced probabilistic model. In the experiments, a total of 750 e-mails (410 spam and 340 normal) were used and a success rate of about 90% was achieved.

In order to determine the root words that will serve as the features in the classification process, a mutual information concept was used. The feature vector can be defined as a group of critical words that are used in classification of e-mails as spam or Legitimate.

Stuart et al. (2004) in his research used a neural network approach on a corpus of e-mail messages from one user. The feature set used to define spam messages is descriptive characteristics of words and messages similar to those that a human reader would use to identify spam. The project used a corpus of 1654 e-mails received by one of the authors over a period of several months. He notes that the NN required fewer features to achieve results similar to the Naïve Bayesian Approach.

Puniškis[9] in his research applied neural network approach to the classification of spam; his method employs the attributes composed of descriptive characteristics of the evasive patterns that spammers employ rather than the context or frequency of keywords in the message. The data used is corpus of 2,788 legitimate and 1,812 spam e-mails received over a period of several months. The result shows that ANN is good and ANN is not suitable for use as alone as a spam filtering tool.

Chuan et al. (2004) presented an anti spam e-mail filter based-LVQ (Learning Vector Quantization (LVQ)) networks, this method combines subclasses into a single class and forms complex class boundaries, to design an anti spam e-mail neural network model and identify spam e-mails which are mainly composed of commercial and political e-mails. LVQ network is a hybrid network, which forms classification through supervise and unsupervised learning. LVQ Model is divided into two layers. The first layer is competitive layer, in which each neuron represents a subclass, while the second layer is the output layer, in which each neuron represents a class. The project makes use of e-mail corpus SpamAssassin. He selects 1,000 pieces e-mails randomly from the corpus, including 580 spam and 420 legitimate. The result proved that the filter is superior to Bayes-based.

[9] Research Report, www.messagelabs.com. Accessed 10 Jan 2008.

Clark (2000) in his PhD thesis "E-mail Classification: A hybrid Approach Combining genetic Algorithm with Neural Networks". Linger is A Neural & Genetic E-mail Reader, the name of the intelligent e-mail classification program in this thesis. It is a recursive acronym; it stands for **L**inger **I**s a **N**eural & **G**enetic **E-mail R**eader. In his thesis, LINGER has two main machine learning algorithms embedded in it; a genetic algorithm and a neural network. LINGER machine learning uses a hybrid approach to classification. GA is used to search for a class of features that would be better for the NN to learn from. The benefit of combining the GA with the NN in this way is to balance the workload placed on the NN. The GA is used for the feature selection, and will select words which give the NN the best information about the different classes being used. This should make the NN faster in training, as NN training time is heavily depending on the dimensionality of the inputs. It may also assist the accuracy of the NNs predictions, having eliminated the less useful words from the input vector. The average accuracy is 71.52% and has a standard deviation of 1.94% .

3 Spam detection using AIS, versus spam detection using genetic optimized AIS

There are several machine learning approaches currently available in spam detection (Bayesian classification, Support Vector Machines SVMs) (Tretyakov 2004), Digest-based filters (de Castro and Von Zuben 1999), Density-based filters (Yoshida et al. 2004), Chi-squared filters (O'Brien and Vogel 2003), global collaboration filters (Hulten et al. 2004) ....etc. however, this part shows a detailed explanation of the Machine learning methods which will be used in this thesis; spam detection using AIS, genetic optimized spam detection using AIS. Results and experimentations are shown and analyzed in part 5.

3.1 Spam Detection using AIS

During the last decade, Artificial Immune System (AIS) field has witnessed a slow and a steady progress as a branch of Computational Intelligence (CI) as shown in Fig. 5 (Dasgupta

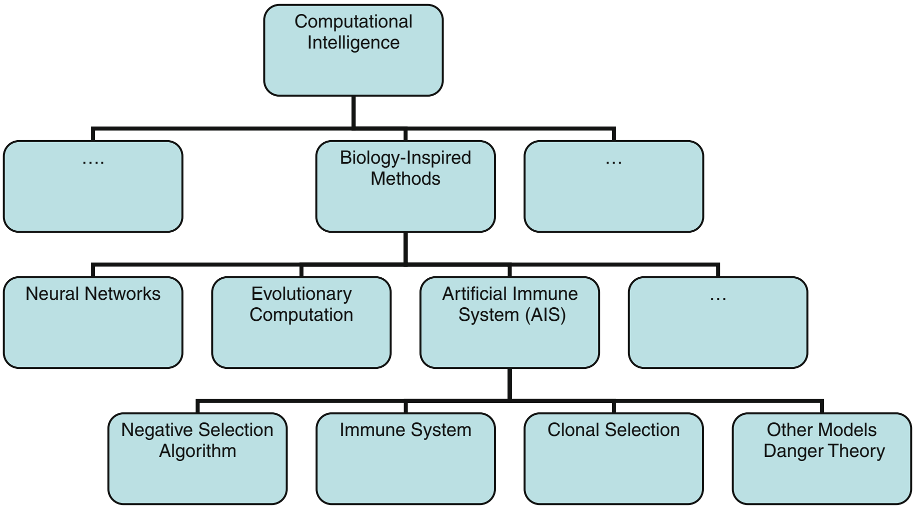

Computational Intelligence
....
Biology-Inspired Methods
...
Neural Networks
Evolutionary Computation
Artificial Immune System (AIS)
...
Negative Selection Algorithm
Immune System
Clonal Selection
Other Models Danger Theory

Fig. 5 Artificial immune system (*AIS*) as a branch of computational intelligence

2006). There has been an increasing interest in the development of computational models inspired by several immunological principles.

The field of artificial immune systems represents a novel computational intelligence paradigm inspired by the biological immune system. Like neural networks and evolutionary algorithms, AIS are extremely conceptual models of their biological counterparts applied to solve problems in different domain areas.

AIS have also been used in combination with other soft computing paradigms so as to create greater models and develop individual performances, supporting the claim that they compose a new and a very useful soft computing approach (de Castro and Timmis 2002).

Artificial immune system (AIS) appeared in the 1990s as a new branch in computational intelligence (CI) (Dasgupta 2006). The fields of AIS have several existed models, and they are used in pattern recognition, computer security, fault detection, and other applications which are being explored by researchers in the field of engineering and science (Timmis et al. 1999; Neal 2003; Forrest et al. 1997).

What is an artificial immune system (AIS)? One answer is that AIS is "a model of the immune system that can be used by immunologists for explanation, experimentation and prediction activities. This is also known as 'computational immunology.' Another answer is that AIS is "an abstraction of one or more immunological processes. Since these processes protect us on a daily basis, from the ever-changing onslaught of biological and biochemical entities that seek to prosper at our expense, it is reasoned that they may be computationally useful" (Garrett 2005).

AIS can be considered a relatively young field, but it is advancing on many fronts, some central themes have become apparent. The arising questions are; what are the benefits of AIS and are they delivering anything powerful, or are they just another addition to the increasing approaches that are biologically inspired? These approaches contain many established paradigms such as genetic and evolutionary computation (GEC), artificial neural networks (ANN) and various forms of artificial life; as well as less established topics such as ant colony dynamics (Garrett 2005; Dorigo 1992, 1999) and cell membrane computing. The intention here is to provide an assessment of prior developments in AIS, its current strengths, weaknesses and its overall usefulness (Garrett 2005).

An "**Artificial Immune System** (AIS) is seen as a type of optimization algorithm inspired by the concepts and processes of the vertebrate immune system. The algorithms in its standard form exploit the Biological immune system's (Sect. 2.5) characteristics of learning and memory to deal with problems. They are coupled to artificial intelligence and closely related to genetic algorithms" (Carpinter and Ray 2006).

There were a restricted number of tries to give the field of artificial immune systems a comprehensive definition. The present work adopts the concept in which artificial immune systems are defined as computational systems inspired by theoretical immunology and observed immune functions, principles and models, applied to solve problems (de Castro 2003; De Castro and Timmis 2002a,b).

Another definition of artificial immune systems is that AIS can be seen as "abstract or metaphorical computational systems developed using ideas, theories, and components, extracted from the immune system" (de Castro and Timmis 2002). Most AIS systems are developed to solve complex computational or engineering problems, such as pattern recognition, elimination, and optimization. This is an essential difference between AIS and theoretical immune system models. While the first is devoted primarily to computing, the Second is focused on the modeling of the IS so as to understand its behavior, in other words many contributions can be added to the biological sciences. It is not exclusive, however, the use of one approach into the other and, indeed, theoretical models of the IS have contributed to the development of AIS (de Castro and Timmis 2002).

To perform pattern recognition using AIS, it usually involves three stages (de Castro and Timmis 2002; de Castro 2003):

1. Defining a representation for the patterns.
2. Adapting (learning or evolving) the system to identify a set of typical data.
3. Applying the system to recognize a set of new patterns (that might contain patterns used in the adaptive phase).

The invention in the field of AIS is still narrow and not easy for many reasons (de Castro and Timmis 2002; de Castro 2003; Dasgupta and Forrest 1996; Cao and Dasgupta 2003; Timmis et al. 2002; Kim and Bentley 2002):

- The number of people who are interested in this research area is still small, but there is an increasing in the last few years.
- The application domains of artificial immune systems are huge.
- Recently the first textbook proposing a general framework to design AIS has been published. And still the knowledge about this area is strict (Dasgupta 1999).
- It is not easy for researchers to identify the difference between an AIS and a work undertaken in theoretical immunology.

There are several AIS models inspired by biological techniques which emphasize designing artifacts–computational algorithms, techniques using basic models of different immunological processes and functionalities. AIS is like other biologically-inspired techniques, such as artificial neural networks and genetic algorithms, AISs also attempt to pull out ideas from the Biological Immune System (BIS) in order to develop computational tools to solve scientific and engineering problems. Although it is still relatively young, the Artificial Immune System (AIS) is rising as an energetic and outstanding field which involves models, techniques and applications of greater diversity (Dasgupta 2006).

Artificial Immune Systems (AIS) are being used in many applications such as anomaly detection (Dasgupta and Forrest 1996; Timmis et al. 1999) pattern recognition (Cao and Dasgupta 2003), data mining (Timmis et al. 2002), computer security (Kim and Bentley 2002; Dasgupta 1999; Hofmeyr and Forrest 2000; Balthrop et al. 2002; Kim and Wilson 2005) adaptive control (Krishnakumar and Neidhoefer 1999) and fault detection (Bradley and Tyrrell 2000; Dasgupta et al. 2004).

By studying the immune system, more specifically, the vertebrate adaptive immune system, and by creating a relation between AIS and spam detection, AIS is found as a complex network of cells. Theses cells can distinguish between harmless and harmful substances that can be recognized by the immune system; when harmful antigens are discovered, an immune system will response to eliminate these harmful antigens (Abi-Haidar and Rocha 2008).

It must be noted that most of researchers propose that there is no single technology which can perform 100% with zero false positives; as a result AIS should be used in conjunction with other filtering systems to minimize errors (Liu and Zhang 2006).

3.1.1 Components of spam detection system (AIS)

There are several components of spam detection system using AIS:

1. **Legitimate (self) and spam (non-self)**

In a spam immune system, our aim is to make a difference between legitimate messages and spam. In a biological system, the non-self elements (known collectively as pathogens) include bacteria and viruses. Like biological pathogens, spam has diversity of forms and some of these forms pathogens will only be slight variations (mutations) of others (Oda 2003; Oda and White 2003, 2005).

One of the most important advantages of using a biological immune system is distinguishing between self and non-self, which means that a biological self does not change in ways which cause an effect to the immune system. The surface proteins used by the biological immune system to distinguish self from nonself do not change over time. However, the immune system is like any other system suffers from some disadvantages. The spam immune system has the same problem as a computer security immune system (Forrest et al. 1997): which is the changing of the self over time. This means that the content of a person's legitimate mails will change over time as they meet new friends, business contacts, develop new interests, discuss current issues, maybe even learn new languages, etc.

This undesirable feature -self changes over time- does not mean the immune system model cannot be used to build a spam detector, only that the model must be used with some care. It does indicate a need for the system to forget as well as to learn things, since features that previously indicated spam could begin to indicate non-spam.

2. **Detectors (lymphocytes)**

In the biological system, there are specialized white blood cells called lymphocytes created to identify and destroy pathogens. Each lymphocyte has a detector (called an antibody), or rather a set of copies of the same antibody.

In the immune system, the antibodies are created through random recombination of a library of genes. Lymphocytes detect pathogens by binding to their surface proteins (called antigens). This binding is inexact, which means exact matching is not necessary: one lymphocyte's antibody may bind to many different (mutated) antigens, although some will bind more closely than others.

The text of the e-mail is considerd (both the headers and the body) as the antigen of a spam message. Approximate binding (inexact matching) is then simulated by the spam system by using regular expressions (patterns that can match a variety of strings) as antibodies.

By using regular expression antibodies, the aim to build a system that can match many possible alternative strings identically. "Enlarge" and "enl4rge" and "enlarg3" are all read in the same way by any human recipient of a message, so it will be promising to develop a method that makes sense to allow the immune system to deal with them as the same string (Oda 2003; Forrest et al. 1997; Oda and White 2003, 2005).

3. **Library**

The partial patterns of the gene library are used to build the full patterns in lymphocytes. These partial patterns, in this case, are small patterns which can be combined with other small patterns to create a larger one, or can stand alone as complete patterns by themselves if necessary. For the spam immune system, the choice is to use small patterns which represent heuristics used for finding spam or non-spam, but this was not the only reasonable choice. Thereis one primary aim in building the library: it should produce lymphocytes which actually detect messages (spam or non-spam) (Oda 2003; Forrest et al. 1997; Oda and White 2003, 2005).

Another approach is to make an electronic library containing every character that could possibly be used in an e-mail message, but it must be noted here that doing so would waste valuable knowledge on the structure –words- of messages. Messages (legitimate or spam) usually contain more words than randomly-concatenated letters. And the words used in spam messages usually represent only a subset of written language. Using such a library has a great drawback of having a great number of undesirable words.

In the other way using a smaller library has a lot of advantages, the speed in getting results, but there are also drawbacks. One of the most significant problems of learning occurs when the messages which have no detector matches. Because the words of this small library will not be sufficient to cover all situations. With a library that is not utterly comprehensive, it may be possible that no gene combination could even produce such a detector. As result, there is a need to build a library that is not huge and not small.

3.1.2 Spam detection using AIS algorithm

The sub-algorithms describe the phases of the lifecycle in more details: Algorithm 2 explains the generation of new lymphocytes, Algorithm 3 describes their initial training phase, Algorithm 4 explains the application of lymphocytes to messages, and Algorithm 5 details the process of culling and ageing of old lymphocytes (Oda 2003; Oda and White 2003, 2005). [10]

Algorithm 1 Spam Immune System:

```
Require: update interval ⬅ a time interval after which the system will age. {Chosen by user} {e.g. 10 days from now}
Repertoire ⬅ (Ø {Initialize repertoire (list) of lymphocytes to be empty}
update time ⬅ ( currenttime + update interval {time of next lymphocyte update}

Generate lymphocytes (See Algorithm 2)
Do initial training (See Algorithm 3)
while Immune System is running do
    if message is received then
    Apply lymphocytes (See Algorithm 4)
    end if

    if current time > update time then
    Cull lymphocytes (See Algorithm 5)
    Generate lymphocytes to replace those lost by culling (See Algorithm 2)
    update time ⬅ ( currenttime + update interval {t}ime of next lymphocyte update
    end if
end while
```

[10] http://terri.zone12.com/doc/academic/crossroads. Accessed 10 Jan 2008.

