The Design and Development of Spam Risk Assessment Prototype: *In Silico* of Danger Theory Variants

Kamahazira Zainal
Faculty of Science and Technology
Universiti Sains Islam Malaysia (USIM)
Nilai, N.Sembilan, Malaysia

Mohd Zalisham Jali
Faculty of Science and Technology
Universiti Sains Islam Malaysia (USIM)
Nilai, N.Sembilan, Malaysia

***Abstract*—Now-a-days, data is flowing with various types of information and it is absolutely enormous and moreover, it is in unstructured form. These raw data is meaningless unless it is processed and analyzed to retrieve all the valuable and meaningful information. In this paper, a design and principal functionalities of the system prototype is introduced. A process of information retrieval by applying the text mining with Artificial Immune System (AIS) is proposed to discover the possible level of severity for a Short Messaging Service (SMS) spam. This is expected to be a potential tool in retrieving an implicit danger that a spam might impact to the recipients. Furthermore, the development of this tool can be considered as an emergence of another data mining tool that also exceedingly possible to be embedded with another existing tool.**

Keywords*—Danger Theory Variants; Text Spam Messages; Severity Assessment; Text Mining; Information Retrieval; Knowledge Discovery

I. INTRODUCTION

At these days, numerous data are disseminating and spreading globally just within seconds. Without any means of analysis, these data possibly flow aimlessly and useless. Through data analysis, enormous data are processed and meant to be applied in various fields. As these days with technology advancement, spam messages in a form of text, images or even videos has been hassling and successfully tricks so many users. The recorded impact loss has also been significantly unpleasant [1][2][3][4]. Various safeguards have been implemented [5][6][7] to protect assets from any further loss caused by this threat but it seems persistent as an unending issue. With this fact, this paper initiative proposed a tool that might aid in lessening the impact by developing an implicit and trusted decision maker instrument.

Combining a text mining methodology with statistical analysis and inspired by Biological Immune System (BIS), this instrument is measuring the risk concentration for a text message that the potential impact likely to occur. With this information, it is to be expected that users will absolutely ignore any allure offer that would draw them to believe it without noticing it is deceitful messages.

The process of assessing the context of text messages is vital whereby the data must be prepared in a very special way before any methods can be applied. In this research, a classifier that imitates the human body defense or immune system is applied. This theory is well known as Artificial Immune System (AIS) and specifically a theory from Danger Theory and its variants are applied. This assessment task is combining text mining and statistical analysis that produced a predictive result to assist users in reacting against spam message positively. Text mining is a discipline that combines data mining and text analytics to use unstructured or textual data along with structured data for the purposes of exploration, discovery, and predictive modeling or classification [8].

In the previous research, there are many studies found for classifying and differentiate messages between legit or ham and spam [9][10][11], but no publication for measuring the possible harm that this threat could convey, especially with the employment of AIS. With the intention to step ahead, this paper will articulate the design and development of a prototype in conducting and implementing a severity assessment for a text spam message. In addition to that, this paper is a continuity study for [12] and [13] which executed to illustrate its applicability. Its aim is to establish an automated method for the experiment simulated in these papers. The developed prototype will then be further tested for a larger size of the dataset, to verify the results claimed in works [12] and [13] are consistent with the initial findings.

In the direction to have a well arrange for a content presentation, this paper is structured as follows. The main literature for the fundamental part of the study is reviewed in Section II. The variant of Danger Theory that has been applied in this study is clarified with its biological abstraction that theoretically appears to be suitable and fit to overcome this threat. Then, an integration of text mining and risk assessment has been developed as the foundation of this study is elaborated in Section III. The proposed prototype with flow diagram and pseudo-code is described in Section IV. In the last section of this paper, conclusion and potential future work are proposed.

II. DANGER THEORY OF ARTIFICIAL IMMUNE SYSTEMS

Computational intelligence has contributed numerous solutions for various fields. This theory is imitating many natural surrounding artificially that is presumed as an intelligent agent, has significantly proven its ability. One of the famous ideas is AIS that imitated the Body Immune System

This work is fully funded by the Ministry of Higher Education of Malaysia via grant research with code USIM/FRGS/FST/32/50315.

(BIS) against antigens and defense the body from any harm and danger pathogens.

The focal point of this paper is to study the Danger Theory, one of the theories that emerged from AIS. This particular theory is impersonating the behavior of dendritic cells (DCs) that is able to sense and detect malicious substance and stimulates the immune system to react [14]. In 1994, Polly Matzinger then introduced Danger Model [15] that the immune system is more concerned with the damage caused by the malignant substance than cell foreignness. In the following paragraph, 2 variants of Danger Theory applied in this study are explicated and related characteristics are identified in its employment for risk assessment task. These 2 variants are compared theoretically and experimentally in [13], particularly for this spam risk assessment study.

A. Dendritic Cell Algorithm (DCA)

The initial version of DCA has been introduced via Danger Project [16] that applied in detecting intrusion. This algorithm basically is about correlating antigen information and signal processing to assess the condition of the surrounding. DCs are antigen presenting cells (APCs) that play a crucial role in detecting danger. It is unique APCs that have the capability for capturing, processing and presenting antigen to T-cells for further action, either to stimulate or depress the immune systems.

