An Immuno-inspired Approach Towards Sentence Generation

Samir Kr Borgohain
Department of Computer Science &Engineering
National Institute of Technology,
Silchar (India)
+9134842242025
samirborg@gmail.com

Shivashankar B. Nair
Department of Computer Science &Engineering
Indian Institute of Technology Guwahati,
Guwahati (India)
+913612582356
sbnair@iitg.ernet.in

ABSTRACT
Though human beings comprehend, imbibe and subsequently generate syntactically and semantically correct languages, the manner in which they do so has hardly been understood or unearthed. Most of the current work to achieve the same is heavily dependent on statistical and probabilistic data retrieved from a large corpus coupled with a formal grammar catering to the concerned natural language. This paper attempts to portray a technique based on an analogy described by Jerne on how his theory of the Idiotypic Network could possibly explain the human language generation capability. Starting with a repertoire of unigrams (antibodies) weaned from a corpus available *a priori*, we show how these can be sequenced to generate higher order *n-grams* that depict full or portions of correct sentences in that language. These sentences or their correct portions form a network similar to the Idiotypic Network that in turn aid in the generation of sentences or portions thereof which are new to the corpus signifying the learning of new and correct sequences. The network is built based on a modified version of the dynamics suggested by Farmer et. al. The paper describes the related dynamics of the network along with the results obtained from a corpus.

Categories and Subject Descriptors
I. 2.7 [**Natural Language Processing**]: Language generation and Language models.

General Terms
Algorithms, Experimentation, Languages

Keywords
Sentence generation; Idiotypic network; Immune; Language processing

1. INTRODUCTION
The problem of processing natural language has been attacked from several frontiers. The earlier techniques used aformal grammar [1, 2, 3, 4] to start with and analysed whether the incoming sentences conformed to the rules of the grammar embedded *a priori*. A Context Free Grammar for instance, is defined by a vocabulary containing a set of non-terminals and a set of productions or transition rules. First an initial non-terminal is considered and is replaced by a set of productions. The productions are applied repeatedly until a terminal sequence of words is generated.

However, natural language with its rich set of virtually countless grammar rules cannot be accommodated by such methods. This forced researchers to try out corpus based statistical methods [2]. Statistical information derived from a large corpus of the language was used to find measures based on probability of the next word in a sequence. Most of these techniques use the term *n-grams* [1,2] where *n* is a non-zero positive integer and the *gram* represents a token or a word in the corpus. The *n-gram* model concentrates on local dependencies and also encodes linguistic information [1] like syntax, semantics, etc.The *n-gram* model is locally dependent because it takes the window size of *n* and predicts only by taking the context of *(n-1)* words. Further the model is trained using a corpus and because a corpus is always finite, some acceptable *n-grams* are bound to be omitted. Addition of smoothing techniques [5] to add non-zero probabilities to zero probability bigrams eases the issue to some extent.Although the empirical based language models [6] are capable of generating sentences, the complexity of a natural language makes it infeasible to generate all the rules within.

In this paper, we propose an immuno-inspired language processing model, a paradigm shift from the conventional techniques of language processing that, when given a set of correct sentences generates a network which in turn can emit new and correct ones. This technique does not suffer from data sparseness and there is no requirement of a grammar base or a large annotated or tagged corpus. The model opens up the possibility of understanding the working of the biological generative grammar [7].

2. LANGUAGE GENERATION IN HUMAN BEINGS
The manner of the origin and evolution of language and how human beings have over several thousands of years seamlessly conveyed their thought processes through language still remains unraveled. The works carried out by Knight, Studdert-Kennedy and Hurford [8], Hurford, Studdert-Kennedy and Knight [9], Harnard andSteklis[10] are notable. According to them, language has evolved in human beings due to their genetic traits. Two distinct schools of thought – one biological and the other based on cultural evolution – exist, which seek the answer to the origin of a language. The first school of thought proposes that human language originated due to biological evolution. Numerous experiments have been carried out by researchers on apes and

GECCO '15, July 11–15, 2015, Madrid, Spain

DOI: http://dx.doi.org/10.1145/2739480.2754805

monkeys to make them learn a language but with hardly any success. On the contrary a human child of around five years of age or even less is capable of acquiring a language from its environment [11] with great ease. This observation led Chomsky [12] to argue that human beings are possibly innately equipped with a set of language rules that aid in the production of sentences.He defines a Universal grammar (UG) which is a formal system within the being and is capable of aligning words (or morphemes) into a pattern which could be grammatically correct or at least acceptable. According to Chomsky these grammatical rules are sort of hard-wired in the human brain. This theory of the existence of a UG is widely accepted by the research community [13,14]. Bickerton [15] describes of the possible existence of a *protolanguage* that exhibits a limited grammar without many of the features of modern day human languages. According to him, the emergence of a fully developed language is due to a sudden and discontinuous 'macro-mutation' of this *protolanguage*. Pinker and Bloom [16] reason that language evolution was a gradual process which possibly took a longperiod of time. Elman [17] questions the innateness of language and concludes that although language is innate the grammar is not encoded in the genomes but has resulted due to many interactions between the genes. Kirby [18] argues that cultural processes may have played a more significant role in the evolution of languages than as the result of evolution due to genes. According to him, languages are passed from one generation to the next resulting in the development of innately possible grammars. Cavalli-Sforza and Feldman [19] have also attempted to develop a general theory for the same, based on cultural evolution.

