

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590018

A Project Report on

**“AUTOMATED PHRASE MINING AND IDENTIFYING SENTENCE BETWEEN ACTIVE AND PASSIVE USING NEURAL NETWORK”**

Project report Submitted in partial fulfillment of the requirements for the Award of degree of

**BACHELOR OF ENGINEERING IN INFORMATION SCIENCE AND ENGINEERING**

**15CSP85**

**Submitted by:**

|  |  |  |
| --- | --- | --- |
| **Indira H k** |  | **[1JT16IS017]** |
| **Anusha V** |  | **[1JT16IS004]** |
| **Chandana Chowdhary** | **K** | **[1JT16IS009]** |
| **Deepika B** |  | **[1JT16IS013]** |

**Under the Guidance of**

**Pallavi C V**

# Assistant Professor, Dept. of ISE

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING JYOTHY INSTITUTE OF TECHNOLOGY, BENGALURU. 560082 2019-2020**



**JYOTHY INSTITUTE OF TECHNOLOGY, BENGALURU-560082 DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

CERTIFICATE

Certified that the project work entitled **“AUTOMATED PHRASE MINING AND IDENTIFYING SENTENCE BETWEEN ACTIVE AND PASSIVE USING**

**NEURAL NETWORK”** carried out by **Ms. Indira H K,**USN: **1JT16IS017, Ms. Anusha V,** USN:**1JT16IS004, Ms. Chandana Chowdhary K,**USN: **1JT16IS009, Ms. Deepika B,** USN:**1JT16IS013, ,** who are bonafide students of Jyothy Institute of Technology, submitted in partial fulfillment for the award of **Bachelor of Engineering** in Information Science and Engineering department of the **Visvesvaraya Technological University**, Belagavi during the year 2019-2020. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

\_ \_

**Signature of the Guide Signature of the HOD Signature of the Principal** Mrs.Pallavi C V Dr. Harshvardhan Tiwari Dr. Gopalakrishna K Assistant Professor, Associate Professor and Head, Principal,

Dept. of ISE, JIT. Dept. of ISE, JIT. JIT, Bengaluru

**Name of the Examiners Signature with date:**

1. \_

2. \_



**DECLARATION**

We, **Indira H K ,Anusha V, Chandana Chowdhary K,Deepika B** students of Eighth semester BE, in the Department of Information Science and Engineering, Jyothy Institute Of Technology, Bengaluru, declare that the project work entitled **“AUTOMATED PHRASE MINING AND IDENTIFYING SENTENCE BETWEEN ACTIVE AND PASSIVE USING NEURAL NETWORK”** which

is being submitted by us in the partial fulfillment for the award of the degree of **Bachelor of Engineering in Information Science and Engineering, from Visvesvaraya Technological University, Belagavi** carried out during the academic year 2019-2020, under the guidance of **Mrs**. **Pallavi C V**, Assistant Professor, Department of Information Science & Engineering, Jyothy Institute Of Technology, Bengaluru. The matter embodied in the project report has not been submitted previously for the award of any degree or diploma by us to any other university or institution.

1. **Indira H k 1JT16IS017**
2. **Anusha V 1JT16IS004**
3. **Chandana Chowdhary K 1JT16IS009**
4. **Deepika B 1JT16IS013**

Department of Information Science and Engineering, Jyothy Institute of Technology,

Bengaluru 560082

# ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise, does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. Firstly, we are very grateful to this esteemed institution “Jyothy Institute of Technology” for providing us an opportunity to complete our project.Our deep and profound gratitude to our guide **Mrs. Pallavi C V**, Asst. Prof. for his keen interest and boundless encouragement in preparing this work.We would like to thank **Dr. Harshvardhan Tiwari,** Associate prof. and head of the Dept., Information science and Engineering for providing us an opportunity and for his valuable support.We express our sincere thanks to **Dr. Gopalakrishna K,** Principal, JIT for providing us with adequate facilities to undertake this project.

We would, also like to take this opportunity to express our gratitude for the support and guidance extended to us by the faculty members of the Information Science and Engineering Department.

Finally, we would thank all our friends who have helped us directly or indirectly in this project.

# ABSTRACT

Parts-of-speech tagging,is the process of assigning POS to words in sentences, has a vast field of applications in natural language processing to identify the active and passive sentence same.

It constitutes an important intermediate step in other tasks such as syntactic analysis or machine translation. Out of the methods that have been employed in solving this problem, neural networks belong to the rather non-typical ones, being often neglected in textbooks.

In this work we provide an overview on some notable attempts that have been made in parts-of- speech tagging with neural networks. Based on these, we also propose our own tagger based on similar principles.

The Objective of this system is touses artificial neural networks it builds the layered network in the form of cluster to compare with the text with the datasets using neurons. As is the case with all parts-of-speech tagging techniques, two factors had to be taken into consideration: accuracy,i.e. the percentage of tokens in the test data which were assigned the correct tag, and time complexity, i.e. how much time is needed to train the tagger, how long the tagging itself takes and how fast the required time grows as the training and test sets get bigger.

In addition to tagging we discuss about identifying the Active and Passive English sentence are same or not using the Artificial neural network for POS tagging that is, Parts-of-speech tagging to identify the subject, action and object.

This proposed theory provides the accuracy more performance in processing so many words or sentences in English by layering and assigning to neuron. This project can be used in identifying grammar, Plagiarism to check sentence turning etc.

