Part-Of-Speech Tagging for Gujarati Language

Group Members:
Dharmeshgiri 2019201025
Smitkumar 2019201021

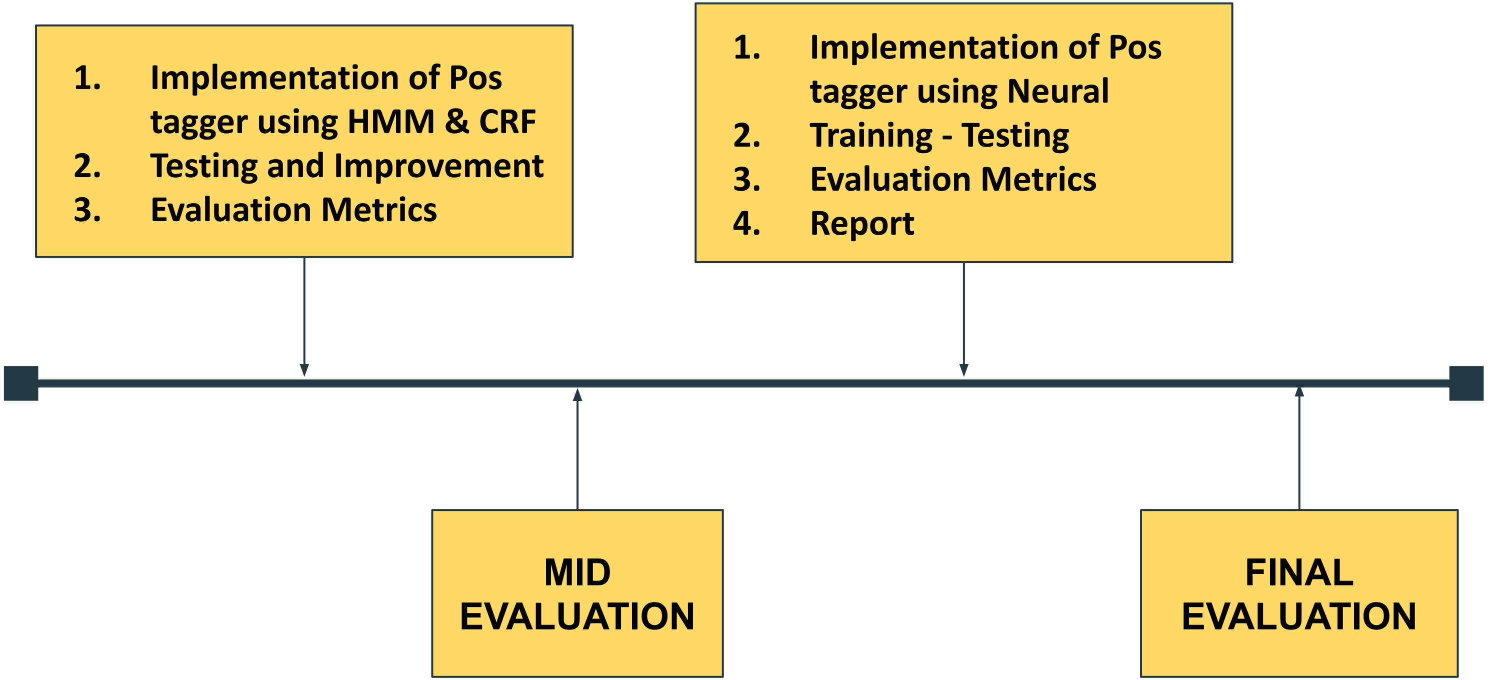
Project Mentor:
Ujwal Narayan

Project Definition

- POS Tagging is a process that attaches each word in a sentence with a suitable tag from a given set of tags..
- Given table contains example of different tag and it’s meaning in part of speech tagging.