Basically, both algorithms of DCA [17][18] and dDCA [19] are correlating data streams in the forms of antigens and signals. The outcome of this correlation will produce a result of the surrounding either it is malignant or benign. Prior to gain this results, there are 3 types of input signals that released by pathogens that could be captured by immature DCs (imDC); Pathogen Associated Molecular Patterns (PAMPs), danger and safe signals. The imDC processed all the captured signals then migrated to the lymph node and divided into 2 types of conditions, with regards to the detected danger concentration. Semi-mature DCs (smDC) brings the safe signals, while mature DCs (mDC) indicate a dangerous context of an antigen. The transformation of imDC to smDC or mDC is assessed via its anomaly metric, Mature Context Antigen Value (MCAV) [18]. This MCAV is the mean value of context per antigen type, in the form of a numerical vector and its value in between 0 to 1. The closer this values to 1, the greater the probability that the antigen is anomalous. This value also reflects the malicious degree or the concentration of the antigen.

```
input : signals from all categories and antigen
output: antigen plus context values
initialiseDC;
while CSM output signal < migration Threshold do
    get antigen;
    store antigen;
    get signals;
    calculate interim output signals;
    update cumulative output signals;
end
cell location update to lymph node;
if semi-mature output > mature output then
    cell context is assigned as 0 ;
else
    cell context is assigned as 1;
end
kill cell;
replace cell in population;
```

Fig. 1. The DCA algorithm [17]. The applied algorithm for this study is depicted in Section IV.*C.3) DCA algorithm application.*

B. Deterministic Dendritic Cell Algorithm (dDCA)

The fundamental theory of DCA and dDCA has no significant difference, except for its simpler calculation to determine the anomalous level. In dDCA, anomaly metric, Kα is executed and uses the magnitudes of k values. The outcome of dDCA is tagged as anomalous when it is returned as a positive value and tagged as normal when the calculated value is negative [19].

```
input  : Antigen and Signals
output:  Antigen Types and cumulative k values
set number of cells;
initialise DCs();
while data do
    switch input do
        case antigen
            antigenCounter++;
            cell index = antigen counter modulus number of cells ;
            DC of cell index assigned antigen;
            update DC's antigen profile;
        end
        case signals
            calculate csm and k;
            for all DCs do
                DC.lifespan -= csm;
                DC.k += k;
                if DC.lifespan <= 0 then
                    log DC.k, number of antigen and cell iterations ;
                    reset DC();
                end
            end
        end
    end
end
for each antigen Type do
    calculate anomaly metrics;
end
```

Fig. 2. The dDCA algorithm [19]. The applied algorithm for this study is depicted in Section IV.*C.4) dDCA algorithm application*

III. TEXT MINING IN SPAM RISK ASSESSMENT

A. An Integration of Danger Theory and Risk Assessment

The fundamental idea about Danger Theory is how the signal processed in measuring the malicious concentration of an antigen. To apply this concept, an input signal which depicted by weight of tokens are used in signal correlation that eventually will give the malignant level of a spam message.

Conceptually comprehend that in Danger Theory; the malicious content is assessed via MCAV for DCA [17][18] and K_α for dDCA [19] which both theories considering that the closer the measured value to 1, the more malicious it is. Hence, to define input signals that eventually meet the characteristics of output signals (assessed value should be in between 0 to 1 and the closer the output signals to 1, the more malicious it would be) in this immune theory, a reliable term weighting schemes should be considered vigilantly. Selecting good features are crucial activity and require extensive domain knowledge from various aspects. As to create input signals in DCA and dDCA, a term weighting schemes are deployed. Details about this scheme are discussed in Section III.C *Feature Extraction via Statistical Analysis*.

According to National Institute of Standards and Technology or NIST [20], risk or malicious concentration is a measure of the extent to which an entity is threatened by a potential circumstance or event, and is typically a function of:

- adverse impacts that would arise if the circumstance or event occurs; and
- the likelihood or probability of occurrence.

The measurement process for this risk calculation is actually one of the crucial parts of risk analysis. With reference to [21], the spam management should be administered as proposed in risk management. In a developed and established standard of risk management [20][22], this process usually consists of 4 essential phase; risk identification, risk assessment, risk response/treatment and risk monitoring.

In this research, a risk assessment is done with regard to the established concept in common risk management, these include:

- the more frequent a term occur in spam messages, the higher the likelihood of a threat will happen; and
- the calculated weight depicts the level of possible impact implicitly.

The probability level and risk impact are depicted in the following Fig. 3 to which this is practical in assessing and prioritizing risk.

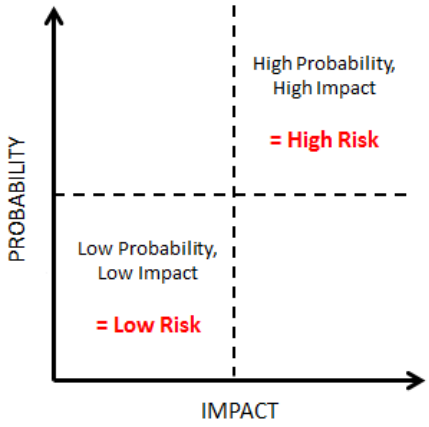

Fig. 3. The Risk Impact / Probability chart

B. The Significance of Text Pre-Processing

There are many research found that applied text mining in SMS spam classification [11][12][13][23][24][25][26][27][28]. However, application of text mining for spam messages is not limited to SMS but also include for email, webpage, and social media platform.

In the process of text mining, pre-processing or also known as the pre-treatment process is one of the important stages. The entire cycle of text categorization that involves all stages includes preparation or collection of data or documents, pre-processing, feature indexing, feature filtering, text classification with algorithm and performance measure. This complete course of action in text categorization has been extensively discussed in [29] and [30]. Messages usually consist of various types of words, which are known as part of speech. These texts may consist of adverbs, articles, conjunctions and many others that possibly not significant for the context assessment. Hence, pre-processing is a process that could distinguish between relevant and irrelevant attributes. An overview of part of speech with examples is tabulated in Table I.