# Table of Contents

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| 1 | **Chapter 1: INTRODUCTION**   * 1. Overview   2. Motivation 1.3Proposed System   1.3.1 Advantages of proposed system   * 1. Artificial Neural Network   2. Summary | 1  1  1  1  1  2  4 |
| 2 | **Chapter 2: LITERATURE SURVEY**   * 1. Simple and effective neural model for joint word segmentation and POS tagging   2. Joint POS tagging and dependency parsing with transition based neural network   3. Automated phrase mining from massive textcorpora   4. Using lexical chains to identify the text difficulty   5. Different types of POS tagger   6. Scalable topical phrase mining from text corpora   7. Paraphrase identification using lexical   8. Existing system      1. Disadvantages of exsisting system | 5  5  5  6  6  6  7  7  8  8 |

|  |  |  |
| --- | --- | --- |
| 3 | **Chapter 3: METHODOLOGY**   * 1. Data sets   2. Artificial neural network construction   3. POS- Guided phrasal segmentation   3.3 Classification using lexical classifier | 9  9  9  10  10 |
| 4 | **Chapter 4: SYSTEM DESIGN**   * 1. System architecture   2. Data flow diagram   3. Use case diagram | 12  13  14  14 |

|  |  |  |
| --- | --- | --- |
| 5 | **Chapter 5: SYSTEM REQUIREMENTS SPECIFICATION**   * 1. Functional requirements   2. Non-Functional requirements   3. Software requirements      1. Java   4. Hardware requirements | 16  16  16  17  17  19 |
| 6 | **Chapter 6: TESTING**   * 1. Types of testing      1. Unit Testing of System      2. Integration Testing of System      3. Functional Testing of system      4. System Testing      5. White-Box Testing of System      6. Black-Box Testing of System      7. Acceptance Testing | 20  20  22  23  23  23  23  23 |
| 7 | **Chapter 7: RESULT ANALYSIS AND SCREENSHOTS** | 24 |

|  |  |  |
| --- | --- | --- |
| 8 | **Chapter 8: CONCLUSION** | 27 |
|  | **REFERENCES** | 28 |

**Table of Figures**

|  |  |  |
| --- | --- | --- |
| **Fig.**  **No.** | **Description** | **Page No.** |
| 1 | Hidden layer | 4 |
| 2 | Bidirectional Dependency Network | 10 |
| 3 | Automated Phrase mining | 10 |
| 4 | Sentiment Classification with Lexicons | 11 |
| 5 | POS Tagging | 13 |
| 6 | System Design | 14 |
| 7 | Flow Chart | 15 |
| 8 | User Interface | 24 |
| 9 | Text Preprocessing | 25 |
| 10 | POS Tagging | 25 |
| 11 | Feature Extraction | 26 |

**Chapter 1**

**INTRODUCTION**

### Overview

The proposed project uses artificial neural networks it builds the layered network in the form of cluster to compare with the text with the datasets using neurons. As is the case with all parts-of- speech tagging techniques, two factors had to be taken into consideration: accuracy, i.e. the percentage of tokens in the test data which were assigned the correct tag, and time complexity, i.e. how much time is needed to train the tagger, how long the tagging itself takes and how fast the required time grows as the training and test sets get bigger.

### Motivation

The project aim is to discuss about identifying the Active and Passive English sentence are same or not using the Artificial network for POS tagging ie., Parts-of-speech tagging to identify the noun, verb and adjective. This proposed theory provides the accuracy more performance in processing so many words or sentences in English by layering and assigning to neuron. This project can be used in identifying grammar, Plagiarism to check sentence turning etc.

### Proposed System

The proposed system uses artificial neural networks it builds the layered network in the form of cluster to compare with the text with the datasets using neurons. As is the case with all part-of- speech tagging techniques, two factors had to be taken into consideration: accuracy, i.e. the percentage of tokens in the test data which were assigned the correct tag, and time complexity, i.e. how much time is needed to train the tagger, how long the tagging itself takes and how fast the required time grows as the training and test sets get bigger.

### Advantages of Proposed System

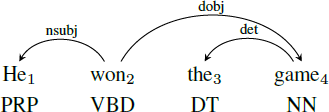
* + - 1. Fast in performance
      2. Time complexity is less
      3. Support the huge dataset
      4. Accurate results

### Artificial Neural Network

Artificial neural networks (ANN) is a biologically influenced concept commonly used within the field of AI (Russel & Norvig, 2010:738). The idea for this concept originates from a highly abstract view of the brain, where information is transmitted through the network (brain). This is done by firing neurons that send their electrical charge through their many axon. The information is then processed by the receiving neuron through their many dendrites. If the cumulative information collected from all the dendrites exceeds the neurons activation threshold, the receiving neuron will fire and propagate information itself. An ANN works in a similar way, where a sending unit, a perceptron, is sending information (output), to the perceptron that it is connected to. The difference is that the modern perceptron lack an activation threshold and will always output information, even though that information might sometimes be zero (not to be confused with null). For the original perceptron, see Rosenblatt (1958).

Parts-of-speech tagging could be viewed as a classification problem, which merits the exploration of methods that are successful in solving similar types of problems. The method of choice in this thesis is the use of artificial neural networks using a purely contextual representation of words (Word2Vec) as a possible solution for parts-of-speech tagging. Artificial neural networks also allow the concept of deep learning, meaning they can create highly abstract representations of the data with the addition of more hidden layers. These networks should be able to excel in parts-of-speech tagging without the drawbacks listed above. For example, no handcrafted features would be required; those features will be automatically discovered by the network. The network itself will not be language dependent, although the model will have to be trained on the new language, but no part of the code will have to be changed for it to be successful.

Phrase mining refers to the process of automatic extraction of high-quality phrases (e.g., scientific terms and general entity names) in a given corpus (e.g., research papers and news). Representing the text with quality phrases instead of n-grams can improve computational models for applications such as information extraction/retrieval, taxonomy construction, and topic modeling.



Almost all the state-of-the-art methods, however, require human experts at certain levels. Most existing methods [13], [31], [38] rely on complex, trained linguistic analyzers (e.g., dependency parsers) to locate phrase mentions, and thus may have unsatisfactory performance on text corpora of new domains and genres without extra but expensive adaption. Our latest domain-independent method Seg Phrase [23] outperforms many other approaches, but still needs domain experts to first carefully select hundreds of varying quality phrases from millions of candidates, and then annotate them with binary labels.