Tag	Meaning	English Examples
ADJ	adjective	*new, good, high, special, big, local*
ADP	adposition	*on, of, at, with, by, into, under*
ADV	adverb	*really, already, still, early, now*
CONJ	conjunction	*and, or, but, if, while, although*
DET	determiner, article	*the, a, some, most, every, no, which*
NOUN	noun	*year, home, costs, time, Africa*
NUM	numeral	*twenty-four, fourth, 1991, 14:24*
PRT	particle	*at, on, out, over per, that, up, with*
PRON	pronoun	*he, their, her, its, my, I, us*
VERB	verb	*is, say, told, given, playing, would*
.	punctuation marks	*. , ; !*
X	other	*ersatz, esprit, dunno, gr8, univeristy*

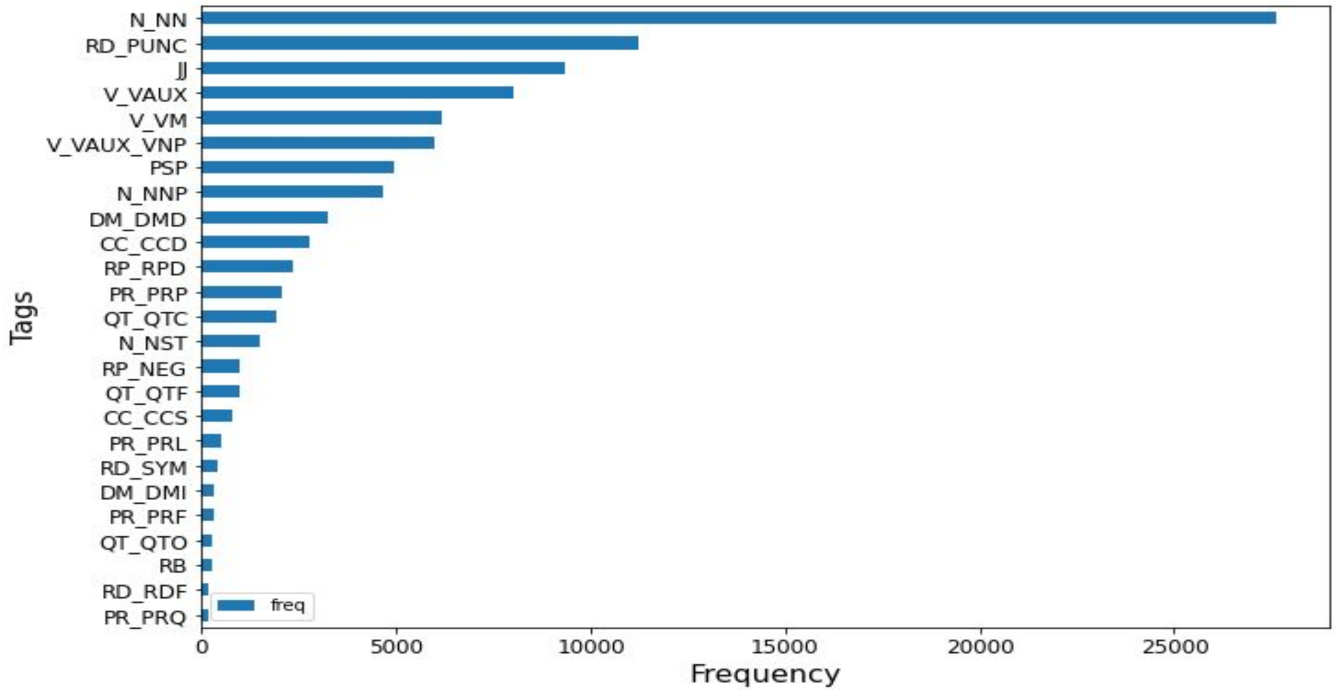
Project Timeline



Dataset Details

Index	Dataset	Size (No. of sentences)	Size (No. of words)	Train (No. of sentences)	Test (No. of sentences)
D1	guj_art and culture_sample1.txt	1000	16050	800	200
D2	guj_economy_sample2.txt	1000	12467	800	200
D3	guj_entertainment_sample3.txt	1000	11879	800	200
D4	guj_philosophy_sample4.txt	1000	12904	800	200
D5	guj_religion_sample5.txt	1000	12247	800	200
D6	guj_science and technology_sample6.txt	1000	16857	800	200
D7	guj_sports_sample7.txt	1000	14836	800	200

Dataset Details



Research Papers

1. Hindi POS Tagger Using Naive Stemming : Harnessing Morphological Information Without Extensive Linguistic Knowledge
   a. https://www.cse.iitb.ac.in/~pb/papers/icon08-hindi-pos-tagger
   b. In this paper, They present a simple HMM based POS tagger, which employs a naive (longest suffix matching) stemmer as a pre-processor to achieve reasonably good accuracy of 93.12%. This method does not require any linguistic resource apart from a list of possible suffixes for the language.

