NSIT & IIITDWD @ HASOC 2020:
Deep learning model for hate-speech identification in Indo-European languages

HASOC FIRE 2020

20TH DEC 2020

Presented By :

Shivangi Srivastava - (B.Tech CSE, Netaji Subhas Institute of Technology, Patna, India)

Sunil Saumya - (Asst. Professor, Indian Institute of Information Technology Dharwad, India)

Roushan Raj - (B.Tech CSE, Netaji Subhas Institute of Technology, Patna, India)

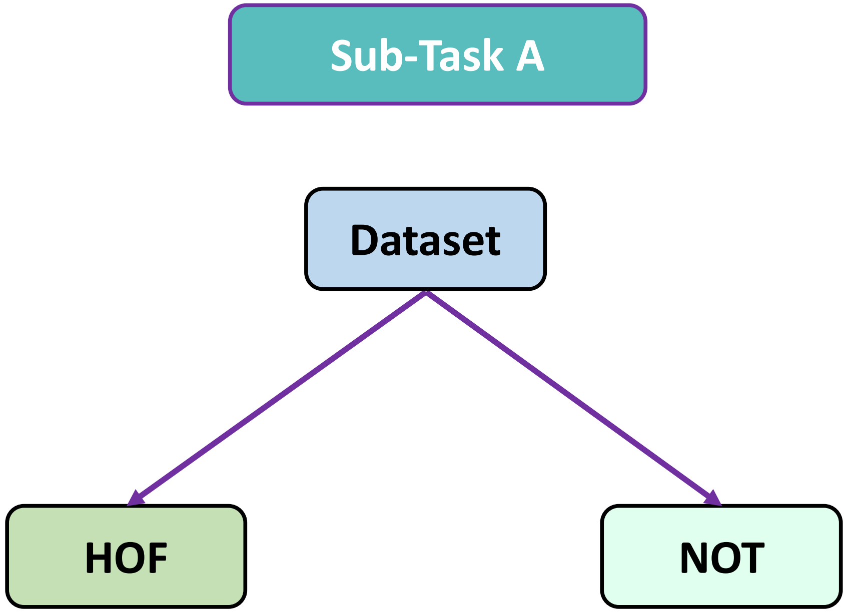
Preview

- Objective
- Task Description
- Data Statistics
- Methodology
- Results
- Conclusion & Future Work

Objective

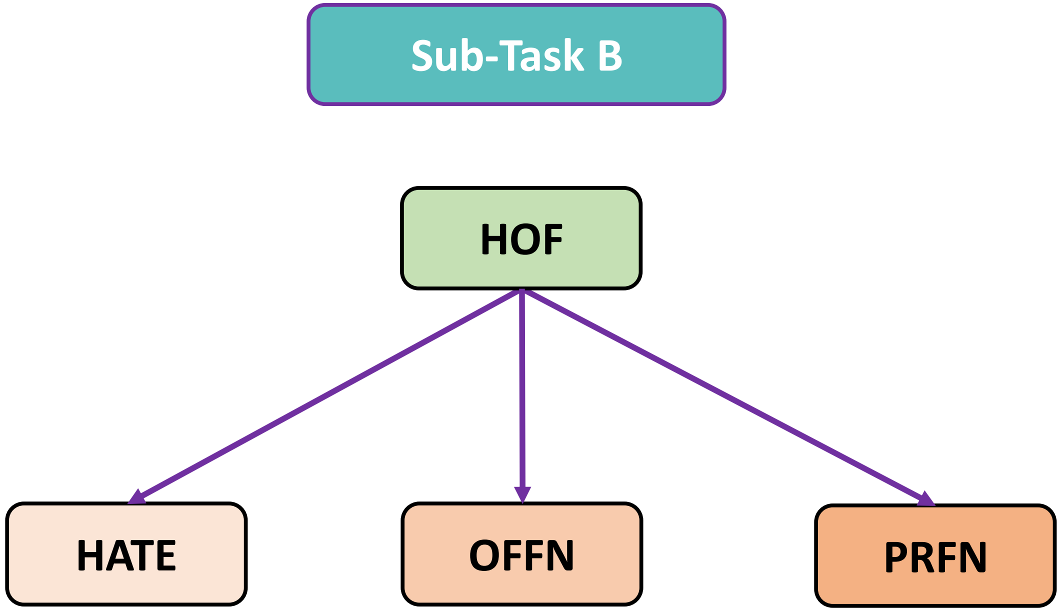
- The low cost, easy accessibility and high effectiveness of social media have changed the way we live.
- But the darker side to this comes with rapid increase in cyberbullying rates and with people spreading hatred & threatening contents.
- Cyberbullying stats 2020 show that 42% of online harassment happens on Instagram which has over a billion active users. Facebook and Snapchat follow closely, with 39% and 31% respectively.
- Its extremely necessary to regulate and monitor such offensive cotent on social media.
- We participated in both subtasks of all three languages (English , Hindi , German).