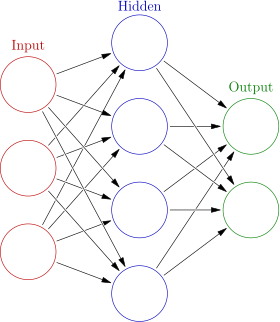
Such reliance on manual efforts by domain and linguistic experts becomes an impediment for timely analysis of massive, emerging text corpora in specific domains.

An ideal automated phrase mining method is supposed to be domain-independent, with minimal human effort 1 or reliance on linguistic analyzers. Bearing this in mind, we propose a novel automated phrase mining framework.

Word segmentation and POS tagging

Word segmentation and POS tagging have been two fundamental tasks for Chinese natural language processing (NLP). As Chinese sentences do not have explicit boundaries among words, word segmentation has been a prerequisite step, and then POS tagging is performed based on segmented sentences in aligning with the models of other languages such as English. State-of-the-art approaches treat both task as structural learning problems, using either sequence labeling models or transition-based incremental models [3]–[6]. The former are able to exploit dynamic decoding, while the later are flexible of feature engineering. As word segmentation and POS tagging are highly correlative tasks, joint models which perform both tasks in a single model have achieved better performances, due to their capabilities of reducing error propagation and making use of mutual interaction between the two tasks [7]–[10]. Among the work of joint word segmentation and POS tagging, Zhang et al. (2010) [11] is one of the representative methods because of its simplicity and

high efficiency. It exploits a transition-based framework, being able to use both character and word- level features. By decoding incrementally to predict a sequence of transition actions, the model behaves quite similarly to the sequence-to-sequence (Seq2Seq) neural models.



### Fig.1.Hidden Layer

* 1. **Summary**

This project is to discuss about identifying the Active and Passive English sentence are same or not using the Artificial network for POS tagging ie., Parts-of-speech tagging to identify the noun, verb and adjective. This proposed theory provides the accuracy more performance in processing so many words or sentences in English by layering and assigning to neuron. This project can be used in identifying grammar, Plagiarism to check sentence turning etc.

# Chapter 2

**LITERATURE SURVEY**

The following papers were referred for our work:

### Simple and effective neural model for joint word segmentation and POS tagging

Zhang ,Yu , and Fu reference paper[1] the authors have worked on joint POS tagging and dependency parsing have focused on developing linear models with feature templates. They introduce transition systems which will perform POS tagging and dependency parsing in an exceedingly joint search space. Our transition system differs from previous add the separation of structural, tagging, and labeling actions. This ends up in three small classifiers with fewer classes (i.e., 𝑗𝑇𝑗 for the tag classifier, 3 for the shift/reduce classifier, and𝑗𝐿𝑗 for the label classifier) instead of one big classifier with rather more classes (i.e.,𝑇𝐽 + 2𝑗𝐿𝑗).More importantly, we use continuous representations instead discrete indicator features to create the classifiers. As indicated by Chen and Manning, lexicalized indicator features crucial for improving parsing accuracy are highly sparse and infrequently incomplete. Alternatively, we resort to neural networks to be told representations from data to bypass the sparsity and incompleteness problems. Another advantage of using neural networks is that there is no need to compose individual features to get more complex features like conventional discriminative dependency parsing. It should even be mentioned that there are interactions between the selections regarding POS tagging and syntactic parsing in research on constituent parsing.

### Joint POS tagging and dependency parsing with transition based neural network

Yang, Zhong, Yong Liu , Masong Sun , Non Yu , Guohong Fuin their paper[2] lexical classifier the authors have worked on recent advances in applying neural networks to POS tagging, dependency parsing and constituency parsing. Among them, our work bears the foremost resemblance, which propose stack-propagation to integrate a tagging model into a neural parser. They propose a stacked pipeline of models and utilize POS tags as a regularized of learned representations. While Zhang and Weiss use the hidden layer of the tagger network because the input for the parser, we have an interest in enabling tagging and parsing to learn one another in an exceedingly joint search space. As a result, the tagger is ready to resolve long-distance tagging ambiguity by exploiting syntactic information.

Meanwhile, the error propagation problem the parser faces is alleviated because of the cascaded error reduction by joint modeling.

### Automated phrase mining from massive text corpora

In this paper, the authors wrote about the large corpus inference paper [3] which states that the tagger provides two rather different training methods which will be chosen freely. The primary method employs a group of recurrent multilayer perceptron networks which learn the foremost likely tags from the word-to-tag probabilities of the words within a context. The second method converts words into feature vectors in an exceedingly multidimensional space; subsequently, the hyper planes separating the data in one class from the opposite ones are explore for using perceptron. An extra statistical method is on market as a baseline to check the performance. Training on the primary 999,998 words within the Brown corpus and evaluating on the remainder, the most effective accuracy was 94.93%, achieved by the primary method. The second method was significantly more successful for smaller training sets, nevertheless, long training times prevented us from determining the accuracy for the most important set. Both methods did systematically better than the baseline.

### Using lexical chains to identify the text difficulty

In this paper the authors wrote about different kinds of POS taggers in the reference paper[4] which states that choosing proper tag for context is a challenge for POS taggers. POS Tagger is could be a software which reads words of specific language and allocates parts of speech to each word. Process of POS tagging is shown in below figure. Initially the input sentence is tokenized into words within the next step. For these tokens POS Tagging method is employed further to assign the tags. Finally, the tagged output is produced. POS tagging methods include supervised, unsupervised, rule-based, stochastic and hybrid approaches. Supervised POS tagging models improve with increase in size of corpus and requires pre tagged models. Unsupervised POS Tagging automatically initiate the tag sets and transform rules. Rule-based POS Tagging utilizes grammatical information and hand-written rules for tagging words. Stochastic models are supported on diverse techniques such as MLE, n- grams or HMM. These authors in [4] have provided the comparison of performance of various tagging techniques. Rule-based, stochastic and hybrid approaches have produced good performance results. Stanford and Tree tagger produces better tag sets.