TABLE. I. AN OVERVIEW OF PART OF SPEECH

Part of Speech	Examples
adverbs	quickly, as
articles	a, an, the
conjunctions	and, but, however
interjections	hooray, ouch
prepositions	on, over, beside
pronouns	she, you, us

In sequence process of risk assessment implementation for a text spam message, pre-processing is one of the highly

considered stages in this task. Its main objective is to obtain the key features and to enhance the relevancy between word and document and also between word and category [31]. Many researchers agreed that this particular stage consists of a few more sub-process which includes tokenization, stop word removal, stemming and capitalization. Further elaborations of stated sub-processes are discussed as follows:

- Tokenization - a document is treated as a string, and then partitioned into a list of tokens [32]
- Capitalization/case folding - it is regularly convenient to lower case every character [33]
- Stemming – word stemming refers to converting words to their morphological base forms, for example, both "clicking" and "clicked" are reduced to root word "click" [34]
- Stop word – stop word removal is a procedure to remove words that are found in a list of frequently used words like "and, for, a" [34]. The stop-words are high frequent words that carry no information (i.e. pronouns, prepositions, conjunctions etc.).

Some authors, [35] and [36] also regard pre-processing as the normalization of the noisy text. This process somehow reduces the high dimensionality of the data that commonly turn out to be the main problem in text mining. This issue can be overcome by executing pre-processing that alleviate the data sparseness problem [34]. Pre-processing also allows an efficient data manipulation and representation [32].

This effect of 'noise' is reduced by eliminating any irrelevant word during the stage of pre-processing that is necessary prior to text mining process. Even though many research claimed that this process will increase the accuracy rate [31][35][36], however in spam classification, some authors have contradicted opinion on the impact of pre-processing. Authors Almeida et al. [37][38] argued that the pre-processing has weakened its effect and degrade the classification rate. However, a simulation [28] done in the same field verified that the pre-treatment of a text would amplify the detection rate in distinguishing spam messages. In addition to that, it is also proven that pre-processing has contributed 5% improvement to accuracy value in opinion mining [35]. Besides pre-processing improve in term of classification accuracy, it also identified that would pick up the speed and tendency to reduce overfitting and overhead to computational cost [32].

C. Feature Extraction via Statistical Analysis

Term weighting methods are used to assign appropriate weights to the term. The term in a document vector must be associated with a value called weight, which measures the importance of this term and denotes how much this term contributes to the classification or categorization task [29]. Different terms have a different level of importance in a text; the term weight is associated with every term as an important indicator [31] and this is the key component applied in Danger Theory algorithm. There are a few types of analysis that can be utilized to calculate the weight, which is orthographic, statistical, semantic, syntactic and usage analysis [39].

This research applies the statistical term weighting where term weighting is based on the discriminative supremacy of a term that appears in a document or a group of documents [40]. It is considering as appropriate for this research that it is discriminating documents in between spam and ham. The higher the value the more relevant the term in spam category also indicates highly or frequently repeated in a spam message. The attributes with higher weight are considered that the messages are more relevant to spam category.

The accuracy of the classification or categorization is largely influenced by the collection of messages especially spam where the statistics are derived from. In this research experiment, 3 pre-selected terms weighting has been chosen to be compared empirically with regard to identifying which scheme is the most suitable and adequate for the immune classifier; DCA and dDCA. This statistical inference is attached to the pre-selected term weighting schemes; Term Frequency (TF), Information Gain Ratio (IG Ratio) and Chi Square (CHI^2) and much dependable on how these schemes are working and functioning. Hence, it is crucial to identify a reliable term weighting scheme which is critical for the performance of the classifier [41]. The higher the weight of an attribute (term), the more relevant it is considered as spam. All weights are normalized in a range from 0 to 1, which this is to adequate the characteristic of anomaly metrics in DCA (MCAV calculation) and dDCA (K_α calculation).

IV. *IN SILICO*: PROTOTYPE OF DCA AND DDCA

A. Computational Immune Classifier

Scientific research usually releases a prototype which intended to represent a working system of an idea rather than a theoretical one. This research objective is to produce a working prototype, which is an initial model of a risk assessment product that is designed and developed to test a concept or process in order to act as an assistant in decision making.

As early in 2006, an early version of DCA has been developed as software system to implement the innate immunity [42][43]. This program is known as `libtissue`, is being used by researchers on a project at the University of Nottingham to explore the application of a range of immune-inspired algorithms to problems in intrusion detection.

The proposed prototype for this work also will be developed as software system to apply the algorithm computationally in real world problem, which in this case is a spam problem. The adaptation of biological ideas into the application of real world problem is involved a few phases as developed by Stepney in 2004 and known as Conceptual Framework [44] approach. This methodology employs an iterative approach to the creation and testing of novel immune-inspired algorithms and consists of 4 stages that are identified as:

- observation of the biological experimentation;
- constructed computational models;
- developed, implemented and studied the biological abstraction as algorithms; and

- applied the algorithm to a specific problem, with feedback for refinement.

Besides Conceptual Framework, there is terminology of broad study categories for biological experiments. They are known in Latin as *in vivo*, *in vitro* and *in silico;* and differentiate as the following [45]:

- *In Vivo* (within the living) – refers to examination using a whole, living organism as opposed to a partial or dead organism;
- *In Vitro* (within the glass) – refers to the technique of performing a given procedure in a controlled environment outside of a living organism; and
- *In Silico* (performed on the computer or via computer simulation) – refers to characterize biological experiments carried out entirely in a computer.

