Multilingual Fine-grained Named Entity Recognition

Thesis submitted in partial fulfilment of the requirements
for the award of the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Prachuryya Kaushik

Under the supervision of

Prof. Ashish Anand

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

January, 2026

Acknowledgements

I would like to express my heartfelt gratitude to my supervisor Prof. Ashish Anand. I am highly obliged for his excellent guidance, perpetual support, and exceptional encouragement. It goes without saying that this journey would not have been possible without the persistent support, the unconditional love, and profound encouragement of my father Dr Pradip Sharma, my mother Dr Chayanika Sharma, and my sister Ar. Praschaya Kaushik. They never doubted my intentions and wholeheartedly supported me in all my endeavours. I express my gratitude towards Prof. Tamarapalli Venkatesh, the Head of the Department, Computer Science and Engineering, in the Indian Institute of Technology Guwahati. The academic journey is incomplete without the blessings of esteemed faculties Prof. Sanasam Ranbir Singh, Prof. Amit Awekar, Prof. V. Vijaya Saradhi, Prof. Prithwijit Guha, Prof. Chandan Karfa, Prof. Diganta Goswami, Prof. Jatindra Kumar Deka, Prof. Purandar Bhaduri, Prof. Siddhartha Deb, Dr. G Muthukumar, Prof. Dipendu Bhunia, Prof. Shamsher Bahadur Singh, Prof. Ravi Kant Mittal, Prof. Anshuman, Dr. M. Lakshmi Vara Prasad, Prof. Sudipta Nandy, Mr Hem Kakati, Ms. Gayatri Goswami, and Ms. Mitali Devi to whom I will always be indebted to. I would like to express my gratitude to officers Mr Bhriguraj Borah, Dr Debajit Sarma, Mr Anurag Tiwari, Mr Nava Kumar Boro, Mr Pranjit Talukdar, Mr Nanu Alan Kachari, Mr Mukunda Madhab Khanikar, Mr Mishu Paul, Ms. Gauri Khutiya Deori, and Mr Raktajit Pathak and my dearest friends, Mr Rahul Malla Buzar Baruah, Mr Suvarthi Sarkar, Dr Nilotpal Biswas, Mr Saurav Gupta, Ms. Ajanta Maurya, Mr Rishabh Mahanta, Mr Kushal Sharma, Mr Gautam Singha, Mr Rohit Raj Rai, Mr Chaitanya Kirti, Mr Telem Joyson Singh, Mr Buddheshwar Bodo, Mr Hrishikesh Bodo, Ms. Wasana Paksaranuwat, Mr Chetan Chinchulkar, Mr Brijesh Yadav, Mr Fugare Ashish Dileep, Mr Alik Pramanick, Mr Nilmadhab Das, Mr Sonal Kumar, Mr Praveen Karmakar, Ms. Laxita Agrawal, Mr Adithya K. Moorthy, Ms. Shifali Agrahari, Mr Saurabh Kumar, Ms. Soumya Bharadwaj, Mr Adittya Gupta, Mr Gautam Sharma, Mr Shivansh Mishra, Ms. Archana K R, Ms. Priya Rajan, Ms. Jia Garodia, Dr Arijit Nath, Dr Alakesh Kalita, Dr Akshay Parekh, Dr Manoj Das, Dr Nazreena Rahman, Dr Pritam Deka, Dr Shaik Hussain, and many others, whom I could not mention here. I fall short of words to express my gratitude to them.

January 6, 2026 Prachuryya Kaushik

Declaration

I certify that

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor(s).
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis can be considered plagiarism to the best of my knowledge and understanding and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor(s) are not in a position to check for any possible instance of plagiarism within this submitted work.

January 6, 2026 Prachuryya Kaushik

Declaration on the Use of AI

I hereby declare that this PhD thesis titled "Multilingual Fine-grained Named Entity Recognition" is my own original work carried out under the supervision of Prof. Ashish Anand.

During the preparation of this thesis, I used AI-based tools (such as Grammarly, ChatGPT, and Google Gemini) solely for non-creative tasks such as correcting the grammar, improving the clarity and organization of the text.

No AI system was used to conduct data analysis, generate scientific insights, design methodologies, or produce research results. All conceptual development, algorithm design, analysis, and conclusions presented in this thesis are entirely my own.

I confirm that no text, code, images, or results have been directly copied from AI outputs without thorough review and modification. I take full responsibility for the accuracy, originality, and integrity of all content in this thesis.

January 6, 2026 Prachuryya Kaushik

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

Ashish Anand
Professor
Email : anand.ashish@iitg.ac.in
Phone : +91-361-258-2374

Certificate

This is to certify that this thesis entitled "**Multilingual Fine-grained Named Entity Recognition**" submitted by **Prachuryya Kaushik**, in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy, to the Indian Institute of Technology Guwahati, Assam, India, is a record of the bonafide research work carried out by him under my guidance and supervision at the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India. To the best of my knowledge, no part of the work reported in this thesis has been presented for the award of any degree at any other institution.