Algorithm 2 Generation of lymphocytes:

```
Require: library ⬅ a gene fragment library (cannot be empty)
Require: repertoire ⬅ the list of existing lymphocytes (may be empty)
Require: p_appending ⬅ the probability of appending to antibody {chosen by user}

while repertoire is smaller than the required size do
    lymphocyte ⬅ a new empty memory structure with space for an antibody, and
    the numbers msg_matched and spam_matched
    antibody ⬅ randomly chosen gene fragment from library {This starts the new
    antibody being created. This will be a regular expression made up of genes and
    wildcards.}
    lymphocyte.msg matched ⬅ 0
    lymphocyte.spam matched ⬅ 0
    repeat
        x ⬅ randomly chosen number between 0 and 1 {uniform distribution}
        while x < p appending do
        newgene ⬅ new randomly chosen gene fragment from library
        antibody ⬅ concatenate antibody, an expression that matches 0 or more
        characters, and newgene
        x ⬅ new randomly chosen number between 0 and 1 {uniform
        distribution}
        end while
    until an antibody is created that does not match any in the repertoire
    lymphocyte.antibody ⬅ antibody
    Add lymphocyte to repertoire of lymphocytes
end while
```

Algorithm 3 Training of lymphocytes:

```
Require: repertoire ⬅ the list of lymphocytes (cannot be an empty list)
Require: message ⬅ a message which has been marked as spam or non-spam
if the message is user-determined spam then
    spam_increment ⬅ 1
    else if the message is user-determined non-spam then
    spam_increment ⬅ 0
    else
    spam_increment ⬅ a number between 0 and 1 indicating how likely the message
    is to be spam {Chosen by user}
end if
for each lymphocyte in the repertoire do
    if lymphocyte.antibody matches the message then
    lymphocyte.msg_matched ⬅ lymphocyte.msg_matched + 1
    lymphocyte.spam_matched ⬅ lymphocyte.spam_matched + spam_increment
    end if
end for
```

Algorithm 4 Application of antibodies with dynamically updated weights:

```
Require: repertoire ⇦ the list of antibodies (cannot be an empty list)
Require: message ⇦ a message to be marked
Require: threshold ⇦ a cutoff point valued between 0 and 1 inclusive; anything with a higher score than this is spam {chosen by user}
Require: increment ⇦ increment used to update lymphocytes
Or...
Require: confidence ⇦ a value between 0 and 1 inclusive, depending upon the user's confidence in the system. {chosen by user}

total_spam_matched ⇦ 0 {initialize # of spams matched to 0}
total_msg_matched ⇦ 0 {initialize # of messages matched to 0}
matching_lymphocytes ⇦ Ø {Initialize empty list of matching lymphocytes}

for each lymphocyte in the repertoire do
    if lymphocyte.antibody matches message then
    total_spam_matched ⇦ total_spam_matched + lymphocyte.spam_matched
    total_msg_matched ⇦ total_msg_matched + lymphocyte.msg_matched
    lymphocyte.msg_matched ⇦ lymphocyte.msg_matched + 1
    {Increment the # of messages matched by this antibody}
    add lymphocyte to matching_lymphocytes
    end if
end for
score ⇦ total_spam_matched / total_msg_matched
{Determine the score using a weighted sum}
if score < threshold then

    Message is spam
    for each lymphocyte in matching lymphocytes do
        if confidence is set then
        increment ⇦ confidence * score
        else
        {increment has been supplied by the user}
        end if
        lymphocyte.spam_matched ⇦ lymphocyte.spam_matched + increment
    end for
else
Message is not spam
end if
```

Algorithm 5 Culling of antibodies: ageing and death

```
Require: matched_threshold ⇦ any lymphocyte with a msg_matched value below this threshold will be killed {chosen by user}
Require: decrement ⇦ amount by which to decrement ageing antibodies {chosen by user}

for each lymphocyte in the repertoire (list of all lymphocytes) do
lymphocyte.spam_matched ⇦
     (lymphocyte.spam_matched / (lymphocyte.msg_matched) *
    (lymphocyte.msg_matched − decrement).
{the ratio between the two weights stays the same as it was before the ageing}
lymphocyte.msg_matched ⇦ lymphocyte.msg_matched − decrement

    if lymphocyte.msg_matched < threshold then
    remove antibody from data store
    end if
end for
```

3.2 How genetic optimized spam detection using AIS works

In the following sections main components of spam detection system are explained which will be modified to enhance the system. Also, our proposed algorithms will be shown.

3.2.1 Genetic optimized spam detection using AIS

GA is an optimization algorithm which can be used in different applications, in standard AIS there are many parameters defined by the user. One of the most important parameters is culling. This parameter is defined by the user in standard AIS. In this work GA is used to determine the culling time. Also, GA is used in determining Rebuild time to solve the problem of users' interests which do not remain the same over time.

Antispam solutions have to increase the frequency of the updates and also to develop more heuristics in less time. The need for an automatic process that would quickly learn the characteristics of the new spam without affecting the accuracy of detection on less recent spam has become vital.

AIS parameter of AIS that is a subject for modification and optimization are the following:

1. **The Self (legitimate)**

The interest of any person is not stable because circumstances are always changed. The problem found in the standard immune system (AIS) is the self (Legitimate) which changes over time. This means that the message content and characteristics that any person would like to receive is changed over time. For example a healthy person might have no interest in any message about medicine. However, if that person is diagnosed with any disease, he may start to receive them. Also any person who has no interest in sport, he does not like to receive any sport messages, however, if that man becomes so fat he may become interested in those messages to become fit. This means that the system must be adapted to this change. This work tries to solve this problem with the help of GA, in this work the system is being rebuilt periodically, so the system has the chance to relearn. This means that the system must accept some types of messages which could be recognized as spam and if the user still recognizes this type of message as spam, as a rsult there is no changing on the self. Moreover, the Genetic Algorithm is used in producing guided random Rebuild time instead of fixed period.

2. **Library**

The standard Artificial Immune system uses a library that does not change over time. But as the system sees more spam messages, the system must have the ability to gather information from these spam messages that could be used to create new useful gene fragment. By adding this extra ability which is important to the system to adapt gene library, it would be possible to make the system adapt to new messages which are not matched by any current gene fragment. In this workinformation is collected from spam messages. Frequent patterns are used to create new antibody or gene library.

3. **Adaptive weight of gene library**

In Standard Artificial Immune system, each gene fragment has an equal chance of being selected, this means that if there is a fragment rarely used, this gene fragment will have an equal chance with any gene that is frequently used. This shows that the system is not adapted. Weights are adapted for each gene fragment based on the previous run.

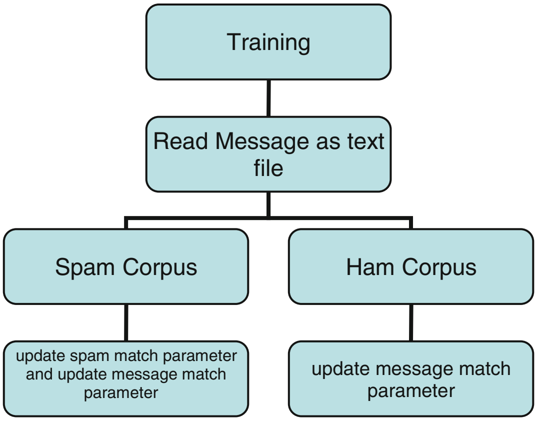

Training
Read Message as text file
Spam Corpus
Ham Corpus
update spam match parameter and update message match parameter
update message match parameter

Fig. 6 Training outline

4. **Lifecycle of lymphocytes**

The main point here is to give the lymphocytes a chance to match and to assign a weight but this time must be bounded so that the system does not waste time on lymphocyte that do not match any message "un-useful lymphocyte". In Standard AIS after some time has passed (the time interval is chosen by the user), the lymphocytes are aged and may be killed. This amount of time is a parameter of the system, defined by the user. The best choice of an update interval will depend upon the number of e-mails received by the users. In this thesis, it is defined by using the Genetic Algorithm. This is done in a fashion similar to the approach used with the self.

The modifications on algorithms (Fig. 8) will be explained in details in the next section.

3.2.2 Genetic optimized spam detection using AIS algorithm

In genetic optimized spam detection using AIS there are several major steps:

- Training:
  The aim of training (Fig. 6) is to build a library and from this library, the best lymphocytes will be chosen to fight against spam. In summary, the steps of training are the following:
  - The system reads each message as a text file and then parsed to identify each header information (such as From, Received, Subject and To).
  - Any lmphocyte which exceeds 20 characters or less than 3 characters is excluded.
  - All accepted lymphocytes are added to the library.
  - Modification is done on (Spam_matched, Msg_matched) parametrs.
  - Perform extra cleaning (if necessary) on library to refine it.
- Testing:
  In testing (Fig. 7) there are several steps:
  - When a message is received, the system compiles it as a text file. Then, the system will look in the innate immune system to search for any matched lymphocyte. Then the system calculates the score for each received message to determine if the message is spam or not.
  - Depending on the score, the system will decide if the message is spam or not (if score is greater than thresold then message is spam).

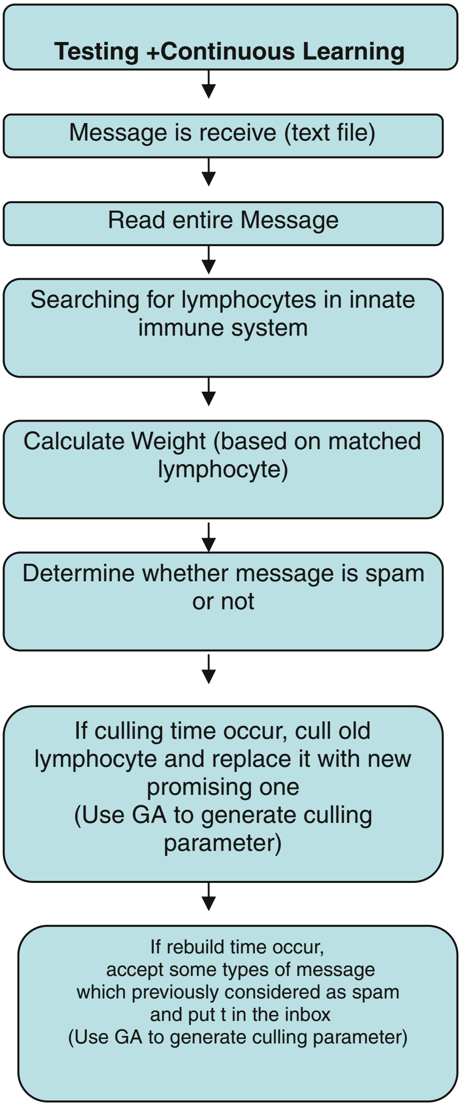

Testing +Continuous Learning
Message is receive (text file)
Read entire Message
Searching for lymphocytes in innate immune system
Calculate Weight (based on matched lymphocyte)
Determine whether message is spam or not
If culling time occur, cull old lymphocyte and replace it with new promising one (Use GA to generate culling parameter)
If rebuild time occur, accept some types of message which previously considered as spam and put t in the inbox (Use GA to generate culling parameter)

Fig. 7 Testing + continuous learning outline (genetic optimized AIS)

- If the message is spam, then the system will add each new lymphocyte to the adaptive immune system (library) to be used in future (learning).
- The system will use GA to generate a Culling parameter. When culling occurs, useless lymphocytes will be deleted and replaced by new promising lymphocytes form the library.
- Also, the system will use GA to generate a rebulid time parameter. When the rebuild time occurs, the system will accept a new type of messages which are considered as spam and will put them in the inbox. Depeneding on the users' feed back, the system will determine if the self is changed or not.

GA in genetic optimized spam detection using AIS is called twice (Fig. 8); firstly, to cull old lymphocytes (useless lymphocytes replaced by new promising ones). Secondly, to check

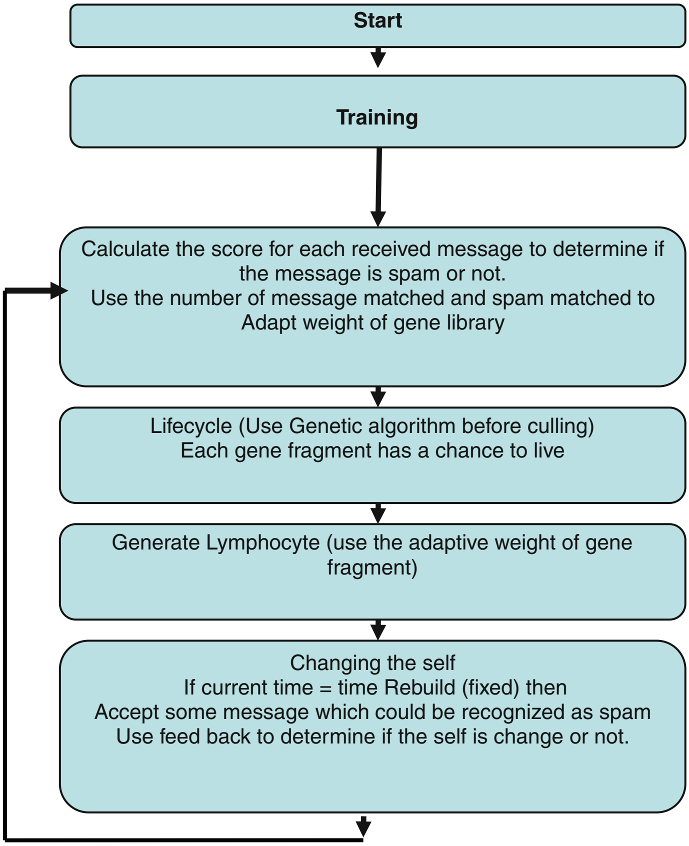

Start
Training
Calculate the score for each received message to determine if the message is spam or not.
Use the number of message matched and spam matched to Adapt weight of gene library
Lifecycle (Use Genetic algorithm before culling)
Each gene fragment has a chance to live
Generate Lymphocyte (use the adaptive weight of gene fragment)
Changing the self
If current time = time Rebuild (fixed) then
Accept some message which could be recognized as spam
Use feed back to determine if the self is change or not.

Fig. 8 Genetic optimized AIS process flow (training + testing)

Algorithm 1 Genetic Optimized Spam Immune System using Genetic Algorithm:

```
Require: Update_Interval for culling old lymphocyte: (update based on time or based on number of message, in both situations, it is generated based on Genetic Algorithm.
Innate_Immune_System ← Ø {Initialize innate immune system (list) of lymphocytes to be empty}
Adaptive_Immune_System ← Ø {Initialize innate immune system (list) of lymphocytes to be empty}
Update: If update based on time used.
Update ← (current time + update time using GA
Or
If update based on number of message.
Update ← number of message using GA
Start Training (Algorithm 2)
While Optimized_Immune_ System is running do
    if message is received then
    Start Application (Algorithm 3)

    end if
        if current time > update time then
        Or
        if number of message received > number of message for update) then
        Start Learning (Algorithm 4)
        end if
end while
```

Algorithm 2 (*Training*):

```
Message ⬅ spam or non spam message. (Training corpus)
Innate_Immune_System ⬅ table (may be empty)
Spam corpus
For each lymphocyte in the spam message corpus do
    If lymphocyte is already exist in Innate_Immune_System then
    lymphocyte.msg_matched ⬅ lymphocyte.msg_matched + 1
    lymphocyte.spam_matched ⬅ lymphocyte.spam_matched + spam_increment
    else
    Add lymphocyte to Innate_Immune_System
    lymphocyte.msg_matched ⬅ lymphocyte.msg_matched + 1
    lymphocyte.spam_matched ⬅ lymphocyte.spam_matched + spam_increment
    end if
end for
Ham corpus
For each lymphocyte in the spam message corpus do
    If lymphocyte is already exist in Innate_Immune_System then
    lymphocyte.msg_matched ⬅ lymphocyte.msg_matched + 1
    end if
end for
End
```

Algorithm 3 *Application*:

```
Innate_Immune_System ⇦ the list of Anti_spam lymphocyte
Adaptive_Immune_System: Empty Table
Message ⇦ a message to be known whether it is spam or ham
Threshold ⇦ a cutoff point valued between 0 and 1 inclusive; anything with a higher
score than this is spam {chosen by user}.
Require: increment ⇦ increment used to update lymphocytes
total_spam_matched ⇦ 0
total_msg_matched ⇦ 0
for each lymphocyte in Innate_Immune_System do
    if lymphocyte.antibody matches message then
    total_spam_matched ⇦ total_spam_matched + lymphocyte.spam_matched
    total_msg_matched ⇦ total_msg_matched + lymphocyte.msg_matched
    lymphocyte.msg_matched ⇦ lymphocyte.msg_matched + 1
end for

Score ⇦ total_spam_matched / total_msg_matched
if score > threshold then
    Message is spam
    lymphocyte.spam_matched ⇦ lymphocyte.spam_matched + increment
        if lymphocyte.antibody does not exist in Innate_Immune_System then
        Add lymphocyte.antibody to Adaptive_Immune_System
        (This is to represent continuous learning)
        end if
else
Message is not spam
end if

End
```

Algorithm 4 (Learning) Cull old lymphocytes and generate new lymphocyte

```
Culling can happened based on
Criteria 1: Number of message calculated used Genetic algorithm
Or
Criteria 2: Update interval calculated used Genetic algorithm
if criteria happened then
Merge Adaptive_Immune_System with Innate_Immune_System and then order it descending based on lymphocyte.spam_matched
        end if
        Select top lymphocytes in Innate_Immune_System
End
```

whether there are any new interests for users (changing the self) in a similar manner. This can be done taking in consideration that the system use different domain ranges for each ones.

3.2.3 Components of the genetic optimized spam detection AIS

3.2.3.1 The library In genetic optimized spam detection AIS training phase is used to create the library which will be used then to generate lymphocytes. This library contains thousands of lymphocytes but the majority number of these lymphocytes are useless. There are thousands of lymphocytes which appear just one time after finishing training phase. So, in order to decrease the huge number of lymphocytes in the library all lymphocytes with spam_matched = 1 and Message_matched = 1 are deleted. useful lymphocytes is one which has matched a large number of spam messages.

In training phase each e-mail message is compiled as a text file, and then parsed to identify each header information (such as From:, Received: Subject: or To:) to distinguish them from the body of the message. Every substring within the subject header and the message body that was delimited by white space was considered to be a token (lymphocyte).

3.2.3.2 Lymphocytes generation For the purposes of using biological immune system in spam detection, the term "digital lymphocyte" to refer to:

- Digital antibody (Sect. 3.3.3.2.1)
- Weighting information (Sect. 3.3.3.2.2)

3.2.3.2.1 Digital antibody The spam immune system use string pattern matching to represent this. Pattern recognition is used in spam detection which mean that any given antibody can be used against more a one infection of spam messages, this is similar to biological immune system which uses the same antibodies against infection and reinfection.

3.2.3.2.2 Weights With each lymphocyte a pieces of information is stored.

- Spam_Matched: the total number or weight of apperance only in spam messages.
- Message_Matched: the total number or weight of apperance of messages by this lymphocyte.

In training: both of these matches are initialized to zero, when the detector matches a message, message_matched is incermented by 1, but if that message which matched is spam, spam_matches will also be incremented by 1.