N. Jerne [7] advocated Chomsky's innateness of generative grammar and drew parallels between language and immunology. Jerne's theory states that the immune system is governed by a set of grammatical rules that allow the formation of new sentences (or antibodies). These antibodies form an interconnected lattice-work. Based on such a network, Jerne draws an analogy with Chomsky's theory of generative grammar that is innate and equally capable of generating new sentences that are recognized by a native speaker. Chomsky's generative grammar results in an "*open-ended*" number of sentences which Jerne correlates to the "*completeness*" of the antibody repertoire. Chomsky and Jerne, both believed [7, 12] that grammatical rules of a human language are carried forward by genomes from one generation to the next.

It is well known that a person fluent in a certain language can wax eloquently in that language. This proficiency makes it extremely difficult for the person to generate ill-formed sentences in the same language, spontaneously. One may thus infer that the person possibly generates a language-based Idiotypic network that impedes the generation of any incorrect language pathogens. With prolonged exposure to a language, a person seems to prototype an in-built immune network capable of distinguishing well-formed sentences and preventing or suppressing the generation of ill-formed ones. It is also interesting to note that grammar by itself is never taught in the initial phases of language learning by a human being. In many a case a person may speak a language fluently without having an iota of knowledge about its grammar. A child, for instance picks up a language very well, oblivious of the associated grammar. It is thus obvious that a collection of sentences, that constitute a corpus, is formed initially within the child. Subsequent sentence generation largely depends on the combinations of the words within this corpus and the inherent grammatical structures contained therein.

3. IMMUNO-INSPIRED LANGUAGE PROCESSING

The existing language models are based on the framework of linguistic rules and statistical methods. While the former rule-based language processing [21] relies heavily on grammatical rules the latter suffers from data sparseness [20]. A sizeable corpus is always necessary while applying the above two techniques of language processing.The immune inspired algorithms have found their use in diverse areas, a few of the notable areas being intrusion detection, pattern recognition and optimization [22]. Immune algorithms have been rarely used in language processing. Kumar and Nair [23] have used the negative selection algorithm [24] to check grammatical errors in English sentences. They treat grammatical errors as pathogens and try to build antibodies that can detect such errors. One of the drawbacks in this approach is the requirement of a large *annotated* corpus. Others have skirted this area by formulating spam filters by combining language with immune algorithm based classifiers [25].

This paper proposes a paradigm shift in the manner of language processing by working out an *Idiotypic Language Network*that, when provided with *n-grams* can eventually emit new and correct sentences.

4. OVERVIEW OF IDIOTYPIC NETWORK

In an immune system the primary role of antibodies is to eliminate the antigens, which constitute foreign substances like the virus, bacteria, etc., from the body. An antibody contains a *paratope* which allows itself to attach onto the *epitope* of an antigen.

According to the *Idiotypic network theory* postulated by Jerne [7], antibodies communicate with different antibodies to form a large scale network even in the absence of antigens [26]. An antibody has its specific antigenic determinant called an *idiotope* through which it binds to the *paratope* of another antibody.The schematic diagram representing Jerne's idiotypic network [7] is shown below as Figure 1.

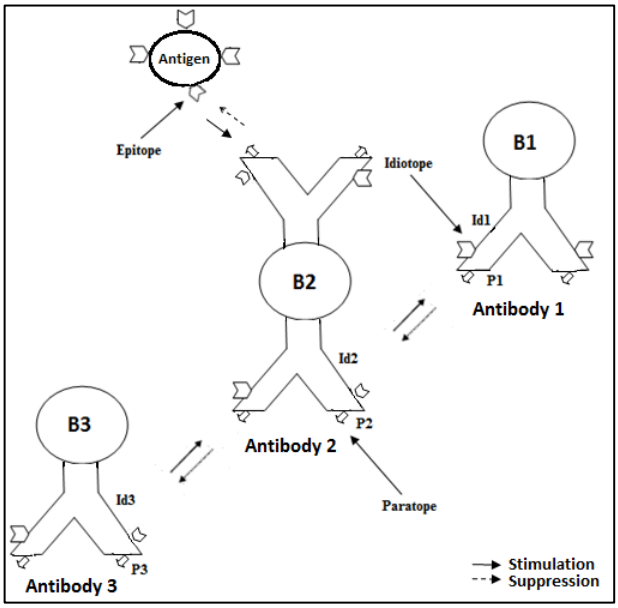

Figure 1. Jerne's Idiotypic Network [7]

The *idiotopeId1* of antibody *B1* binds to the *paratopeP2* of antibody *B2* and stimulates the latter. On the other hand, antibody *B2* recognizes *Id1* of antibody *B1* as an antigen and therefore suppresses antibody *B1*. Stimulations and suppressions between

the antibodies, constitute the formation of an *idiotypic* network even in the absence of an antigen.