Date: January 6, 2026
Place: Guwahati

Prof. Ashish Anand
(Thesis Supervisor)

Abstract

The extraction of domain-specific entity information, particularly via Fine-grained Named Entity Recognition (FgNER), is crucial for advancing Natural Language Processing (NLP) applications such as knowledge base construction and relation extraction. However, the creation of high-quality FgNER datasets is severely hampered by the prohibitive cost of manual annotation and the inherent noise in dataset generated through weak supervision paradigms such as distant supervision. This is even more challenging for languages with a diverse landscape and a scarcity of resources. This thesis addresses these critical quality and resource scarcity challenges by developing novel, scalable frameworks for generating vast, high-quality FgNER datasets across numerous languages and diverse entity type taxonomies. In addressing the challenges, this work makes four key contributions. In the first contribution, a Taxonomy Adaptable Fine-grained Entity Recognition through Distant Supervision framework, TAFSIL, is proposed. TAFSIL leverages the high interlink between WikiData and Wikipedia through multi-stage heuristics, fuzzy matching, and quality sentence selection. The proposed framework is then used to create robust FgNER datasets in six Indian languages: Hindi, Marathi, Sanskrit, Tamil, Telugu, and Urdu. With approximately three million samples, the TAFSIL dataset achieves an 83% relative improvement in average F1 score over zero-shot baselines. Although TAFSIL is a highly scalable framework across multiple languages and taxonomies, its dependence on the availability of Wikipedia is a bottleneck. Since Wikipedia is not available for many low-resource and vulnerable languages, this thesis tries to exploit alternative resources and methodologies. The second contribution is an application of the annotation projection method to create FgNER resources for three vulnerable languages of India's North Eastern Region: Bodo, Manipuri, and Mizo. Existing FgNER resources in English and parallel corpora from English to target vulnerable languages are utilized via the annotation projection method to create these high-quality datasets. Moreover, in this work, cross-lingual zero-shot FgNER analyses are discussed substantially. These analyses suggest the importance of script similarity in FgNER, which paves the way to leverage this property to reduce noise and improve dataset quality. The third contribution is CLASSER, a framework for Cross-lingual Annotation Projection enhancement through Script Similarity for Fine-grained Named Entity Recognition. CLASSER employs a two-stage process: an initial projection from source language annotations to target language annotations, followed by a crucial refinement step that leverages datasets from script-similar languages. CLASSER is applied to five low-resource Indian languages: Assamese, Bodo, Marathi, Nepali, and Sanskrit. The dataset created through the CLASSER framework is of very high quality, demonstrating substantial performance gains, with F1 score improvements of up to 26% in Marathi and 46% in Sanskrit over the current state-of-the-art. Although the CLASSER framework is very effective across script-similar languages, its scalability is limited for languages written using different scripts. Therefore, the fourth contribution consolidates the resource creation efforts by introducing the highly scalable framework, Entity-Anchored Machine Translation (EaMaTa) framework. The EaMaTa framework is then used to create SampurNER, a large-scale, high-quality FgNER dataset covering all 22 scheduled Indian languages, and its extension FiNE-MiBBiC, an FgNER dataset targeting four extremely low-resource languages. Rigorous analyses confirm the superior quality of these resources, demonstrating an up to 9% increase in F1-score over the current state-of-the-art. Extensive insights are highlighted regarding the influence of language family and script similarity

on cross-lingual FgNER model performance across one of the most diverse linguistic groups. The collective work presented in this thesis significantly mitigates the resource gap for FgNER in Indian and low-resource languages, providing high-quality, large-scale datasets and establishing effective, scalable frameworks for future resource creation efforts.

Keywords: Named entity recognition, Fine-grained Named entity recognition, Cross-lingual transfer, Multilingualism, Endangered languages, Resources for less-resourced languages, Automatic creation and evaluation of language resources, Datasets for low resource languages

Contents

List of Figures

List of Tables

List of Algorithms

Glossary of Terms

Terms	Details
Anchor Symbols	Special symbols used in the entity-anchored sentence format to mark the start and end boundaries of an entity mention, with the end symbol being distinct based on the entity type.
Annotation Projection	A technique for multilingual data creation that uses parallel corpora and word alignment to transfer annotations from a high-resource source language to a low-resource target language.
Auxiliary Annotation Sequence	The sequence of predictions generated for the target sentence by the multilingual encoder model, which was fine-tuned on the script-similar auxiliary language dataset.
Auxiliary Language	A language used in the refinement stage that has an existing Fine-grained Named Entity Recognition dataset and uses a script similar to the target language, but generally lacks parallel data with the target language.
Boundary Mismatch Errors	Errors that occur when the entire entity mention cannot be detected correctly or when the predicted entity span does not perfectly match the correct boundary.
CLASSER	Cross-lingual Annotation Projection enhancement through Script Similarity for Fine-grained Named Entity Recognition.
Co-prediction	An observation in error analysis where entity types are often predicted together (confused) by the model, especially when the underlying concepts are closely related.
Confidence Threshold	A predefined probability value used in the refinement function. The initial annotation is only replaced by the auxiliary annotation if the auxiliary model predicts the new label with a probability greater than or equal to the Confidence Threshold.
Contextual Embeddings	Vector representations of words or tokens that capture their meaning based on the surrounding words in the sentence.