The two numbers; spam_matched and message_matched can be used to give a weighted percentage of the time an antibody detects spam. The field message_matched gives an 62 indication of how often this antibody has been used, which helps to determine how important it should be in the final weighting

In this research the weighted average is applied as follows:

$$\text{Weighted Average} = \frac{\sum_{i=1}^{n} matching\_Lymphocytes(Spam\_Matched)}{\sum_{i=1}^{n} matching\_Lymphocytes(Message\_Matched)} \tag{3.1}$$

where

$matching\_Lymphocytes(Spam\_Matched)$	the total number of apperance only in spam messages.
$matching\_Lymphocytes(Message\_Matched)$	total number of apperance of messages by this lymphocyte.

3.2.3.3 Lifecycle of lymphocytes The lymphocyte which has the large number of spam_matched has the greatest chance to be selected and used against any new message. Also, the lymphocyte which has the lowest number of spam_Matched has the greatest chance to be culled and repleaced with a new acquired lymphocyte.

There are many choices used to select the update interval (*see Algorithm 1 Optimized Spam Immune System using Genetic Algorithm)* such as the number of messages receives, the update interval based on time, the user request,…etc. However, in this work the aim of using Genetic Algorithm is to determine the update interval. Also the system gives the selected lymphocytes a chance to fight against infection, However, when any lymphocyte becomes useless which means that the spam_matched value will remain static (no change) and there is a new lymphocyte in (Adaptive_Immune_System) that has spam_matched score greater than this lymphocyte this eventually means that old lymphocytes will be culled and new lymphocytes will be added to the list of Innate_Immune_System.

3.2.3.4 Innate immune system In bilogical immune system there is an innate immune system that has the capability to defeat infections. Also, in spam immune system there is an innate Immune system built from a library and must be able to defeat against spam. It is known that there is a great number of lymphocytes in this library and by using this number of lymphocytes which is not small the system will be exhaustive. The best choice to solve of this problem is to select the best lymphocytes which has the capability to defeat the greatest number of spam.This does not mean that all lymphocytes in the Innate immune system are useful and will remain for ever. Any new promising lymphocyte created in adaptive immune system will be moved to the innate immune system and will take the place of an useless lymphocyte.

3.2.3.5 Adaptive immune system In bilogical immune system huamns can get an external defeat against infection from medicine or other methods. In spam immune system there is an adaptive immune system built from spam messages which contain lymphocytes that do not exist in the innate immune sytem.

3.2.3.6 Culling the useless lymphocytes As mentioned before that the system suffers from containing some useless lymphocytes in the innate immune system. So the best solution to keep only useful lymphocytes in innate immune system is culling. Culling in AIS occurs after a parameter defined by the user and this is not the best solution. In this work, a new creative method is used to perform culling based on a Genetic Algorithm. Genetic algorithm is used to determine an update interval taking in consideration either the time or the number of messages received by the user. In this work, the genetic algorithm is used to determine the culling according to number of messages receicved.

The success of AIS and genetic optimized AIS spam filtering techniques is determined by classic measures of precision, recall, False Positive,and false negative (Secker et al. 2003; Yue et al. 2006; Sarafijanovic and Le Boudec 2007; Abi-Haidar and Rocha 2008).

Spam Precision: the percentage of messages classified as spam that actually are spam.

$$\text{Relevant Retrieved}/\text{Retrieved} \times 100\%$$

Legitimate Precision: the percentage of messages classified as legitimate that are indeed legitimate.

$$\text{Relevant Retrieved}/\text{Retrieved} \times 100\%$$

Spam recall: the proportion of the number of correctly-classified spam messages to the number of messages originally categorized as spam.

$$\text{Relevant Retrieved}/\text{Relevant} \times 100\%$$

legitimate recall: the proportion of correctly-classified legitimate messages to the number of messages originally categorized as legitimate.

$$\text{Relevant Retrieved}/\text{Relevant} \times 100\%$$

N(Sp_Sp): the number of spam messages correctly classified as spam.
N(Sp_Le): the number of spam messages incorrectly classified as legitimate
N(Le_Le): the number of legitimate messages correctly classified as legitimate
N(Le_Sp): the number of legitimate messages incorrectly classified as spam

$$\text{Spam Precision} = \frac{\text{N(Sp\_Sp)}}{\text{N(Sp\_Sp\_)+(n(Le\_Sp)}} \times 100\% \tag{3.2}$$

$$\text{Spam Recall} = \frac{\text{N(Sp\_Sp)}}{\text{N(Sp\_Sp)+N(Sp\_Le)}} \times 100\% \tag{3.3}$$

$$\text{Legitimate Precision} = \frac{\text{N(Le\_Le)}}{\text{N(Le\_Le)+N(Sp\_Le)}} \times 100\% \tag{3.4}$$

$$\text{Legitimate recall} = \frac{\text{N(Le\_Le)}}{\text{N(Le\_Le)+N(Le\_Sp)}} \times 100\% \tag{3.5}$$

$$\text{False Positive} = \frac{\text{Number\_of\_emails\_wrongly\_identified\_as\_spam}}{\text{Total\_number\_of\_emails}} \times 100\% \tag{3.6}$$

$$\text{False Negative} = \frac{\text{Number\_of\_emails\_wrongly\_identified\_as\_ham}}{\text{Total\_Number\_of\_emails}} \times 100\% \tag{3.7}$$

4 Spam detection using ANN

Most e-mail readers and users spend a non-trivial amount of time regularly deleting junk e-mail (spam) messages. This preliminary part talks about an alternative approach using a neural network (NN) classifier on a corpus of e-mail messages. The feature set uses descriptive characteristics of words and messages similar to those that a human reader would use to identify spam (Stuart et al. 2004).

Neural networks can be used to solve the problem of classifying e-mails into e-mails that internet users wish to receive and spam e-mails that internet users do not wish to receive. Neural networks is a technology that attempt to mimic the way how the human brain works.

Application areas include system identification and control (vehicle control, process control), game-playing and decision making (backgammon, chess, racing), pattern recognition (radar systems, face identification, object recognition and more), sequence recognition (gesture, speech, handwritten text recognition), medical diagnosis, financial applications (automated trading systems), data mining (or knowledge discovery in databases, "KDD"), visualization and e-mail spam filtering. [11]

In order to block spam messages, filters are used. Filters are one of the main ways that can be used to reduce the amount of spam that may arrive into user's e-mails. One of the main method used to combat spam are neural networks. Neural networks can be used to solve the problem of classifying e-mails into e-mails that internet users wish to receive (ham or legitimate) and spam e-mails that internet users do not wish to receive. Artificial neural networks are made of simple neurons that are interconnected together. These simple processing units mimic the behavior of a simple human neuron. There are many types of neural networks, but the most commonly used are back-propagation. These networks are required to be trained by applying training data to the inputs, and telling the network what the result at the output layer should be. The network is then trained and the weights between neurons adjusted till the output matches what is required. Neural networks are very useful in classifying spam as they are able to generalize well. A well trained neural network should be able to recognize spam e-mails haven't been seen before.

Although no single technology can achieve one hundred percent spam detection with zero false positives (despite vendor claims), machine-learned heuristics in general and neural networks in particular have proven extremely effective and reliable at accurately identifying spam and minimizing errors to an acceptable minimum (Miller 2008).

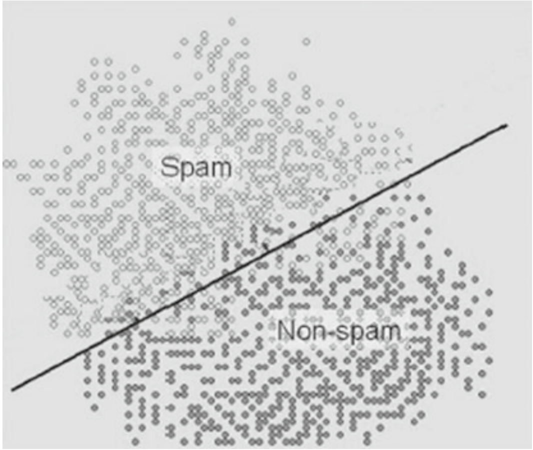
Hermann von Helmholtz, Ernst Mach, and Ivan Pavlov made significant contributions to neural research at the beginning of the 20^{th} Century. These contributions had led to ANN development. ANN is inspired by the brain. The research and works done by these early leaders were the fundamental block for the development of the concepts used later in ANN (Bailey et al. 2006; Hagan et al. 2002; Pogula Sridhar 2005; Abu-Nimeh et al. 2007; Puniškis et al. 2008; Chuan et al. 2005; Vinther 2002).

One of the definitions of ANN is "An artificial neural network (ANN), often just called a "neural network" (NN), is a mathematical model or a computational model based on biological neural networks. It consists of interconnected groups of artificial neurons and it processes information using a connectionist approach. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase". [12]

[11] http://en.wikipedia.org/wiki/Neural_network#Characterization. Accessed 10 Jan 2008.

[12] http://en.wikipedia.org/wiki/Neural_network#Characterization. Accessed 10 Jan 2008.

Fig. 9 Distinctive patterns of good and spam messages



As AIS is inspired by the biological immune system also the Artificial Neural Network (ANN) is a machine learning algorithm that is inspired by the biological brain systems. There are two phases in ANN as in many other learning algorithms: The training phase: a training dataset is used to determine the weight parameters that define the neural model. The weight of each neuron or each interneuron connection is calculated during this phase based on experience or previous results or mathematical equation. The test phase: the trained neural model is used to process real test patterns and give classification results. Knowledge is acquired by the network through these weights and the values of inputs. (Bart and Binargrl 2003; Clark et al. 2003; Ozgur et al. 2004; Miller 2008).

Neural networks are based on pattern recognition; so the key solution is that each message can be evaluated on the bases of its pattern. This is represented below in Fig. 9 Each point on the graph (also known as a "vector") represents an e-mail message. Although this 2-D example is a simple example, it helps to determine visually the idea used in neural networks (Miller 2009).

Neural network such as most machine learning methods is employed to identify these patterns. Also, the neural network must first be "trained" as most machine learning methods. This training method has a computational analysis for each message content and characteristics using large representative samples (dataset) of both spam and non-spam messages. The result of this training and analysis is that the neural network will "learn" to recognize by "spam" and "non-spam" (Miller 2009).

As any filtering method; some legitimate e-mails were incorrectly classified as spam because they are sharing a lot of characteristics of spam. It is accepted that a spam filter would not be able to correctly filter spam 100% all the time.

The neural network was able to produce a result very quickly; time is a very important factor with regard to filters, because taking a long time to produce a result could result in considerable delays to the user.

In the next section our new approach called Continuous Learning Approach Artificial Neural Network (CLA_ANN) is introduced.

4.1 CLA_ANN

The following perceptron learning algorithm approach is developed.

Algorithm 1 CLA_ANN:

```
Require: Update_interval: a time interval after which the system will update its input
layer [Defined by user]
Input_layer: identify number of layer used for defining spam.
Update_time: current time+ Update_interval

Start Training (Algorithm 2)

While CLA_ANN System is running do
        if message is received then
        Start Application (Algorithm 3)
        end if
                if current time > update time then
                Or
                if number of message received > number of message for update then
                Start Learning (Algorithm 4)
                end if
end while
```

Algorithm 2 (Training): creation of input layer

```
Require: Message ⬅ spam or non spam message. (Training corpus)
Innate_neurons ⬅ table (may be empty)
Spam corpus
For each token in the spam message corpus do
        If layer is already exist in Innate_Input_Layer then
        Innate_neurons.msg_matched ⬅ Innate_neurons.msg_matched + 1
        Innate_neurons.spam_matched ⬅ Innate_neurons.spam_matched +
        spam_increment
        else
        Add token to Innate_neurons
        Innate_neurons.msg_matched ⬅ Innate_neurons.msg_matched + 1
        Innate_neurons.spam_matched ⬅ Innate_neurons.spam_matched +
        spam_increment
        end if
end for

Ham corpus
For each token in the spam message corpus do
        If token is already exist in Innate_neurons then
        Innate_neurons.msg_matched ⬅ Innate_neurons.msg_matched + 1
        end if
end for
token.weight= Innate_neurons.spam_matched / Innate_neurons.msg_matched

End
```

Algorithm 3 *Application*:

```
Innate_neurons ⇦ the list of tokens for spam detection
Adaptive_neurons: Empty Table
Learning_rate: defined by user
Message ⇦ a message to be known whether it is spam or ham
Threshold ⇦ a cutoff point valued between 0 and 1 inclusive; anything with a higher
score than this is spam {chosen by user}.
Require: increment ⇦ increment used to update lymphocytes
number_of_token_matched ⇦ 0
for each token in Innate_Input_Layer do
        if token matches message then
        total_weight ⇦ total_weight + token.weight
        number_of_token_matched= number_of_token_matched+1
                if token.weight > threshold then
                Desired_Output ⇦ 0.9999 (Spam)
                else
                Desired_Output ⇦ 0.1111 (Ham)
                end if
        end if
end for
Score ⇦ total_weight / number_of_token_matched
{Determine the score using a weighted sum}
if score > threshold then
        Message is spam
        Error_rate ⇦ Absolute (Desired_Output – Score)
        Correction ⇦Error_rate*Learning_rate
        Token.weight ⇦ Token.weight + Correction
        If token does not exist in Innate _ Input_Layer then
        Add token to Adaptive_neurons
        (This is to represent continuous learning)
else
        Message is not spam
        Error_rate ⇦ Absolute (Desired_Output – Score)
        Correction ⇦Error_rate*Learning_rate
        Token.weight ⇦ Token.weight - Correction
end if

End
```

Algorithm 4 (*Learning*) Delete Old input layer and replace it with anew promising *input layer*

```
Delete can happened based on
Criteria 1: Number of message calculated used Genetic algorithm
Or
Criteria 2: Update interval calculated used Genetic algorithm
if criteria happened then
        Merge Adaptive_neurons with Innate_neurons and then order it descending based
        on Token.spam_matched
        end if
        Select top Innate_neurons
End
```

4.2 The training phase

First, preparing the neurons where the system create the library which will be used then to generate neurons. This library contains thousands of neurons but it is noticed that most of these neurons are useless since there are rubbish words (more than 20 characters, address of sites,… etc.) So, in order to decrease the huge number of neurons in the library all layers

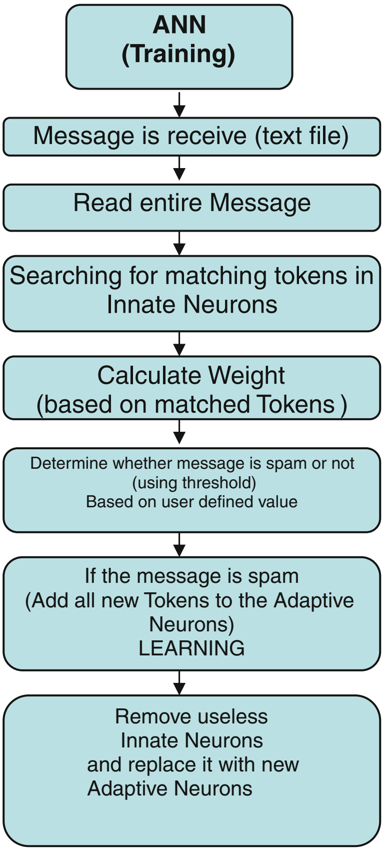

ANN
(Training)
Message is receive (text file)
Read entire Message
Searching for matching tokens in Innate Neurons
Calculate Weight
(based on matched Tokens)
Determine whether message is spam or not
(using threshold)
Based on user defined value
If the message is spam
(Add all new Tokens to the Adaptive Neurons)
LEARNING
Remove useless
Innate Neurons
and replace it with new
Adaptive Neurons

Fig. 10 CLA_ANN outline

with spam_matched = 1 and Message_matched = 1 are deleted. Useful neurons are ones which have matched a large number of spam messages.

In training phase, each e-mail message was compiled as a text file, and then parsed to identify each header information (such as From:, Received: Subject: or To:) to distinguish them from the body of the message. Every substring within the subject header and the message body that delimited by white space was considered to be a token.

In training phase a (1075) spam and (710) ham are used for training (Section 5.6). Also, in training phase the system is evaluated for several times each one with different neurons.

4.2.1 How ANN work

The CLA_ANN (Fig. 10) works as the following:

- When a message is received, the system compiles it as a text file. Then, the system will look in the innate neurons to serach for any matched token. Bases on the weight of each token, the system will calculate the score for each received message to detemine if the message is spam or not.
- The system will compare the score with a threshold value, if the score is greater than the threshold value then the message will be considered as spam. Otherwise, it will be considered as ham.

Table 2 ANN input

Input		Input layer	Initial Layer (weight)	Desired output	Weights Calculated
Threshold TH	Learning rate LR		W i	Z	N
User defined	User defined	Neuron 1	W 1	0.9 or 0.1	W n
User defined	User defined	Neuron 2	W 2	0.9 or 0.1	W n
User defined	User defined	Neuron 3	W 3	0.9 or 0.1	W n
User defined	User defined	Neuron 4	W 4	0.9 or 0.1	W n
User defined	User defined	Neuron 5	W 5	0.9 or 0.1	W n
User defined	User defined	Neuron 6	W 6	0.9 or 0.1	W n
User defined	User defined	Neuron 7	W 7	0.9 or 0.1	W n
User defined	User defined	Neuron …	W…	0.9 or 0.1	W n
User defined	User defined	Neuron …	W…	0.9 or 0.1	W n
User defined	User defined	Neuron …	W…	0.9 or 0.1	W n
User defined	User defined	Neuron n	W n	0.9 or 0.1	W n

- If the message is spam, then the system will add each new token to the adaptive neurons to be used in future (learning).
- The system will use a user define value to determine when to remove old layers and replace them with new adaptive neurons. This method will guarantee that there is a continuous learning.
- When the system deletes useless innate neurons,they will be replaced with new adaptive neurons which means that there is a continuous learning.