HASOC 2020 Task Description



HOF :- Hate and Offensive

NOT :- Non Hate-Offensive



HATE :- Hate

OFFN :- Offensive

PRFN :- Profane

Data Statistics

Language	Task-1		Task-2			
	HOF	NOT	HATE	OFFN	PRFN	NONE
English	1856	1852	158	321	1377	1852
German	673	1700	146	140	387	1700
Hindi	847	2116	234	465	148	2116

Table-1 : Class division of both sub-tasks for Train Dataset

Language	Task-1		Task-2			
	HOF	NOT	HATE	OFFN	PRFN	NONE
English	423	391	25	82	233	414
German	134	392	24	36	88	378
Hindi	197	466	56	87	27	493

Table-2 : Class division of both sub-tasks for Test Dataset

Data Statistics for sub-task A

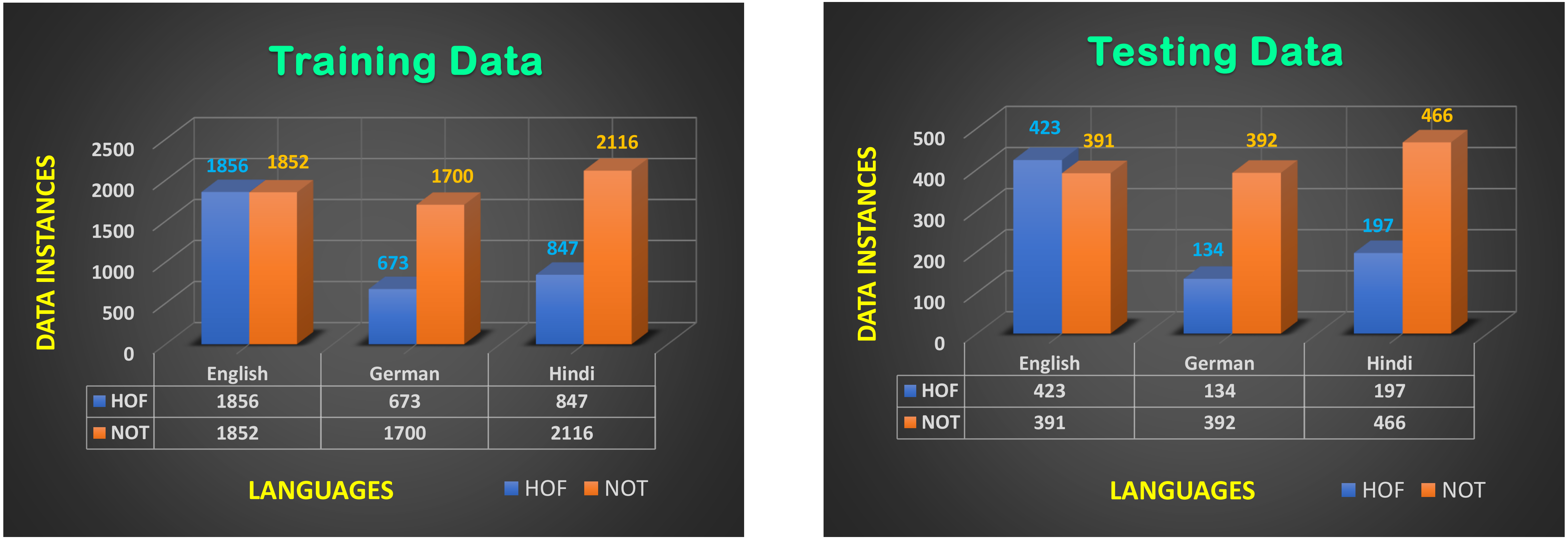
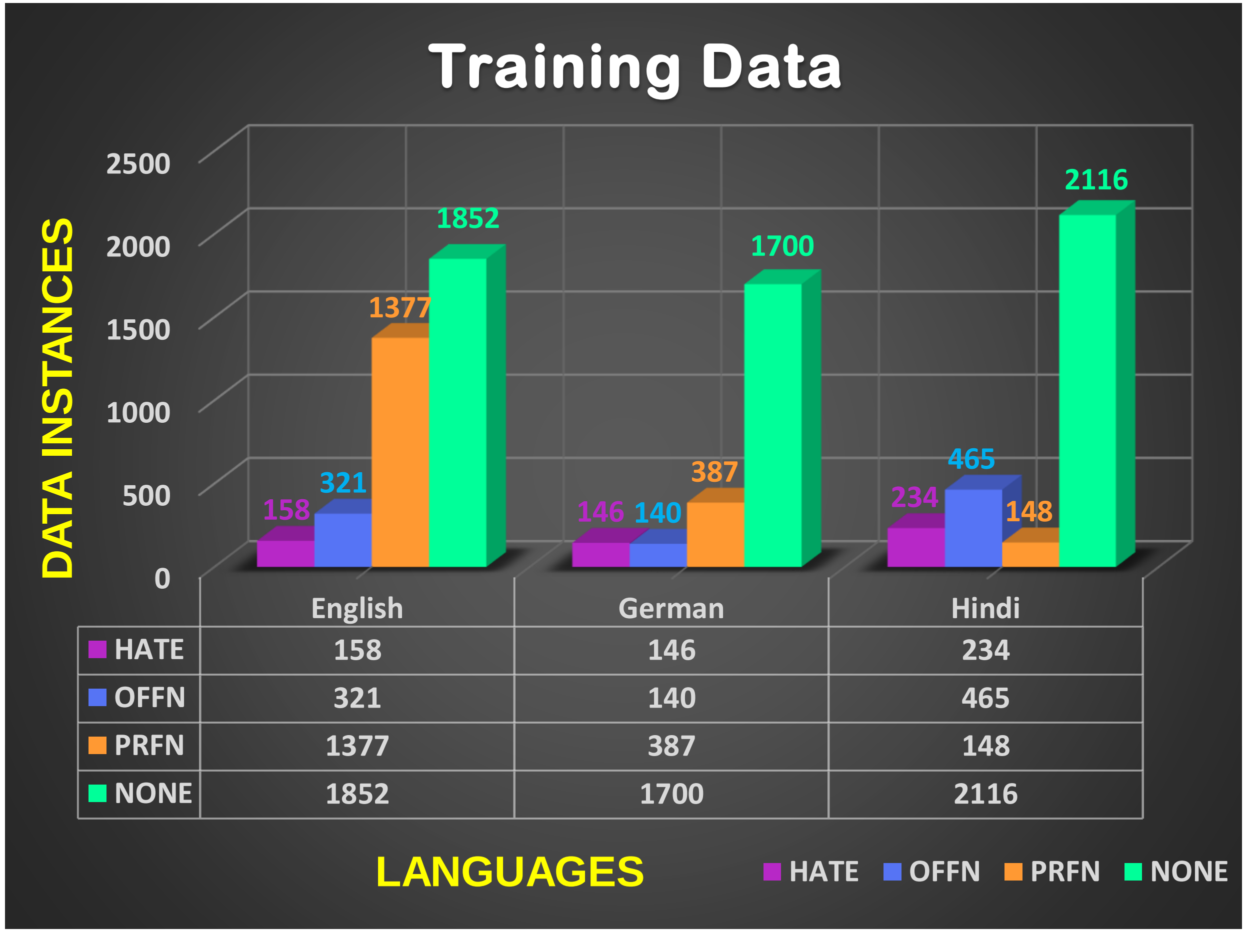
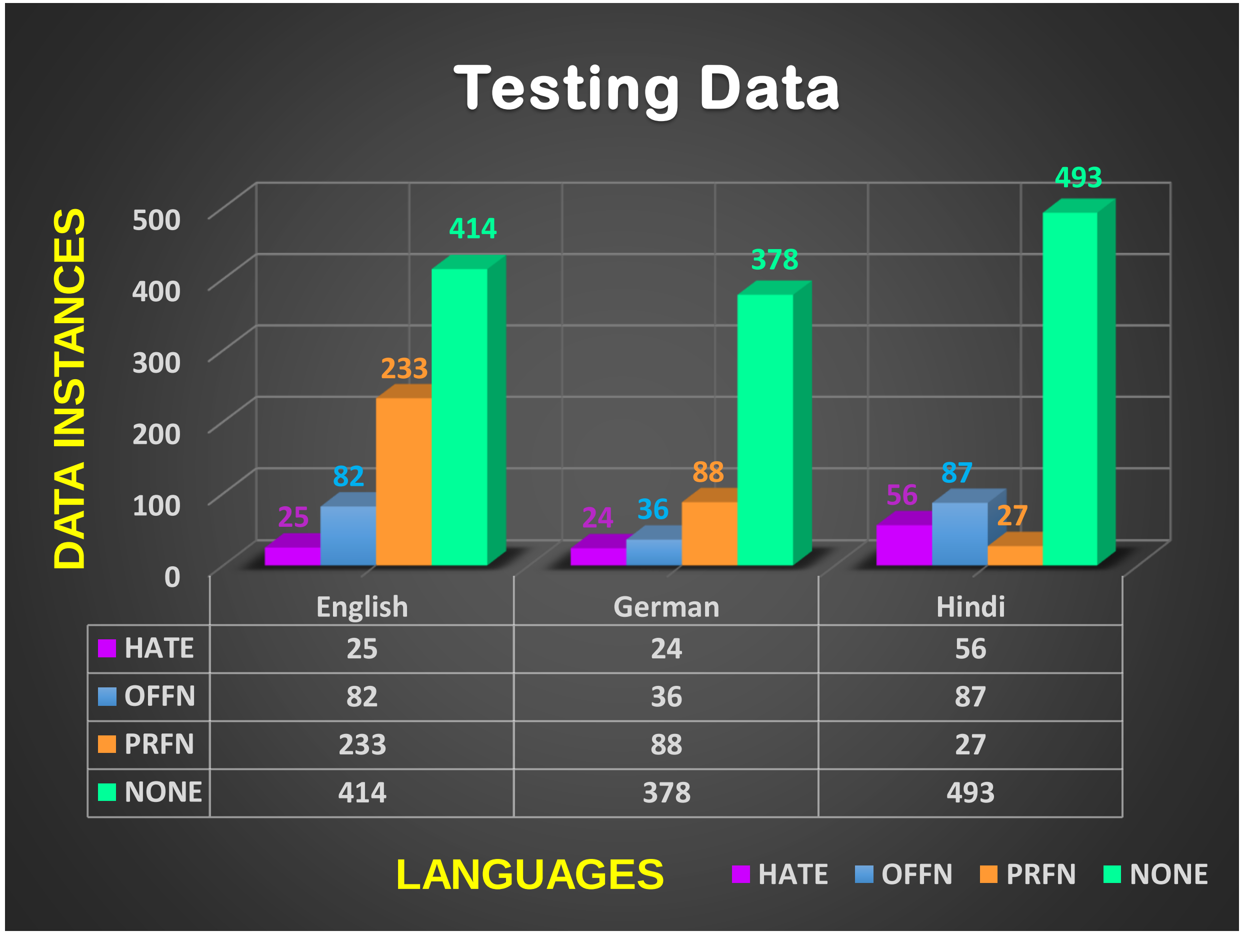