Terms	Details
Cross-lingual Annotation Projection	A method where annotations from a source language are transferred to a target language using a source-to-target language parallel corpora and word alignment tools.
Cross-lingual Fine-grained Named Entity Recognition	The task of performing Fine-grained Named Entity Recognition in one language by leveraging resources (models or data) trained or fine-tuned primarily on a different language.
Cross-lingual Transfer	The process of applying a model trained or fine-tuned on one language to a different language.
Crosslingual Semantic Text Similarity (XSTS)	A human evaluation methodology used to rate translated sentences (typically on a scale of 1 to 5) to ensure that the semantic meaning of the source sentence is preserved in the target language.
Distant Supervision	A method for dataset creation that links textual mentions to knowledge-base (KB) entries and inherits the corresponding type information.
Edit Distance	The number of single-character edits (insertions, deletions, substitutions) required to change one word into the other.
Entity	An identifiable object that exists within a defined universe.
Entity Detection	The task of focusing on locating spans that correspond to predefined categories or types.
Entity Recognition	The task that involves performing Entity Detection and Entity Typing jointly.
Entity Typing	The task of assigning specific types or categories to entity mentions that have already been detected.
Entity-anchored Machine Translation (EaMaTa)	A proposed framework for high-quality multilingual Fine-grained Named Entity Recognition dataset creation, converting the source Named Entity Recognition dataset into an "anchored" format before translation.
Entity-anchored Sentence Format	A sentence format where each entity mention is enclosed by unique anchors.
Entity-anchors Mismatch Detection	The cleaning process in the EaMaTa framework where sentences missing a corresponding start or end anchor for an entity mention are removed.

Terms	Details
False Negative	An entity mention is present in the ground truth, but could not be detected by the system.
False Positive	An entity mention is not present in the ground truth, but falsely detected by the system.
FewNERD	A large-scale, human-annotated English Fine-grained Named Entity Recognition dataset used as the source for the EaMaTa framework, covering 8 coarse-grained and 66 fine-grained entity types.
Final Refined Annotation	The resulting high-quality annotation sequence after applying the refinement function to the initial annotation sequence.
Fine-grained Named Entity Recognition (FgNER)	Entity recognition that uses a more sophisticated and detailed classification taxonomy.
Fuzzy Match	An empirical heuristic applied to detect mentions even when they have multiple valid spellings or variations (like agglutinations) within a calculated maximum edit distance.
Gold Dataset	A dataset that is manually annotated by human annotators and considered the ground truth for evaluation.
Gold Test Set	A small subset of the test data that is manually annotated by human experts for rigorous evaluation, serving as the ground truth.
Initial Annotation	The annotation sequence obtained after the direct annotation projection of labels from the source language to the target language.
Inter-Annotator Agreement (IAA)	A measure of the consistency or reliability between human annotators on the assigned labels.
Knowledge Base (KB)	A structured repository of information (like WikiData) that contains facts and relationships in the form of property-key values.
Long Tail Distribution	A statistical distribution pattern (also observed in entity mention frequencies) where a few items have very high frequencies, and many items (for example, rare entity types) have very low frequencies.

Terms	Details
Macro-Average	An averaging method that calculates the metric (Precision, Recall, F1-score) independently for each entity type and then takes the unweighted mean across all types.
Mapping	The output of the word alignment service, defining which token(s) in the source sentence correspond to which token(s) in the target sentence.
Master Dictionary	A resource created in the preprocessing step of TAFSIL framework, containing mappings of mentions and their entity types, built by leveraging the aliases and structured knowledge from Wikidata and Wikipedia.
Micro-Average	An averaging method that computes the overall metrics by aggregating the total True Positives, False Positives, and False Negatives across all entity types before applying the formulas.
Multilingual Encoder	A pre-trained neural network capable of processing and generating contextual embeddings for multiple languages.
Multilingualism	The property to process or generalize across multiple languages, often demonstrated by improved zero-shot performance on unseen languages.
Named Entity Recognition (NER)	A core natural language processing task that involves identifying and classifying text spans into specific, predefined semantic categories (e.g., Person, Organization).
Natural Language Inference (NLI)	A core natural language processing task to logically interpret natural language.
Natural Language Processing (NLP)	Also known as Computational Linguistics. The field of study concerned with the interactions between computers and human (natural) languages.
Noise-Aware Models	Models for fine-grained entity typing that explicitly cope with label noise from distant supervision, by applying techniques like attention-based denoising or label refinement.
Parallel Corpora	A collection of texts or sentences, where each sentence in the source language is aligned with its corresponding sentence in the target language with similar meaning.
Plain Sentence Format	The standard sentence representation without any Named Entity Recognition annotations or special symbols.

Terms	Details
Parts-of-Speech (POS) tagging	The process of marking up a word in a text as corresponding to a particular part of speech, based on both its definition, as well as its context.
Quality Sentence Selection	A step in TAFSIL (Stage 4) that uses a Parts-of-Speech (POS) tagger to verify if a detected entity mention has a valid noun tag and removes erroneous sentences.
Rule-based Systems	Early Named Entity Recognition systems that relied on handcrafted patterns and domain knowledge to identify specific categories of entities.
Script Similarity	The visual resemblance between the writing systems (scripts) of two different languages.
Script Similarity Enhancement	The refinement step in the CLASSER framework (Stage 2) that improves projected annotations by leveraging an auxiliary language dataset that shares the same writing system (script) as the target language.
Sentence Mismatch Detection	The cleaning process in the EaMaTa framework where the anchors are ignored in the translated entity-anchored sentence and the result is compared to the plain translated sentence.
Sequence Labeling Task	A common formulation of a natural language processing task, where a label is assigned to each word or token in a sequence.
Semi-supervised Methods	Approaches that begin with a small set of seed examples and iteratively expand coverage by learning contextual patterns.
Silver Dataset	A large-scale dataset created through automated or semi-automated processes, generally considered less reliable than a Gold dataset.
Spurious Entities	False Positive entity mentions, which are detected by the system for undefined entity types.
Supervised Methods	Machine learning approaches that are heavily dependent on annotated corpora.
TAFSIL	Taxonomy Adaptable Fine-grained Entity Recognition through Distant Supervision for Indian Languages.
Taxonomy	Predefined entity types and hierarchy.