4.2.2 Testing ANN

This is done by subjecting ANN to neurons that were not used in training without adjusting the weights (Sects. 5.3.2 and 5.3.3). Dataset used for testing consist of (682) spam and (3435) ham.

Perceptron Neural Network is used to evaluate the system as shown in Tables 2 and 3.

5 Evaluation and testing of the different spam detection techniques

This part describes the libraries and the parameters used in testing the spam immune system, genetic optimized spam immune system and spam detection using ANN. Experimentation, simulation results, are depicted in this chapter.

5.1 Spam corpus

The task of selecting a corpus for the evaluation of any learning algorithms of spam detection is difficult. One challenge is that private e-mails are rarely available for public studies.

In order to test the system, it is necessary to have a public available corpus of e-mails. There are many researchers who use their own personal e-mails or any other available e-mails

Table 3 ANN Output

Output Result	Network	Error (correction)	New weight (if spam)	Final New weight (if ham)
IF (N > TH, spam, Ham)	E (Error)	R	W1	W1
Result	Z-N	LR x E		
Spam or ham	Z-N	LR x E	W1 + R	W1 − R
Spam or ham	Z-N	LR x E	W1 + R	W1 − R
Spam or ham	Z-N	LR x E	W2 + R	W2 − R
Spam or ham	Z-N	LR x E	W3 + R	W3 − R
Spam or ham	Z-N	LR x E	W4 + R	W4 − R
Spam or ham	Z-N	LR x E	W5 + R	W5 − R
Spam or ham	Z-N	LR x E	W6 + R	W6 − R
Spam or ham	Z-N	LR x E	W7 + R	W7 − R
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…
Spam or ham	Z-N	LR x E	Wn + R	Wn − R

for training and for the evaluation of their systems. But this will not give a broad ranging evidence to prove that the system will work efficiently.

5.2 Spamassassin public corpus

The SpamAssassin (SA) corpus is a larger collection made available by spamassassin.org.[13] The SpamAsassin corpus has been used in some research (Oda 2003; Zhang et al. 2004; Zhan et al. 2005; Yao 1999; Whitley 2009; Balakrishnan and Honavar 2008). It contains 4147 legitimate and 1764 (only in 2002) spam messages collected from public or donated by individual users.

5.3 Preparing the corpus

The corpus was divided depending on the information found in the date field “e-mail header”. This information is not necessarily accurate, since it relies on the time of clock of the sender, which may not be correct.

All the messages whose data fields were outside 2002 were discarded. The breakdowns of the dataset used in this work are shown in Table 4. The SpamAssassin public corpus is divided into seven parts: 20021010 easy ham, 20021010 hard ham, 20030228 easy ham, 20030228 easy ham 2, and 20030228 hard ham contains the non spam messages. 20030228 spam, 20030228 spam 2, 20050311 spam 2 and 20021010 spam contains the spam messages. The parts marked 2 indicate more recent additions to the collection. The easy and hard ham indicates which messages have features which make them seem more like spam. Only five parts of the corpus were selected for the study; 20030228 easy ham, 20030228 easy ham 2 and 20030228 hard ham contain the ham messages. Whereas, 20030228 spam and 20050311 spam 2 contain the spam messages. It must be noted that the SpamAssassin

[13] www.spamassassin.org. Accessed April 2008.

Table 4 SpamAssassin public corpus by month

	Non spam				Spam		
	Easy ham	Easy ham 2	Hard ham	Total non spam	Spam	Spam 2	Total spam
Jan	0	0	1	1	0	0	0
Feb	0	0	0	0	0	0	0
Mar	0	0	0	0	0	0	0
Apr	0	0	0	0	0	0	0
May	0	0	1	1	0	1	1
Jun	0	0	6	6	0	502	502
Jul	0	550	152	702	6	566	572
Aug	423	843	34	1, 300	157	169	326
Sep	1, 283	0	28	1, 311	321	0	321
Oct	726	0	9	735	6	0	6
Nov	16	6	14	36	0	7	7
Dec	52	1	2	55	11	18	29

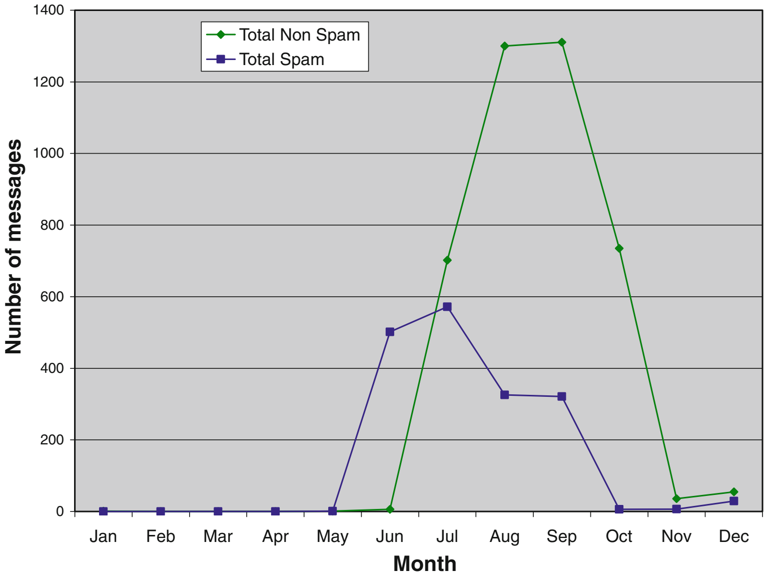


Fig. 11 (SpamAssassin public corpus by month)

public corpus is difficult to be classified and does not reflect a normal ration of spam to ham. Looking at Fig. 11 the corpus contain few non spam messages from January to June, then the number increased during the next months before going back to smaller numbers. It is probably not a typical behavior for an individual mailbox. This pattern is more similar to the way in which the mails were collected by the SpamAssassin public corpus team.

Table 5 Training data by month

	Non spam				Spam		
	Easy ham	Easy ham 2	Hard ham	Total non spam	Spam	Spam 2	Total spam
Jan	0	0	1	1	0	0	0
Feb	0	0	0	0	0	0	0
Mar	0	0	0	0	0	0	0
Apr	0	0	0	0	0	0	0
May	0	0	1	1	0	1	1
Jun	0	0	6	6	0	502	502
Jul	0	550	152	702	6	566	572
				710			1, 075

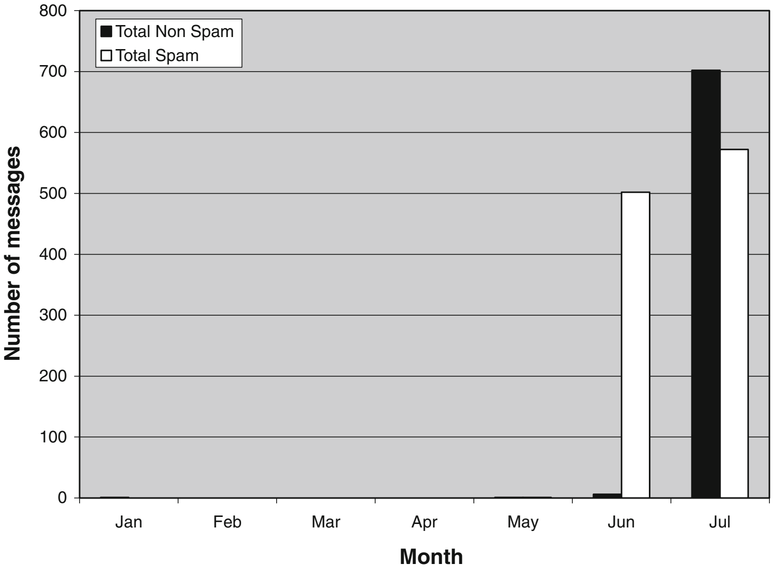


Fig. 12 Training data by month

5.3.1 Training data

The messages from January to July were chosen to be the training set. It was only in July and June that there were enough non spam and spam messages in the corpus for sufficient training. Training data contains 710 non-spam messages and 1,075 spam messages. The breakdowns are shown in Table 5 and Fig. 12.

5.3.2 Testing data

The messages from August to December were chosen to be the Testing set. It was in August and September that there were enough non spam and spam messages in the corpus for sufficient Testing. Testing data contains 3,437 non-spam messages and 689 spam messages. The breakdowns are shown in Table 6 and Fig. 13.

Table 6 Testing data by month

Month	Non spam				Spam		
	Easy ham	Easy ham 2	Hard ham	Total non spam	Spam	Spam 2	Total spam
Aug	423	843	34	1, 300	157	169	326
Sep	1, 283	0	28	1, 311	321	0	321
Oct	726	0	9	735	6	0	6
Nov	16	6	14	36	0	7	7
Dec	52	1	2	55	11	18	29
				3, 437			689

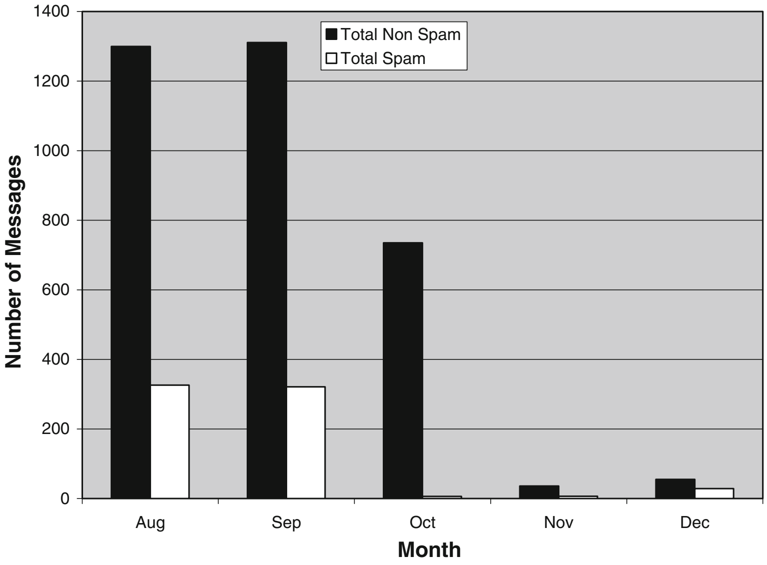


Fig. 13 Testing data by month. Note: Some files are corrupted so they are excluded

5.4 Spam detection using AIS

5.4.1 Initial training

In training phase, a number of lymphocytes were generated from the chosen library. These lymphocytes were trained on the messages form January to July (710 non-spam messages and 1,075 spam messages). Several sets of lymphocytes were created, trained and saved so that the same initial sets could be used with varying runtime parameters (Figs. 14, 15, 16).

5.4.2 Spam detection using AIS testing

In the testing phase, the messages form August to December were used in testing (Testing data contains 3,435 non-spam messages and 682 spam messages). In standard spam detection using AIS the following is not applied:

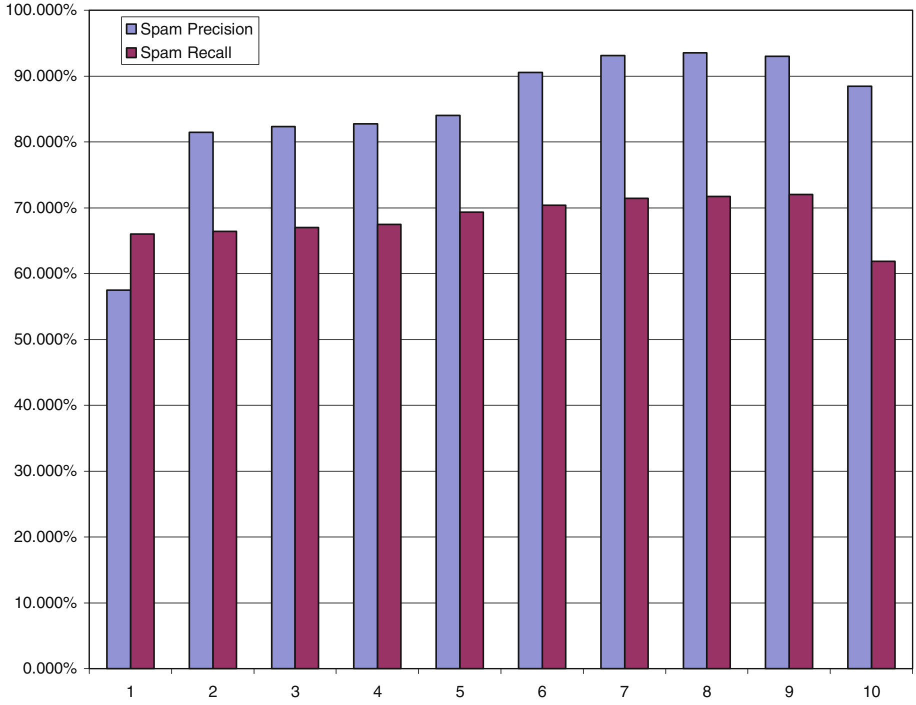


Fig. 14 Spam precision, recall results (AIS standard)

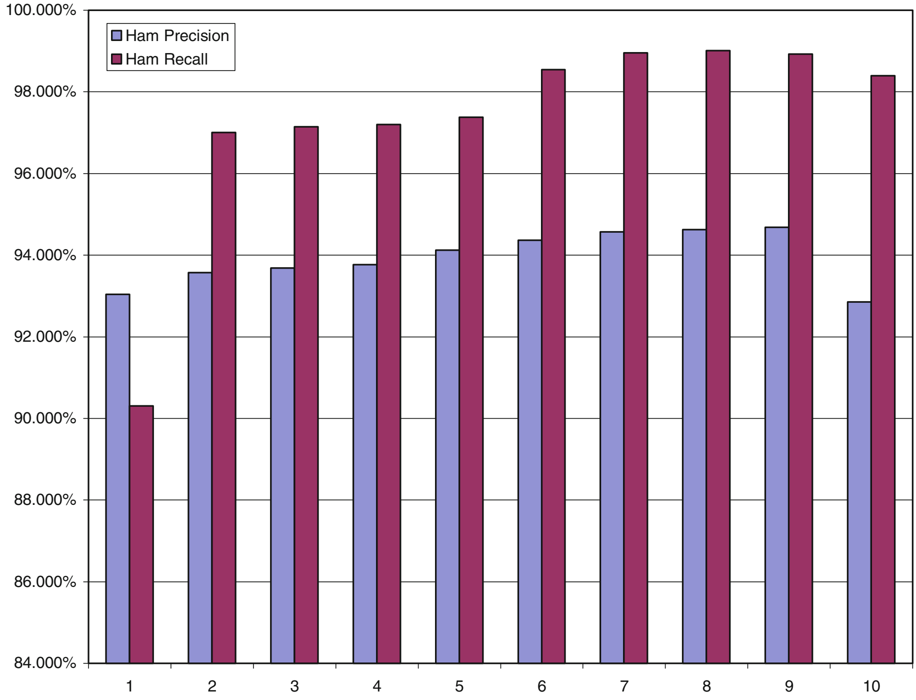


Fig. 15 Ham precision, recall results (AIS standard)

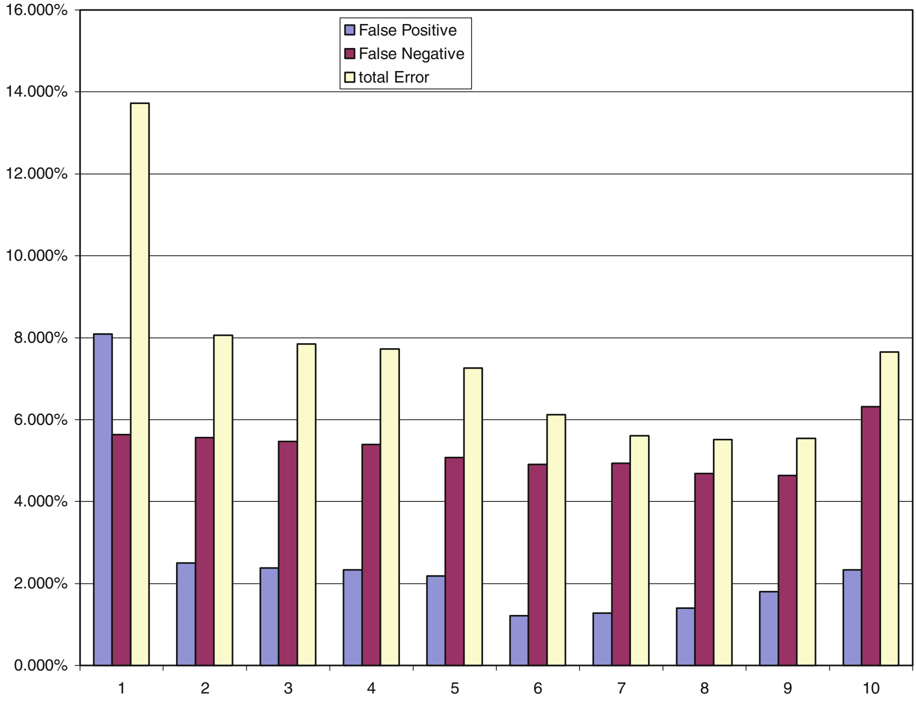


Fig. 16 False positive, false negative, total error results (AIS standard)

- Culling Based on GA.
- Rebuilding the system Based on GA.
- Adaptive weight of lymphocytes to replace useless lymphocytes with new promising lymphocytes.