Research Papers

2. POS Tagging For Resource Poor Indian Languages Through Feature Projection
   a. https://www.researchgate.net/publication/323174678_POS_Tagging_For_Resource_Poor_Indian_Languages_Through_Feature_Projection
   b. They used feature transfer from a resource rich language to resource poor languages. Across 8 different Indian Languages, they achieved encouraging accuracies without any knowledge of the target language and any human annotation For Indian Languages, they considered the following morph features The prefix characters up to 7 characters, The suffix characters up to 4 characters, Length of the word, Context Window size of 3 (Previous word,Current word and Next word)

Research Papers

3. Character-level Supervision for Low-resource POS Tagging
   a. https://pdfs.semanticscholar.org/a93a/3799ba977aa2393cad8cd260d6f778a495e0.pdf?_ga=2.249332687.936938462.1614443610-381175129.1614443610
   b. In this paper experiment with three auxiliary tasks: lemmatization, character-based word autoencoding, and character-based random string autoencoding. They have used bidirectional LSTM.

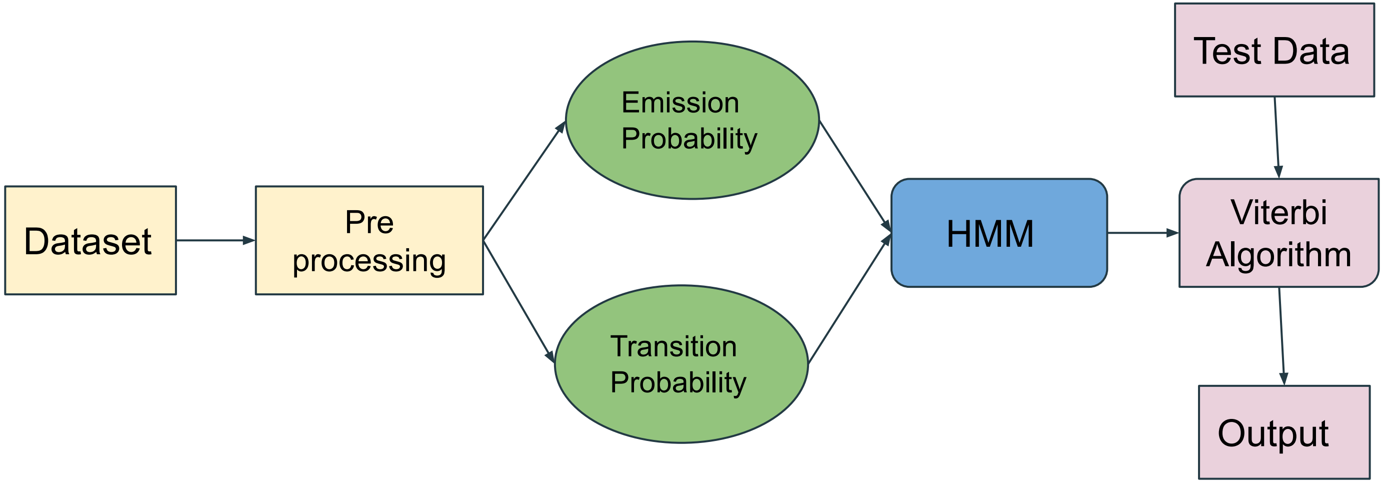
Research Papers

4. Part-Of-Speech Tagging for Gujarati Using Conditional Random Fields
   a. https://www.aclweb.org/anthology/I08-3019.pdf
   b. They used dataset where where the training corpus is of 10,000 words and the test corpus is of 5,000 words in Gujarati. The algorithm has achieved an accuracy of 92%. . The machine learning part is performed using a CRF model.

Baseline : Hidden Markov Model (HMM)

- **HMM** (**Hidden Markov Model**) is a Stochastic technique for **POS tagging**.
- HMMs are a standard generative probabilistic model for sequence labeling that allows for efficiently computing the globally most probable sequence of labels and supports supervised, unsupervised and semi-supervised learning.
- HMM approach was used for this task since it does not need detail linguistic knowledge of the language as rule based approach.
- Hidden Markov models are known for their applications to reinforcement learning and temporal pattern recognition such as speech, handwriting, gesture recognition, musical score following, partial discharges, and bioinformatics.