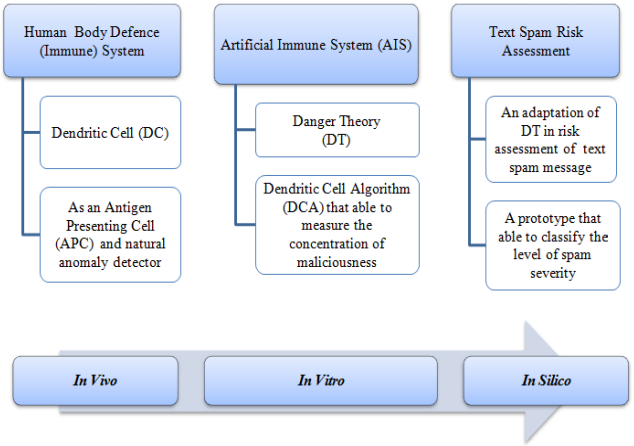

Fig. 4. A concept mapping between the biological perspectives of Human Immune System (*in vivo*); computational theory, Artificial Immune System (AIS); and the real implementation of AIS in computer security field, text spam risk assessment (*in silico*)

B. Risk Scale for The Classifier

DCA use 3 input signal– PAMP, Danger, Safe to produce semi-mature, *O[smDC]* and mature, *O[mDC]* as an output signal in identifying the malicious and normal cell. While dDCA only manipulates minimum 2 signals which are Danger and Safe signals. In this case, the signal value of PAMP is considered as Danger signal for the task of anomalous measurement.

Comparison to each other of semi-mature and mature output signal is used to calculate the risk into 3 distinct levels (mature signals counted as high and medium risk level, semi-mature signal as low risk). A risk scale is used in distinguishing these 3 distinct levels both for input and output signals. This scale also shows the concentration of risk level which the closer the value to 1, the more malicious it is and value that closer to 0 indicate highly safe or normal.

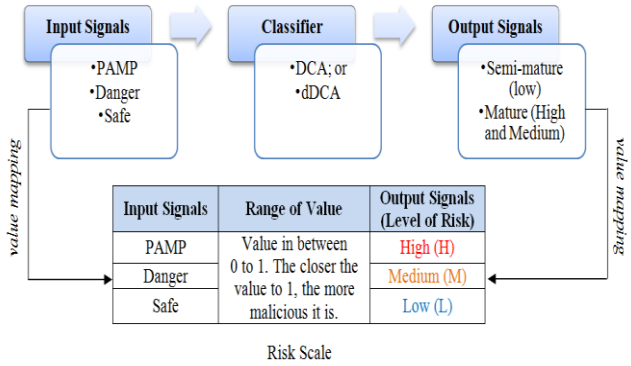

Fig. 5. The mapping of input signals and output signals with the associated risk scale

This risk scale for this experiment is developed based on Likert scale principle with 3 points, high, medium and low level to assess the severity intensity. With regard to verifying the sensitivity of different range for a risk scale, 2 different risk scales have been examined via a simulation which can be found in [12][13]. The same suggestion in executing sensitivity analysis of different weight also has been proposed in [17], to ensure that the empirically derived weights are suitable for the signal processing.

C. Design and Development of A Prototype

Design, develop and model evaluation of a prototype is part of data mining process. As this research is basically about retrieval information of severity intensity from text spam message, a prototype is a must tool to execute this mining task. A summary overview of the flow sequence of the entire process is as follows and the detailed explanation about this process in practical data mining can be found in [46].

An algorithm is normally described in a semiformal notation such as pseudo-code and flowcharts. Flowcharts are used mainly for the high-level description of the algorithms and pseudo-code for describing the details. Pseudo-code is a notation that uses a few simple rules and describes the algorithm that defines a problem solution. It can be used to describe relatively large and complex algorithms. It is relatively easy to convert the pseudo-code description of an algorithm to a computer implementation in a high-level programming language [47].

In this research experiments, all the processes are elaborated via flowchart diagram; the pictorial representation of the whole logic. Then, an advance description is intricate via pseudo-code to illustrate the entire process. Eventually, the whole process will be developed as a set of the prototype, implemented using programming language, which is considered as our future works. The developed prototype of the model then will be embedded in a data mining tool and consequently will be evaluated in terms of its practical functionalities and performance.

The proposed prototype of measuring the intensity or degree of severity for a text spam message consists of 6 phases, as shown below:

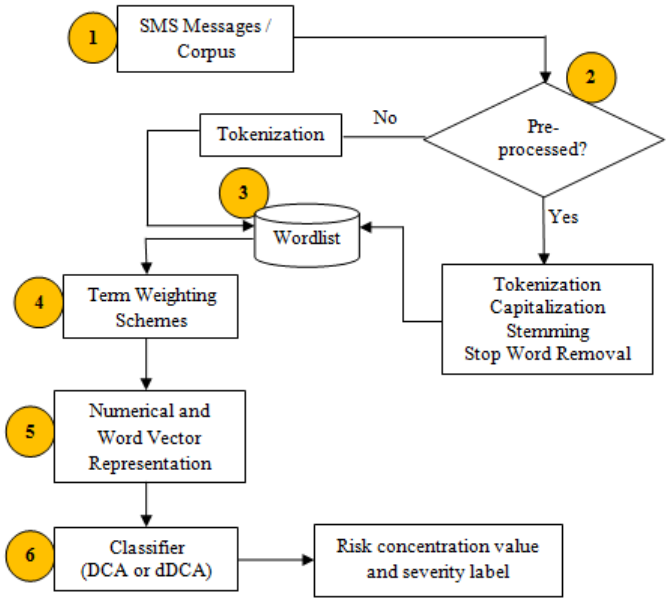

Fig. 6. An overview of the framework for implementation of a risk assessment in text spam message