5.4.3 Spam detection using AIS results

By testing the system using standard AIS the following results are got: Tables 7, 8 and 9).

Tables 7 and 8 show spam precision and recall, ham precision and recall respectively. With a smaller number of lymphocytes the percentage of spam precision and spam recall are low. However, an accepted value of percentage when the number of lymphocytes is between 500 and 900.

Based on the results in Table 9; an accepted false positive and false negative rates when the number of lymphocytes is between 500 and 900. The best results occur when the number of lymphocytes is 600.

5.5 Genetic optimized AIS spam detection

5.5.1 Testing

In testing phase, the messages form August to December were used in testing (Testing data contains 3,435 non-spam messages and 682 spam messages).

Table 7 Spam precision, recall results (AIS standard)

No of lymphocytes	Spam precision (%)	Spam recall (%)
100	57.471	65.982
200	81.475	66.422
300	82.342	67.009
400	82.734	67.449
500	84.014	69.355
600	90.566	70.381
700	93.117	71.408
800	93.499	71.701
900	92.992	71.994
1,000	88.470	61.877

Table 8 Ham precision, recall results (AIS standard)

No of lymphocytes	Ham precision (%)	Ham recall (%)
100	93.041	90.306
200	93.569	97.001
300	93.683	97.147
400	93.766	97.205
500	94.119	97.380
600	94.369	98.544
700	94.574	98.952
800	94.630	99.010
900	94.678	98.923
1,000	92.857	98.399

Table 9 False positive, false negative, total error results (AIS standard)

No of lymphocytes	False positive (%)	False negative (%)	Total error (%)
100	8.088	5.635	13.72
200	2.502	5.562	8.06
300	2.380	5.465	7.85
400	2.332	5.392	7.72
500	2.186	5.077	7.26
600	1.214	4.906	6.12
700	1.274	4.936	5.61
800	1.399	4.688	5.51
900	1.799	4.639	5.54
1,000	2.336	6.315	7.65

- For each message the lymphocytes were applied to the message and the weights associated with the lymphocytes were updated appropriately.
- If testing happened, culling may occur according to:

  - The number of message received.
  - The update interval defined by the user
  - Other parameters defined by the user.

- If culling occurred, new lymphocytes would be generated

In this research culling is applied based on:

- Number of messages received (using genetic algorithm).

The system was evaluated through using different numbers of lymphocytes and culling parameters.

Testing scores:

After testing the system Tables 10, 11, and 12 show the following results using a specific threshold:

Getting high values of spam precision means that there are few messages which are incorrectly classified as spam.

Figures 17 and 18 show spam precision and recall, ham precision and recall respectively. With fewer numbers of lymphocytes, the error is high for spam recall. This is not true with spam precision rate which remains significantly more constant.

Table 10 Spam precision, recall results (genetic optimized)

No of lymphocytes	Spam precision (%)	Spam recall (%)
100	97.603	65.689
200	93.028	74.340
300	92.014	77.713
400	78.451	81.672
500	96.538	69.501
600	91.986	77.419
700	93.333	73.900
800	93.333	73.900
900	93.333	73.900
1,000	97.863	67.155

Table 11 Ham precision, recall results (genetic optimized)

No of lymphocytes	Ham precision (%)	Ham recall (%)
100	93.603	99.680
200	95.101	98.894
300	95.707	98.661
400	96.331	95.546
500	94.264	99.505
600	95.653	98.661
700	95.024	98.952
800	95.024	98.952
900	95.024	98.952
1,000	93.861	99.709

Table 12 False positive, false negative, total error result (genetic optimized)

No of lymphocytes	False positive (%)	False negative (%)	Total error (%)
100	0.267	5.684	5.95
200	0.923	4.251	5.17
300	1.117	3.692	4.81
400	3.716	3.036	6.75
500	0.413	5.052	5.47
600	1.117	3.741	4.86
700	0.874	4.324	5.20
800	0.874	4.324	5.20
900	0.874	4.324	5.20
1,000	0.243	5.441	5.68

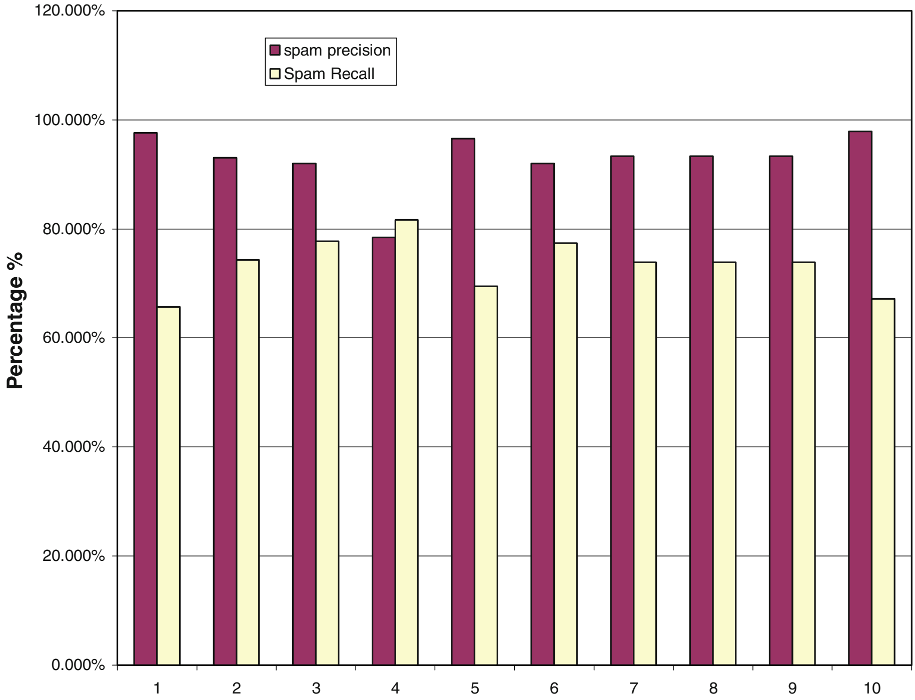


Fig. 17 Spam precision, recall results (genetic optimized)

The following table shows false positive and false negative Results.

According to Table 12. The false positive rate is low which is more important than the false negative rate. Getting a false negative rate which is greater than the false positive is acceptable for any users since all messages will appear in his regular mail. Misclassifying good messages are easy to be missed because the messages will be kept in the spam folder, which the user will not almost open or he may delete the messages before seeing them (Figs. 19, 20).

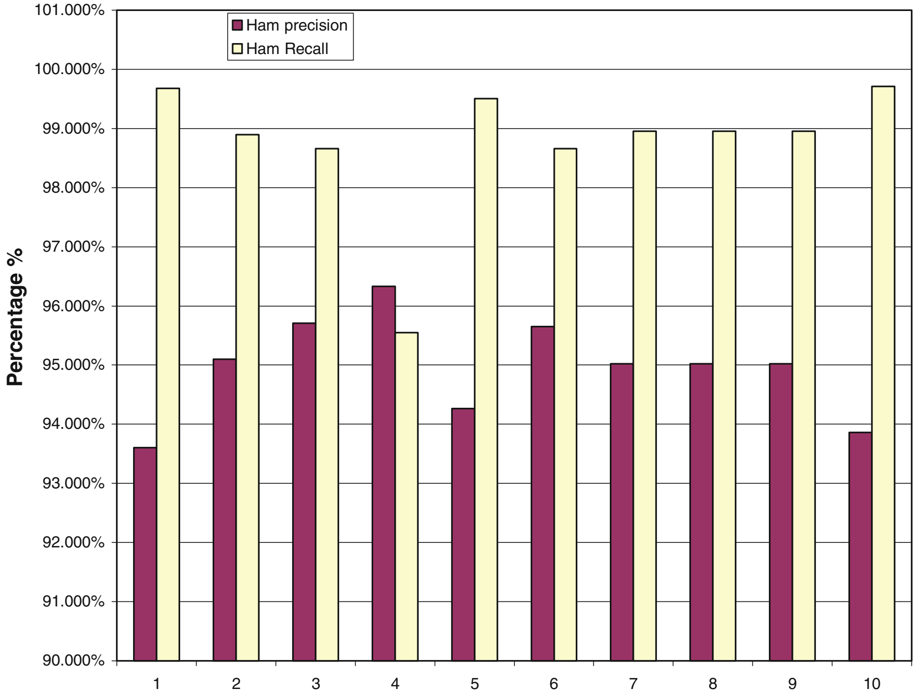


Fig. 18 Ham precision, recall results (genetic optimized)

5.5.2 Lymphocyte and accuracy

False positives are legitimate messages which have been tagged as spam by the system. False negatives are spam messages which have been tagged as ham messages by the system. This means that a false positive is more important than a false negative (Table 13). But this does not mean that a false negative is not important. If there is a classifier or a filter which can classify messages with a low ratio of a false positive and a high ratio of a false negative, which means that thousands of messages will appear in the inbox and this is a problem. The ultimate goal is to adapt between a false positive and a false negative to have minimum values of false positive and an accepted value of false negative.

According to Tables 12 and 13 the best results are when the number of lymphocytes used for spam detection is 600 Lymphocytes.

5.5.3 Rebuilding the system using GA (changing the self)

As mentioned before, one of the main contributions of genetic optimized AIS is that the ability that the self will change. In other words, the interests of any person are not stable because circumstances always change. Also, the message content and characteristics that any person would like to receive changed over time. So the system must be adapted to this change. The Genetic Algorithm is used to produce random Rebuild time instead of fixed period. For the evaluation purposes genetic algorithm is used to generate a number of messages which will be used for testing if there is a change in the self. Evaluation of the rebuilding is as the following:

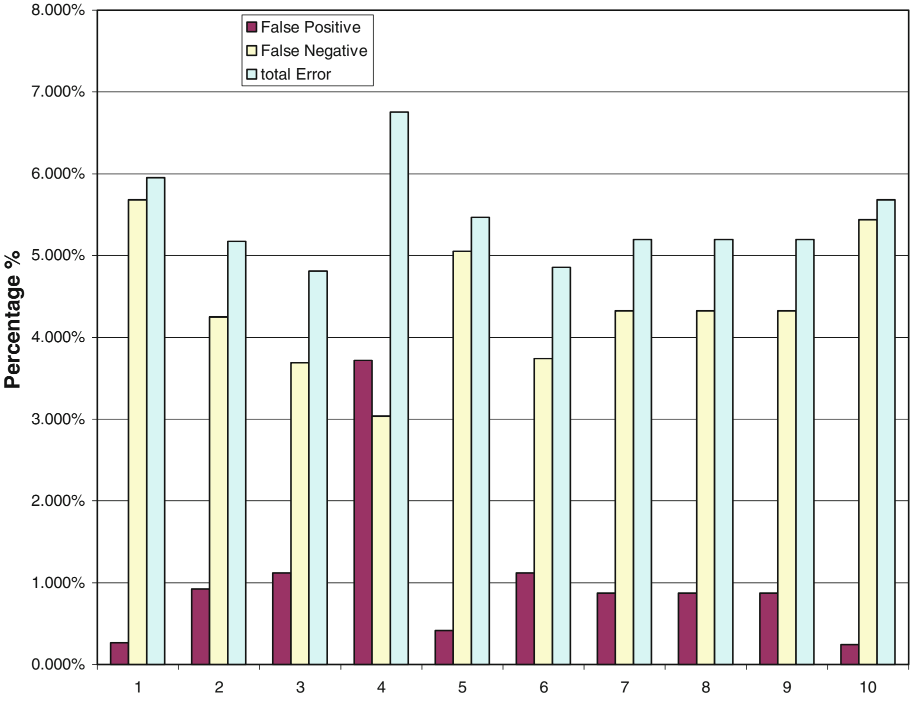


Fig. 19 False positive, false negative, total error results (genetic optimized)

- Running the system in a normal situation.
- Using GA to generate a guided random number of messages which will be used to check if there is a change on the self.
- When the system reaches that number of messages created by GA, Firstly, the threshold value is incremented to allow more messages to get into inbox. Then, the first 100 messages are taken as a sample for the evaluation purposes. All these messages which have a score between the old threshold (normal) and the new threshold (temporary) within these 100 messages will be considered as ham in the future. After that, these messages are put in a specific folder to be tested again using the old threshold (normal) to see whether the system will classify them as ham or not.
- Msg_Matched is incremented by the user define values as shown in Table 14.

Testing is done on 100 messages and the results as following:

Figure 21 shows the effect on accuracy when threshold is incremented by 0.03.

Figure 21 shows that there is no big effort of decrement values on the accuracy of results.

5.6 Spam detection using ANN results

After testing the system (Tables 15, 16, 17) the following results are appeared using a specific threshold:

Getting low values of spam recall means that there are several messages classified as legitimate incorrectly. This means that high values of false negative will appear.

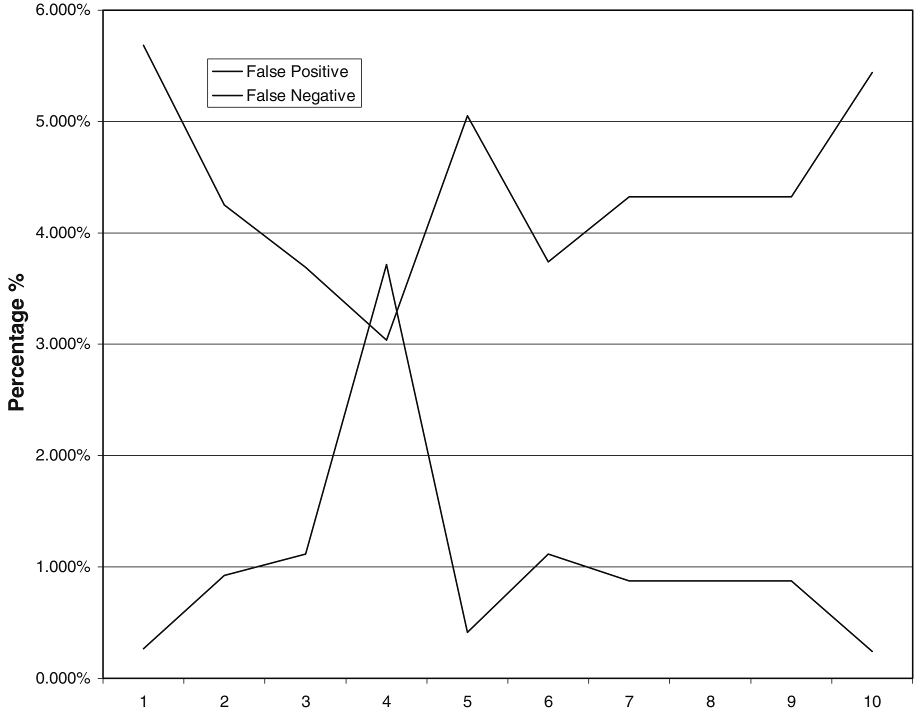


Fig. 20 Relations between the numbers of lymphocytes, false positive and false negative (genetic optimized)

Table 13 Relations between the number of lymphocytes, false positive and false negative (genetic optimized)

No of lymphocytes	False positive (%)	False negative (%)
100	0.267	5.684
200	0.923	4.251
300	1.117	3.692
400	3.716	3.036
500	0.413	5.052
600	1.117	3.741
700	0.874	4.324
800	0.874	4.324
900	0.874	4.324
1,000	0.243	5.441

Figures 22 and 23 show spam precision and recall, ham Precision and recall respectively. By using a smaller number of the neurons, the results are excellent for spam precision, ham precision, and ham recall. This is not true with spam recall rates (Figs. 24, 25).

This section shows using the generated neurons from a library of batches of 200, 300, 400, 500, 600, 700, 800, 900, and 1,000. Each one was tested against all the messages in the testing dataset. Table 16 summarizes the false positive and the false negative results using ANN.

Table 14 Rebuilding the system using GA (changing the self) (genetic optimized)

Threshold (increment)	No of lymphocytes	Increment (Msg_matched)	Number of messages which will be considered a ham in future	N (Leg-Leg)	N (Leg-Spam)	Accuracy (%)
0.05	600	3	85	84	1	98.82
0.05	600	2	93	92	1	98.92
0.05	600	1	97	92	5	94.85
0.04	600	3	88	83	5	94.32
0.04	600	2	81	80	1	98.77
0.04	600	1	77	76	1	98.70
0.03	600	3	57	56	1	98.25
0.03	600	2	88	87	1	98.86
0.03	600	1	78	76	2	97.44
0.02	600	3	40	39	1	97.50
0.02	600	2	78	77	1	98.72
0.02	600	1	55	53	2	96.36
0.01	600	3	40	39	1	97.50
0.01	600	2	33	32	1	96.97
0.01	600	1	37	35	2	94.59

Increment (Msg_matched) Incrementing Msg_Matched (see algorithm 3 in section 3.3.2)
Number of messages which will be considered a ham in future The number of messages which are classified as spam within the 100 messages used for evaluation
N (Leg-Leg) The number of legitimate messages which are classified as Legitimate
N (Leg-Spam) The number of legitimate messages which are classified as spam

Depending on the results in Table 17 the false positive rates which are very low are acceptable. On the other hand, false negative rates are tolerable.