Baseline : Hidden Markov Model (HMM)



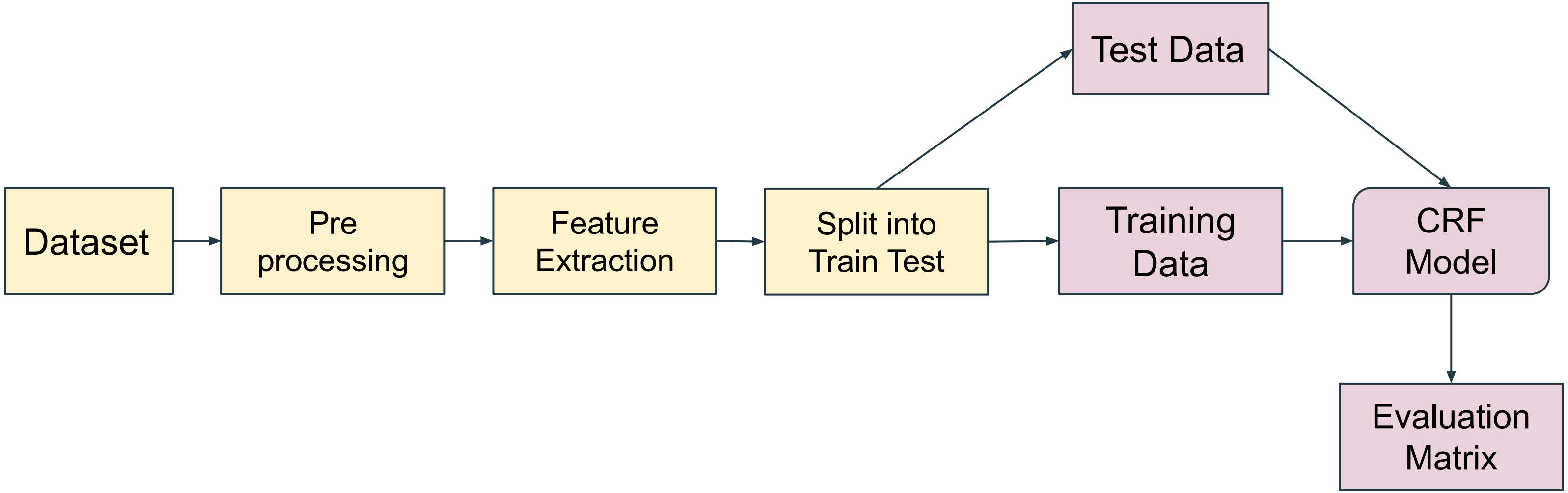
Baseline : Conditional Random Field (CRF)

- A **Conditional Random Field (CRF)** is a sequence modeling algorithm which is used to identify entities or patterns in text, such as POS tags.
- This model not only assumes that features are dependent on each other, but also considers future observations while learning a pattern.
- Since these models take into account previous data, we use features which are modelled from the data to feed into the CRF.
- CRF is described as:

$$P_w(y|x) = \frac{1}{Z_w(x)} exp\left(\sum_{j=1}^{n} \sum_{i=1}^{m} w_i f_i(y_{j-1}, y_j, x, j)\right)$$

where $Z_w(x) = \sum_{y \in Y} exp\left(\sum_{j=1}^{n} \sum_{i=1}^{m} w_i f_i(y_{j-1}, y_j, x, j)\right)$

Baseline : Conditional Random Field (CRF)



Feature Selection

- ‘word’ : Word
- 'is_first' : Is it first word of sentence ? (True / False)
- ‘is_last’ : Is it last word of sentence ? (True / False)
- ‘prefix-1’ : Prefix of word of sizes - 1
- ‘prefix-2’ : Prefix of word of sizes - 2
- ‘prefix-3’ : Prefix of word of sizes - 3
- ‘suffix-1’ : Suffix of word of sizes - 1