Terms	Details
Total Entity Counts Mismatch Detection	The cleaning process in the EaMaTa framework where sentences are discarded if the total number of entities (marked by anchors) in the translated sentence does not match the count in the original source sentence.
Translation-based Approaches	Methods for multilingual data creation, where entire sentences and their annotations are translated into target languages.
Trie	A tree-like data structure used for efficient term lookup.
True Positive	An entity mention is present in the ground truth, and it is correctly detected by the system.
Type Mismatch Errors	Errors that occur when an entity mention is categorized to the wrong entity type.
Unsupervised Methods	Approaches that exploit co-occurrence statistics and distributional regularities across corpora to identify entity mentions without labeled data.
Vulnerable Languages	Languages, like Bodo, Manipuri, and Mizo, that are at risk of disappearing, often due to a limited number of speakers or lack of resources.
White Space Error	A component in the maximum edit distance formula used to accommodate variations like agglutination (conjoined words).
Word Alignment Service	A tool or model that generates a mapping between source and target language tokens.
Word-based Tokenization	The process of segmenting a sentence into individual word units.
Zero-shot Performance	Evaluating a model on a target language without any training data for that language

List of Abbreviations

Abbreviations	Details
AAAI	AAAI Conference on Artificial Intelligence
ACE	Automatic Content Extraction shared evaluation
all3	Balanced train set comprising samples from Bodo, Manipuri, and Mizo languages
all5	Balanced train set comprising samples from Assamese, Bodo, Marathi, Nepali, and Sanskrit languages
All26	Balanced train set comprising samples from Assamese, Bengali, Bhojpuri, Bishnupriya, Bodo, Chhattisgarhi, Dogri, Gujarati, Hindi, Kannada, Kashmiri, Konkani, Maithili, Malayalam, Manipuri, Marathi, Mizo, Nepali, Odia, Punjabi, Sanskrit, Santali, Sindhi, Tamil, Telugu, and Urdu languages
as	Assamese language (ISO 693-1 Language code)
BIO	A widely used labeling format where B indicates the Beginning of a span, I the Inside of a span, and O denotes tokens Outside these categories.
BIOES	A widely used labeling format where B indicates the Beginning of a span, I the Inside of a span, E the End of a span, S a Single-word span, and O denotes tokens Outside these categories.
BM	Boundary Mismatch (Entity Error Type)
bn	Bengali language (ISO 693-1 Language code)
bho	Bhojpuri language (ISO 693-2 Language code)
bpy	Bishnupriya Manipuri/ Bishnupriya language (ISO 693-3 Language code)
brx	Bodo language (ISO 693-3 Language code)
CLASSER	Cross-Lingual Annotation Projection enhancement through Script Similarity for Fine-grained Entity Recognition

Abbreviations	Details
CoNLL 2002/2003	Shared tasks focused on Named Entity Recognition and Chunking
CRF	Conditional Random Fields
D_IB	DECENT model with IndicBERTv2-MLM-Sam-TLM encoder
D_mB	DECENT model with BERT-base-multilingual-cased encoder
D_MR	DECENT model with MuRIL encoder
D_XLM	DECENT model with XLM-RoBERTa-large encoder
DECENT	Decoupled Encoding and Cross-Attention for Efficient Named Entity Typing
doi	Dogri language (ISO 693-2 Language code)
EaMaTa	Entity-anchored Machine Translation
en	English language (ISO 693-1 Language code)
en†	FewNERD dataset (English)
Ent	Number of Entities
FgNER	Fine-grained Named Entity Recognition
FIGER	Fine-grained Entity Recognition
FiNE-MiBBiC	Fine-grained Named Entity Recognition for Mizo, Bhojpuri, Bishnupriya Manipuri and Chhattisgarhi
FiNERVINER	Fine-grained Named Entity Recognition dataset for Vulnerable languages of India's North Eastern Region
FIRE-2014	Forum for Information Retrieval Evaluation (included NER for Tamil and Malayalam)
gom	Konkani language (ISO 693-2 Language code)
gu	Gujarati language (ISO 693-1 Language code)
HAnDS	Heuristics Aligned with Distant Supervision
HAREM	Shared task for NER in Portuguese
hi	Hindi language (ISO 693-1 Language code)

Abbreviations	Details
HMM	Hidden Markov Models
hne	Chhattisgarhi language (ISO 693-3 Language code)
HUB-4	Shared task following MUC-6
IAA	Inter-Annotator Agreement
IB	IndicBERTv2 (‘IndicBERTv2-MLM-Sam-TLM‘)
IO	A widely used labeling format where I indicates Inside of a span and O denotes tokens Outside these categories.
IREX	Information Retrieval and Extraction shared task
IJCNLP-2008	International Joint Conference on Natural Language Processing shared task (focused on Indian languages Named Entity Recognition)
IJCNLP-AACL	International Joint Conference on Natural Language Processing and the Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics
KB	Knowledge-Base
kn	Kannada language (ISO 693-1 Language code)
ks	Kashmiri language (ISO 693-1 Language code)
L_mD	LITE model with mDeBERTa-v3-base-mnli-xnli encoder
L_XLM	LITE model with XLM-RoBERTa-large-xnli encoder
LITE	Language Inference based Typing of Entities
LLMs	Large Language Models
Lng	Language
LREC	Language Resources and Evaluation Conference
lus	Mizo/ Lushai language (ISO 693-2 Language code)
mai	Maithili language (ISO 693-2 Language code)
mBERT	Multilingual BERT (‘bert-base-multilingual-cased’)
MCN2	MultiCoNER2 Taxonomy or MultiCoNER2 dataset

