

|  |  |  |
| --- | --- | --- |
| **Natural Language Processing** | | |
| Lab Manual | | |
| **Department of Computer Science and Engineering**  **The NorthCap University, Gurugram** | | |
|  | |
|  | |
|  | |
|  | |
|  |  | | |

**Natural Language Processing**

**Laboratory Manual**

**CSL448**



Department of Computer Science and Engineering

The NorthCap University, Gurugram- 122001, India

Session 2024-25

*Published by:*

**Department of Computer Science & Engineering**

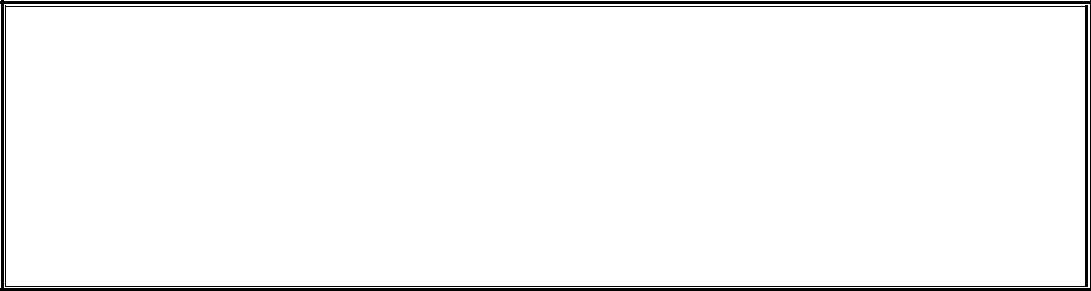
**School of Engineering and Technology**

**The NorthCap University Gurugram**

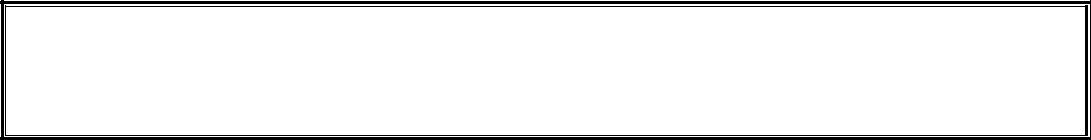
**• Laboratory Manual is for Internal Circulation only**

**© Authors**

*No part of this Practical Record Book may be reproduced, used, stored without prior permission of The NorthCap University*



Copying or facilitating copying of lab work comes under cheating and is considered as use of unfair means. Students indulging in copying or facilitating copying shall be awarded zero marks for that particular experiment. Frequent cases of copying may lead to disciplinary action. Attendance in lab classes is mandatory.



Labs are open up to 7 PM upon request. Students are encouraged to make full use of labs beyond normal lab hours.

**PREFACE**

Applied Computational Statistics Laboratory Manual is designed to meet the course and program requirements of NCU curriculum for B.Tech. eigth semester students of CSE branch. The concept of the lab work is to give brief practical experience for basic lab skills to students. It provides the space and scope for self-study so that students can come up with new and creative ideas.

The Lab manual is written on the basis of “teach yourself pattern” and expected that students who come with proper preparation should be able to perform the experiments without any difficulty. A brief introduction to each experiment with information about self-study material is provided. The laboratory exercises will help students to provide a hands-on each exercise that will help them to understand thoroughly. The students are expected to come thoroughly prepared for the lab. General disciplines, safety guidelines and report writing are also discussed.

The lab manual is a part of curriculum for the The NorthCap University, Gurugram. Teacher’s copy of the experimental results and answer for the questions are available as sample guidelines.

We hope that lab manual would be useful to students of CSE branch and author requests the readers to kindly forward their suggestions / constructive criticism for further improvement of the work book.

Author expresses deep gratitude to Members, Governing Body-NCU for encouragement and motivation.

**Authors**

**The NorthCap University**

**Gurugram, India**

|  |  |  |  |
| --- | --- | --- | --- |
| **CONTENTS** | | | |
| S.No. | **Details** | Page No. | |
|  |  |  | |
| 1 | **Introduction** | vi | |
| 2 | **Lab Requirement** | vii |
| 3 | **General Instructions** | viii |
| 4 | **List of Experiments** | x |
| 5 | **List of Flip Experiments** | xi |
| 6 | **List of Projects** | xii |
| 7 | **Rubrics** | xiv |
| 8 | **Annexure 1 (Format of Lab Report)** | xv |
| 9 | **Annexure 2 (Format of Project Report)** | xlviii |

1. **INTRODUCTION**



That ‘learning is a continuous process’ cannot be over emphasized. The theoretical knowledge gained during lecture sessions need to be strengthened through practical experimentation. Thus, practical makes an integral part of a learning process. ­­­­­­­­­­­­­­­­­­­­­