4.1 Modeling the idiotypic network

Farmer et. al[27] provide a mathematical model for the Idiotypic network which closely approximates its biological counterpart. The dynamics of the model consists of a set of differential equations that changes the concentration of the antibodies w.r.t stimulations and suppressions. The natural death of antibodies is also taken into account. The mathematical equation to calculate the concentration of i^{th} antibody is given by

$$\frac{dx_i}{dt} = \sum_{j=1}^{N} M_{ji}x_j - \alpha_1 \sum_{j=1}^{N} M_{ij}x_j + \sum_{j=1}^{n} M_i y_j - \alpha_2 \, .x_i \quad (1)$$

where N is the number of antibodies and x_i (or x_j) is the concentration of antibody i (or j). The affinity between antibody j and antibody i is denoted by M_{ji} and the affinitybetween antibody i and the detected antigen is represented by M_i. The first and second terms on the R.H.S of the equation (1) denote the stimulations and suppressions from other antibodies while the third term represents the stimulation from the antigen. The fourth specifies the natural death of the antibody.

Jos´e Pacheco et. al [28] have proposed an Abstract Immune Systemalgorithm inspired by the Jerne's idiotypic network model and clonal selection wherein he ignores the antigens which results in a modified version of Farmer's equation given below –

$$\frac{dx_i}{dt} = \sum_{j=1}^{N} M_{ji}x_j - \alpha_1 \sum_{j=1}^{N} M_{ij}x_j - \alpha_2 \, .x_i \quad (2)$$

Researchers have employed various techniques to calculate the affinity values M_{ji} and M_i, depending on the problem at hand and the representation of entities viz. antibodies and antigens in immune algorithms. They have represented entities within, using binary strings, strings over finite set of alphabet (other than binary), real-valued vectors and even hybrid representations which contain continuous and categorical data [29].The concentrations of antibodies have been calculated using a non-linear differential equation [30]. Affinity which is the distance between two antibodies has been calculated using different algorithms such as Weighted kappa [31] and Kendall's Tau [32, 33]. Ishiguro et al. [34] have proposed a decentralized consensus-making system using Farmer's equation to calculate the concentration of affinities. The affinity values or degree of disallowance are calculated by using the following two equations.

$$M_{ji} = \frac{S_p^{Abj} + S_r^{Abi}}{S_{Abi}^{Abj}} \quad (3)$$

$$M_{ij} = \frac{S_p^{Abi} + S_r^{Abj}}{S_{Abi}^{Abj}} \quad (4)$$

where S_p^{Abj} and S_r^{Abj} denote the number of times penalty and reward signals were received when antibody Ab_j was selected. The denominator denotes the number of times when both Ab_j and Ab_i react with their specific antigens.

5. IDIOTYPIC LANGUAGE NETWORK (ILN)

5.1 A Typical ILN

In order to make the explanation simpler and lucid, we illustrate an ILN using an example. To start with, assume that a small child has heard and gathered a lexicon containing the following words as shown in Table 1. Using these words as a base, the child generates combinations that are subsequently uttered during a phase termed the training phase. The different combinations of words generated from these words could be looked upon as candidate antibodies (from the immunological perspective) and have been depicted in Figure 2.

Table 1. A small lexicon of words generated by the child

Sl. No.	Words learnt by the child
1	*went*
2	*I*
3	*home*
4	*all*
5	*alone*

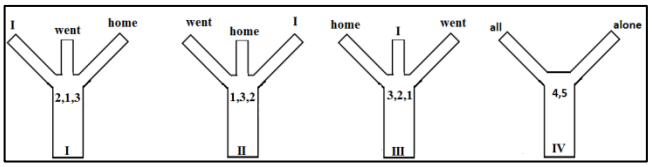

Figure 2. Randomly generated antibodies

During the training phase, the child picks antibodies randomly and utters them in some sequence. A human listener, assumed to be the tutor, verifies it and provides a reward or a penalty based on whether the sequence is correct or incorrect.

Let us now consider that the child uttered the following sequence of antibodies in the order - antibody#II followed by antibody# IV. The sequence that is generated is –

"*went home Iall alone*".

The tutor in turn informs that antibody#II is incorrect but antibody#IV is correct. The peers of antibody#II are I and III since they all contain the same words but in different orders. Since antibody#II is incorrect, it will be suppressed by I and III. This will ensure that amongst antibodies I, II and III, antibody#II will be selected more infrequently in subsequent times. Figure 3 represents the stimulations and suppressions that occur between the set of antibodies.

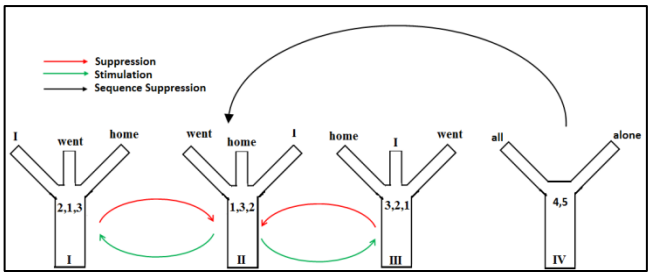

Figure 3. The stimulations and suppressions of antibodies

In Figure 3, the green coloured arrows represent the stimulations while the red ones indicate suppressions. The antibody#II is suppressed by both its peer antibodies numbered I and III as antibody#IIwas found to be incorrect. On the other hand, antibody#II stimulates both the antibodies numbered I and III. As a result, antibody#II will not be selected in the subsequent iteration as it is being suppressed and thus has effectively a lower concentration. The thick black arrow between antibody#IV and antibody#II with the arrow pointing to the latter indicates the suppression of the sequence viz. –<"*went home I*"> followed by <"*all alone*">.