### Different types of POS tagger

In this paper, the author wrote about the lexical classifier, the sentences are supported featured classification that ranges on many bag of words. The lexical chain features performed significantly better than bag of words across all the classifier and achieves the accuracy. Finally we found it provide the precise Information for identifying difficult sentences of text, it’s semantically related words during the document which we found difference between easy and difficult text for lexical average chain length and also the cross chains. These authors in[5] have provided the lexical classifier difference between the straightforward and difficult text by using significant performed featured lexical chain.

### Scalable topical phrase mining from text corpora

In this paper, the author wrote about the foremost modeling algorithms model text corpora with unigrams. unigram: in field of linguistics and probability an n gram is contiguous sequence of n items from given sample of text or speech , n gram of size 1 is referred as unigram. Human interpretation often relies on inheriting grouping of terms into phrases. So during this paper they consider the matter of discovering topical phrases of assorted lengths. Existing work either performs post processing to the results of unigram based topic models or utilizes complex n gram discovery topic models. These methods generally produce inferiority topical phrases or poor scalability on even moderately sized datasets. Hence this paper proposes different approach that’s both computationally efficient and effective. Solution combines a completely unique phrase mining framework to segment a document into single and multi word phrases and new topic model that operates on induced document partition. Topic model: form of statistical model for locating abstract topics that occur in collection of documents. And this approach provides prime quality topical phrases with negligible extra cost to bag of words topic model in an exceedingly very variety of datasets including search publication titles, abstracts, reviews and new articles.

### Paraphrase identification using lexical

In his paper, the author wrote about the linguistic communication, which understand the complex task humans learn from childhood to automate the tactic of meaning understanding for computer is great real word applications. From the unstructured linguistic communication text the easy text processing the meaning, the machine learning system uses lexical features for sentence level paraphrase identification. These authors in [6] have provided the identifying the meaning which is

comparable between two text segments, results with dependency features are sensitive to minor syntactic change.

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference** | **Paper Name** | **ShortCome** | **Author** |
| **Paper 1** | Simple and effective neural model for joint word segmentation and POS tagging | POS TAGGING | Meishon Zhang, NonYu, Guohong Fu |
| **Paper 2** | Joint POS tagging and dependency parsing with transition based neural network | LEXICAL CLASSIFIER | Liner Yang,Meishon Zhong,Yong Liu, Masong Sun, Non Yu,Guohong Fu |
| **Paper 3** | Automated phrase mining from massive  text corpora | MASSIVE CORPUS | Jingbu Shang, Jailu Liu, Meng  Jiang |
| **Paper 4** | Using lexical chains to identify the text difficulty | LEXICAL CLASSIFIER | Kanakaraddi |
| **Paper 5** | Different types of POS tagger | POS TAGGING | Partha Mukherje |
| **Paper 6** | Paraphrase identification using lexical | Classifier | Rutal S. Mahajan |

### Existing System

The existing system uses text mining or data mining concept where all the English words should be stored either in the text file or data base it extract sequentially comparing with the training data set to give the results.

### Disadvantages of Existing System

* + - 1. Huge space is required
      2. Lack of performance
      3. More complex
      4. Processing Is sequential
      5. Time consuming

# Chapter 3

**METHODOLOGY**

### Data sets

The data used for the purpose of training and evaluating the ANN for this thesis originate from the Stockholm-Umeå corpus also known as SUC (Gustafson-Capková and Hartmann, 2006). This corpus is a balanced corpus composed of text from the 1990’s and contain a total of one million words. Each word in the corpus were tagged with one of the 25 POS tags in the data set (for a complete list, see appendix A) by experts. The data were shuffled and split into a training set (85%), a validation set (5%) and an evaluation set (10%). In addition to these data sets, a Word2Vec set was created composed of SUC and the first 80 million words of the Swedish Wikipedia corpus (Språkbanken, n.d.), resulting in a total of 81 million token.

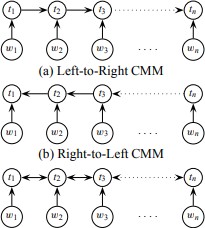
### Artificial neural network Construction

An ANN1 was created in Python in a way that allowed matrix multiplications to be used in the updating of the weights and biases. The reason for this was to take advantage of the computational friendly framework supplied by the numpy library. Other dependencies were gensim (Word2Vec) and codecs (reading Swedish letters from file).

The perceptron were modeled as two n-dimensional matrixes (depending on the amount of layers). The first matrix contained all the connecting weights while the second matrix contained all the biases of the perceptron, reducing the information associated with the perceptron to two arrays of numbers. As an activation function for the network, a sigmoid function (Eq. 5) was used.



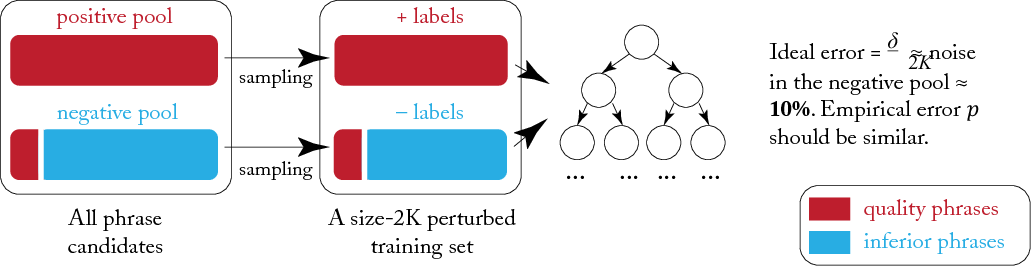
The pseudo code containing the creation of the network, prediction procedure, back propagation calculation and network training can be seen below in Figure 1.