Abbreviations	Details
MEMM	Maximum Entropy Markov Models
MISC	Stands for Miscellaneous (used in sequence labeling examples).
ml	Malayalam language (ISO 693-1 Language code)
mni	Manipuri/ Meitei language (ISO 693-2 Language code)
mr	Marathi language (ISO 693-1 Language code)
MT	Machine Translation
MUC-6	Message Understanding Conference-6 (1996)
MuRIL	Multilingual Representation for Indian Languages (‘muril-large-cased‘)
MultiCoNER	Multilingual Complex Named Entity Recognition
ne	Nepali language (ISO 693-1 Language code)
NER	Named Entity Recognition
NLI	Natural Language Inference
NLP	Natural Language Processing
NN	Noun (Part-of-speech tag)
NNP	Proper Noun (Part-of-speech tag)
or	Odia language (ISO 693-1 Language code)
pa	Punjabi language (ISO 693-1 Language code)
PER	Stands for Person, an entity type
PID	Wikidata Identifier for a property (e.g., P31 for ‘instance of’)
PLM	Pre-trained Language Models
POS	Part-of-Speech or Parts-of-Speech
QID	Wikidata Identifier for an item (e.g., Q62 for San Francisco)
sa	Sanskrit language (ISO 693-1 Language code)
sat	Santali language (ISO 693-2 Language code)

Abbreviations	Details
sd	Sindhi language (ISO 693-1 Language code)
SemEval	Semantic Evaluation shared task
Sent	Number of Sentences
SIGIR	ACM SIGIR Conference on Research and Development in Information Retrieval
SOTA	State-of-the-Art
SE	Spurious Entity (Entity Error Type)
SVM	Support Vector Machines
ta	Tamil language (ISO 693-1 Language code)
TAFSIL	Taxonomy Adaptable fine-grained entity recognition through distant supervision for Indian languages
TALLIP	The ACM Transactions on Asian and Low-Resource Language Information Processing
te	Telugu language (ISO 693-1 Language code)
TFSL-M	TAFSIL dataset in MultiCoNER2 taxonomy
tgt	Target language
Tkn	Number of Tokens
TM	Type Mismatch (Entity Error Type)
UFET	Ultra-Fine Entity Typing
ur	Urdu language (ISO 693-1 Language code)
XL	XLM-RoBERTa
XLM-RoBERTa	Cross-lingual Language Model - RoBERTa ('XLM-RoBERTa-large')
XSTS	Crosslingual Semantic Text Similarity

List of Symbols

Symbol	Details
aux	Auxiliary language
$\mathcal{A}(s_{src}, s_{tgt})$	Annotation alignment service producing a mapping map or *map*
α_j	Start anchor symbol for the j-th token
β_{je}	End anchor symbol for the j-th token of entity type e
C	Linked corpus (Wikipedia) (Input to TAFSIL framework)
D_l	Fine-grained Named Entity Recognition dataset of size N in language l
D_{aux}	Fine-grained Named Entity Recognition dataset of size N in auxiliary (aux) language
D_l	Fine-grained Named Entity Recognition dataset of size N in language l
D_l	Fine-grained Named Entity Recognition dataset of size N in language l
D_{tgt}^{ref}	Final refined target Fine-grained Named Entity Recognition dataset, $D_{tgt}^{ref} = \{(s_{tgt}^{(i)}, \bar{y}_{tgt}^{(i)})\}_{i=1}^{N}$
$d(j)$	Maximum edit distance for token j, used as the empirical formula for the maximum edit distance for fuzzy matching in stage 3 of TAFSIL framework. $d(j) = 1 + white\_space(j) + length(j)/6$
e_d(j, t)	Edit distance between token j and term t
e	Entity type
E	Set of entity types
$\mathbf{EB}_{src}$	Contextual embeddings for the source sentence s_{src}
$\mathbf{EB}_{tgt}$	Contextual embeddings for the target sentence s_{tgt}

Symbol	Details
F1	**F1-score**: $$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$
$f(\hat{y}_{tgt}^{(j)}, \ddot{y}_{tgt}^{(j)})$	Refinement function, determines the final annotation for token j
h	Hyperlink in a sentence
H	Set of hyperlinks
j	Individual token of a sentence s
J	Set of tokens of a sentence s
k_i	Key in Wikidata (W)
κ	Cohen's kappa coefficient (Inter-Annotator Agreement)
m	Mention detected in sentence s
M_s	Set of mentions in sentence s
mapping	Alignment mapping from source-sentence tokens to target-sentence tokens
$mapping^{-1}(j)$	The inverse mapping, giving the source token index for a target token j
MD	Master dictionary of mentions and entity types
ME	Pretrained multilingual encoder (for projection)
ME_{aux}	Multilingual encoder fine-tuned on D_{aux} (for Stage 2 refinement)
N	Number of sentences per dataset
NN	Noun POS tag
NNP	Proper Noun POS tag
PC	Parallel corpora of size N having source-to-target language aligned-sentence pairs, $PC = \{(s_{src}^{(i)}, s_{tgt}^{(i)})\}_{i=1}^{N}$
P	**Precision**: $$Precision = \frac{True\,Positive}{True\,Positive + False\,Positive}$$

