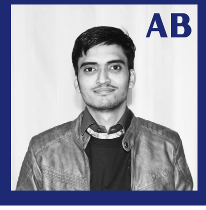
Univ.AI

Named Entity Recognition (NER)

Rachit Jain, Anshul Bhardwaj



ABOUT

The project focuses on creating a NER model that identifies key tokens and classifies them into set of predefined entities. The data would involve scientific publications in the **WIESP Dataset**.

NER helps us extract key information from scientific papers which can help search engines to better select and filter articles.

MODELS & RESULTS

The basic architecture of any model remains the same. What defines its success is how well it is put into use!

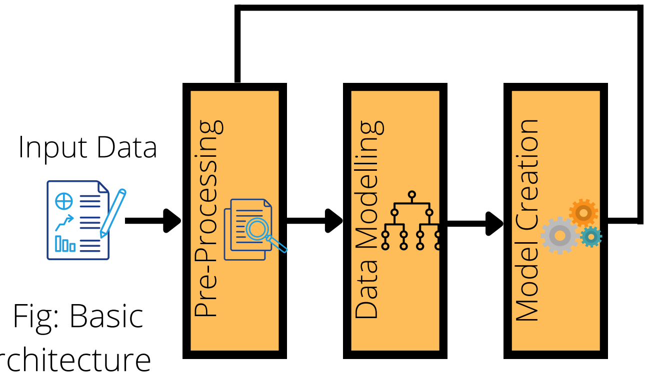


Fig: Basic Architecture

USING SIMPLE RNN

Accuracy maxed out at **92%** with multiple design changes

USING WORD2VEC

The trained embeddings **do not do well** for our task

BASELINE: 2 LSTM STACKED

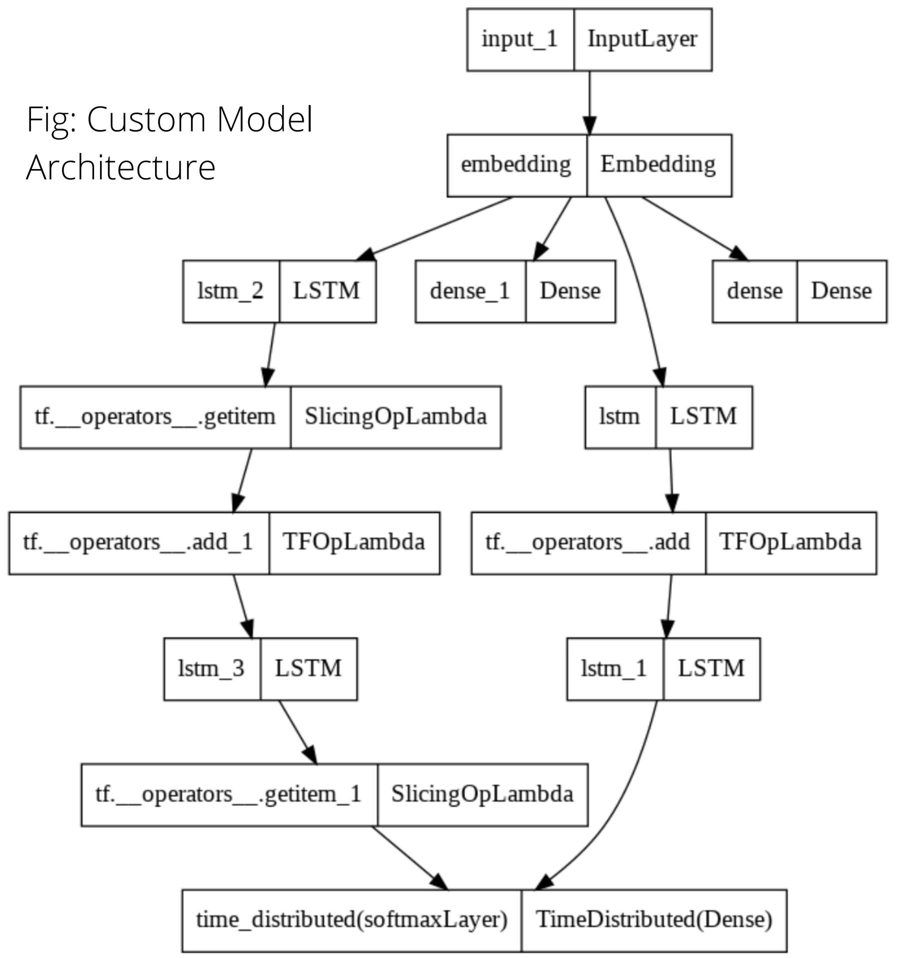
Improvement over RNNs; **92.5%** accuracy after fine-tuning