### Fig.1.Bidirectional Dependency Network

* 1. **POS-Guided Phrasal Segmentation**

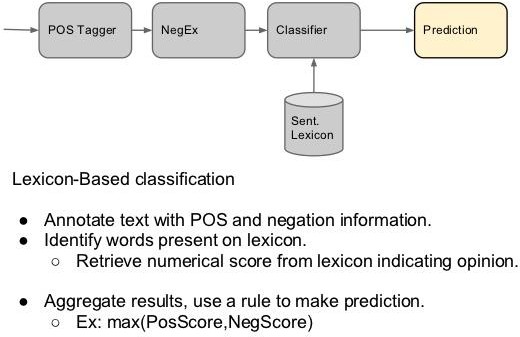
Phrasal segmentation addresses the challenge of measuring completeness (our fourth criterion) by locating all phrase mentions in the corpus and rectifying their frequencies obtained originally via string matching.



### Fig.2.Automated Phrase mining

* 1. **Classification using Lexical Classifier**

**Tag Classification**: Resolving the tag conflict is a jT j- class classification problem. Instead of using conventional feature templates that are highly sparse and inevitably incomplete, we leverage a neural network based classifier. To determine the POS tag of the last word added to the stack, which is represented as xS0 ,



### Fig.3.Sentiment Classification with Lexicons

**Chapter 5**

**SYSTEM DESIGN**

The purpose overdue the layout stage is to make mastermind an answer to issue dictated by the need record. Underlying stage in moving from the territory to the course of action space. The design of the framework is perhaps the most segregating component impacting the way of the item. The blueprint of the framework is possibly the most fundamental component affecting the way of the item and note worthily affects the later stages, particularly testing and upkeep.

Framework plot means to recognize the components that should be in the framework, these modules and to work together with each other to convey the fancied results. Around the end of the framework layout all the genuine data structures, record associations, yield bunches furthermore genuine modules in the framework and their points of interest are picked.

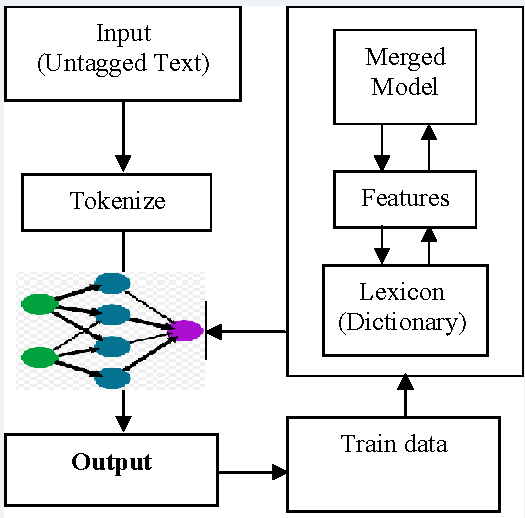
The configuration stage is to position answer of the matter determined by the precondition report. This stage is the underlying stage in moving from the issue space to the plan range. The setup of the system is possibly the most fundamental variable impacting the way of the item. The layout of the system is perhaps the most essential segment impacting the way of the item and significantly influences the later stages, particularly testing and upkeep.

System arrangement hopes to perceive the modules that should be in the structure, the particulars of these modules and to associate with each other to make the looked for results. Around the end of the system arrange all the genuine data structures, record outlines, yield bunches and what's more genuine modules in the system and their points of interest are picked.

### The model contains of following separate level:

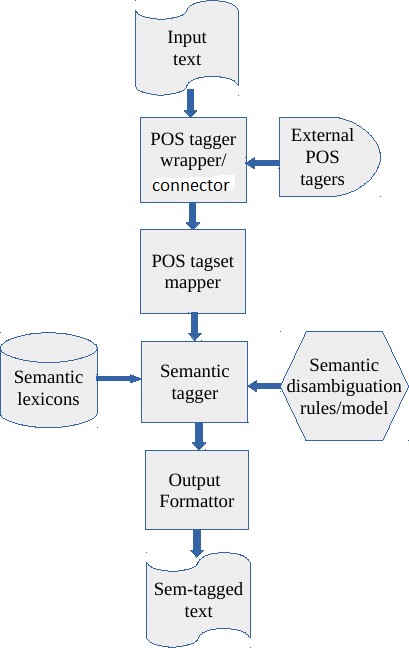
* + - * Requirement and Definition: In this level the reprobate is indicated alongside the needed administration targets and the imperatives are distinguished
      * System and Software Designing: In this level the plan details are deciphered into a product depiction. The product engineer at this level is on edge with information structure, programming design, calculation subtle elements and interface depictions.
      * Implementation: In this level the outlines are deciphered into the product space.
      * Component, Integration and System Testing: Testing at this level concentrates on ensuring that any mistakes are perceived and that the product meets its required necessity. After this stage the product is conveyed to the client
      * Processes and Maintenances: In this level the product is effective to meet the changing client needs, adjusted to oblige varieties in the outside environment, right mistakes and exclusions beforehand undetected in the testing stages, upgrading the productivity of the product.

### System Architecture



**Fig.4. POS Tagging**

* 1. **Data Flow Diagram**



**Fig.5. System Design**

* 1. **USE CASE DIAGRAM**

Use cases are used in the midst of necessities elicitation and examination to address the handiness of the system. A Use case diagram focus on the behavior of the structure from an outside point of view.



POS Tagging

Read

Upload

Feature Extraction

Word Sense

Text to voice

### Fig.6.Flow Chart

**Chapter 6**

**SYSTEM REQUIREMENTS SPECIFICATION**

* 1. **Functional Requirements**

Utilitarian Requirement portrays a segment of an item scheme and structure must portable when given detailed inputs and conditions. In this structure taking after are the reasonable requirements are

* + - Parts of speech tagging to tag sentences to verb and noun
    - Feature Extraction to segregate the words
    - Word sense occurrences
    - Text to voice recognition

### Non-Functional Requirements

Nonfunctional requirements describe how a system must behave and establish constraints of its functionality. This type of requirements is also known as the system’s *quality attributes*. Attributes such as performance, security, usability, compatibility are not the feature of the system, they are a required characteristic. They are "developing" properties that emerge from the whole arrangement and hence we can't compose a particular line of code to execute them. Any attributes required by the customer are described by the specification. We must include only those requirements that are appropriate for our project.