Symbol	Details
p_i	Property in Wikidata (W)
PK	Set of property-key values
POS(j)	Part-of-speech tag of token j
$prob_{aux}(j)$	The annotation probability distribution for the j-th token in s_{tgt} after refinement using ME_{aux}
$prob^{max}_{aux}(j)$	Maximum probability of $\ddot{y}^{(j)}_{tgt}$ for j-th token
R	**Recall**: $$Recall = \frac{True\,Positive}{True\,Positive + False\,Negative}$$
S4	Dataset created after Stage 4 of TAFSIL framework
¬S4	Dataset created without Stage 4 of TAFSIL framework
s	Sentence
src	Source language
$s^{(i)}_l$	i-th sentence in language l
S_{src}	Source dataset sentences in BIO format
s^{anchor}_{src}	Source sentence in entity-anchored format
s^{plain}_{src}	Source sentence in plain format
S^{clean}_{tgt}	Clean Fine-grained Named Entity Recognition dataset in target language tgt
s^{anchor}_{tgt}	Translated entity-anchored sentence
s^{plain}_{tgt}	Translated plain sentence
$\bar{s}^{\mathbf{plain}}_{tgt}$	Translated entity-anchored sentence converted to plain sentence after ignoring the anchors
T	Trie for mention lookup
t	Term in Trie T
tgt	Target language
τ	Confidence Threshold (set to 0.85)

Symbol	Details
$\mathcal{TLT}$	Translation service
V	Mapping of property values to entity types
W	Knowledge base (Wikidata)
$y_l^{(i)}$	Fine-grained Named Entity Recognition annotations for tokens of the i-th sentence in language l
$y_{tgt}^{(j)}$	Annotation of the token j in the target sentence
$\ddot{y}_{tgt}$	Auxiliary annotation sequence for s_{tgt} (from ME_{aux})
$\hat{y}_{tgt}$	Initial annotation sequence for s_{tgt} (from Stage 1)
$\bar{y}_{tgt}$	Final refined annotation sequence
$\sum(m \in s_{src}^{anchor})$	Total count of entity mentions in the source anchored sentence
$\sum(m \in s_{tgt}^{anchor})$	Total count of entity mentions in the target anchored sentence
$\uparrow \Delta\%$	Percentage increase (relative improvement in F1 score)
$\uparrow \%$	Percentage increase (relative improvement in F1 score)
$\downarrow \Delta\%$	Percentage decrease (relative reduction in F1 score)
$\oplus$	Used in ablation study tables to denote refinement/enhancement using an auxiliary language

"Data is potentially the most undervalued and deglamorized aspect of today's AI ecosystem."

Aroyo, Lease, Paritosh, & Schaekermann (2022) [1]

1

Introduction

The pursuit of knowledge is a fundamental and enduring human endeavor. Over millennia, written information has served as a primary mechanism for storing and disseminating knowledge, leading to the development of diverse scripts and natural languages globally. Key historical milestones, notably the invention of the printing press in the fifteenth century and the subsequent technical revolution of the twentieth century (driven by personal computers and computer networks), have significantly accelerated the growth of knowledge sharing in the form of natural language text. However, the wealth of unstructured information contained within these texts poses a persistent challenge: efficiently accessing structured knowledge for computational algorithms. This difficulty has necessitated substantial research within Natural Language Processing (NLP), also known as Computational Linguistics.

Named Entity Recognition (NER) is a core NLP task that involves identifying and classifying text spans into specific, predefined semantic categories [2]. For instance, in this sentence, *Jude Bellingham scored a goal for Real Madrid.*, an NER system would identify *Jude Bellingham* as a *Person* and *Real Madrid* as an *Organization*. These detected and classified text spans are known as *Named Entities*.

To extract more sophisticated and insightful knowledge, a fine-grained classification is essential. For example, in these two sentences, *Jude Bellingham scored a goal in Santiago Bernabéu. Santiago Bernabéu became president of Real Madrid in 1943.*, a fine-grained system would classify *Jude Bellingham* as *person/player*, *Real Madrid* as an *organization/sports_club*, and contextually differentiate *Santiago Bernabéu* in the first sentence as a *location/stadium*, which is named after the *person/president* mentioned in the second sentence. One of the most used entity type hierarchies was proposed by Ling and Weld (2012) [2] in which 113 fine types were defined at two levels. As shown in Figure 1.1, the first-level types are *person*, *organization*, *location*, *product*, *building*, *art*, and *event*. The second level types are fine-grained types such as *actor*, *architect*, *coach*, etc., which are hierarchically defined under the *person* type. Several other contributions are made in Fine-grained Named Entity Recognition (FgNER), especially for the English language, which are discussed in detail in section 2.3.1 in Chapter 2. The extraction of such structured knowledge via FgNER is crucial for powering numerous downstream NLP applications, including question answering, machine translation, information retrieval, relation extraction, automatic text summarization, opinion mining, and knowledge base construction [3, 4, 5].

person	doctor	**organization**	terrorist_organization
actor	engineer	airline	government_agency
architect	monarch	company	government
artist	musician	educational_institution	political_party
athlete	politician	fraternity_sorority	educational_department
author	religious_leader	sports_league	military
coach	soldier	sports_team	news_agency
director	terrorist		

Figure 1.1: Examples of fine-grained entity types proposed by Ling and Weld (2012) [2]. The **coarse-grained** types are in **bold**, and the other types within each box are the corresponding fine-grained types belonging to that coarse-grained type.