Model Type											
Mainframe	Layers	Sentence Length	Bidirectional	Hidden Size	Batch Size	Epochs	Dataset	Loss	Accuracy	Validation Loss	Validation Accuracy
SimpleRNN	3	10 to 30	FALSE	32	8	30	After extra pre-processing	0.1811	0.9378	0.2933	0.9195
SimpleRNN	3	10 to 30	FALSE	32	8	30	Without extra pre-processing	0.2388	0.9249	0.2897	0.9166
SimpleRNN	3	10 to 30	TRUE	32	8	30	After extra pre-processing	0.2414	0.9228	0.287	0.9155
SimpleRNN	3	10 to 30	TRUE	32	8	30	Without extra pre-processing	0.2332	0.9255	0.284	0.9142
SimpleRNN	3	10 to 30	TRUE	32	8	8	After extra pre-processing	0.2031	0.9308	0.2797	0.9169
SimpleRNN	3	10 to 30	TRUE	32	8	10	Without extra pre-processing	0.218	0.9289	0.2844	0.9182
SimpleRNN	3	10 to 60	TRUE	32	8	9	Without extra pre-processing	0.1352	0.9278	0.1651	0.9191

Fig: Different experiments with basic model architectures by tuning parameters

2 BIDIRECTIONAL LSTM LAYERS

Fig: Custom Model Architecture



This allowed model to **learn context over long sentences** and have enough trainable parameters to increase its robustness. Accuracy rose to **95.75%** without any pre-training system

BERT-BASED MODEL

After multiple BERT versions, the best accuracy of **97.2%** came on the model based on **SciBERT**: A Pretrained Language Model for Scientific Text

DATA & LABELS

1. **bibcode**: Can be used to extract the texts.
2. **label_studio_id**: Can also be used to extract texts.
3. **ner_ids**: Can be used as label encoded values for the ner_tags.
4. **section**: Two types: fulltext or acknowledgement.
5. **tokens**: The units whose entities need to be found by the model.
6. **unique_id**: This could also be used to separate out units of texts.

Fig: Raw Data Representation

bibcode	label_studio_id	ner_ids	ner_tags	section	tokens	unique_id
2019MNRAS.486.5558S	487	62	O	fulltext	fit	fulltext_487_2019MNRAS.486.5558S
2019MNRAS.486.5558S	487	62	O	fulltext	uncertainty.	fulltext_487_2019MNRAS.486.5558S
2019MNRAS.486.5558S	487	62	O	fulltext	Photometric	fulltext_487_2019MNRAS.486.5558S
2019MNRAS.486.5558S	487	62	O	fulltext	data	fulltext_487_2019MNRAS.486.5558S
2019MNRAS.486.5558S	487	62	O	fulltext	are	fulltext_487_2019MNRAS.486.5558S
2019MNRAS.486.5558S	487	62	O	fulltext	from	
2019MNRAS.486.5558S	487	17	B-Mission	fulltext	K2,	
2019MNRAS.486.5558S	487	15	B-Instrument	fulltext	SMEI,	
2019MNRAS.486.5558S	487	27	B-Telescope	fulltext	Hipparcos	
2019MNRAS.486.5558S	487	62	O	fulltext	(	
2019MNRAS.486.5558S	487	30	B-Wavelength	fulltext	H	
2019MNRAS.486.5558S	487	61	I-Wavelength	fulltext	p	

Fig: Extracted Sentence Lengths

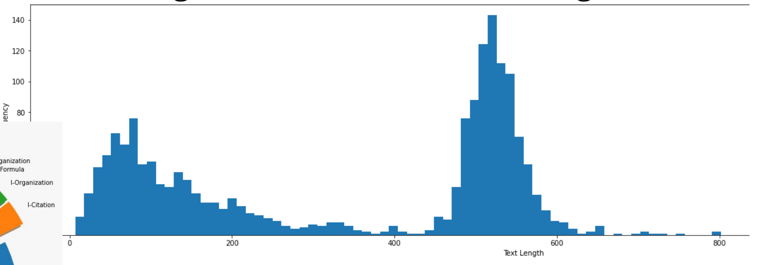
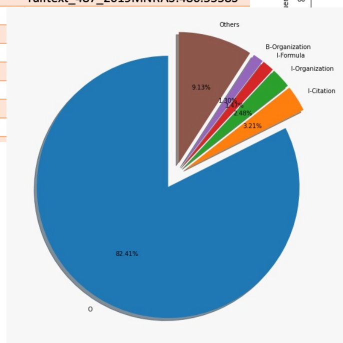


Fig:Distribution of NER Tags in the dataset



CONCLUSION

Right Batch Size, Early Stopping, More Trainable Parameters, Deep Architectures and Transfer Learning truly boost the performance of models.

SCOPE OF IMPROVEMENT

1. Class Imbalance: use custom loss function
2. More data for under-represented tokens
3. Use embeddings trained for Scientific NER