5.6.1 Number of neurons and accuracy

There is a relationship between the number of neurons and accuracy (Table 18). You can notice that when the number of the neurons is increasing the results will be better. However, there is no need for having a high number of neurons since this will affect the performance. In our core modifications on ANN, excellent promising results with accepted values of neurons are got.

Modifications on ANN give excellent results even with a small number of neurons. Table 18 shows that false positive values are acceptable in all situations, whereas, false negatives swing within low boundaries.

According to Table 18 the best results are when the number of neurons used for spam detection, in relation with false positive and false negative, is 300 neurons. Also, good results are appeared when the number of the neurons is 600 while most other results for false negatives swing within a small boundary.

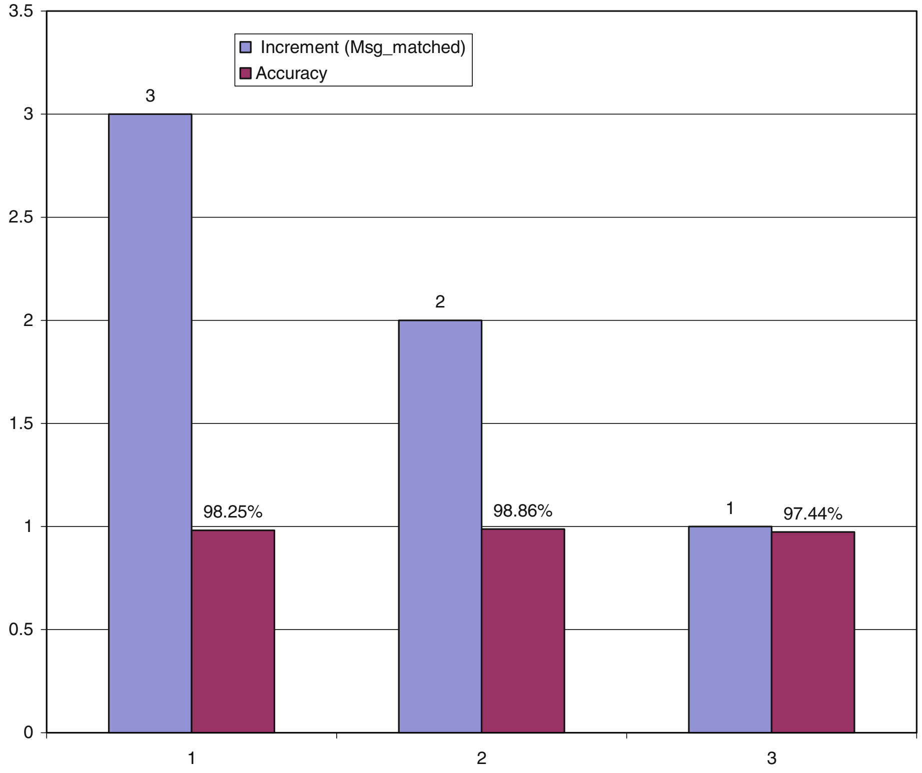


Fig. 21 Rebuilding the system using GA (changing the self) (genetic optimized)

5.7 Comparing standard AIS, genetic AIS and CLA_ANN

In this section a comparison of the results are shown (Table 19) of our experiments on SpamAssassin corpus. You can find that genetic optimized spam detection gives the best results using 600 lymphocytes with 1.117% false positive and 3.741% false negative. Furthermore, spam detection using AIS gives the best results when the number of lymphocytes is 600 with 1.214% false positive and 4.906 false negative.

The Table 20 shows that our modifications on ANN give promising results that could be used in the process of fighting against spam. The accepted false positive value is when the number of the neurons is 300. The best false positive value is found when the number of the neurons is 700.

In the following paragraph (Table 21) shows some commercial antispam solutions and their accuracy.[14], [15], [16]

[14] http://www.abaca.com/pr_2009_04_20.html. Accessed 1 Jun 2009.

[15] http://www.cloudmark.com/en/serviceproviders/authority-spamassassin.html. Accessed 1 Jun 2009), (https://trial.securecloud.com/imhs/. Accessed 1 Jun 2009.

[16] http://www.spamtitan.com/antispam?gclid=CKqc2qDu25oCFYuB3godDwJbdQ. Accessed 1 Jun 2009.

Table 15 Spam precision, recall result (ANN)

No of neurons	Spam precision (%)	Spam recall (%)
200	97.325	69.355
300	96.022	77.859
400	98.673	65.396
500	98.969	70.381
600	98.586	71.554
700	98.918	67.009
800	98.569	70.674
900	99.149	68.328
1,000	98.765	70.381

Table 16 Ham precision, recall result (ANN)

No of neurons	Ham precision (%)	Ham recall (%)
200	94.244	99.622
300	95.763	99.360
400	93.561	99.825
500	94.438	99.854
600	94.644	99.796
700	93.844	99.854
800	94.487	99.796
900	94.077	99.884
1,000	94.437	99.825

Table 17 False positive, false negative, total error results (ANN)

No of inputs	False positive (%)	False negative (%)	Total error (%)
200	0.316	5.077	5.392
300	0.534	3.668	4.202
400	0.146	5.732	5.878
500	0.121	4.906	5.028
600	0.170	4.712	4.882
700	0.121	5.465	5.587
800	0.170	4.858	5.028
900	0.097	5.247	5.344
1,000	0.146	4.906	5.052

5.8 Discussion

The spam immune system successfully adapts the biological immune system model to be used in spam detection. The overall results are good enough to be accepted (Sects. 5.4, 5.5 and 5.6).

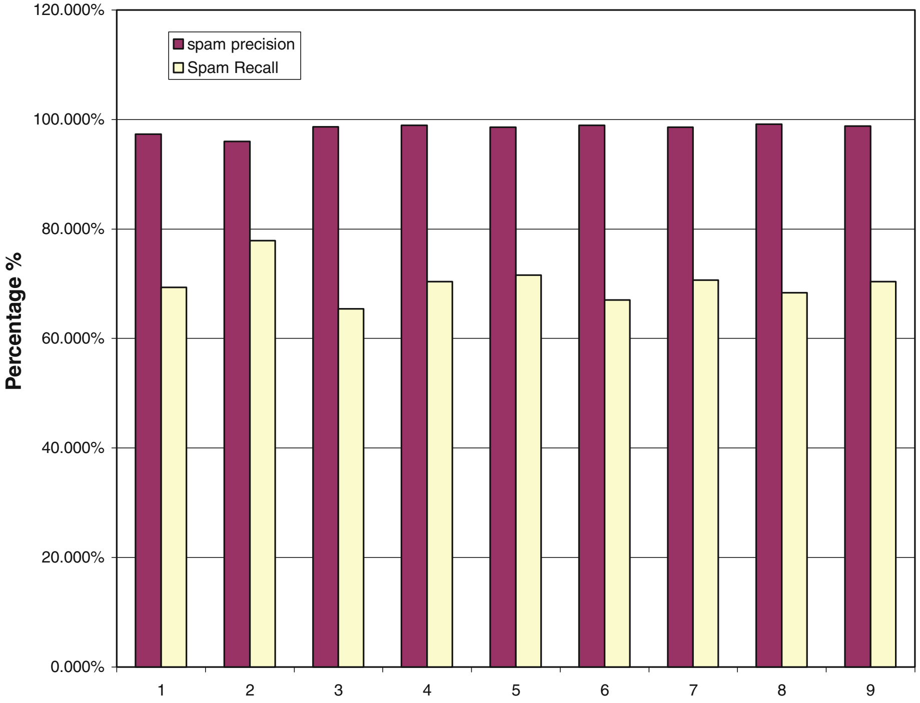


Fig. 22 Spam precision, recall results (ANN)

Moreover, by using just one single approach (the Immune system approach) to achieve such accuracy is promising, since most of the commercial antispam systems use several approaches such as origin based and content filtering methods. A combination of several distinct approaches can achieve a higher accuracy.

The immune system will tend to perform better while applying in user's personal e-mails. If the immune system is adapted to work on users' personal e-mails, with the help of genetic algorithm that gives the system the chance to rebuild (learn new thing) and accept some types of messages classified as spam to be classified as ham, it is thought that the system will perform in an amazing way since it will be adapted to work according to the users' interests.

Using GA in genetic optimized spam detection was helpful in determining the culling time and the rebuild time parameters instead of using fixed parameters. This means that GA gives us an opportunity to get different culling time and rebuild time. This is more realistic in real life because users' interests do not change regularly but can be suddenly changed according to any new happening.

Concerning the adaptive weight of lymphocytes; the adaptive weight of lymphocytes enables the system to replace old lymphocytes with new promising ones. This merit is necessary, useful and helpful in making the system work better; the results got using this approach are promising as it is succeeded in modifying the system to be able to work in similar way to the biological human system. As a result, it can learn new things about spam and therefore, it can develop itself without any intrusion from human. However, this advantage can not guarantee 100% of success without running the system for long periods.

Testing the system using different numbers of lymphocytes shows that the accuracy can perform in a good manner while using specific numbers of lymphocytes. In addition, it is

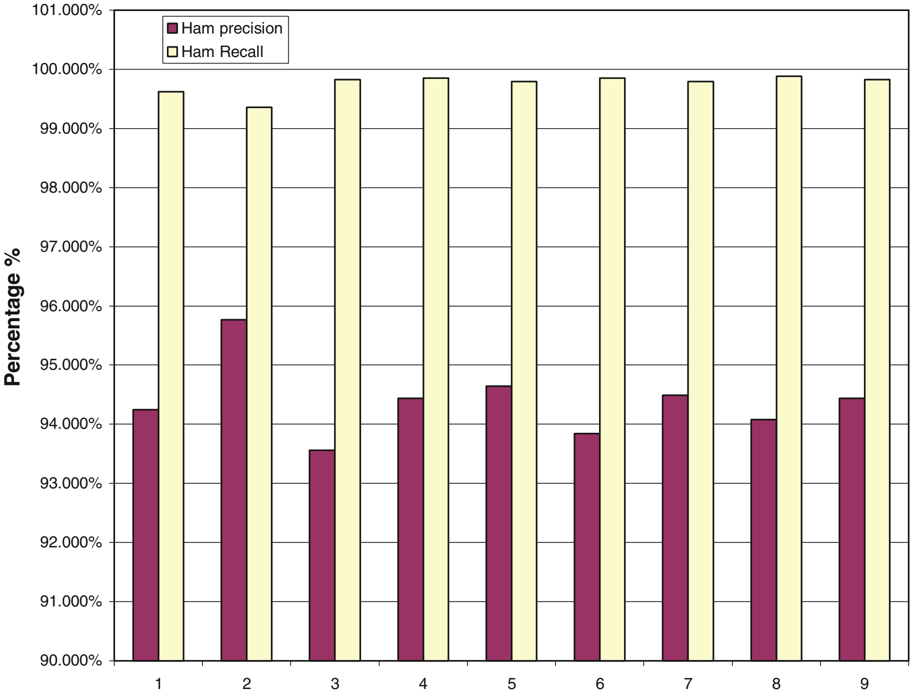


Fig. 23 Ham precision, recall results (ANN)

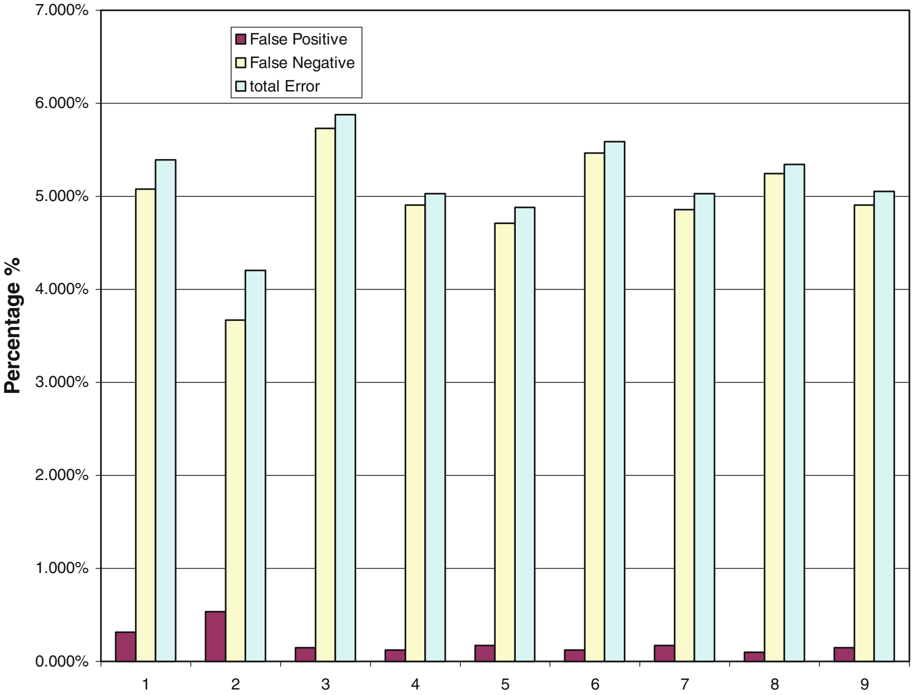


Fig. 24 False positive, false negative, total error results (ANN)

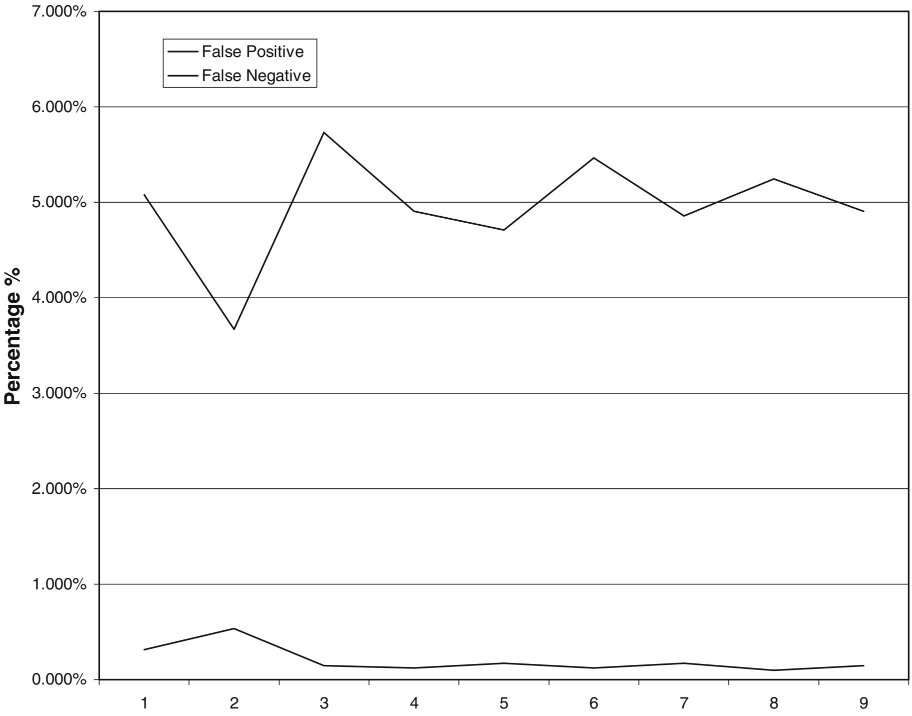


Fig. 25 Relations between neurons, false positive and false negative (ANN)

Table 18 Relations between neurons, false positive and false negative (ANN)

No of inputs	False positive (%)	False negative (%)
200	0.316	5.077
300	0.534	3.668
400	0.146	5.732
500	0.121	4.906
600	0.170	4.712
700	0.121	5.465
800	0.170	4.858
900	0.097	5.247
1,000	0.146	4.906

discovered that when the number of lymphocytes is small, a high percentage of numbers of false positive which is not acceptable at all are got. The ultimate goal of any research of spam detection search for an optimum value that balances between false positive and false negative.

In this work, the results are excellent since tables' shows very amazing results for false positive and an accepted value for false negative (Tables 19, 20). Taking in consideration that a balance between a false positive and a false negative must be done. The results summarized

Table 19 AIS, genetic optimized AIS summary

No of lymphocytes	AIS standard (%)		Genetic optimized AIS (%)	
	False positive	False negative	False positive	False negative
200	2.502	5.562	0.923	4.251
300	2.380	5.465	1.117	3.692
400	2.332	5.392	3.716	3.036
500	2.186	5.077	0.413	5.052
600	1.214	4.906	1.117	3.741
700	1.274	4.936	0.874	4.324
800	1.399	4.688	0.874	4.324
900	1.799	4.639	0.874	4.324
1,000	2.336	6.315	0.243	5.441

Table 20 CLA_ANN summary

No of neurons	CLA_ANN (%)	
	False positive	False negative
200	0.316	5.077
300	0.534	3.668
400	0.146	5.732
500	0.121	4.906
600	0.170	4.712
700	0.121	5.465
800	0.170	4.858
900	0.097	5.247
1,000	0.146	4.906

in the Tables 19 and 20 show that accepted values for both of false positives and false negatives are appeared.

The modifications on ANN give excellent results. Promising values when the number of neurons is 300. When the system can achieve such results with this low number of neurons this refers to a perfect performance which is amazing since everyone always look for a high performance with a minimum CPU usage.