Non-Functional Requirements are as follows:

* + - **Usability:** The connection is accommodated every edifice. The client has the volume view and makes sections in the structures. Acceptances are given in every document to stay away from conflicting or invalid passage in the application/system.
    - **Availability:** System will be accessible all the time with the exclusion of the time needed for the strengthening of information.
    - **Portability:** The application is produced in Java. It is compact to other working system gave JDK is available to the working system.
    - **Integrity:** The task work is fundamentally outlined in an incorporated development environment, where every class, individuals, traits is planned under java bundle. Building and examining the principle class will incorporate every one of the classes in like manner for the best possible assemblage of the undertaking work.
    - **Extensibility:** The venture work is similarly open for any future alteration.

### Software Requirements

* + - Operating System : Windows
    - IDE : Net Beans 7.3.1
    - Technology : Java and J2EE
    - Java Version : JDK1.7

### JAVA

Java technology is high level programming language, can be characterized by following

* + - * **Simple** based on OOP standard and designed in such way that people can learn easily if they recognize the basic concept of C++.
      * **Object Oriented** In everything is treated as object and it can be easily protracted since it is founded on the object model.
      * **Distributed** process of making a facility or a product that is available for use of the customer and business users. By direct means of work or indirect means of work with the help of mediators. The three parts of marketing mixes are product, pricing.
      * **Multithreaded** this feature is conceivable to write plans that can do multiple tasks concurrently. This plan allows programmers to hypothesis smoothly running cooperative system.
      * **Dynamic** is intended to adjust to creating environment. Java programs convey more measure of runtime confirmation that can be utilized to check and resolve gets to objects on run time.
      * **Architecture Neutral** compiler generates construction impartial object code which makes the byte code to be executable on many platforms, with attendance of java run time system.
      * **Portable** operation dependent aspects of the requirement make java portable.
      * **High Performance** use of Just In Compilers increases high performance.
      * **Robust:** Java removes error disposed to situations by highlighting mainly on compile time error and run time checking.
      * **Secure:** Secure feature allows to develop virus free systems.

### Java Platform Discussion

A Java stage is things or environment in which a framework runs. Without a doubt the most surely understood stages are Windows 2000, Linux, Solaris and Mac OS. The Java stages are showed as a mixture at work framework and hardware. It is not the same as other stage is that it is simply programming organize that continues running on top of other gear based stages.

### The Java has modules:

* + The Java-Virtual-Machine (JVM)
  + The Java-Application-Programming-Interface (JAPI)

### Naming Convention

Enlightening names are one of the finest implements for making effectively justifiable code. Dodge theoretical names that are prone to be reused by different parts of the framework.

### Function Naming

Functions names must recognize the activity performed or the data gave by the capacity. Steps are taken to see that the names will for the most part start with a verb.

### Variable Naming

The sort and motivation behind every variable ought to be obvious the code is utilized. E.g. per user will anticipate that username will be no less than 6 characters. Names are shaped from delimited by lower case letters, composite words, with underscores permitted.

Variable names are given long, and important. Variables are named with the substance of the variable. All variables contains lower case letters for e.g., which is the cluster comprising of the landing values.

The sort and purpose behind each variable is evident inside the code in which it is used. Per user will foresee that username will be no under 6 characters.

* + - * Variable names are given long and imperative.
      * Variables are termed with the substance of the variable.
      * It contains lower case letters, which is the group involving the passage values.

### Hardware Requirement

* + - Hardware Dual Core
    - Hard Disk 50 GB
    - Speed 1.4 GHz
    - RAM 1GB
    - Key Board Standard Keyboard
    - Touch Pad Button Mouse
    - Monitor

# Chapter 7

**TESTING**

The inspiration driving is to find defaults in the errand. The testing is the explanation behind attempting to discover every lack or deficiency in a working things. It gives the best way to check the suitability of sections, sub-assemblies and a completed thing. It is the method of working with the arrangement of confirming that the item framework meets its chucks and the customer covets and does not miss the mark in an unacceptable way. There are characteristic sorts of test. Every test category discourses a precise testing need.

### Types of Testing

* + 1. Unit
    2. Integration
    3. Functional
    4. System
    5. White-Box
    6. Black-Box
    7. Acceptance

### Unit Testing of System

Testing fuses the sketch of examinations that endorse that the inner venture reason is working truly, and that program inputs extensive yields. All inside code and branches ought to be insisted. It is utilization to test the individual programming unit of an application. Unit testing is done before breaker. It is the central testing that depends heaps of its change and is intruding.

Unit testing guarantees that each exceptional strategy for a occupational system perform correctly to the recorded purposes of interest and covers obviously depicted inputs and output result.

This performs indispensable tests part by part level and test a specific occupational framework, application, structure course of action. Unit testing is for the most part decided as mutual code and unit test of the thing lifecycle.

### Test plan and method

* + - * + It will be performed physically and practical and will be composed in point of interest.

### Objectives

* + - * + Field accesses essentially work properly.
        + Folios must be triggered from the recognized link.
        + The entrance display, responses must not be late.

### Features

* + - * + Link’s should take to the correct page.
        + Confirm that the accesses are of the format.
        + No same entries should be allowed.

### TEST CASES

|  |  |
| --- | --- |
| Name | Preprocessing the text |
| Description of test  cases | To identify text like ion, sion , and recursing word |
| Sample Input | Test Files |
| Expected Output | Processing word with removal |
| Actual Output | Same as Expected |
| Result | Pass |

|  |  |
| --- | --- |
| Name | POS Tagging |
| Description of test  cases | To identify the noun, verb and adjective |
| Sample Input | Words |
| Expected Output | Words assign to their respective tagger |
| Actual Output | Same as Expected |
| Result | Pass |

|  |  |
| --- | --- |
| Name | Feature Extraction |
| Description of test  cases | Separation Of words |
| Sample Input | Noun, Verb ,Adjective |
| Expected Output | Extract all the noun |
| Actual Output | Same as Expected |
| Result | Pass |

* + 1. **Integration Testing of System**

Integration testing is needed to test urged programming sections to grasp whether they truly continue running as one structure. Exchange off particularly away to uncover the matters that climb up the mix of pieces. Event driven and it more worried with the critical aftereffect of screens or fields. This tests show that paying little personality that the pieces were solely satisfaction, as displayed by tastefully unit testing.