Since a part of the sentence is wrong, the child once again attempts to correct the incorrect portion. If it chooses antibody# I

to generate the following sequence: The sequence would then be expressed as -

I, IV →*I went homeall alone*

The tutor verifies and rewards the correct antibodies. This new information causes a change in the above network (Figure 3) of antibodies which is reflected in Figure 4. Here a new set of stimulations and suppressions are generated between antibodies I and III. Also since antibodies I and IV are correct, they stimulate one another which are shown by bidirectional arrows. This however means that "*I went home all alone*" and "*allalone I went home*" are correct at the moment. However, the latter will be rejected due to suppressions that will appear at a later stage when this incorrect sentence is generated.

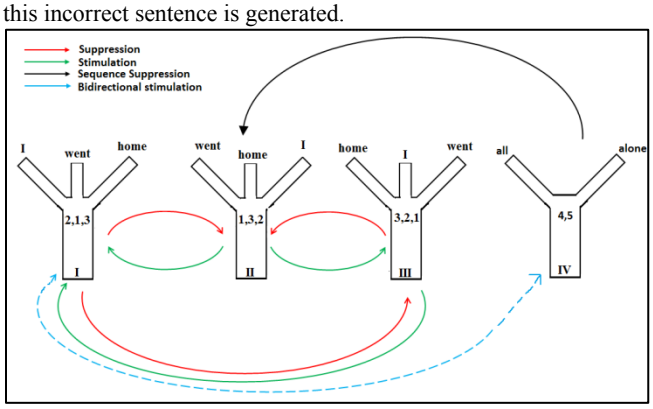

Figure 4.Addition of more stimulation and suppression links between the antibodies

At the end of the training phase, we find that an antibody network is established as a result of interactions between the antibodies. With the arrival of new antibodies which are randomly selected and created from the given lexicon, more such suppressions and stimulations will appear. It may be noted that over a period of time antibodies II, III and V will disappear since they will either be suppressed greatly (reduced concentration) or possibly because they will never be used (activated). Thus only antibodies I and IV will survive to generate two sentences viz. in the present stage of learning:

I went home.

I went home all alone.

This has been illustrated graphically in Figure 5 as an *idiotope-paratope* linkage between two antibodies.

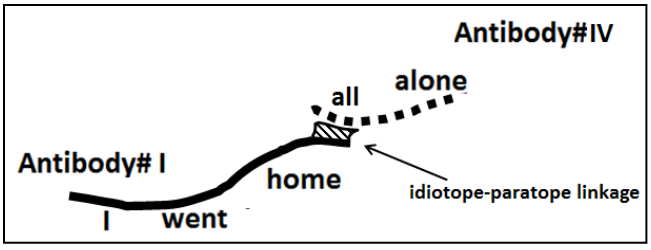

Figure 5. Two segments (antibodies)forming an *idiotope-paratope* linkage

With more words in the lexicon, the network would cater to the generation of more combinations and eventually retain only the correct ones. If we imagine the network in this manner where an antibody forms part of a sentence (a correct sequence of words) then antibodies could concatenate or adhere to parts of another antibody (*idiotope-paratope* like linkage) to make a complete and correct sentence. Thus a more complex set of sentences could be viewed as shown in Figure 6.

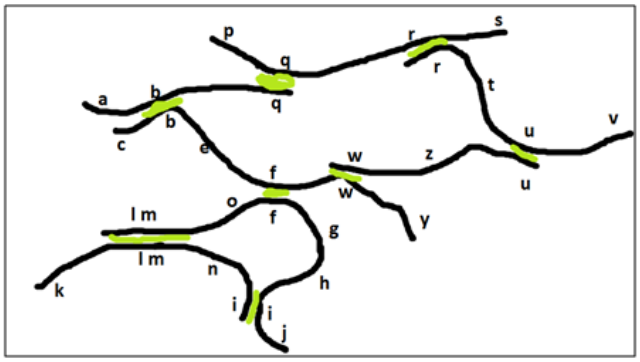

Figure 6. A complex ILN (Each alphabet designates a unique antibody viz. a word or a sequence of words)

In this figure, the green portions act as *joiners*or *bonds* or *idiotope-paratopelinkages*that fuse together two or more antibodies. Assuming the lowercase alphabet are actually valid words in the lexicon (or a sequence of words or n-grams), there could be many valid sentences that could now be generated from such a network; for e.g. *abq, abefghij, klmnij, klmofghij*and so on. The subsequent section deals with the dynamics that drive the formation and evolution of the ILN.

5.2 ILN Generation and Dynamics

The basic flow in the generation of an ILN is shown in Figure 7. Each of the components within has been described separately inthe following sections.

