|  |  |
| --- | --- |
| **VAC NLP PROJECT** | |
| **COURSE NAME:** | **Basics of NLP** |
| **COURSE CODE:** | **CSE V 077** |
| **SESSION AND SEM:** | **5 CSE 12** |
| **OUTCOMES FROM PROJECT:** | **The research goal of this work is to address Sentiment Analysis by constructing a classifier that can categorize positive and negative movie reviews.** |
| **BUDDING RESEARCHER NAME:** | **Vinay Lakshman M** |
| **BUDDING RESEARCHER ID:** | **20181CSE0791** |

|  |  |
| --- | --- |
| **SUBMITTED TO** | |
| **SUPERVISOR:** | **Prof. Mitali Halder** |
| **SIGNATURE:** |  |

|  |
| --- |
| **SYNOPSIS OF VAC NLP PROJECT** |

|  |
| --- |
| **PROJECT TITLE:** |
| *Sentiment analysis for Movie Review Data with Natural Language Processing.* |

|  |
| --- |
| **PROBLEM STATEMENT:** |
|  |

|  |
| --- |
| **METHODOLOGY:** |
| 1. **DATA SET AQUISTION** |
| It is one of the most important steps in designing a classifier. After lots of research, we eventually selected the “IMDB Dataset” for experimentation. The dataset contains a lot of reviews, both positive and negative, as the website is pretty famous for movie reviews and ratings. The better the dataset, the more accurate and efficient the classifier performs. |
| 1. **SPLITTING TRAIN AND TEST SETS** |
| Planning to divide the total dataset into 80% training data and 20% validation data. The classifier will be tested based on comparative study of review analysis, as in how relevant the predicted review sentiments are to the actual sentiments. |
| 1. **TEXT TOKENIZATION** |
| Text tokenization is segmenting text into sentences and words by specifying the basic linguistic units; words,  numbers and punctuations. In English language words are usually separated by white spaces. |
| 1. **DATA PRE- PROCESSING** |
| This step involves cleaning the dataset such as removing unwanted characters (- or – or # or $ ), html tags or stopwords that don’t add much value, and replace them with blank spaces. |
| 1. **CREATING BAG OF WORDS AND TDF/IF MODELS** |
| **Bag of Words:** This method analyzes the entire corpus of text, builds a dictionary of all words, and translates every instance in the dataset into a list of numbers, counting how many times each word appears in the document.  **TDF/IF models:** TDF-IFnormalizes the word counts based on the frequency of how often each word appears across the documents. The main idea is that common words get smaller weighting factors, and relatively rare words get larger weighting factors, which enables you to dig deeper into the (often highly informative) words that appear less often in the dataset. |
| 1. **CLASSIFIERS (LOGISTIC REGRESSION AND NAIVE BAYES)** |
| **Naïve Bayes (NB)**: This classifier has two probabilities: P(class) which is the probability an input will produce a certain class, and P(input\_condition|class) is the probability an input feature has a certain value, given the class. Otherwise, default probability is 0.  **Logistic regression:** logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables. |
| 1. **CLASSIFICATION** |
| Finally, the classifier can classify the reviews present in the dataset and the same would be plotted for better visual representation. |

|  |
| --- |
| **PYTHON PACKAGES** |
| The code will be compatible with Python 3.5 and above.   * Numpy (generally Pre installed) * Nltk (generally Pre installed) * Matplotlib (to plot the results) (generally Pre installed) * WordCloud (to plot results) * Sklearn (generally Pre installed) * Bs4 (use BeautifulSoup to remove html tags and parsers) * pandas (generally Pre installed) |

|  |
| --- |
| **SOFTWARE REQUIREMENTS:** |
| * Language Used: Python * Web Browser: Google Chrome, Mozilla, IE8, Microsoft Edge. |

|  |
| --- |
| **HARDWARE REQUIREMENTS:** |
| * Processor (CPU) with 2 gigahertz frequency or above. * Minimum of 2 GB of RAM. * A minimum of 10 GB of available space on the hard disk. |

|  |
| --- |
| **DATASET:** |
| IMDB Dataset |

|  |
| --- |
| **Project Plan with Timeline** |
| The project plan is briefly divided into 4 milestones.   1. Data Pre Processing Phase (upto 5th November 2020) 2. Coding and Training phase (Upto 12th November 2020) 3. Testing (15th November 2020) 4. Document with Coding Submission (20th November 2020) |

|  |
| --- |
| **EXPECTED OUTCOME(S):** |
| This Project aims to achieve the following outcomes:   * to address Sentiment Analysis by constructing an approach that can classify movie reviews. * Separate positive instances from negative instances in the reviews. * Compare two classifier models and two word-filtering models and pick the combination that performs the best. |

|  |  |
| --- | --- |
| SUBMITTED BY: | |
| SCHOLAR NAME: | Vinay Lakshman M |
| SCHOLAR ID: | 20181CSE0791 |
| SIGNATURE: | Vinay |
| DATE OF SUBMISSION: | 22-10-2020 |