TABLE. II. RISK ASSESSMENT PHASES IN TEXT SPAM MESSAGES

Phase	Process	Description
1	Preparation	Data, which consists of SMS messages (ham and spam) are collected and prepared for an initial database of the prototype. A new spam message can be processed once this initial database has been developed. Messages can be in a .txt, .doc, .docx, .xls or .xlsx files format.
2	Pre-processing	The corpus (for initial population) or text message have an option either to run through text pre-treatment or not. Tokenization is a must process if decided not to have the pre-processing in place. Otherwise, the full cycle of text pre-processing will be executed, which include tokenization, capitalization, stemming or also known as lemmatization and removal of stop word from the data.
3	Wordlist	This wordlist database contains tokenized words with the information of its total and document occurrences and its frequency in spam and ham document category.
4	Term Weighting Schemes	TF, IG Ratio, and CHI^2 are the available term weighting schemes and act as feature selection methods. These schemes will calculate the importance of every word in the corpus that indicates its relevance to the spam (risk) category. The value derived statistically from this method is further implemented as an input signal in the classifier. This information is stored as an internal database library. All schemes calculate tokenized word as in between 0 to 1 which suggested that the closer the value to 1, the closer it is to malicious level.
5	Numerical and Word Vector Representation	Calculated value for every term in the previous phase then will be mapped to the risk scale. Every derived value will represent the term's severity degree, i.e high, medium, low. The range for risk scale is user-defined.
6	Classification	This phase consists of 3 sub-process which are: Signal processing – identified input signals in a spam message will be correlated and processed depends on chosen classifier i.e DCA or dDCA; Context assessment – the content of the spam message is assessed for its risk concentration level in numerical value; and Risk flagging – classified risk level i.e high, medium, low will be marked to spam message accordingly.

The following paragraphs will elaborate on prototype processes which cover the algorithm design, general process, DCA and dDCA immune classifier. These processes of designing and development of a research prototype would include flow chart and pseudo-code. Later this design will be utilized in developing the prototype using programming language.

1) Algorithm Design For The Prototype

As explained in the previous section, an algorithm is described in a semiformal notation in a form of flow charts and pseudo-code. Prior to having these 2 items established, a step by step process or a brief review is required in order to guide the whole process in design and development of a prototype.

The following is the wide-ranging steps to implement the design and development of the entire system, which also directly related to Fig. 6 and Table II.

- Create initial population

a) collect dataset corpus

b) text pre-processing

c) assign input signals value for antigen (via statistical analysis)

d) store antigen value

- Calculate the severity level

a) receive a new spam message

b) text pre-processing

c) map the identified antigen with the stored input signals value (refer library database)

d) calculate the severity level using classifier (DCA or dDCA)

e) map the calculated output signals with developed risk scale and identify the level of risk

f) mark the spam messages with the risk-level flag accordingly

- Action to response

a) according to identified risk level, response against spam could be deleted, escalate to authority body, re-calculated the risk level (for the case of false positive), or do nothing

2) General Process of the Entire Proposed System

a) Flow Diagram

The entire process for the proposed prototype which has been explained in the previous paragraph is depicted as a flow diagram in Fig. 7.

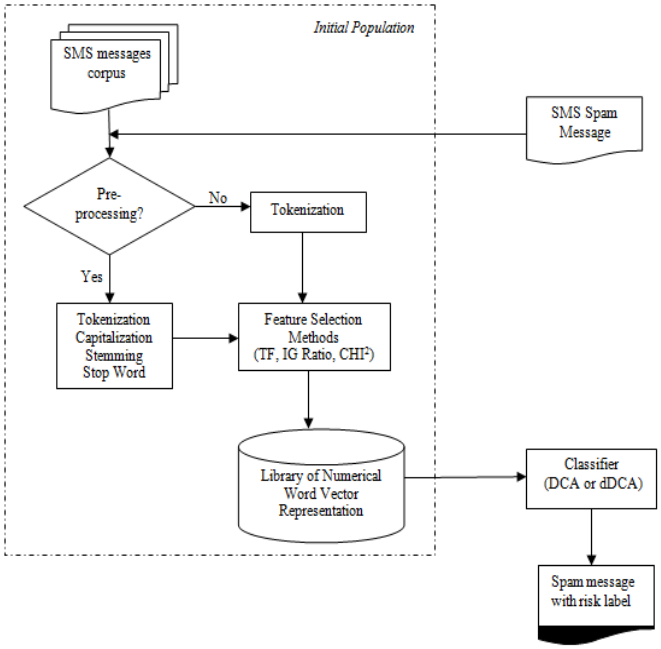

Fig. 7. A flow diagram for general architecture of the entire proposed prototype

b) Pseudo-code

The following is the pseudo-code that describing the flow diagram in Fig. 7 for the whole processes involved in the proposed prototype in general.