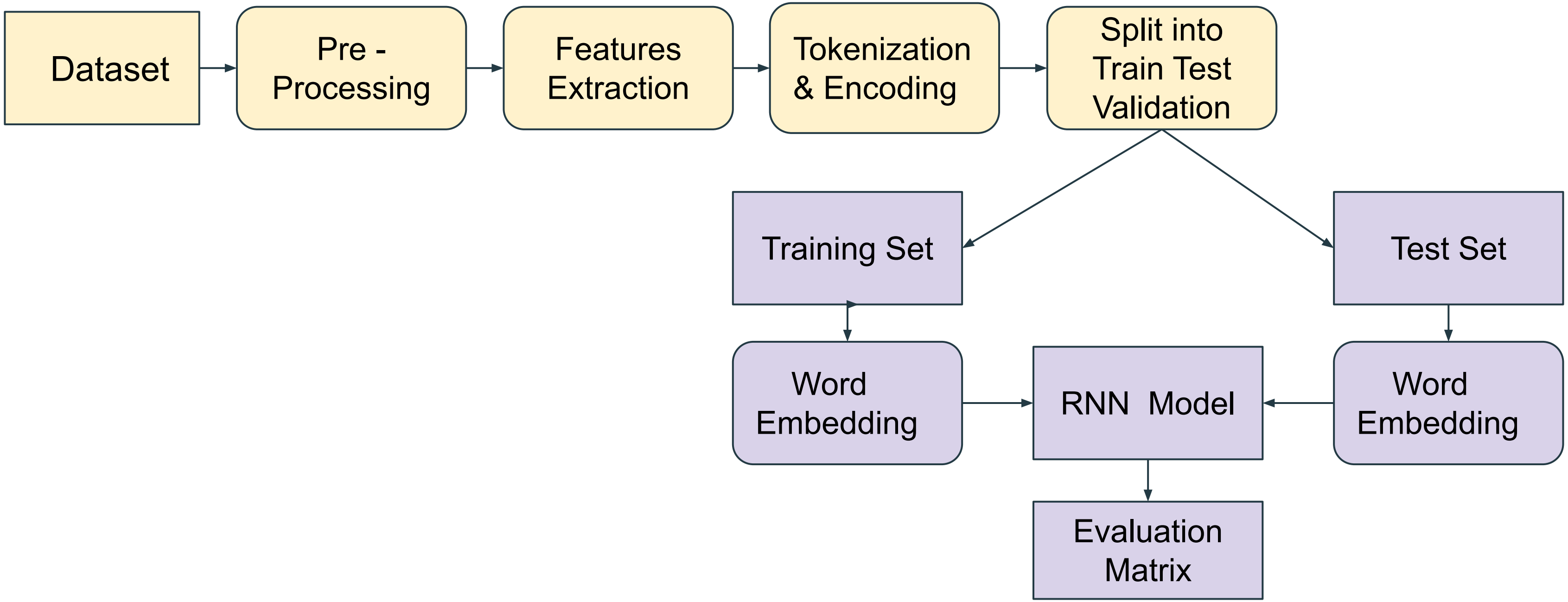
Feature Selection

- ‘suffix-2’ : Suffix of word of sizes - 2
- ‘suffix-3’ : Suffix of word of sizes - 3
- ‘prev_word’ : Previous word
- ‘pprev_word’ : Previous of previous word
- ‘next_word’ : Next word
- ‘nnext_word’ : Next to next word
- ‘Is_numeric’ : Is it numeric ? (True / False)

Neural Part

- Recurrent neural networks, or RNNs, are a type of artificial neural network that add additional weights to the network to create cycles in the network graph in an effort to maintain an internal state.
- The promise of adding state to neural networks is that they will be able to explicitly learn and exploit context in sequence prediction problems, such as POS Tagging.
- Neural Based Models used for PoS Tagging.
  - RNN
  - LSTM
  - Bidirectional LSTM