Fig. : Class distribution of sub-task A for training and testing data

Data Statistics for sub-task B



	English	German	Hindi
HATE	158	146	234
OFFN	321	140	465
PRFN	1377	387	148
NONE	1852	1700	2116



	English	German	Hindi
HATE	25	24	56
OFFN	82	36	87
PRFN	233	88	27
NONE	414	378	493

Fig. : Class distribution of sub-task B for training and testing data

Pre-processing Steps

1. Cleaning and Filtering texts:

   - convert texts to lowercase.
   - removed the redundant texts such as punctuation symbols e.g. !"#$%&´()*+,./:;<=>?@[/]ˆ{|}.
   - removed the retweet symbol (RT) of Twitter data.
   - removed URLs.
   - removed alphanumeric characters and apostrophes.

2. Removing stopwords

3. Stemming

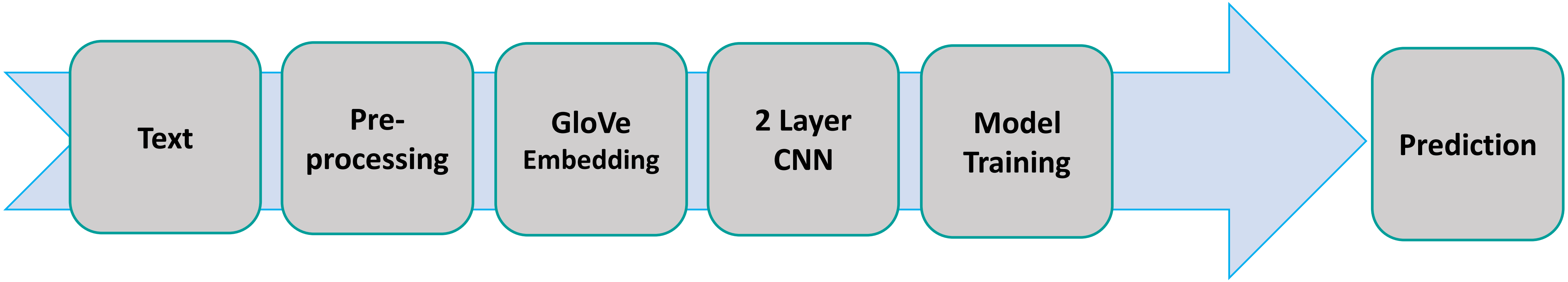
4. Tokenization and Creating vocabulary

5. Encoding