Input; M: SMS messages (S: Spam messages; H: Ham messages)
Output; O: Spam messages with marked risk level

1 Begin (while TRUE)
2 Read SMS messages
3 For without pre-processing, execute only tokenization on SMS messages
4 Or else for with pre-processing, execute the full-text pre-treatment (tokenization, capitalization, stemming and remove stop word)
5 Calculate and assign value for tokenized term (provided as input signals in classifier)
6 Determine spam's risk level (High, Medium or Low)
7 Display the flagged spam message with its associated risk label (print the result)
8 End

3) DCA algorithm application

a) Flow Diagram

The processes involved for the classifier DCA is depicted as a flow diagram in Fig. 8.

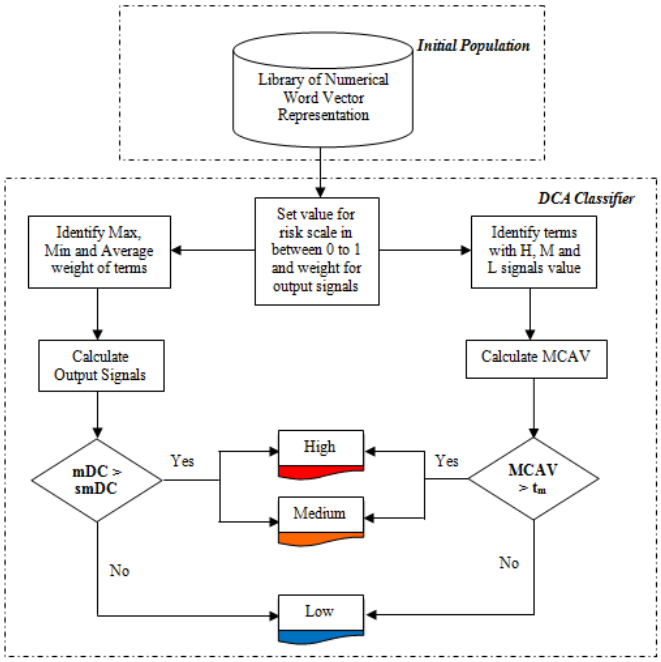

Fig. 8. A flow diagram for DCA classifier process

b) Pseudo-code

The following is the pseudo-code that describing the flow diagram in Figure 8 for the processes involved in DCA classifier.

Input; M: SMS messages (S: Spam messages; H: Ham messages)
Output; O: Spam messages with marked risk level

1 Begin (while TRUE)
2 Read SMS messages
3 For without pre-processing, execute only tokenization on SMS messages
4 Or else for with pre-processing, execute the full-text pre-treatment (tokenization, capitalization, stemming and remove stop word)
5 Calculate and assign a value for tokenized term via term weighting schemes method (provided as input signals in classifier). Store all derived antigen and signal values in library
6 Set value for risk scale with 3 different levels in between 0 to 1 (as in Figure 5) and weight to transform input signal to output signal
7 Identify maximum, minimum and average value for input signal and calculate output signals, *O[mDC]* and *O[smDC]*
8 Identify words with high, medium and low signals value and calculate MCAV
9 Identify anomaly threshold, t_m and compare with MCAV
10 Determine spam's risk level (High, Medium or Low) whenever these conditions are both fulfilled.
- *O[mDC]* > *O[smDC]* and MCAV > t_m . this is to indicate that spam message is malicious and MCAV result should return a value as High or Medium
- *O[mDC]* < *O[smDC]* and MCAV < t_m - this is to indicate that spam message is benign and MCAV result should return a value as Low

Refer Fig. 5 for mapping the MCAV value with the level of risk.
11 Display the flagged spam message with its associated risk label (print the result)
12 End

4) dDCA algorithm application

a) Flow Diagram

The processes involved for the classifier dDCA is depicted as a flow diagram in Fig. 9.

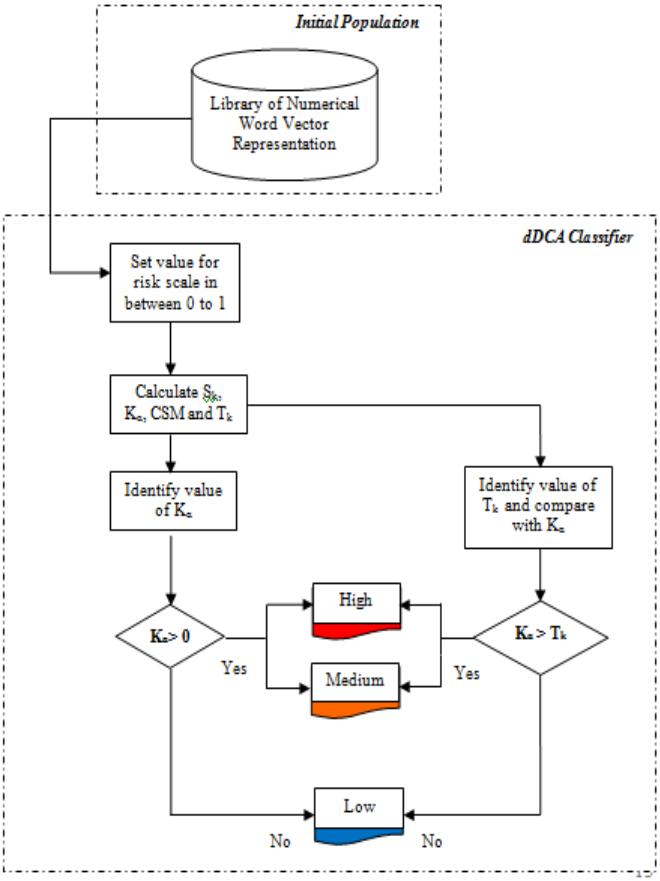

Fig. 9. A flow diagram for dDCA classifier process

b) Pseudo-code

The following is the pseudo-code that describing the flow diagram in Fig. 9 for the processes involved in dDCA classifier.