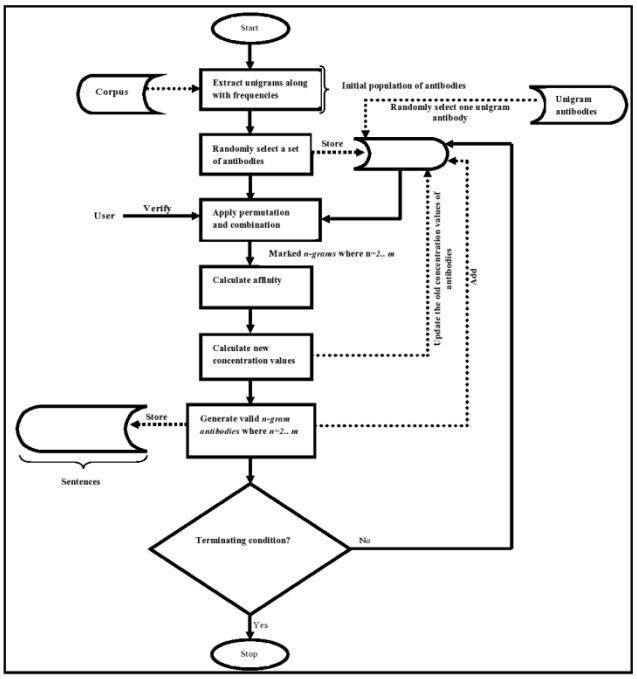

Figure 7. Architecture of the sentence generation model

a) Pre-processing: From a given corpus, unigrams are initially extracted along with their frequencies. The collection of all unigrams along with their frequencies forms the initial population of candidate antibodies and their respective concentrations.

b) Random selection of unigram antibodies: A set of random unigram antibodies are selected from the initial population. This random selection was done only to ensure that the processing time is not too high. It may be noted however that the entire population could be taken in. We thus start off by building a small ILN with this initial sub-population and eventually add more candidate antibodies to scale the derived ILN.

c) Permutation and Combination: The candidate antibodies stored in the selected population are permuted and combined in all possible ways to generate all probable *n-gram* antibodies where n>1. Some of the probable *n-gram* antibodies may not be correct. The user verifies the same and is appropriately marked at this stage of processing. The marked *n-gram* antibodies (both correct as well as incorrect ones) are preserved. The incorrect ones are preserved along with the correct ones because the correct *n-grams* fetch rewards whereas incorrect *n-grams* fetch penalties which are required for calculating the affinity values.

d) Affinity: The binding of an antibody to an antigen (or antibody) is known as affinity. The mutual affinity values of every marked bigram antibody with another in the same set are found. As a candidate bigram antibody is composed of two individual units i.e. unigram antibodies, each of the unigram antibodies are either rewarded or penalizedbased on their correctness. By using the reward and penalty parameters, the affinity between the antibodies is calculated. For our convenience, we have assumed that reward and penalty parameters will be represented using positive numerical values. Initially, both the reward and penalty parameters are initialized to 0. If an antibody receives a reward, the reward parameter is set to 2 otherwise the penalty parameter is set to 1. The same method is applied to calculate the affinity values for higher-order *n-gram* antibodies in the subsequent steps of the iteration. The equations (3) and (4) mentioned in sub-section 4.1 were tailored to calculate the mutual affinity values. The denominators in R.H.S. of the equations (3) and (4) term in the present context denote the degree of the strength of juxtaposition. If either antibody Ab_ifollowed by antibody Ab_jor antibody Ab_jfollowed by antibody Ab_iyield a correct bigram antibody, then theS_{Abi}^{Abj} =1. If these pairs of antibodies (Ab_i,Ab_j)followed by (Ab_j, Ab_i)generates a correct tetragram then the value of the denominator is taken to be less than unity in order to increase their mutual stimulations between the antibodies. Let us consider two randomly selected candidateantibodies viz. Ab_1=*a:12* and Ab_2=*lion:7* where Ab_1 and Ab_2 are the symbolic representations of the antibodies. The R.H.S terms are the actual unigram values along with their respective frequencies (concentrations). After permutation and combination, we get four bigram pairs namely (Ab_1, Ab_1), (Ab_1, Ab_2), (Ab_2, Ab_1) and (Ab_2, Ab_2). The pair (Ab_1, Ab_2) i.e. *a lion* is the only pair that is correct while the remaining three pairs are incorrect. Thus, for the pair (Ab_1, Ab_2), we have S_p^{Ab1}=0, S_r^{Ab1}=1, S_p^{Ab2}=0 and S_p^{Ab2}=1.By substituting the values in equations(3) and (4), we get the following affinity values M_{21} =1 and M_{12} =1. Also, we have M_{11} =1 and M_{22} =1.

e) Concentration: The affinity values are substituted in Farmer's equation to calculate the new concentration values of antibodies. The old concentration values of the antibodies are regularly updated with new ones. To make things clear, we take the same example as mentioned in sub-section 5.2 (d).By using Farmer's equation (1), the concentration values of antibodies Ab_1 and Ab_2 respectively can now be calculated. We have Ab_1 = 12, Ab_2 =7, M_{12}=1, M_{21}=1, M_i=0 and y_i=0 (due to absence of antigen), α_1 = α_2 =0.1, N=2, n=0. If an antibody does not interact with any of the antibodies for a longer period of time, then its concentration value is decreased by using the death rate factor i.e. α_2. Otherwise, α_2 is ignored during the initial stages of calculation of concentrations.By replacing the values in the above equation (1), we have

$\frac{dAb_1}{dt}$ =151.2 and $\frac{dAb_2}{dt}$ =152.4.