**COURSE OBJECTIVES:**

1. **Outline** linguistic phenomena, the field of natural language processing and the capabilities and limitations of current natural language technologies.
2. **Understand** language models and **apply** simple pre-processing tasks imperative for a computer to understand natural language.
3. **Apply** natural language processing tools to **analyze** the data through basic natural language processing tasks such as syntactic parsing, dependency parsing.
4. **Demonstrate** computational skills to **create** NLP pipelines using existing NLP libraries, retrain models and extend existing NLP tools
5. **Apply** and **evaluate** the use of existing statistical and deep learning techniques in language applications such as machine translation.
6. **LAB REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Requirements** | **Details** |
| **1** | **Software Requirements** | Python 3.x, Numpy, NLTK, Spacy, Pandas, Matplotlib, Seaborn, statistics, sci-kit learn |
| **2** | **Operating System** | Windows 7 onwards or Linux (32 or 64 bit) |
| **3** | **Hardware Requirements** | 4 GB RAM (Recommended)  2.60 GHz (Recommended) |
| **4** | **Required Bandwidth** | NA |

1. **GENERAL INSTRUCTIONS** 
   1. **General discipline in the lab**
   * Students must turn up in time and contact concerned faculty for the experiment they are supposed to perform.
   * Students will not be allowed to enter late in the lab.
   * Students will not leave the class till the period is over.
   * Students should come prepared for their experiment.
   * Experimental results should be entered in the lab report format and certified/signed by concerned faculty/ lab Instructor.
   * Students must get the connection of the hardware setup verified before switching on the power supply.
   * Students should maintain silence while performing the experiments. If any necessity arises for discussion amongst them, they should discuss with a very low pitch without disturbing the adjacent groups.
   * Violating the above code of conduct may attract disciplinary action.
   * Damaging lab equipment or removing any component from the lab may invite penalties and strict disciplinary action.
   1. **Attendance**

* Attendance in the lab class is compulsory.
* Students should not attend a different lab group/section other than the one assigned at the beginning of the session.
* On account of illness or some family problems, if a student misses his/her lab classes, he/she may be assigned a different group to make up the losses in consultation with the concerned faculty / lab instructor. Or he/she may work in the lab during spare/extra hours to complete the experiment. No attendance will be granted for such case**.**
  1. **Preparation and Performance**
* Students should come to the lab thoroughly prepared on the experiments they are assigned to perform on that day. Brief introduction to each experiment with information about self -study reference is provided on LMS.
* Students must bring the lab report during each practical class with written records of the last experiments performed complete in all respect.
* Each student is required to write a complete report of the experiment he has performed and bring to lab class for evaluation in the next working lab. Sufficient space in work book is provided for independent writing of theory, observation, calculation and conclusion.
* Students should follow the Zero tolerance policy for copying / plagiarism. Zero marks will be awarded if found copied. If caught further, it will lead to disciplinary action.
* Refer **Annexure 1** for Lab Report Format

1. **LIST OF EXPERIMENTS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Title of the Experiment** | **Software  based** | **Unit covered** | **Time Required** |
| 1. | Explore Python NLTK and all the text corpora available in NLTK. | Python (Jupyter) | 1 | 2 hours |
|  | Find the most frequently occurring five words from a piece of text | Python (Jupyter) | 1 | 2 hours |
|  | Write a program to list all instances of the word ‘the’ in the Brown corpus using regular expressions. | Python (Jupyter) | 2 | 2 hours |
|  | Write a program to print all the bigrams and trigrams from a given corpus and sorting them by frequency. | Python (Jupyter) | 2 | 2 hours |
|  | Write a program to perform add-one smoothing on a sparse bigram table. | Python (Jupyter) | 2 | 2 hours |
|  | Write a program to read text data from a file and perform tokenization on sentence and word level using diferent tokenizers in Python NLTK. | Python (Jupyter) | 2 | 2 hours |
|  | Write a program to read text data from a file and perform Part of Speech Tagging, stop-word removal, stemming (using different stemmers), lemmatization and identify the difference between stemming and lemmatization. | Python (Jupyter) | 2 | 2 hours |
|  | Write a program to find the unusual words in a given language. | Python (Jupyter) | 2 | 2 hours |
|  | Write a program to read text data from a file and perform chunking and chinking. | Python (Jupyter) | 3 | 2 hours |
|  | Write a program to read text data from a file and implement Named Entity Recognition. | Python (Jupyter) | 3 | 2 hours |
|  | Write a program to compute the similarity between any two sentences of text using different metrics for analyzing textual similarity. | Python (Jupyter) | 3 | 2 hours |
|  | Write a program to read text data from a file and perform pre-processing, Word Sense Disambiguation and list of synonyms, antonyms, hypernyms and hyponyms of every word as obtained from the lexical ontology WordNet. | Python (Jupyter) | 3 | 2 hours |
|  | Write a Python program to implement Co-Reference resolution. | Python (Jupyter) | 3 | 2 hours |
|  | Write a program to perform text classification in Python using TF-IDF. | Python (Jupyter) | 3 | 2 hours |
|  | Write a program to perform Sentiment Analysis of hotel reviews using SentiWordNet. | Python (Jupyter) | 3 | 2 hours |
|  | Develop a simple LSTM network to learn sequences of characters from *Alice in Wonderland*. Then, use this model to generate new sequences of characters. | Python (Jupyter) | 4 | 4 hours |
|  | Write a program to perform Sentiment Analysis using BERT. | Python (Jupyter) | 4 | 4 hours |
|  | Dialogue summarization using generative AI: perform prompt engineering and compare zero shot, one shot, and few shot inferences and compare the output. | Python (Jupyter) | 5 | 4 hours |
| Value Added Experiments | | | | |
|  | Project – Name Gender Identifier | Python(Jupyter) | 1,2,3,4,5 | 4 hrs |
|  | Project – Estimating Similarity between two text documents | Python(Jupyter) | 1,2,3,4,5 | 4 hrs |
|  | Project – Sentiment Analysis of Tweets using ML algorithms | Python(Jupyter) | 1,2,3,4,5 | 4 hrs |
|  | Project- Building a text generation model. | Python(Jupyter) | 1,2,3,4,5 | 6 hrs |
|  | Project- Implementing a generative AI application in a creative domain. | Python(Jupyter) | 1,2,3,4,5 | 6 hrs |