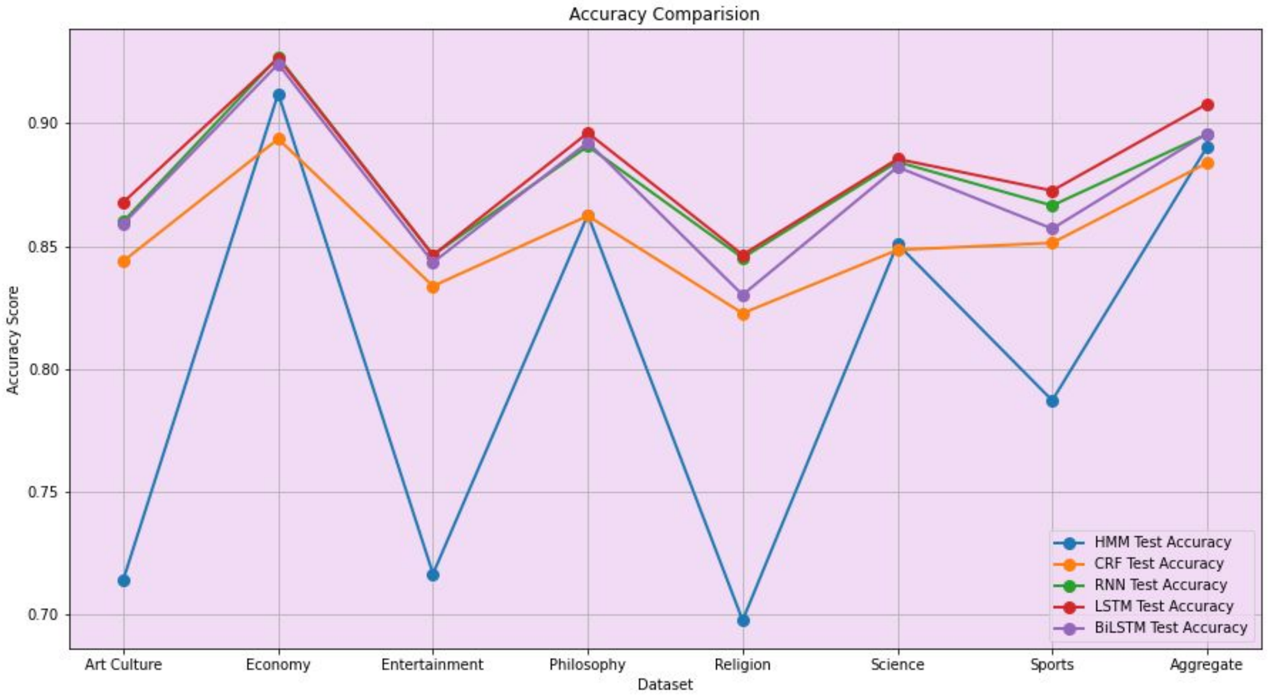
Architecture



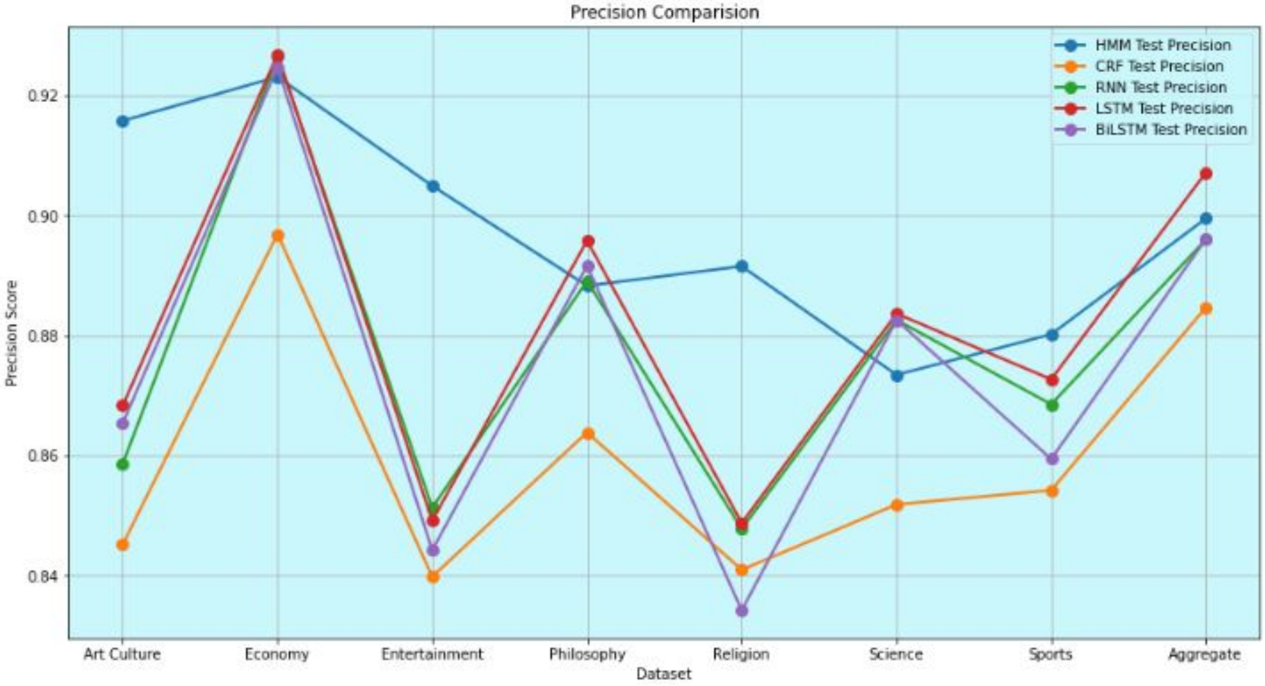
Architecture

- **Preprocessing :** Data cleaning, Remove non useful characters, words and output list of sentence
- **Feature Selection :** convert each word into feature
- **Tokenization & Encoding :** encoding of features of using inbuilt Tokenizer of nltk
- **Split into Train Test Validation :** 65 - 20 - 15 ratio
- **Word Embedding :** More meaningful vector representation for neural model
- **RNN Model :** RNN, LSTM and Bidirectional LSTM
- **Evaluation Metrics :** Accuracy, F1-score