f) Generation of valid *n-gram* antibodies: A valid *n-gram* antibody is generated only when the probable bigram or higher-order *n-gram* antibodies are marked as correct and the individual units of the *n-gram* have concentration values higher than the pre-specified threshold. In sub-section 5.2 (e), we have seen that the new concentration values of Ab_1 and Ab_2 is 151.2 and 152.4 respectively. As the bigram pair (Ab_1, Ab_2) is correct and the concentration values of both the antibodies have increased, a new *n-gram* (bigram) antibody is generated viz. Ab_3= "*a lion*". The initial concentration value of the newly generated antibody *Ab3* is set to 1. It is likely that when Ab_3 interacts with other antibodies in future, its concentration value will increase. A copy of the newly generated *n-gram* antibody is added to the selected population of candidate antibodies. As a result the number of antibodies in the selected population increases. It now contains a mixture of unigrams and *n-gram* antibodies where n>1. After the completion of each iteration, again a single random antibody is selected from the initial population of the candidate antibodies and inducted into the selected population of mixed candidate antibodies. This is to introduce diversity in the selected population of candidate antibodies. Prior to induction of a new antibody, the older version of selected population of candidate antibodies has already participated in the training and has formed an ILN. The iteration continues until all the candidate antibodies in the initial population are exhausted or when the user terminates further generation.

g) Generation of New Sentences: In sub-section 5.2 (f),we have shown the process of generating new *n-gram* antibodies. An *n-gram* antibody is said to be valid if it is a syntactically and grammatically acceptable sentence or part thereof. This feature of finding valid *n-grams* is left to the user who verifies it manually. Only the grammatically *n-gram* antibodies where n>2 are stored separately and the collection of these *n-grams* forms the set of partial or complete sentences. The first and second term of Farmer's equation (1) represent the stimulations and suppressions from other antibodies. In section 5.2 (e), we have used Farmer's equation first to calculate the stimulation and suppression values between the antibodies and finally found the new concentration values of the antibodies. An ILN is thus formed due to the stimulations and suppressions. The ILN becomes a part of a larger network when more number ofantibodiesare introducedinto the system. Due to this reason the antibody population grows rapidly. To control the growth of the antibody population, measures are initiated to eliminate specific antibodies depending on a pre-specified threshold concentration value τ. The choice of value for τ plays a significant role. If τ is low, the rate of elimination of some candidate antibodies decreasesresulting in an increase inantibody population. A higher value ofτ increases the rate of elimination of antibodies resultingin the loss ofsome candidate antibodies that could have contributed to the construction of higher order *n-gram* antibodies. The choice of τ should be done judiciously and may requireseveral trials.The antibodies with concentration values greater than τ survive and become part of ILN.The ones having concentration values less than τ die and are thus flushed out from the network. In each of the iterations, we keep introducing a fresh supply of antibodies to the system. The existing antibodies that are part of the ILN, interact (based on the stimulations and suppressions) with these incoming antibodies. The incoming antibodies get stimulated and suppressed from other antibodies and as a result the existing ILN gets modified over the iterations.

6. RESULTS AND DISCUSSIONS

We have portrayed herein the results of the experiments based on an initial population of 80 unigram antibodies extracted from a small untagged corpus whose contents are shown in Table 2. The initial values of the parameters α_1 andα_2 wereset to 0.1.Since an antigen is absent in this case the value of $y_i$was set to 0.

Table 2. The contents of the corpus

Once upon a time there lived a lion in a forest. One day after a heavy meal, it was sleeping under a tree. After a while, there came a mouse and it started to play on the lion. Suddenly the lion got up with anger and looked for those who disturbed its nice sleep. Then it saw a small mouse standing trembling with fear. The lion jumped on it and started to kill it. The mouse requested the lion to forgive it. The lion felt pity and left it. The mouse ran away. On another day, the lion was caught in a net by a hunter. The mouse came there and cut the net. Thus it escaped. Thereafter, the mouse and the lion became friends. They lived happily in the forest afterwards. Moral: A friend in need is a friend indeed.

The list of nine randomly selected unigram antibodies along with their initial and new concentration values after six iterations of the training, are listed in Table 3. These nine antibodies comprise the selected population of candidate antibodies.

Table 3. The unigram antibodies and concentrations

Unigram antibodies	Initial concentrations (frequencies)	New concentrations calculated using equation 1
a	12	7948.674
lion	7	146117.08125
in	4	129472.252
there	4	60.8
lived	2	39.5
time	1	22.85
forest	1	22.85
upon	1	14.75
once	1	20839.359

The statistics of the number of valid *n-grams* generated at the end of the trainingare shown in Table 4. It is to be noted that these valid *n-grams* are added to the initially selected population of candidate antibodies. Thus, the updated selected population of candidate antibodies now contains a diverse population of antibodies (different types of *n-grams*).

Table 4. Statistics of *n-gram*antibodies generated after six iterations

Valid *n-gram* antibodies generated	Count
bigram antibodies	33
trigram antibodies	30
quadrigram antibodies	11
pentagram antibodies	9
hexagram antibodies	5
heptagram antibodies	3
octagram antibodies	2
n-gram (where *N*=9) antibodies	0
n-gram (where *N*=10) antibodies	1
n-gram (where *N*=11) antibodies	11

The partial listof correct and incorrect sentences that are generated after the training is shown in Table 5 and Table 6 respectively.