1. **LIST OF FLIP EXPERIMENTS**

|  |  |  |
| --- | --- | --- |
| **Exp. No.** | **Title of the Experiment** | **Mapped CO** |
|  | Estimate similarity between two text documents | CO 1, 2,3,4,5,6 |
|  | Explore and use BERT and BART algorithms for any NLP application | CO1,2,3,4,5,6 |

1. **LIST OF PROJECTS**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Project Title** | **Mapped CO** |
|  | Name Gender Identifier | CO 1,2,3,4,5,6 |
|  | Implementing and working with Large Language Models | CO1,2,3,4,5,6 |
|  | Sentiment Analysis of Tweets using ML algorithms | CO1,2,3,4,5,6 |

1. **RUBRICS (Only for Lab components)**

|  |  |
| --- | --- |
| **Marks Distribution** | |
| **Continuous Evaluation (25 Marks)** | **Project Evaluations (20 Marks)** |
| Each experiment shall be evaluated for 5 marks and at the end of the semester proportional marks shall be awarded out of total 25. | Project shall be evaluated for 20 marks and at the end of the semester viva will be conducted related to the project. |
| **Viva and Reporting (25 Marks)**  Following is the breakup of 25 marks for each  **10 Marks**: Observation & conduct of experiment. Teacher may ask questions about experiment in mid-term viva.  **10 Marks:** Observation & conduct of experiment.  **5 Marks:** For report writing |

**Annexure 1**

**INDEX**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S.No** | **Experiment Title** | **Date**  **of Experiment** | **Date of Submission** | **Sign student** | **CO Covered** | **Sign Faculty** |
| **1** |  |  |  |  |  |  |
| **2** |  |  |  |  |  |  |
| **3** |  |  |  |  |  |  |
| **4** |  |  |  |  |  |  |
| **5** |  |  |  |  |  |  |
| **6** |  |  |  |  |  |  |
| **7** |  |  |  |  |  |  |
| **8** |  |  |  |  |  |  |
| **9** |  |  |  |  |  |  |
| **10** |  |  |  |  |  |  |
| **11** |  |  |  |  |  |  |
| **12** |  |  |  |  |  |  |
| **13** |  |  |  |  |  |  |
| **14** |  |  |  |  |  |  |
| **15** |  |  |  |  |  |  |
| **16** |  |  |  |  |  |  |
| **17** |  |  |  |  |  |  |
| **18** |  |  |  |  |  |  |

**EXPERIMENT NO. 1**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** Explore Python NLTK and all the text corpora available in NLTK. |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshots**

**EXPERIMENT NO. 2**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section: |** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s): Write a program to list all instances of the word ‘the’ in the Brown corpus using regular expression** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshots**

**EXPERIMENT NO. 3**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section: |** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s): Write a program to print all the bigrams and trigrams from a given corpus and sorting them by frequency** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshots**

**EXPERIMENT NO. 4**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section: |** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s): Write a program to perform add-one smoothing on a sparse bigram table** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshots**

**EXPERIMENT NO. 5**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section: |** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s): Write a program to read text data from a file and perform tokenization on sentences and word level using different tokenizers in python NLTK** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshot**

**EXPERIMENT NO. 6**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section: |** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s): Write a program to read text data from a file and perform Part of Speech Tagging, stop-word removal, stemming (using different stemmers), lemmatization and identify the difference between stemming and lemmatization.** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshots**

**EXPERIMENT NO. 7**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section: |** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s): Write a program to find the unusual words in a given language.** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**Screenshot**

**EXPERIMENT NO. 8**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 9**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 10**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 11**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 12**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 13**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 14**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 15**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objectives(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 16**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 17**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**

**EXPERIMENT NO. 18**

|  |
| --- |
| **Student Name and Roll Number:** |
| **Semester /Section:** |
| **Link to Code:** |
| **Date:** |
| **Faculty Signature:** |
| **Marks/Grade:** |
| **Objective(s):** |

**Student Work Area**

**Algorithm/Flowchart/Code/Sample Outputs**