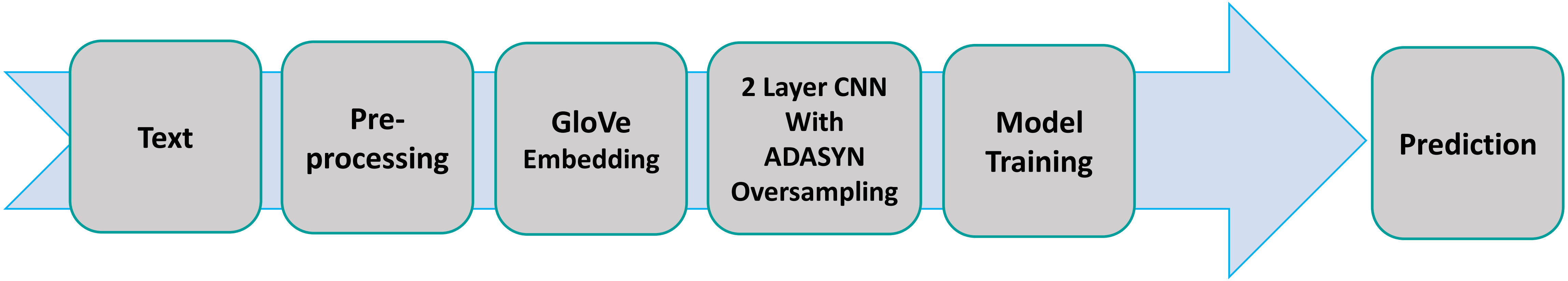
6. Pre-Padding

Our Approach for English language

English Sub-Task A



English Sub-Task B

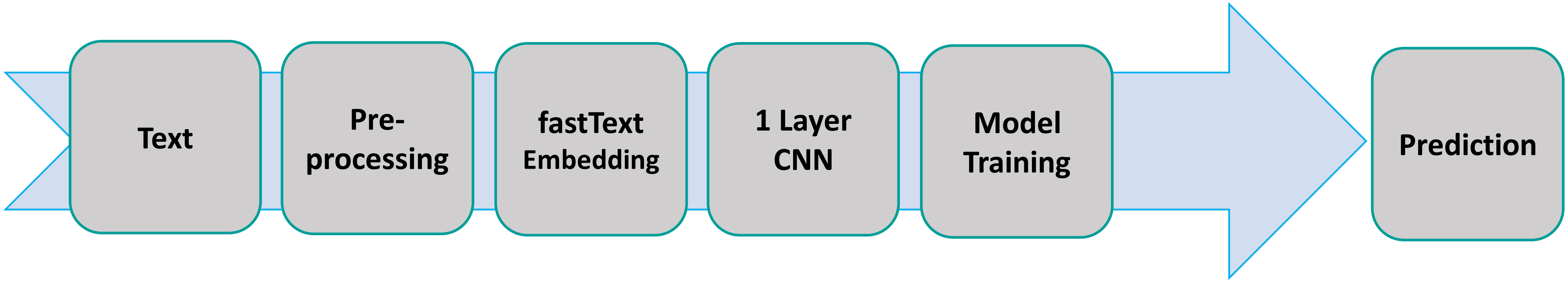


English sub-tasks results

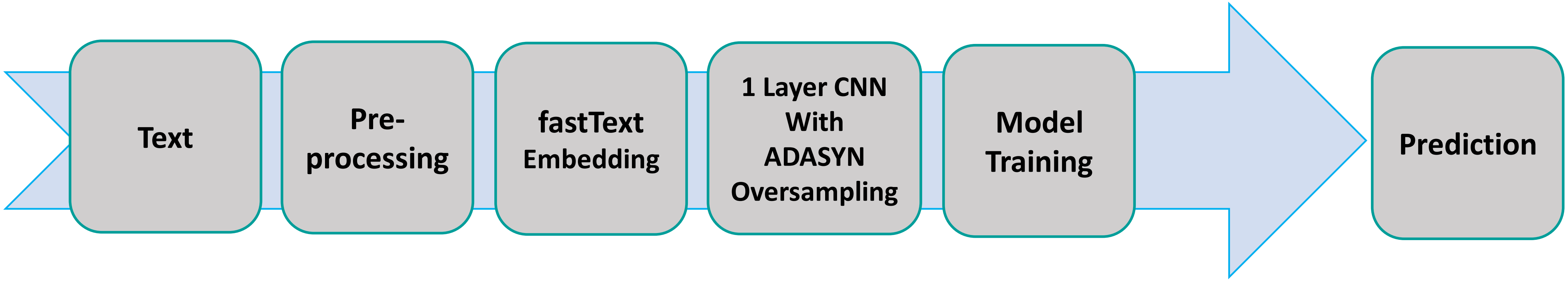
	Sub-task	Model	Embedding	f1 macro-avg
English	A	CNN 1 layer	GloVe	0.84
		CNN 2 layer	**GloVe**	**0.86**
		BiLSTM 1 layer	GloVe	0.84
		BiLSTM 2 layer	GloVe	0.83
		Hybrid Model	GloVe	0.84
	B	CNN 1 layer	GloVe, Unbalanced dataset	0.49
			GloVe, SMOTE	0.49
			GloVe, ADASYN	0.53
		CNN 2 layer	GloVe, Unbalanced dataset	0.49
			GloVe, SMOTE	0.51
			GloVe, ADASYN	**0.54**
		BiLSTM 1 layer	GloVe, Unbalanced dataset	0.48
			GloVe, SMOTE	0.50
			GloVe, ADASYN	0.51
		BiLSTM 2 layer	GloVe, Unbalanced dataset	0.48
			GloVe, SMOTE	0.49
			GloVe, ADASYN	0.51
		Hybrid Model	GloVe, ADASYN	0.51