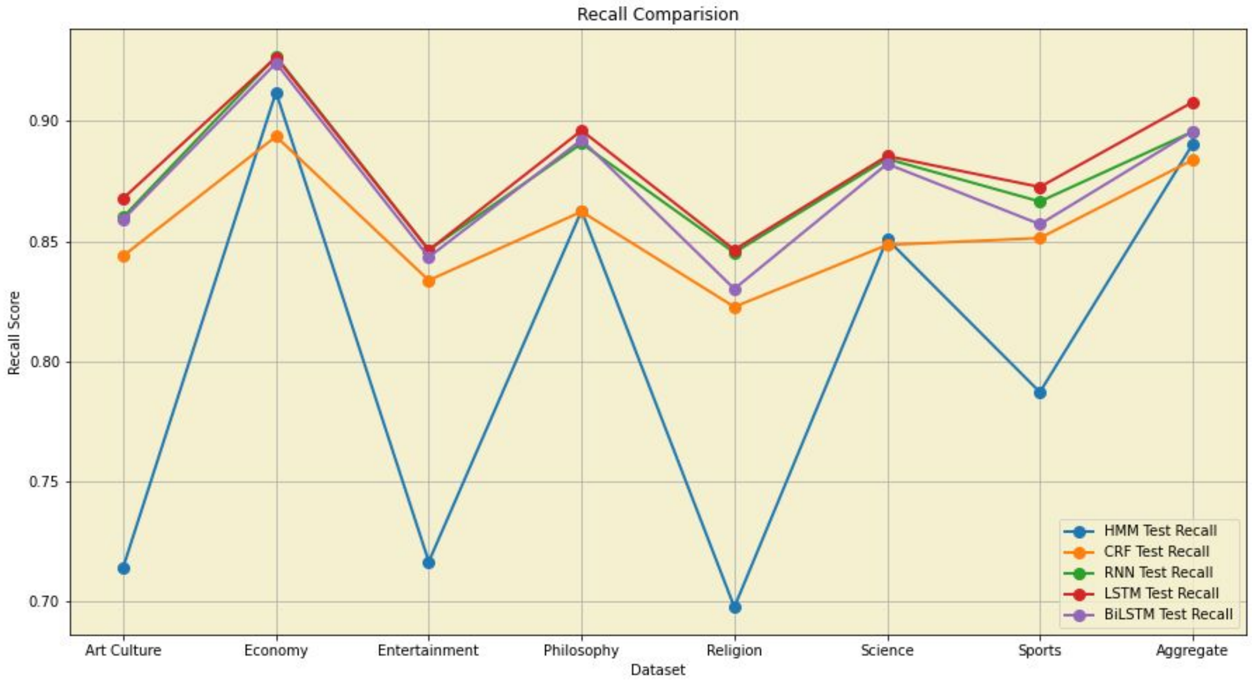
Results & Comparisons : Accuracy



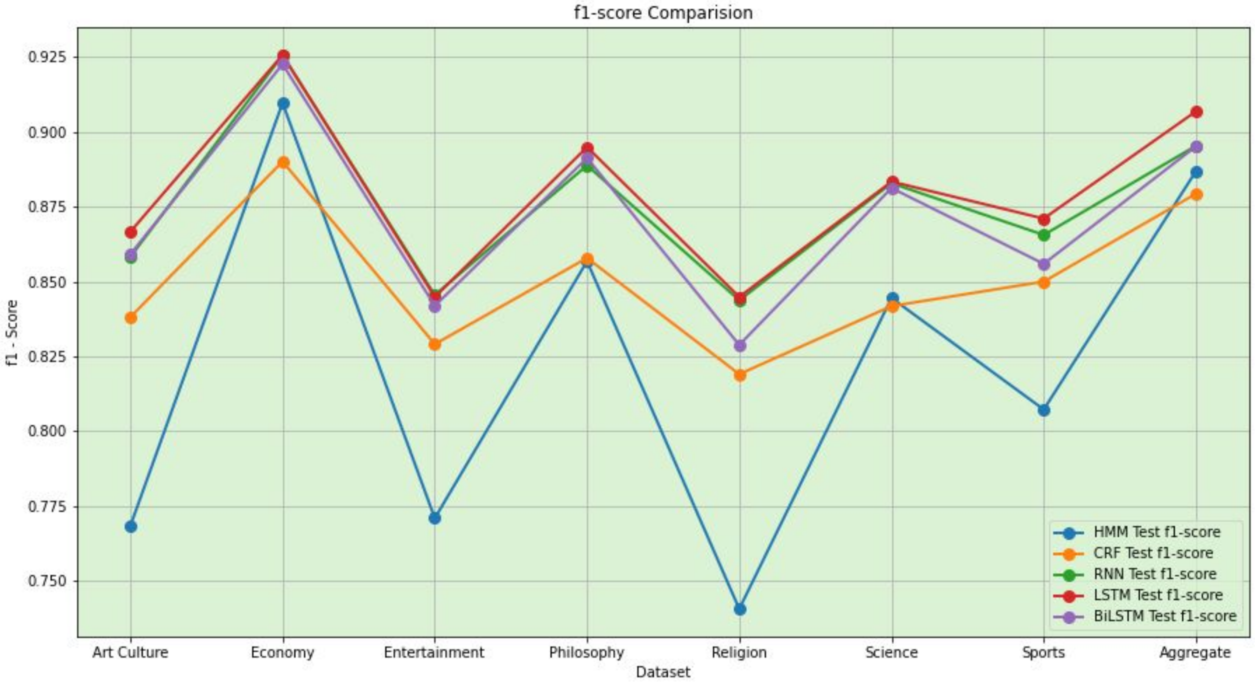
Results & Comparisons : Precision



Results & Comparisons : Recall



Results & Comparisons : F1-score



Error Analysis - HMM

Original Tag	Assigned Tag	Error Count
N_NNP	N_NN	481
JJ	N_NN	325
V_VAUX_VNP	N_NN	290
V_VM	N_NN	173
QT_QTC	N_NN	80
PR_PRP	DM_DMD	70
V_VAUX	V_VM	67

Error Analysis - CRF

Original Tag	Assigned Tag	Error Count
N_NNP	N_NN	427
JJ	N_NN	334
V_VAUX_VNP	N_NN	123
N_NN	JJ	122
PR_PRP	DM_DMD	72
N_NN	N_NNP	63
V_VM	N_NN	54

Error Analysis - LSTM

Original Tag	Assigned Tag	Error Count
N_NN	N_NNP	252
N_NN	JJ	182
N_NNP	N_NN	163
JJ	N_NN	157
V_VAUX_VNP	V_VM	80
N_NN	V_VAUX_VNP	70
V_VM	V_VAUX	57
PR_PRP	DM_DMD	56

Conclusion

- HMM model has more precision but less recall compare to other models.
- CRF accuracy does not fluctuate much with change in dataset.
- LSTM model has good Accuracy and F1 score compare to other models.
- An adjective (JJ) is tagged as a noun (N_NN) with a high error count because, in Gujarati language, adjectives may or may not occur before the nouns.
- All models are struggling to make differences between common(N_NN) and proper noun (N_NNP).

THANK YOU