1.1 Motivation

Despite its importance, NER remains an extremely challenging task. Early research in the 1990s focused on utilizing language-specific syntactic [6] and lexical features (e.g., capitalizing English words) [7], and rule-based systems [8]. The inherent lack of scalability and adaptability across different languages and domains led researchers to adopt machine learning techniques [9]. While supervised learning systems [10] achieved encouraging performance, they were hampered by the requirement for extensive, costly, and time-consuming annotated training data [11], often necessitating the involvement of domain experts and linguists [4, 8]. To address these limitations, researchers explored unsupervised [12, 13] and semi-supervised [14, 15] learning algorithms to automatically recognize entities from large volumes of available unlabeled data, such as Wikipedia [2], Freebase [16], and news articles [17]. More recently, distantly supervised [18, 19] and deep learning-based [20, 21] systems have reduced the dependence on manual annotation; however, these approaches still face limitations in the quality of weakly labeled datasets due to the presence of noise and bias [22, 23]. Some other challenges in NER task include: data scarcity in low-resource languages (e.g. vulnerable languages such as Bodo and Manipuri) and in some domains (e.g. confidential defense-related data), ambiguity in entity mentions (e.g. *Santiago Bernabéu* may be a *person* or *location* depending on the context), boundary detection errors (e.g. *Real Madrid* is an *organization* but *Madrid* is a *location*), nested entities (e.g. *Colon Cancer* is a *disease* which contains the nested entity mention *Colon* as a *body_part*), complex entities (e.g. *Uncoupled* is a verb form but also an *art/movie* name) and so on [3, 24]. With the expansion of social media and messaging platforms, some new challenges are emerging in terms of informal and misspelled texts, along with newly generated unseen entities [25]. Considering these limitations and challenges, this research is dedicated to addressing key issues in Multi-Lingual Fine-Grained Named Entity Recognition.

1.2 Research Gaps

- **Data scarcity in low-resource languages:** Fine-grained NER datasets and models have been limited to English and other high-resource languages [26]. The only available Fine-grained NER dataset in Indian languages is MultiCoNER2 dataset [3] in Hindi and Bengali.

- **Noise in annotated datasets:** Automatic dataset construction methods based on heuristics and distant supervision often introduce task-dependent noise patterns [22]. For example, Gillick et al. (2014) [27] observed that false-positive labels constitute a major source of noise in automatically generated FgNER training data.

- **Scalability across taxonomies:** A key research gap in FgNER lies in developing scalable methods that can generalize across diverse domains and evolving taxonomies, as existing approaches typically assume prior knowledge of the target domain and label set for a given test instance, whereas entity typing without such assumptions remains a largely unexplored problem [28].

1.3 Contributions

To address the challenges of data scarcity, label noise, and taxonomy rigidity in Indian languages, this thesis presents a series of progressively refined frameworks for Multi-Lingual Fine-Grained Named Entity Recognition. The primary contribution of this thesis is the development of scalable methodologies, including distant supervision, annotation projection and refinement, and entity-anchored translation. Collectively, these contributions provide the NLP community with over 30 million tokens of annotated FgNER dataset across 26 Indian languages in five different taxonomies, offering significant improvements in F1-score performance over existing state-of-the-art baselines. The contributions are demonstrated in the Figure 1.2 and briefly described in the following sub-sections.

