Practical Tasks List

No	Date	Practical Title	Sign
1		a. Install NLTK. b. Convert the given text to speech. c. Convert audio file Speech to Text.	
2		a. Study of various Corpus - Brown, Inaugural, Reuters, udhr with various methods like fields, raw, words, sents, categories. b. Create and use your own corpora (plaintext, categorical). c. Study Conditional frequency distributions. d. Study of tagged corpora with methods like tagged_sents, tagged_words and write a program to find the most frequent noun tags. e. Map Words to Properties Using Python Dictionaries. f. Study DefaultTagger, Regular expression tagger, UnigramTagger.	
3		a. Study of Wordnet Dictionary with methods as synsets, definitions, examples, antonyms. b. Study lemmas, hyponyms, hypernyms, entailments. c. Write a program using python to find synonym and antonym of word 'active' using Wordnet. d. Compare two nouns. e. Handling stopwords. - Using nltk to add or remove stopwords in NLTK's default stopword list. - Using Gensim to add or remove stopwords in the default Gensim stopword list. - Using Spacy to add or remove stopwords in the default Spacy stopword list.	
4		Text Tokenization a. Tokenization using Python's split() function. b. Tokenization using Regular Expressions (RegEx).	

		c. Tokenization using NLTK. d. Tokenization using the spaCy library. e. Tokenization using Keras. f. Tokenization using Gensim.
5		Illustrate part of speech tagging. a. Part of speech tagging and chunking of user-defined text. b. Named Entity recognition of user-defined text. c. Named Entity recognition with diagram using NLTK corpus - treebank.
6		a. Define grammar using nltk. Analyze a sentence using the same. b. Accept the input string with Regular expression of FA: 101+. c. Accept the input string with Regular expression of FA: (a+b)*bba.
		d. Implementation of Deductive Chart Parsing using context free grammar and a given sentence.
7		a. Study PorterStemmer, LancasterStemmer, RegexpStemmer, SnowballStemmer. b. Study WordNetLemmatizer.
8		a. Implement Naive Bayes classifier.