In few words,both genetic optimized spam detection using AIS and spam detection using CLA_ANN are good enough to be used as an anti spam effective detection methods to fight spam and the program is ready to be adopted and used for commercial purposes.

6 Conclusions and future work

In spite of the considerable efforts to reduce and stop spam, e-mail users are still facing significant numbers of unsolicited and unwanted e-mails arriving in their inboxes. The difficulties in identifying spam e-mails arise mainly from the fact that spam is constantly changing; spammers try to make it resembles legitimate e-mails to allow it to bypass the filters used to

Table 21 Some commercial anti spam accuracy

Product name	Accuracy	Note
ABACA	99%	Abaca's Industry-Leading Anti
Cloudmark	Greater than 98% Accuracy	Cloudmark Authority
TREND Microo	Stop up to 99% of spam	Trend Micro Incorporated
SpamTitan	Antispam with 98.5% catch rate	SPAM TITAN

trap it. So current spam fighting solutions are far from perfect; for example, spam filters will always have the problem of false positives.

The strive to find more effective tools for distinguishing between interesting and non-interesting materials is still increasing. The observations and the results based on using the artificial immune system can help us guide or they can further deepen our understanding of the natural immune system.

It is now well known to every researcher that there are no techniques which can be claimed alone to be the ideal perfect solution with truly 0% false positive and 0% false negative. So currently used antispam systems couples several machine learning techniques for effective content classifications. This is for one main reason which is the definition of unsolicited E-mails varies from one to another.

Many of the approaches used in spam detection use multiple approaches, such as black-listing, white-listing...etc. However, if these approaches were added into a complete system combined with spam immune system, the results may then achieve a higher accuracy.

The results presented were encouraging but there are still a number of available options to optimize such a system. An increase in accuracy may be achieved by enhancing the system; a longer-term project which can be done would be to hybrid this method with a more traditional information retrieval technique such as a rule-based system.

This work shows that the spam problem is a complex system, and should be dealt with by developing strategies to holistically interact with it. Such strategies must embrace both technical and legal realities simultaneously in order to be successful. It has also been shown that a combination of more than one technique will perform better than any single technique.

One of the main advantages of this system is that genetic optimized AIS spam filtering allows the detection of not-previously-seen spam content, by exploiting its bulkiness.

The detection capability found of the ANN is good, but given that it has low but nonzero false positive spam recognition result rates (real messages incorrectly classified as spam) the ANN is not suitable to be used alone as a spam elimination tool. In fact any nonzero false positive spam detection rates are unacceptable because the rejected e-mails could be an important message for the recipient. Strategies that apply a combination of techniques, such as a NN with a whitelist, pattern recognition would yield better results (Puniškis et al. 2006).

The following paragraph summarizes the overall results:

- The standard AIS spam detection system achieves 1.214% false positive and 4.906% false negative using 600 lymphocytes.
- The optimized genetic spam immune using AIS system achieves a promising result using SpamAssassin public corpus with 1.117% false positive and 3.741% false negative using 600 Lymphocytes.
- The CLA_ANN also achieves a promising result using SpamAssassin public corpus with 0.534 % false positive and 3.668% false negative using only 300 neurons.

Genetic optimized spam detection using AIS results is better than standard spam detection using AIS since an adaptive weight of lymphocytes is used in genetic optimized approach. Also, GA is used in culling and in calculating rebuild time.

According to spam detection using CLA_ANN promising results are got because of using an adapted learning rate based on previous runs of the system.

By making a comparison among these three used methods in this workthat concentrates on three things which are: weakness, opportunity and future modifications as follows:

- Standard AIS
  - Standard AIS is not suitable when the number of lymphocytes is low. This means that accepted value of lymphocytes is necessary to get good results. Increasing the number of lymphocytes will affect the system performance since each time a message received the system must search for a match in the entire list of lymphocytes.
  - Standard AIS has an opportunity to be used and enhanced especially if it is combined with another approach.
  - According to future modifications on standard AIS. There are several modifications already done on this method in this work.
- Genetic Optimized Spam Detection using AIS
  - Genetic optimized AIS is not also suitable enough when the number of lymphocytes is small.
  - It functions well without being integrated with other methodologies such as source based methods or content filtering methods. Similar to what is being used in commercial antispam.
  - Future modifications on this method could be achieved by:
    - Dealing with images within the message and attachment files.
    - The To, From and Date fields: Contain information that can be useful.
    - Combining genetic optimized spam detection using AIS with source based filtering methods such as whitelists or blacklists could be helpful.
    - Taking in consideration the length of the e-mail could be helpful, since most of spam messages are long.
- Spam detection using CLA_ANN
  - It shows that it can give promising results at any number of neurons even if that number is small.
  - It has an excellent opportunity to be used as an antispam fighter even if it is not combined with another approach.
  - Modifications of future work could use a machine learning method to delete old and useless neurons and replace them with new adapted layers.

In general, the results are promising if compared with the commercial anti spam methods (Table 21) which almost used more than one single approach to combat spam.

This research study of e-mail classification should prove to be very useful for future research in the area of intelligent e-mail filtering and classifications. It has been found that the hybrid approach of a GA with AIS is very useful. Using the GA for optimizing some parameters can not only compete the existing approaches but can also perform better in some circumstances.

The following directions seem promising on the way to have more effective and efficient automatic e-mail classifications.

Future research on ANN will look for two directions. Firstly, including more useful features to be included within the system that are difficult for a generative model. Secondly, combining ANN with a generative model that use source base filtering techniques instead of content filtering could be promising.

In genetic optimized spam detection using AIS; promising false positive values are got when the number of lymphocytes is low, the related genetic optimized spam detection future research will pursue two directions. Firstly, trying to enhance the false negative results by combing this method with other methods such as the source based methods which would be helpful here and could enhance the value of false negative. Secondly, developing the system continuously to make it ready for any new tricks of spammers who do not stop thinking of how to defeat the system in order to achieve their goals. This can be achieved by including more features to the system so as to not be cheated by the various methods of spammers.

References

Abi-Haidar A, Rocha LM (2008) Adaptive spam detection inspired by the immune system. In: Bullock S, Noble J, Watson RA, Bedau MA (eds) Artificial life XI: Eleventh International Conference on the simulation and synthesis of living systems. MIT Press (in press)

Abu-Nimeh S, Nappa D, Wang X, Nair S (2007) A comparison of machine learning techniques for phishing detection, ACM, Pittsburgh

Agrawal R, Bayardo RJ, Srikant R (2000) Athena: mining-based interactive management of text databases. In: Proceedings of 7th international conference on extending database technology (EDBT'00). Konstanz, Germany, pp 365–379

Alexandru C, Researcher C (2009) BitDefender Antispam NeuNet Whitepaper, BitDefender AntiSpam Laboratory, www.bitdefender.com. Accessed 15 Jan 2009

Androutsopoulos I, Koutsias J, Chandrinos K, Paliouras G, Spyropoulos C (2000a) An evaluation of naive Bayesian antispam filtering. In: Potamias G, Moustakis V, van Someren M (eds) Proceedings of the workshop on machine learning in the new information age, 11th European conference on machine learning (ECML 2000). Barcelona, Spain, pp 9–17

Androutsopoulos I, Paliouras G, Karkaletsis V, Sakkis G, Spyropoulos C, Stamatopoulos P (2000b) Learning to filter spam e-mail: a comparison of a naive Bayesian and a memorybased approach. In: Zaragoza H, Gallinari P, Rajman M (eds) Proceedings of the workshop on machine learning and textual information access, 4th European conference on principles and practice of knowledge discovery in databases (PKDD 2000), Lyon, France, pp 1–13

Androutsopoulos I, Koutsias J, Chandrinos KV, Spyropoulos CD (2000c) An experimental comparison of naïve Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In: Proceedings of SIGIR, 2000. Annual ACM conference on research and development in information retrieval proceedings of the 23rd annual international ACM SIGIR conference on research and development in information, pp 160–167, 2000. ISBN:1-58113-226-3

Androutsopoulos I, Koutsias J, Chandrinos KV, Paliouras G, Spyropoulos CD (2000d) An evaluation of Naïve Bayesian antispam filtering. In: Proceedings of the workshop on machine learning in the new information age

Atkins S (2003) Size and cost of the problem. In: Proceedings of the fifty-sixth internet engineering task force (IETF) meeting. SpamCon Foundation

Bailey SP, Lieutenant, United States Navy B.S. (2006) Master Thesis, Illinois Institute of Technology, 1997 Submitted in partial fulfillment of the requirements for the degree of master of science in electrical engineering from the naval postgraduate school, neural network design on the src-6 reconfigurable computer

Balakrishnan K, Honavar V (2008) Evolutionary design of neural architectures, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.46.8973. Accessed 1 Jan 2008

Balthrop J, Forrest S, Glickman M (2002) Revisiting lisys: parameters and normal behavior. In: Proceedings of the congress on evolutionary computation, pp 1045–1050

Bart T, Binargrl S (2003) Learning spam: simple techniques for freely-available software. In: Proceedings of the Usenix annual technical conference, Freenix Track 2003, http://web.cecs.pdx.edu/~bart/papers/spam.pdf. Accessed 20 Jun 2008

Batista E (2001) A fight to Ban Cellphone Spam, WiredNews, 6 July 2000, available from http://www.wired.com/news/business/0,1283,37376,00.html; Spam: a new nuisance for wireless users, USAToday, 13 April 2001. Accessed 10 Jan 2008

Bekker S (2003) Spam to Cost U.S. Companies $10 Billion in 2003, ENT News, viewed May 11 2005, http://www.entmag.com/news/article.asp?EditorialsID=5651. Accessed 10 Feb 2009

Boulevard A, Ramon S (2008) An introduction to neural networks a white paper visual numerics, Inc. December 2004 Visual Numerics, Inc. 12657, CA 94583 USA, www.vni.com. Accessed 10 Oct 2008

Bradley D, Tyrrell A (2000) Immunotronics: Hardware fault tolerance inspired by the immune system. In: Proceedings of the 3rd international conference on evoluable systems (ICES2000). Springer, Berlin, vol 1801, pp 11–20

Brutlag JD, Meek C (2000) Challenges of the E-mail domain for text classification. In: Proceedings of the 17th international conference on machine learning. Stanford University, USA, pp 103– 110

Burnet FM (1959) The clonal selection theory of acquired immunity. Cambridge University Press. http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1522512. Accessed 10 Oct 2008

Burnet FM (1978) Clonal selection and after. In: Bell GI, Perelson AS, Pimbley GHJr (eds) Theoretical immunology. Marcel Dekker Inc., London, pp 63–85

Cao Y, Dasgupta D (2003) An immunogenetic approach in chemical spectrum recognition. In: Ghosh, Tsutsui (eds) Chap. 36 in the edited volume, Advances in Evolutionary Computing. Springer, Berlin

Cardoso-Cachopo A, Oliveira A (2003) An empirical comparison of text categorisation methods. In: Nascimento MA, de Moura ES, Oliveira AL (eds) Proceedings of conference on string processing and information retrieval. Springer, Berlin, pp 183–196

Carneiro J, Leon K, Caramalho Í, Dool C van den , Gardner R, Oliveira V, Bergman M, Sepúlveda N, Paixão T, Faro J et al (2007) When three is not a crowd: a cross regulation Model of the dynamics and repertoire selection of regulatory CD4 T cells. Immunol Rev 216(1):48–68

Carol MC, Prodeus AP (1998) Linkages of innate and adaptive immunity. Curr Opin Imm 10:36–40

Carpinter J, Ray H (2006) Tightening the net: a review of current and next generation spam filtering tools. Department of Computer Science and Software Engineering, University of Canterbury, Christchurch, New Zealand Computers & Security 566–578, Journal homepage: www.elsevier.com/locate/cose, 25. Elsevier

Carreras X, Andm'Arquez L (2001) Boosting trees for antispam e-mail filtering. In: Proceedings of RANLP-2001, 4th international conference on recent advances in natural language processing. www.lsi.upc.edu/~carreras/pub/boospamev.ps. Accessed 15 Dec 2008

Chhabra S (2005) Master Thesis, "Fighting Spam, Phishing and E-mail Fraud". A Thesis submitted in partial satisfaction of the requirements for the degree of Master of Science in Computer Science, University of California, Riverside

Chhabra S, Yerazunis WS, Siefkes C (2004) Spam filtering using a markov random field model with variable weighting schemas. In: Proceedings of the fourth IEEE international conference on data mining, pp 347–350

Christianini N, Shawe-Taylor J (2000) An introduction to support vector machines: and other kernel-based learning methods. Cambridge University Press, Cambridge. http://cambridge.org/uk/catalogue/catalogue.asp?isbn=9780521780193. Accessed 10 Jan 2009

Chuan Z, Xianliang L, Mengshu H, Xu Z (2004) A LVQ-based neural network anti-spam e-mail approach. College of Computer Science and Engineering of UEST of China, Chengdu, China 610054

Chuan Z, Xianliang L, Mengshu H, Xu Z (2005) A LVQ-based neural network anti-spam e-mail approach January ACM SIGOPS. Oper Syst Rev 39(1)

CipherTrust Inc (2004) Controlling Spam The IronMail® Way, http://wwwciphertrustcom/files/forms/landing_template.php?sp. Accessed Jan 2008

Çiltik A (2006) Master Thesis, Time efficient spam e-mail filtering for turkish. Submitted to the Institute for Graduate Studies in Science and Engineering in partial fulfillment of the requirements for the degree of Master of Science in Computer Engineering

Clark J (2000) PhD thesis, University of Sydney, Australia. E-mail classification: a hybrid approach combining genetic algorithm with neural networks

Clark J, Koprinska I, Poon J (2003) A neural network based approach to automated e-mail classification. Proc IEEE/WIC Int Conf 13(17):702–705

Cohen WW (2008) Learning rules that classify e-mail. In: Proceedings of the 1996 AAAI spring symposium on machine learning in information access, California. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.4129. Accessed 20 Dec 2008

Colaco C (1998) Acquired wisdom in innate immunity. Imm Today 19(1):50, http://www.sciencedirect.com/science?. Accessed 10 Jan 2009

Cook D (2006) Catching spam before it arrives: domain specific dynamic blacklists. In: Proceedings of the 2006 Australasian workshops on grid computing and e-research. ACM, Australian Computer Society, vol 54, pp 193–202. ISBN~ISSN:1445-1336, 1-920-68236-8

Cristiani N, Scholkopf B (2002) Support vector machines and kernel methods: the new generation of learning machines. AI Mag 23(3):31–41

Dalvi N, Domingos P, Mausam, Sanghai Sumit, Verma Deepak (2004) Adversarial classification. In: KDD '04: Proceedings of the 2004 ACM SIGKDD international conference on knowledge discovery and data mining. ACM Press, New York, pp 99–108

Damiani E, De Capitani di Vimercati S, Paraboschi S, Samarati P (2004) An open digest-based technique for spam detection. In: Proceedings of the 2004 international workshop on security in parallel and distributed systems

Damiani E, Vimercati SDCD, Paraboschi S, Samarati (2004) An open digest-based technique for spam detection. Paper presented to the 2004 international workshop on security in parallel and distributed systems, San Francisco, CA

Dasgupta D (2006) University of Memphis, USA. IEEE Comput Intell Mag 1(4):40–49

Dasgupta D (1999) Immune-based intrusion detection system: a general framework. In: Proceedings of the 22nd national information systems security conference (NISSC), http://scholar.google.com/scholar?hl=ar&lr=&q=Immune-based+intrusion+detection+system%3A+A+general+framework. Accessed 10 Oct 2008

Dasgupta D, Forrest S (1996) Novelty detection in time series data using immunology. In: ISCA 5th international conference on intelligent systems, Reno, Nevada

Dasgupta D, KrishnaKumar K, Wong D, Berry M (2004) Negative selection algorithm for aircraft fault detection. In: The proceedings of the third international conference, ICARIS 2004 on artificial immune systems, Catania, Sicily, Italy

de Castro LN (2001) An introduction to the artificial immune systems, ICANNGA, 2001, Prague

de Castro LN (2003) Artificial immune system as novel soft computing paradigm. Soft Comput J 7(7)

de Castro LN, Von Zuben FJ (1999) Artificial immune Systems: part I—basic theory and applications, Technical Report TR—DCA 01/99

de Castro, Timmis J (2002) Artificial immune systems: a novel paradigm to pattern recognition. Computing Laboratory University of Kent at Canterbury Kent, Canterbury, UK, In: Corchado JM, Alonso L, Fyfe C (eds) Artificial neural networks in pattern recognition, SOCO-2002, University of Paisley, UK, pp 67–84

de Castro LN, Von Zuben FJ (2009) Technical report DCA-RT 02/00 February, 2000 artificial immune systems: part II. A survey of applications, http://scholar.google.com/scholar?hl=ar&lr=&q=Technical+Report+DCA-.pdf. Accessed 10 Jan 2009

De Castro LN, Timmis (2002) Artificial immune system, a new computational intelligence approach. Springer, Berlin, http://books.google.com/books?hl=. Accessed 10 Jan 2008

De Castro LN, Timmis (2002) Artificial immune system, a novel paradigm for pattern recognition, In: Alonso L, Corchado J, Fyfe C (eds) Artificial neural network in pattern recognition. University of Paisley, pp 67–84

Deepak P, Parameswaran S (2005) Spam filtering using spam mail communities. In: Proceedings. The 2005 symposium on applications and the internet, pp 377–383