Table 5. Partial list of correct sentences generated after the training

Sl. No	Partial list of generated sentences which were correct
1	once upon a time there lived a lion in a forest
2	once in a forest there lived a lion
3	there lived a lion in a forest
4	there lived a lion
5	a lion lived in a forest
6	in a forest there lived a lion

Table 6. Partial list of incorrect sentences that are generated after the training

Sl. No	Partial list of generated sentences or portions thereof which were incorrect
1	a forest lived a lion
2	lion lived once in forest
3	there lived a forest
4	once in time a lion
5	once there forest lived a lion

When six numbersof sentences were generated from the updated selected population of candidate antibodies, a new unigram antibody was randomly selected from the initial population of candidate antibodies and added to the selected population of candidate antibodies. Table 7 and Table 8 show the list of correct and incorrect sentences generated after the addition of new unigram antibodies viz. "*was*"and *"sleeping"*with initial concentration values (frequencies) of 2 and 1 respectively.

Table 7. Partial list of correct sentences generated with the inclusion of new unigram antibodies "*was*" and "*sleeping*"

Sl. No	New unigram antibody with its initial concentration	Partial list of generated sentences which are correct
1	was:2	there was a lion
2		there was a forest
3		there was a lion in a forest
4		in a forest there was a lion
5		once upon a time there was a forest
6	sleeping:1	a lion was sleeping

Table 8. Partial list of incorrect sentences generated with the inclusion of new unigram antibodies "*was*" and "*sleeping*"

Sl. No	New unigram antibody with its initial concentration	Partial list of generated sentences which are incorrect
1	was:2	there lived a lion was in a forest
2		forest was a lion
3		once upon a time was a lion
4		once upon a time was a forest
5	sleeping:1	there lived a lion was sleeping in a forest
6		forest was sleeping
		lion in a forest was sleeping

Figure 8 shows a portion of the ILN when the whole of the corpus presented in Table 2 was used. The green bonds facilitate a change or jumping of tracks along the black lines while reading out the sentences contained in the network. The ILN is obviously yet to saturate since it carries a few incorrect sentences. By tracing out the paths of the ILN in Figure 8, we are able to generate the following sentences which are listed separately in Table 9.

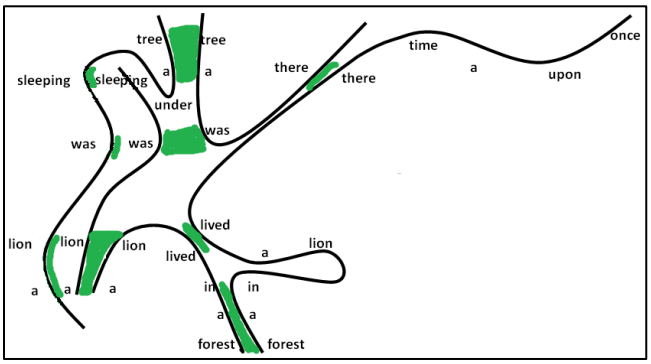

Figure 8. A portion of the ILN formed from the corpus in Table 2

Table 9. Partial list of generated sentences which were correct obtained from the ILN of Figure 8.

Sl. No	Partial list of generated sentences which were correct obtained from the ILN of Figure 8
1	once upon a time there lived a lion in a forest
2	a lion lived in a forest
3	a lion was sleeping
4	a lion was sleeping under a tree
5	once upon a time there was a tree

7. CONCLUSIONS

Using an existing corpus of words,we have been able to generate anIdiotypicLanguage network (ILN) from initially disconnected *n-grams* that form metaphors for the antibodies.With this network, new correct sentences or parts thereof were generated. With more interaction or effective addition of new *uni-* or *n-grams* the network strengthens itself and generates newer sentences. The generation of the ILN throws light on the possible mechanism of the biologicalequivalent of the generative grammar. The verification process is currently being done by the human user who in turn delivers the reward and penalty. To make the system autonomously generate the ILN, a larger and correct corpus could be used for such verification. The generated sentences or parts thereof could be verified for their correctness if they exist within this larger corpus. This would be analogous to how a child who continuously reads several books (the larger and correct corpus) gains in his/her sentence generation capability.In future, we intend to emulate and use this immuno-inspired approach in a distributed manner as described in [35] to enable the system to learn and converge faster.

8. REFERENCES

[1] Rosenfeld, R. 2000. Two decades of statistical language modeling: where do we go from here?, Proceedings of the IEEE, 88(8), 1270-1288.

[2] Jurafsky, D. and Martin, J. H. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics, 2nd edition, Prentice-Hall.

[3] Sleator, D and Temperley, D. 1991. Parsing English with a link grammar. Technical Report CMU-CS-91-196, Computer Science Department, Carnegie Mellon University, Pittsburgh,PA.

[4] Lafferty, J. D, Sleator, D. and Temperley, D. 1992. Grammatical trigrams: a probabilistic model of link grammar. In Proceedings of the AAAI Fall Symposium on Probabilistic Approaches to Natural language, Cambridge, MA.

[5] Chen, S. F. and Goodman, J. T. 1999.An empirical study of smoothing techniques for language modeling. Computer, Speech and Language.

[6] Waterfall, H.R., Sandbank, B., Luca Onnis, L. and Edelman, S. 2010. An empirical generative framework for computational modeling of language acquisition. Journal of Child Language, 37, 671-703.

[7] Jerne, N. K. 1974. Towards a network theory of the immune system. Ann. Immunol. Paris, 125C, 373.

[8] Knight, Chris, Kennedy, M. S. and Hurford, J. 2000. Editors. The evolutionary emergence of language: Social function and the origins of linguistic form, Cambridge: Cambridge University Press.