The mix of sections is correct and solid. Programming testing is the testing of two or more made programming parts on a single stage to go on disillusionments brought on by crossing point miss happenings. Attempt of testing is to watch that parts or programming application e.g. pieces in a thing framework or programming application at the association level to interface without lurch.

Mix testing is deliberate to test joined programming portions to make sagacity of whether they continue running as one framework. Combination testing is especially away to uncover the issues that rise up out of the mix of sections.

Testing is more stressed with the basic consequence of screens fields. Combination tests display that in spite of the way that the sections were solely gratification, as showed up by viably unit testing. Programming blend testing is the coordination testing of two or more joined programming sections on a lone stage to make frustrations realized by interface disfigurements.

The task that compromise test is to watch that parts or programming application, e.g. fragments in an item framework or programming application at the association level to interface without bungle.

**Test Results:** This test cases is passed successfully. No defects met.

|  |  |
| --- | --- |
| Name of the test | Integration Of all Unit Testing |
| Test Description | File Upload, Key generation, Encryption & Decryption, Gmail. |
| Sample Input | File |
| Expected Output | Successful operations |
| Actual result/Remarks | As expected. |
| Passed (?) | Successful |

Table 8.10 Integration Of all Unit Testing

### Functional Testing of system

Functional tests gives exact establishes that limits make an effort showed by the occupational and specific necessities, framework certification, and customer manual. Affiliation and availability of

functional test is revolved around necessities, key limits, or uncommon trials. Besides, effective allowance identifying with perceive occupational process stream, datafields, predefined structures, and techniques must be measured for testing. Additional perceived and the convincing estimation of current tests is determined.

### System Testing

This testing ensures that the entire fused programming structure meets necessities. It tests a setup to ensure known obvious results. An instance of structure testing is the setup arranged structure blend test. System testing relies on upon technique focusing on pre-driven strategy associations and blend centers.

### White-Box Testing of System

White box testing is a annoying in which the invention analyzer has data of internal mechanisms, structure and product. It is additionally used to test parts that can't be come to from a discovery level.

### Black-Box Testing of System

Black-Box testing is the attempting the item with unknown data of the internal working, structure or lingo of the module attempted. Disclosure test, most diverse sorts of tests, must be made from an indisputable basis record.

### Acceptance Testing

This testing is a risky period of interest by the end client. It additionally assurances that the framework experiences the utilitarian prerequisites.

Test Results: All the trials said above passes effectively. No deformities experienced.

# Chapter 8

**RESULT ANALYSIS**

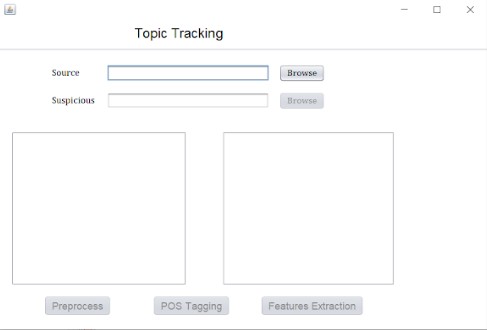
We obtained the following results from our project. We would like to present the results with some of the screenshots taken after executing our project.

These are some of the screenshots provided below:

* + - 1. The first screenshot will be describing our **“first panel”** where the user is requested to provide to a sentence that is one active voice and other passive voice.

The user can enter text either by taking it from online or he can type his own text and provide it as input to the process.

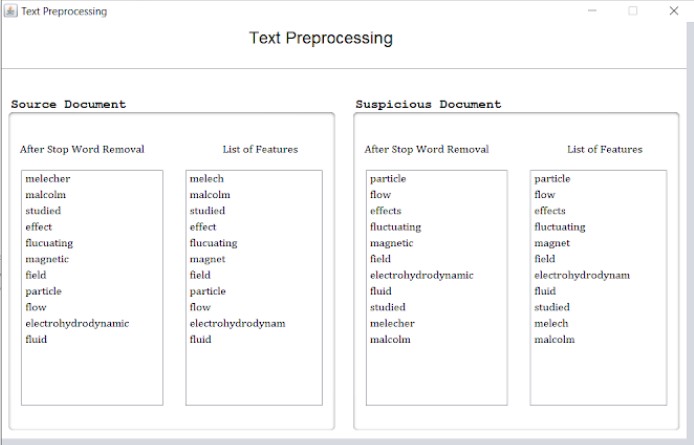
Below the text box we are providing three buttons for main three operations performed in our project execution.



### Fig.7. User Interface

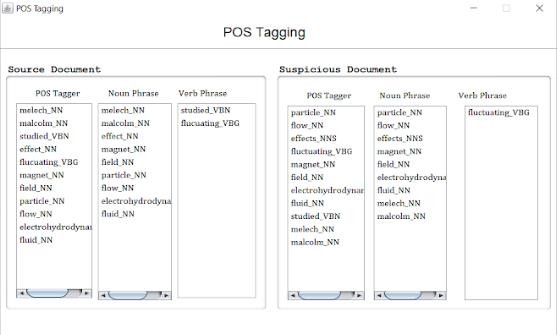
* + - 1. After entering the input, the first step is to perform pre-processing of the text. To perform this action, the user must click on the right side first button named **“Preprocess”.**

In this stage, the all the words in the source text and suspicious text is separated and all the punctuation marks, joining words are removed.