Delany SJ (2006) PhD Thesis. Using case-based reasoning for spam filtering. A thesis submitted to the Dublin Institute of Technology in fulfillment of the requirements for the degree of Doctor of Philosophy School of Computing, Dublin Institute of Technology

Dimmock N, Maddison I (2004) Peer-to-peer collaborative spam detection. ACM Student magazine, Issue 11.2, Spam, winter

Dorigo M (1992) Optimization, learning and natural algorithms. PhD thesis, DEI, Politecnico di Milano, Italia (in Italian)

Dorigo M (1999) Ant algorithms for discrete optimization. Artif Life Spring 5(2):137–172

Drakeand CE, Oliver JJ, Koontz EJ (2004) Anatomy of a phishing E-mail. In: Proceedings of the first conference on E-mail and Anti-spam (CEAS)

Drewes R (2002) An artificial neural network spam classifier, Rich Drewes, viewed May 8 2005, http://www.interstice.com/drewes/cs676/spamnn/spam-nn.html. accessed 1 Jan 2009

Drucker H, Wu D, Vapnik VN (1999) Support vector machines for spam categorization. IEEE transactions on neural networks (FANO, R. 1961). Transmiss Inf 10(5):1048–1054, doi:10.1109/72.788645

Drucker H, Wu D, Vapnik V (1999) Support vector machines for spam categorisation. IEEE Trans Neural Netw 10(5):1048–1055

Dumais S, Platt J, Heckerman D, Sahami M (1998) Inductive learning algorithms and representations for text categorisation. In: Proceedings of ACM 7th international conference on information and knowledge management (CIKM 98). ACM Press, London, pp 148–155

Fearon DT, Locksley RM (1996) The instructive role of innate immunity in the acquired immune response. Science 272:50–53

Forrest S, Hofmeyr S, Somayaji A (1997) Computer immunology. Commun ACM 40(10):88–96

Garrett SM (2005) Computational biology group. How do we evaluate artificial immune systems? By the Massachusetts Institute of Technology Evolutionary Computation. Department of Computer Science, University of Wales, Aberystwyth, Wales, vol 13(2), pp 145–178

Gauthronet S, Drouard E (2001) Unsolicited commercial communications and data protection, Commission of the European Communities. http://books.google.jo/books?id=_H9sBl9eN5sC&pg. Accessed 20 Dec 2008

Gee K (2003) Using latent semantic indexing to filter spam, SAC 2003, Florida, USA. In: Proceedings of the 2003 ACM symposium on applied computing, pp 460–464, 2003. ISBN: 1-58113-624-2

Goodman J, Cormack GV, Heckerman D (2007) Spam and the Ongoing Battle. Commun ACM 50(2)

Graham (2003) Better Bayesian filtering. Paper presented to 2003 spam conference, vol 167. In: Proceedings of the 2006 Australasian workshops on Grid computing and e-research. ACM international conference proceeding series, vol 54, pp 193–202

Graham P (2009) A plan for spam by, http://www.paulgraham.com/spam.html. Accessed 2 Jan 2009

Graham-Cumming J (2005) People and spam. In: MIT spam conference, 2005. http://www.jgc.org/pdf/spamconf2005.pdf. Accessed 1 Jan 2009

Gulyás C (2006) Master Thesis. Creation of a Bayesian network-based meta spam filter, using the analysis of different spam filters, Budapest, 16th May 2006

Hagan MT, Demuth HB, Beale MH (2002) Neural network design. University of Colorado at Boulder, Boulder

Hawley AE (1997) Taking spam out of your cyberspace diet: common law applied to bulk unsolicited advertising via electronic mail. Univ Missouri Kansas City Law Rev 66:381–423

Hershkop S (2006) PhD Thesis. Partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Graduate School of Arts and Sciences, Columbia University, Behavior-based e-mail analysis with application to spam detection (submitted)

Hofmeyr SA, Forrest S (2000) Architecture for an artificial immune system. Evolut Comput 8(4):443–473

Hulten G, Goodman J, Rounthwaite R (2004) Filtering spam e-mail on a global scale'. In: Proceedings of the 13th international world wide web conference on alternate track papers \ posters. ACM Press, New York, pp 366–367

Itskevitch J (2001) Master Thesis. A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in the School of Computing Science, Simon Fraser University, Automatic hierarchical e-mail classification using association rules

Janeway CA Jr (1992) The immune system evolved to discriminate infectious nonself from noninfectious self. Imm. Today 13(1):11–16

Janeway CA Jr (1993) How the immune system recognizes invaders. Scientif Am 269(3):41–47

Janeway CA Jr, Travers P (1997) Immunobiology the immune system in health and disease. Artes Médicas (in Portuguese), 2nd edn

Joachims T (1998) Text categorization with support vector machines: learning with many relevant features. In: Nedellec C, Rouveirol C (eds) Proceedings of ECML-98, 10th European conference on machine learning, number 1398 in LNCS. Springer, Heidelberg, pp 137–142

Joachims T (2001) A statistical learning model of text classification with support vector machines. In: Proceedings of the 24th ACM international conference on research and development in information retrieval. ACM Press, London, pp 128–136

Khong W-K (2001) Master Thesis. The law and economics of junk e-mails (spam). Thesis submitted to the Erasemus Programme In Law And Economics In Partial Fulfilment of the Degree of European Master In Law And Economics

Khorsi A (2007) An overview of content-based spam filtering techniques. Informatica 31:269–277

Kim J, Bentley P (2002) Toward an artificial immune system for network intrusion detection: an investigation of dynamic clonal selection. In: Proceedings of the 2002 congress on evolutionary computation (CEC 2002), Honolulu, pp 1244–1252

Kim J, Wilson WO (2005) Uwe Aickelin, and Julie McLeod. Cooperative Automated Worm Response and Detection Immune ALgorithm (CARDINAL) Inspired by T-cell Immunity and Tolerance. In the proceedings of the fourth international conference, ICARIS 2005 on Artificial Immune Systems, Banff, Alberta, pp 168–181

Kolcz A, Alspector J (2001) Svm-based filtering of e-mail spam with content-specific misclassification costs. In: Proceedings of the TextDM'01 workshop on text mining—held at the 2001 IEEE international conference on data mining, TextDM'2001 (IEEE ICDM-2001 workshop on text mining), IEEE, pp 123–130

Krishnakumar K, Neidhoefer J (1999) Immunized adaptive critic for an autonomous aircraft control application. In: Artificial immune systems and their applications, Chap. 20. Springer, Berlin, pp 221–240

Livingston B (2001) Inside the spammer's world. http://news.com.com/2010-1071-281499.html. Accessed 15 Jan 2009

Liu X-b, Zhang N (2006) Incremental immune-inspired clustering approach to behavior-based anti-spam technology. Int J Inf Technol 12(3)

María J, Cajigas G, Puertas E (2006) Content based SMS spam filtering, DocEng'06, October 10–13. ACM, Amsterdam, 1-59593-515-0/06/0010

María J, Cajigas G, Puertas E (2006) Content based SMS spam filtering, DocEng'06, October 10–13, 2006. ACM, Amsterdam, 1-59593-515-0/06/0010

Medlock B (2003) Master Thesis, A Generative, Adaptive Language model Approach to spam filtering. Thesis Submitted in part fulfillment of the University of Cambridge MPhil degree in computer speech, text and internet technology

Medzhitov R, Janeway CA Jr (1997) Innate immunity: impact on the adaptive immune response. Curr Opin Imm 9:4–9

Medzhitov R, Janeway CA Jr (1997) Innate immunity: the virtues of a nonclonal system of recognition. Cell 91:295–298

Medzhitov R, Janeway CA Jr (1998) Innate immune recognition and control of adaptive immune responses. Semin Imm 10:351–353

Messaging Anti-Abuse Working Group (2006) MAAWG E-mail metrics Program, First Quarter 2006 Reports. www.maawg.org/about/FINAL_1Q2006_Metrics_Report.pdf. Accessed 10 Jan 2009

Mitchell T (1997) Machine learning. Commun ACM 42(11):30–36. ISSN: 0001-0782, 1999

Millerk C (2009) Group product manager, enterprise e-mail security, neural network-based antispam heuristics, http://neuro.bstu.by/our/antispam.pdf. Accessed 10 Jan 2009

Miller C (2008) Group product manager, enterprise e-mail security, neural network-based antispam heuristics, http://wwwciphertrustcom/files/forms/landing_template.php?sp. Accessed 10 Jan 2008

Neal M (2003) Meta-stable memory in an artificial immune network. In: The proceedings of second international conference on artificial immune systems (ICARIS), Napier University, Edinburgh

O'Brien C, Vogel C (2003) Spam filters: bayes vs. chi-squared; letters vs. words. In: ISICT '03: Proceedings of the 1st international symposium on information and communication technologies. Trinity College, Dublin, pp 291–296

Oda T (2003) Tony White, increasing the accuracy of a spam-detection immune system. In: Proceedings of the congress on evolutionary computation (CEC 2003), Canberra, Australia. Proceedings, vol 1, pp 390–396, http://terri.zone12.com/doc/academic/spam_cec2003.pdf. Accessed 1 Feb 2008

Oda T, White T (2003) Developing an immunity to spam, Genetic and evolutionary computation conference, Chicago, IL. In: Proceedings, part I series: lecture notes in computer science, vol 2723, pp 231–242. Springer, Berlin

Oda T, White T (2005) Immunity from spam: an analysis of an artificial immune system for junk e-mail detection. In: Proceedings of artificial immune systems: 4th international conference, ICARIS, Banff, AB, Canada, August 14–17, 2005. Lecture notes in computer science 3627. Springer, Berlin, pp 276–289

Ozgur L, Gungor T, Gurgen F (2004) Adaptive anti-spam filtering for agglutinative languages: a special case for Turkish, http://www.cmpe.boun.edu.tr/~gungort/papers/Adaptive%20Anti-Spam.pdf. Accessed 10 Oct 2008

Pantel P, Lin D (1998) Spamcop: a spam classification & organization program. In: Learning for text categorization: Papers from the 1998 Workshop. AAAI Technical Report WS-98-05, Madison, WI

Parish CR, O'Neill ER (1997) Dependence of the adaptive immune response on innate immunity: some questions answered but new paradoxes emerge. Imm Cell Biol 75:523–527

Pascucci A (2006) Toward a PhD Thesis on Pattern Recognition http://www.dis.uniroma1.it/~dottorato/db/relazioni/relaz_pascucci_1.pdf. Accessed 10 Jan 2009

Pelletier L, Almhana J, Choulakian V (2004) Adaptive filtering of spam. In: Proceedings. Second annual conference on communication networks and services research, pp 218–224

Pogula Sridhar S (2005) Master Thesis developing neural network applications using labview, A thesis presented to the faculty of the Graduate School University of Missouri-Columbia. In Partial Fulfillment Of the Requirements for the Degree Master of Science, Dr. Robert W. McLaren, Thesis Supervisor

Postel J (1975) RFC706: on the junk mail problem. Technical report, Network Working Group, http://www.faqs.org/rfcs/rfc706.html. Accessed 20 Feb 2008

Puniškis D, Laurutis R, Dirmeikis R (2006) An artificial neural nets for spam e-mail recognition. Electron Electric Eng 5(69):1392–1215. http://www.ee.ktu.lt/journal/2006/5/1392-1215-2006-05-02-73.pdf. Accessed 10 April 2008

Puniškis D, Laurutis R, Dirmeikis R (2008) An artificial neural nets for spam e-mail recognition, 2006, http://www.ee.ktu.lt/journal/2006/5/1392-1215-2006-05-02-73.pdf. Accessed 10 Jan 2008

Saarinen J (2003) Spammers hit below men's belts. http://www.nzherald.co.nz/storydisplay.cfm?storyID=3518097. Accessed 20 Feb 2009

Sahami M, Dumais S, Heckerman D, Horvitz E (1998) A Bayesian approach to filtering junk e-mail. In learning for text categorization: Papers from the 1998Workshop. AAAI Technical Report WS-98-05, Madison, WI. http://www.aaai.org/Library/Workshops/1998/ws98-05-009.php. Accessed 1 Dec 2008

Sarafijanovic S, Le Boudec J-Y (2007) Artificial immune system for collaborative spam filtering, EPFL, Switzerland, Technical Report LCA-REPORT-2007-008, EPFL

Scavenger (2003) Master Thesis. A junk mail classification program. A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science, Department of Computer Science and Engineering, College of Engineering, University of South Florida, January 20, 2003

Schneider K (2003) A comparison of event models for naive bayes antispam e-mail filtering. In: Proceedings of the 11th conference of the European chapter of the association for computational linguistics (EACL'03) http://scholar.google.com/scholar?hl=ar&lr=&q=Schneider.+K.+2003.+A+comparison+of+event+models+for+naive+bayes+antispam+e-mail+filtering. Accessed 20 Nov 2008

Sebastiani F (2002) Machine learning in automated text categorization. ACM Comput Surv 34(1):1–47

Secker A, Freitas AA, Timmis J (2003) Computing Laboratory University of Kent AISEC. An artificial immune system for e-mail classification, vol 1, pp 131–138, ISSN: ISBN: 0-7803-7804-0

Simon H (1983) Why should machines learn? An artificial intelligence approach. Mach Learn 1:392–399, http://books.google.com/books?hl=ar&lr=&id. Accessed 1 Jan 2009

Soonthornphisaj N, Chaikulseriwat K, Tang-On P (2002) Anti-spam filtering: a centroid-based classification approach. In: 2002 6th International conference on signal processing, vol 2, pp 1096–1099, 26–30 Aug. 2002

Stuart I, Cha S-H, Tappert C (2004) A neural network classifier for junk e-mail. In: Proceedings of student/faculty research Day, CSIS, Pace University

Timmis J, Knight T, De Castro LN, Hart E (2004) An overview of artificial immune systems. In: Paton R, Bolouri H, Holcombe M, Parish JH, Tateson R (eds) Computation in cells and tissues: perspectives and tools for thought. Natural computation series. Springer, Berlin, pp 51–86

Timmis J, Neal M, Hunt J (1999) An artificial immune system for data analysis. In: The proceedings of the international workshop on intelligent processing in cells and tissues (IPCAT)

Timmis J, Neal M, Knight T (2002) AINE: machine learning inspired by the immune system. IEEE Trans Evolut Comput 1(4):40–49

Tonegawa S (1983) Somatic generation of antibody diversity. Nature 302:575–581

Tretyakov K (2004) Machine learning techniques in spam filtering. Institute of Computer Science, University of Tartu, Data mining problem-oriented Seminar, MTAT.03.177, pp 60–79

Vapnik V (1999) The nature of statistical learning theory. Statistics for engineering and information science, 2nd edn. Springer, New York (Originally published as a monograph 2nd ed., 2000, XIX, 314 p. 48 illus., Hardcover ISBN: 978-0-387-98780-4, 2000)

Vinther M (2002) Intelligent junk mail detection using neural networks, http://www.logicnet.dk/reports/JunkDetection/JunkDetection.htm. Accessed 10 Jun 2008

Weinberg GR (2005) Master Thesis. To the engineering systems division in partial fulfillment of the requirements for the Degree of Master of Science in Technology And Policyat The Massachusetts Institute of Technology. A system analysis of the spam problem, Massachusetts Institute Of Technology (submitted)

Whitley D (2009) Genetic algorithms and neural networks. Genetic Algorithms Eng Comput Sci, http://scholar.google.com/scholar?hl=ar&lr=&q=Genetic+Algorithms+and+Neural+Networks.+Genetic+Algorithms+in+Engineering+and+Computer+Science. Accessed 10 Jan 2009

Wiehes A (2005) Master Thesis, Comparing anti spam methods, Master of Science in Information Security, Department of Computer Science and Media Technology, Gjøvik University College

Yao X (1999) Evolving artificial neural networks. Proc IEEE 87(9):1423–1447

Yoshida K, Adachi F, Washio T, Motoda H, Homma T, Nakashima A, Fujikawa H, Yamazaki K (2004) Density-based spam detector. In: Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining. ACM Press, Seattle, pp 486–93

Yue X, Abraham A, Chi Z-X, Hao Y-Y, Mo H (2006) Artificial immune system inspired behavior-based anti-spam filter. Springer, Berlin, vol 11(8)

Zhang I, Yao T (2003) Filtering junk mail with a maximum entropy model. In: Proceedings of 20th international conference on computer processing of oriental languages (ICCPOL03), pp 446–453. Received January 2004; revised August 2004; accepted August 2004. ACM Transactions on Asian language information processing, vol 3, No. 4, Dec. 2004

Zhang L, Zhu J, Yao T (2004) Natural Language Processing Laboratory, Institute of Computer Software & Theory Northeastern University, An Evaluation of Statistical Spam Filtering Techniques. ACM Trans Asian Language Inf Process 3(4):243–269

Zhang L, Zhu J, Yao T (2004) An evaluation of statistical spam filtering techniques. ACM Trans Asian Language Inf Process (TALIP) 3(4):243–269

Zhan C, Xianliang L, Mengshu H, Xu Z (2005) A lvq-based neural network anti-spam e-mail approach. SIGOPS Oper Syst Rev 39(1):34–39

Zhou D, Burges CJC, Tao T (2007) Transductive link spam detection. Microsoft research one microsoft way Redmond, WA 98052. ACM, AIRWeb '07, Banff, Alberta 978-1-59593-732-2