Our Approach for German language

German Sub-Task A



German Sub-Task B

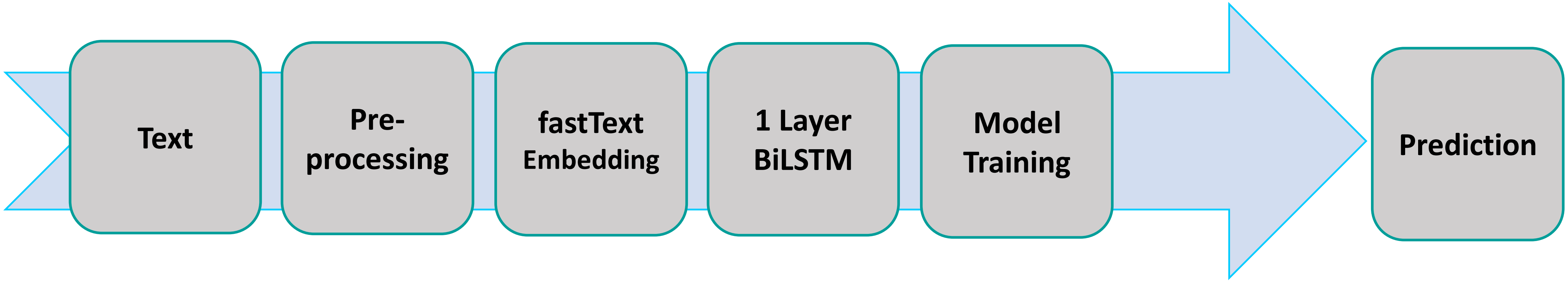


German sub-tasks results

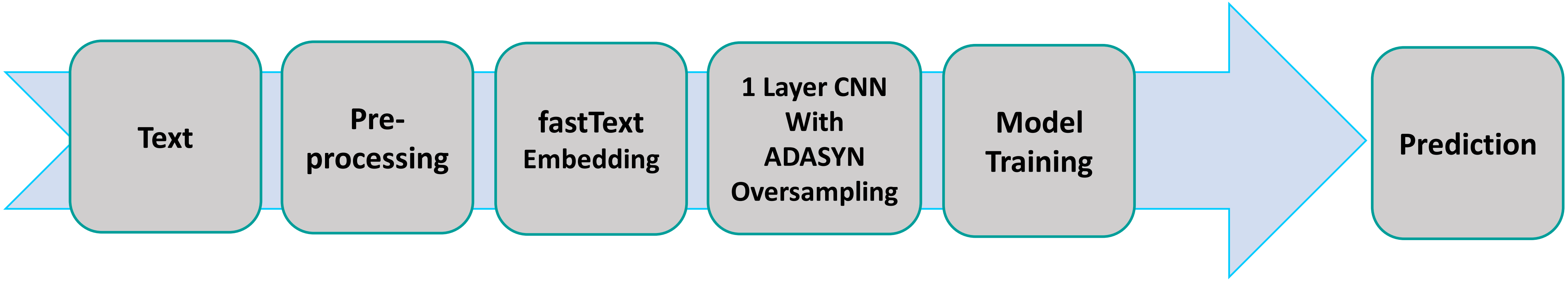
	Sub-task	Model	Embedding	f1 macro-avg
German	A	**CNN 1 layer**	**fastText**	**0.75**
		CNN 2 layer	fastText	0.73
		BiLSTM 1 layer	fastText	0.74
		BiLSTM 2 layer	fastText	0.70
		Hybrid Model	fastText	0.72
	B	**CNN 1 layer**	fastText, Unbalanced dataset	0.39
			fastText, SMOTE	0.43
			fastText, ADASYN	**0.45**
		CNN 2 layer	fastText, Unbalanced dataset	0.39
			fastText, SMOTE	0.40
			fastText, ADASYN	0.43
		BiLSTM 1 layer	fastText, Unbalanced dataset	0.38
			fastText, SMOTE	0.41
			fastText, ADASYN	0.42
		BiLSTM 2 layer	fastText, Unbalanced dataset	0.37
			fastText, SMOTE	0.33
			fastText, ADASYN	0.35
		Hybrid Model	fastText, ADASYN	0.41