Input; M: SMS messages (S: Spam messages; H: Ham messages)
Output; O: Spam messages with marked risk level

1 Begin (while TRUE)
2 Read SMS messages
3 For without pre-processing, execute only tokenization on SMS messages
4 Or else for with pre-processing, execute the full-text pre-treatment (tokenization, capitalization, stemming and remove stop word)
5 Calculate and assign a value for tokenized term via term weighting schemes method (provided as input signals in classifier). Store all derived antigen and signal values in library
6 Set value for risk scale in between 0 to 1 (Fig. 5)
7 Calculate:
- the sum of all input signal, S_k;
- K_α, magnitudes of k value;
- Costimulation output signal, CSM; and
- Threshold, T_k.

8 Determine spam's risk level (High, Medium or Low) whenever these conditions are both fulfilled.
- $K_\alpha > 0$ and $K_\alpha > T_k$ - spam message is tagged as the malicious message; or
- $K_\alpha < 0$ and $K_\alpha < T_k$ - spam message is tagged as the benign message.

9 Display the flagged spam message with its associated risk label (print the result)
10 End

D. Performance Measurement

All further experiments are measured using True Positive (TP) value as the measurement metric for accuracy classification rate. TP value reflected the correct classification of spam messages according to its risk level and context of the spam message. TP calculated as a ratio in percentage (%), whereby its total number of correctly risk-classified messages is proportional to the total number of all messages that has been assessed. All messages are referring to the total number of truly-classified and falsely classified spam messages.

$$TP = \frac{\sum correctly\ risk{-}classified\ messages}{\sum all\ messages} \quad (1)$$

V. Conclusion and Future Works

This developed prototype of a data mining tool can be considered as unsupervised as it is designed and developed inspired by Danger Theory idea that does not require training or learning process. Text data is generally viewed as categorical data and its weight derived from statistical analysis is the signal value in numerical value. In this task, text data is the antigen and the weight is signal value correlated to assess and predict the maliciousness of a message. Since the manual simulation as clarified in [12] and [13] has been executed using RapidMiner for its weight value of terms (input signals) and text pre-processing, this proposed prototype will be developed in a high-level programming language and further embed in RapidMiner. Subsequently, the developed prototype will be tested with a larger dataset of SMS and deploy with another form of text messages such as Twitter, to verify its consistent and reliable results as discovered in [12] and [13]. Development of this prototype in the mobile platform is also encouraged to demonstrate the performance and functionalities.

This potentially discovered knowledge through this mechanism is valuable to be applied in various fields. The remarkable findings are potential to be enhanced in another field of data mining study, such as computational linguistics.

References

[1] A. Ivanov, "Damage caused by spam," Securelist – Information about Viruses, Hackers and Spam, 2016. [Online]. Available: https://securelist.com/threats/damagecausedbyspam/.

[2] J. M. Rao and D. H. Reiley, "The Economics of Spam," J. Econ. Perspect., vol. 26, no. 3, pp. 87–110, 2012.

[3] T. Takemura and H. Ebara, "Economic Loss Caused by Spam Mail in Japanese Industries," 2008.

[4] E. O. Yeboah-Boateng and P. M. Amanor, "Phishing, SMiShing & Vishing: An Assessment of Threats Against Mobile Devices," J. Emerg. Trends Comput. Inf. Sci., vol. 5, no. 4, pp. 297–307, 2014.

[5] "Combating Spam: Policy, Technical and Industry Approaches," 2012.

[6] "Anti-Spam Technical Alliance Technology and Policy Proposal," 2004.

[7] Government of Canada, "5 Things to Look for - Worried It's Spam?" 2012.

[8] J. Dean, Big Data, Data Mining, and Machine Learning. John Wiley & Sons, 2014.

[9] M. A. Balubaid, U. Manzoor, B. Zafar, A. Qureshi, and N. Ghani, "Ontology Based SMS Controller for Smart Phones," Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 1, pp. 133–139, 2015.

[10] G. Sethi and V. Bhootna, "SMS Spam Filtering Application Using Android," Int. J. Comput. Sci. Inf. Technol., vol. 5, no. 3, pp. 4624–4626, 2014.

[11] G. Song, Y. Ye, X. Du, X. Huang, and S. Bie, "Short Text Classification: A Survey," J. Multimed., vol. 9, no. 5, pp. 635–643, 2014.

[12] K. Zainal and M. Z. Jali, "The Significant Effect of Feature Selection Methods in Spam Risk Assessment using Dendritic Cell Algorithm," 2017.

[13] K. Zainal and M. Z. Jali, "Comparative Analysis of Danger Theory Variants in Rating Risk Concentration via Context Assessment of Text Spam Messages," 2017.

[14] E. Wieder, "Dendritic Cells: A Basic Review," Int. Soc. Cell. Ther., no. May, 2003.

[15] P. Matzinger, "Tolerance, Danger and The Extended Family," Annu. Rev. Immunol., vol. 12, pp. 991–1045, 1994.

[16] U. Aickelin, P. Bentley, S. Cayzer, K. Jungwon, and J. McLeod, "Danger Theory: The Link Between AIS and IDS?," Int. Conf. Artif. Immune Syst., pp. 147–155, 2003.

[17] J. Greensmith, "The Dendritic Cell Algorithm," University of Nottingham, 2007.

[18] J. Greensmith, U. Aickelin, and J. Twycross, "Articulation and Clarification of the Dendritic Cell Algorithm Dendritic Cells," 2009.