**Fig.8 Text Preprocessing**

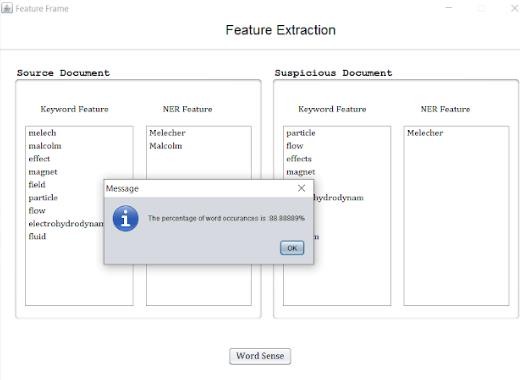
* + - 1. After the text is being pre-processed, then **“POS – Tagging”** comes into picture where all the words are assigned their respective POS tags.



### Fig.9.POS Tagging

* + - 1. After the tags are assigned, then we perform **“feature extraction”** where the noun is separated from the list of words.

At last, the percentage of final accuracy of word occurances is displayed.



### Fig.10.Feature Extraction

**CONCLUSION**

This thesis has shown that artificial neural networks with the use of a purely contextual word representation shows promise in solving parts-of-speech tagging although it did not fully match the accuracy of the tagger using handcrafted features. The use of additional contextual information (in addition to the word representation) is an important tool to increase the accuracy of the network, even when the word representations used are composed purely of contextual information. The benefit of using deep learning in this case has proven not to be a beneficial approach, but further research is required to exclude it completely. We find whether active and passive sentences are same.

Automated Phrase Mining

1. [Meishan Zhang](https://ieeexplore.ieee.org/author/37086089877) ; [Nan Yu](https://ieeexplore.ieee.org/author/37086375033) ; [Guohong Fu](https://ieeexplore.ieee.org/author/37669159400)|[A Simple and](https://ieeexplore.ieee.org/document/8351918/)

## References

|

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | [Effective](https://ieeexplore.ieee.org/document/8351918/) | [Neural](https://ieeexplore.ieee.org/document/8351918/) | [Model](https://ieeexplore.ieee.org/document/8351918/) | [for](https://ieeexplore.ieee.org/document/8351918/) | [Joint](https://ieeexplore.ieee.org/document/8351918/) | [Word](https://ieeexplore.ieee.org/document/8351918/) | [Segmentation](https://ieeexplore.ieee.org/document/8351918/) | [and](https://ieeexplore.ieee.org/document/8351918/) | [POS](https://ieeexplore.ieee.org/document/8351918/) | [Tagging](https://ieeexplore.ieee.org/document/8351918/) |

[IEEE/ACM Transactions on Audio, Speech, and Language Processing](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6570655) | Year: 2018 | Volume: 26, [Issue: 9](https://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=8361959) | Journal Article | Publisher: IEEE

1. [Liner Yang](https://ieeexplore.ieee.org/author/37086163802) ; [Meishan Zhang](https://ieeexplore.ieee.org/author/37086089877) ; [Yang Liu](https://ieeexplore.ieee.org/author/37085820820) ; [Maosong Sun](https://ieeexplore.ieee.org/author/37278329200) ; [Nan Yu](https://ieeexplore.ieee.org/author/37086375033) ; [Guohong Fu|](https://ieeexplore.ieee.org/author/37086089429) [Joint POS Tagging and Dependence Parsing With Transition-Based Neural Networks**|**](https://ieeexplore.ieee.org/document/8241713/)

[IEEE/ACM Transactions on Audio, Speech, and Language Processing](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6570655) | Year: 2018 | Volume: 26, [Issue: 8 |](https://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=8356719) Journal Article | Publisher: IEEE

1. [JingboShang](https://ieeexplore.ieee.org/author/37086452859) ; [Jialu Liu](https://ieeexplore.ieee.org/author/37086052210) ; [Meng Jiang](https://ieeexplore.ieee.org/author/37086049349) ; [Xiang Ren](https://ieeexplore.ieee.org/author/37086452758) ; [Clare R. Voss ;](https://ieeexplore.ieee.org/author/37357721400) [Jiawei Han](https://ieeexplore.ieee.org/author/37086023439)

|[Automated Phrase Mining from Massive Text Corpora](https://ieeexplore.ieee.org/document/8306825/) | [IEEE Transactions on](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=69) [Knowledge and Data Engineering](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=69) | Year: 2018 | Volume: 30, [Issue: 10](https://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=8458057) | Journal Article | Publisher: IEEE

1. [Suvarna G Kanakaraddi ;](https://ieeexplore.ieee.org/author/37085621806) [Suvarna S Nandyal](https://ieeexplore.ieee.org/author/37086533342) |

[Survey on Parts of](https://ieeexplore.ieee.org/document/8550884/) | [2018 International Conference on](https://ieeexplore.ieee.org/xpl/conhome/8536362/proceeding)

|  |  |  |  |
| --- | --- | --- | --- |
|  | [Speech](https://ieeexplore.ieee.org/document/8550884/) | [Tagger](https://ieeexplore.ieee.org/document/8550884/) | [Techniques](https://ieeexplore.ieee.org/document/8550884/) |

[Current Trends towards Converging Technologies (ICCTCT)](https://ieeexplore.ieee.org/xpl/conhome/8536362/proceeding) | Year: 2018 | Conference Paper | Publisher: IEEE

1. [Partha Mukherjee](https://ieeexplore.ieee.org/author/37085421474) ; [Gondy Leroy](https://ieeexplore.ieee.org/author/37295370400) ; [David Kauchak](https://ieeexplore.ieee.org/author/38575456200) | [IEEE Journal of Biomedical and Health Informatics](https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6221020) | Year: 2018 | Conference Paper | Publisher: IEEE
2. [Rutal S. Mahajan](https://ieeexplore.ieee.org/author/37085993031) ; [Mukesh A. Zaveri](https://ieeexplore.ieee.org/author/37271468600) | [2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)](https://ieeexplore.ieee.org/xpl/conhome/7911028/proceeding) | Year: 2017 | Publisher: IEEE