Our Approach for Hindi language

Hindi Sub-Task A



Hindi Sub-Task B



Hindi sub-tasks results

	Sub-task	Model	Embedding	f1 macro-avg
Hindi	A	CNN 1 layer	fastText	0.55
		CNN 2 layer	fastText	0.57
		BiLSTM 1 layer	**fastText**	**0.67**
		BiLSTM 2 layer	fastText	0.59
		Hybrid Model	fastText	0.53
	B	**CNN 1 layer**	fastText, Unbalanced dataset	0.23
			fastText, SMOTE	0.35
			fastText, ADASYN	**0.36**
		CNN 2 layer	fastText, Unbalanced dataset	0.22
			fastText, SMOTE	0.35
			fastText, ADASYN	0.34
		BiLSTM 1 layer	fastText, Unbalanced dataset	0.29
			fastText, SMOTE	0.33
			fastText, ADASYN	0.35
		BiLSTM 2 layer	fastText, Unbalanced dataset	0.28
			fastText, SMOTE	0.32
			fastText, ADASYN	0.32
		Hybrid Model	fastText, ADASYN	0.34

Hyper-parameters

➢ Epochs	– 200
➢ Batch size	– 32
➢ Activation function	– ReLU, Sigmoid
➢ Optimizer	– Adam
➢ Dropout rate	– 0.2
➢ Pre-padding (max-length)	– 100
➢ Reduce lr	(Patience - 2)
➢ Earlystopper	(Patience - 8)

Result and Observations

Languages	f1 macro-avg in Sub-tasks A/B	Rank in sub-tasks A/B
Hindi	0.5337 / 0.2667	1st / 2nd
German	0.4919 / 0.2468	18th / 14th
English	0.4879 / 0.2361	32nd / 16th

- **Ranks** and **f1 macro-avg score** of our best six models are calculated by the organization with approximately 15% of the private test data.

- Sub-task B achieved a lower f1 macro-avg score than sub-task A irrespective of the language. Reasons could be :

  - ✓ The heavily unbalanced dataset.
  - ✓ Miniature differences in the three classes leading to predicting lot more false-positive classes.
  - ✓ Hindi dataset was a code-mixed data with a lot of English words, while the embedding used was only for the Hindi language, which could probably be a reason for the poor performance in sub-task B.

Conclusion & Future Work

- We proposed different CNN and BiLSTM architecture developed using word vectors of the relevant pre-trained corpus.

- Future scope could be improving dataset balancing as sub-task B gave a lower f1 macro-avg score even after applying SMOTE and ADASYN over-sampling techniques.

- Further improvisation could be to tackle the identification of hate speech in multilingual tweets and posts on social media

- Open source implementation of our best models :
  https://github.com/roushan-raj/HASOC-2020

References

- S. Mishra, S. Mishra, 3idiots at hasoc 2019: Fine-tuning transformer neural networks for hate speech identification in indo-european languages., in: FIRE (Working Notes), 2019, pp. 208–213.
- S. Kamble, A. Joshi, Hate speech detection from code-mixed hindi-english tweets using deep learning models, arXiv preprint arXiv:1811.05145 (2018).
- Z. Waseem, Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter, in: Proceedings of the first workshop on NLP and computational social science, 2016, pp. 138–142.
- I. Alfina, R. Mulia, M. I. Fanany, Y. Ekanata, Hate speech detection in the indonesian language: A dataset and preliminary study, in: 2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS), IEEE, 2017, pp. 233–238.
- S. Kamble, A. Joshi, Hate speech detection from code-mixed hindi-english tweets using deep learning models, arXiv preprint arXiv:1811.05145 (2018).
- S. Hinduja, J. W. Patchin, Connecting adolescent suicide to the severity of bullying and cyberbullying, Journal of school violence 18 (2019) 333–346.

Acknowledgment

Thomas Mandl,
University of Hildesheim, Germany

Sandip Modha,
DA-IICT & LDRP-ITR, Gandhinagar, India

Gautam Kishore Shahi,
University of Duisburg-Essen, Germany

Amit Kumar Jaiswal,
University of Bedfordshire, UK

Durgesh Nandini,
University of Bamberg, Germany

Prasenjit majumder,
DA-IICT, Gandhinagar, India

Daksh Patel,
Dalhousie University, Halifax, Canada

Johannes Schäfer,
University of Hildesheim, Germany

Thank You!