[9] Hurford, James, Kennedy, M. S. and Knight, C. 1998 Editors. Approaches to the evolution of language, Cambridge: Cambridge University Press.

[10] Steklis, H. and Harnad, S. 1976. From hand to mouth: Some critical stages in the evolution of language Editors. Origins and Evolution of Language and Speech. Annals of the New York Academy of Sciences, Volume 280, pages 445—455.

[11] Cangelosi A. andParisi D. 2002. Computer simulation: A new scientific approach to the study of language evolution. In: Cangelosi A, Parisi D, editors. Simulating the evolution of language. Springer; London, pp. 3–28.

[12] Chomsky, N. 1972. Language and Mind. San Diego, Harcourt Brace Jovanovich.

[13] Slobin, D. I. 1979. Psycholinguistics. Gelnview, Illinois, Scott, Foresman and Company.

[14] Pinker, S. 1994. The Language Instinct, Penguin.

[15] Bickerton, D. 1990. Species and Language. Chicago, Chicago University Press.

[16] Pinker, S. and Bloom, P. 1992. Natural Language and Natural Selection. In The Adapted Mind. J. H. Barkow, L. Cosmides and J. Tooby (Eds.), Oxford University Press.

[17] Elman, J. L. 1999. The Emergence of Language: A Conspiracy Theory. In The Emergence of Language. B. MacWhinney (Ed.). Mahwah, New Jersey, Lawrence Erlbaum Associates**:** 1-28.

[18] Kirby, S. 1998. Fitness and the Selective Adaptation of Language. In Approaches to the Evolution of Language. J. R. Hurford, M. Studdert-Kennedy and C. Knight (Eds.), Cambridge University Press**:** 359-383.

[19] Cavalli-Sforza, L. L. and Feldman, M. W. 1978. Towards a Theory of Cultural Evolution. Interdisciplinary Review 3: 99-107.

[20] Peng, F. 2001. The sparse data problem in statistical language modeling and unsupervised word segmentation. PhD proposal, http://citeseer.ist.psu.edu/489036.html.

[21] Nadkarni, P.M, Ohno-Machado, L. and Chapman, W.W. 2011. Natural language processing: an introduction. J Am Med Inform Assoc. 18(5):544–51,DOI=http://10.1136/amiajnl-2011-000464.

[22] Zheng J, Chen Y. and Zhang W. 2010. A survey of artificial immune applications, Artificial Intelligence Review, Springer, Volume 34, Issue 1, pp 19-34.

[23] Kumar, A. and Nair, S. B. 2007. Artificial Immune Systems, 6th International Conference, ICARIS 2007, Santos, Brazil, Proceedings. Lecture Notes in Computer Science 4628, Springer, ISBN 978-3-540-73921-0 pp. 348 – 357.

[24] Forrest, S., Perelson, A., Allen, L. andCherukuri, R. 1994. Selfnonself discrimination in a computer. Proceedings of the IEEE Symposium on Security and Privacy, IEEE Computer Society Press, Los Alamitos.

[25] Yue, X., Abraham, A., Chi, Z-X., Hao, Y. Y. and Mo, H. 2007. Artificial immune system inspired behavior-based anti-spam filter, Soft Computing, 11, pp. 729-740.

[26] de Castro and L.N., Timmis, J. 1992. Artificial immune systems: A new computational intelligence approach, Springer, London.

[27] Farmer, J.D, Packard, N.H. and Perelson, A.S. 1986. The immune system, adaptation, and machine learning,Physica, vol. 22, pp. 187-204.

[28] Pacheco, J and Costa, J.F. 2007. The abstract immune system algorithm, 6th International Conference on Unconventional Computation, Kingston, Canada.

[29] Hart, E and Ross, P. 2005. The impact of the shape of antibody recognition regions on the emergence of idiotypic networks, International JournalofUnconventional Computing, 1 (3), pp. 281–313.

[30] Dipankar, D. and Fernando, N. 2008. Immunological Computation: Theory and Applications, Auerbach Publications.

[31] Harmer, P. G, Williams, P. D., Gnush and Lamont, G. 2002. An artificial immune system architecture for computer security applications. IEEE Transaction on Evolutionary Computation, 6(3), 252–280.

[32] Percus, J. K., Percus, O. E. and Perelson, A. S. 1993. Predicting the size of the T-cell receptor and antibody combining region from consideration of efficient self-nonself discrimination. Proceedings of National Academy of Sciences USA 90, Las Vegas, pp. 1691–1695.

[33] Balthrop, J. F., Esponda, Forrest S. and Glickman, M. 2002. Coverage and generalization in an artificial immune system. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002), pp. 3–10, Morgan Kaufmann Publishers, New York.

[34] Ishiguro, A., Shirai, Y., Kondo, T. and Uchikawa, Y. 1996. Immunoid: An architecture for behavior arbitration based on the immune networks, Proc. Int. Conf. Intelligent Robotics and Systems, pp.1730 -1738.

[35] Jha, S.S., Shrivastava, K. and Nair, S.B. 2013. On Emulating Real-world Distributed Intelligence using Mobile Agent based Localized Idiotypic Networks, The First International Conference on Mining Intelligence and Knowledge Exploration (MIKE 2013), Virudhunagar, India, Springer International Publishing, LNAI, Vol. 8284, pp. 487-498.