[19] J. Greensmith and U. Aickelin, "The Deterministic Dendritic Cell Algorithm," 2008.

[20] R. M. Blank and P. D. Gallagher, "Guide for Conducting Risk Assessments," no. September. pp. 1–39, 2012.

[21] K. Zainal and M. Zalisham, "A Perception Model of Spam Risk Assessment Inspired by Danger Theory of Artificial Immune Systems," Int. Conf. Comput. Sci. Comput. Intell., vol. 59, pp. 152–161, 2015.

[22] G. Locke and P. D. Gallagher, "NIST- Guide for Applying the Risk Management Framework to Federal Information Systems," vol. 1. p. 93, 2010.

[23] J. M. G. Hidalgo, G. C. Bringas, and E. P. Sánz, "Content Based SMS Spam Filtering," 2003.

[24] A. Mosquera, L. Aouad, S. Grzonkowski, and D. Morss, "On Detecting Messaging Abuse in Short Text Messages using Linguistic and Behavioral patterns," Arxiv - Soc. Media Intell., 2014.

[25] H. Xia, Y. Fu, and J. Zhou, "Intelligent spam filtering for massive short message stream," Int. J. Comput. Math. Electr. Electron. Eng., vol. 32, no. 2, pp. 586–596, 2013.

[26] D. Belém and F. Duarte-Figuriredo, "Content Filtering for SMS Systems Based on Bayesian Classifier and Word Grouping," IEEE, 2011.

[27] K. Zainal, N. F. Sulaiman, and M. Z. Jali, "An Analysis of Various Algorithms For Text Spam Classification and Clustering Using RapidMiner and Weka," Int. J. Comput. Sci. Inf. Secur., vol. 13, no. 3, pp. 66–74, 2015.

[28] K. Zainal and M. Z. Jali, "A Review of Feature Extraction Optimization in SMS Spam Messages Classification," Int. Conf. Soft Comput. Data Sci., vol. 545, pp. 158–170, 2016.

[29] A. Patra and D. Singh, "A Survey Report on Text Classification with Different Term Weighting Methods and Comparison between Classification Algorithms," Int. J. Comput. Appl., vol. 75, no. 7, pp. 14–18, 2013.

[30] O. Ardhapure, G. Patil, D. Udani, and K. Jetha, "Comparative Study of Classification Algorithm for Text Based Categorization," Int. J. Res. Eng. Technol., pp. 217–220, 2016.

[31] V. Srividhya and R. Anitha, "Evaluating Preprocessing Techniques in Text Categorization," Int. J. Comput. Sci. Appl., pp. 49–51, 2010.

[32] A. Khan, B. Baharudin, L. H. Lee, and K. Khan, "A Review of Machine Learning Algorithms for Text-Documents Classification," J. Adv. Inf. Technol., vol. 1, no. 1, pp. 4–20, 2010.

[33] X. Zhu, "Basic Text Process - Advanced Natural Languange Processing." pp. 1–3, 2010.

[34] L. Zhang, J. Zhu, and T. Yao, "An Evaluation of Statistical Spam Filtering Techniques," ACM Trans. Asian Lang. Inf. Process., vol. 3, no. 4, pp. 243–269, 2004.

[35] N. Samsudin, M. Puteh, A. R. Hamdan, and M. Z. A. Nazri, "Normalization of Common NoisyTerms in Malaysian Online Media," Proc. Knowl. Manag. Int. Conf., no. July, pp. 515–520, 2012.

[36] N. Samsudin, A. R. Hamda, M. Puteh, and M. Z. A. Nazri, "Mining Opinion in Online Messages," Int. J. Adv. Comput. Sci. Appl., vol. 4, no. 8, pp. 19–24, 2013.

[37] T. A. Almeida, J. M. G. Hidalgo, and A. Yamakami, "Contributions to the study of SMS spam filtering," Proc. 11th ACM Symp. Doc. Eng. - DocEng '11, p. 259, 2011.

[38] T. A. Almeida, J. María, G. Hidalgo, and T. P. Silva, "Towards SMS Spam Filtering : Results under a New Dataset," Int. J. Inf. Secur. Sceince, vol. 2, no. 1, pp. 1–18, 2012.

[39] P. Losiewicz, D. W. Oard, and R. N. Kostoff, "Textual Data Mining to Support Science and Technology Management," J. Intell. Inf. Syst., vol. 15, no. 2, pp. 99–119, 2000.

[40] M. Abdel Fattah, "New term weighting schemes with combination of multiple classifiers for sentiment analysis," Neurocomputing, vol. 167, pp. 434–442, 2015.

[41] V. Pekar, M. Krkoska, and S. Staab, "Feature Weighting for Co-occurrence-based Classification of Words," 20th Int. Conf. Comput. Linguist., 2004.

[42] J. Twycross and U. Aickelin, "Experimenting with Innate Immunity," 2006.

[43] J. Twycross and U. Aickelin, "Libtissue - Implementing Innate Immunity," 2010.

[44] J. Greensmith, A. Whitbrook, and U. Aickelin, "Artificial Immune Systems," pp. 1–29, 2010.

[45] "Differences between in vitro, in vivo, and in silico studies," The Marshall Protocol Knowledge Base-Autoimmunity Research Foundation, 2012.

[46] J. Monte F. Hancock, Practical Data Mining, 1st ed. Taylor & Francis Group, 2012.

[47] J. M. Garrido, Introduction to Elementary Computational Modeling-Essential Concepts, Principles and Problem Solving. 2012.