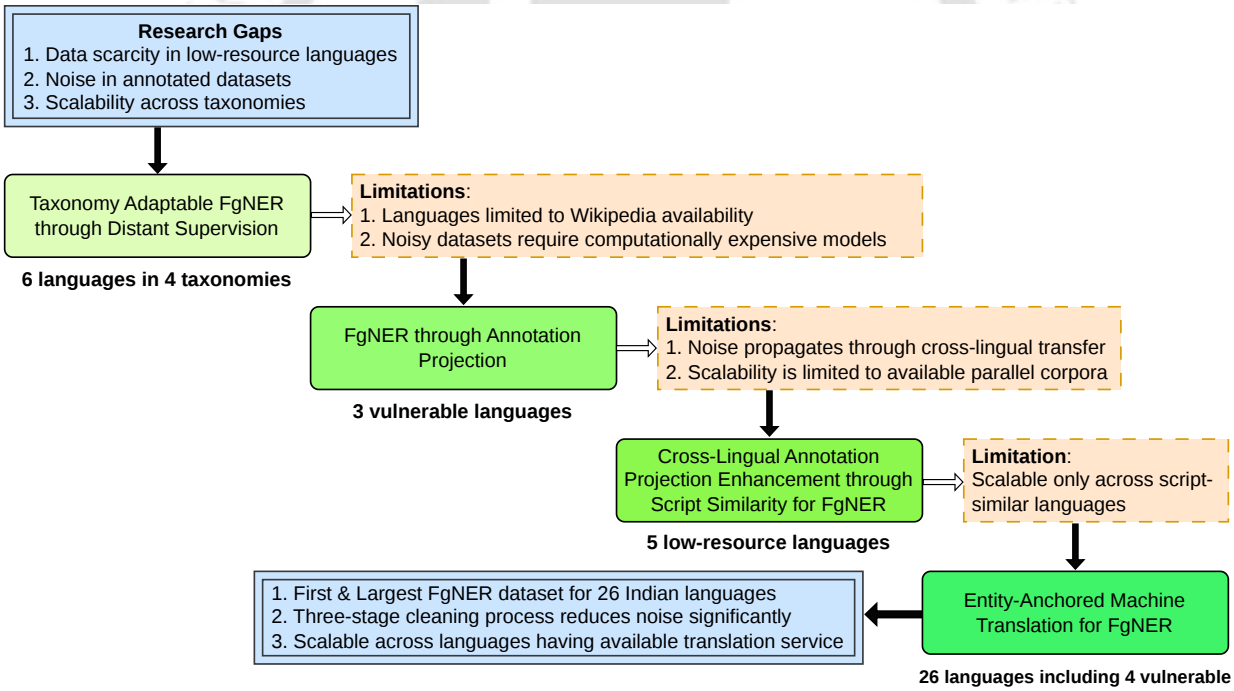

Figure 1.2: Identified Research Gaps, corresponding contributions & their inter-connections.

1.3.1 Taxonomy Adaptable Fine-grained Named Entity Recognition through Distant Supervision

The first major contribution of this thesis is the Taxonomy Adaptable Fine-grained Named Entity Recognition through Distant Supervision framework: TAFSIL [29]. The distant supervision method employs external knowledge bases to automatically label corpora [22]. While distant supervision is an effective way to mitigate manual annotation costs, existing methods are often restricted

to a single taxonomy and tailored for English. TAFSIL framework addresses this by leveraging the interlinkage between the knowledge base WikiData and the linked corpora Wikipedia through multi-stage heuristics and fuzzy matching. This framework was utilized to create FgNER datasets for six Indian languages: *Hindi, Marathi, Sanskrit, Tamil, Telugu,* and *Urdu*, across four distinct taxonomies: *FIGER* [2], *OntoNotes* [27], *HAnDS* [22], and *MultiCoNER2* [3]. The experiments demonstrate a relative improvement of 83% in average F1-score over zero-shot performance, confirming the robustness of the framework.

1.3.2 Fine-grained Named Entity Recognition through Annotation Projection

While the TAFSIL framework effectively utilizes the existing knowledge base and linked corpora, its scalability is inherently limited by the availability of Wikipedia in the target languages (Figure 1.2). For extremely low-resource and vulnerable languages such as *Bodo, Manipuri,* and *Mizo*, Wikipedia does not exist, limiting the application of the TAFSIL framework for the FgNER dataset creation. To bridge this gap, annotation projection is employed from the available source FgNER dataset to the target languages by utilizing source-to-target language parallel corpora. A multilingual encoder-based projection tool is used to create a significant resource comprising over 282k annotated entity mentions across each of these three vulnerable languages [30]. This methodology enables the development of FgNER systems for linguistic communities that lack the structured linked corpora, which are required by previous distant supervision techniques. Moreover, this work extensively discusses cross-lingual zero-shot FgNER analyses, highlighting the importance of script similarity in FgNER and showing how it can be leveraged to reduce noise and improve dataset quality.

1.3.3 Cross-Lingual Annotation Projection Enhancement through Script Similarity for Fine-grained Named Entity Recognition

Although annotation projection provides a viable solution for dataset creation for low-resource languages, the process is often hampered by alignment errors and noise during the transfer from the source language to the target language. To improve the quality of these projected labels, a framework for the enhancement of annotation through script similarity is proposed [31]. In addition to the annotation projection used in the previous contribution, this framework introduces a second stage of refinement that leverages script similarities among Indian languages. By refining projected annotations using datasets from script-similar languages, the proposed framework achieves substantial performance gains, including a 26% and 46% improvement in F1-score for *Marathi* and *Sanskrit* over the first contribution TAFSIL. The framework is successfully applied to generate a substantial dataset of over 1.8 million sentences for five low-resource Indian languages, including *Assamese*, *Marathi*, *Nepali*, *Sanskrit*, and vulnerable language *Bodo*. This contribution demonstrates that linguistic commonalities can be systematically exploited to correct the inherent noise of cross-lingual projection.

1.3.4 Entity-Anchored Machine Translation for Fine-grained Named Entity Recognition

Despite the refinement of projected annotations through script-similarity yields a very high-quality dataset, its scalability remains restricted to the availability of FgNER datasets in languages written using the same script. To achieve a comprehensive and high-quality resource, the Entity-Anchored Machine Translation (EaMaTa) framework is proposed [32]. This framework leverages the largest manually annotated English FgNER dataset, FewNERD [33], and translates it into 26 Indian languages. The entity mentions are anchored with special symbols, and the three-stage cleaning

process proposed in this framework preserves the crucial information after the translation process. This framework mitigates the limitations of the lack of FgNER datasets in languages written using different scripts and the noise induced by the distant supervision framework. The resulting dataset covers 26 languages spoken by over two billion people across the Indian subcontinent and demonstrates a 9% increase in F1-score compared to the current state-of-the-art. The proposed framework and the generated dataset provide significant resources for multilingual fine-grained named entity recognition.

1.4 Organization of the Thesis

The thesis is organized as follows:

- Chapter 2 is dedicated to the background and literature survey.
- Chapter 3 describes the proposed "Taxonomy Adaptable Fine-grained Named Entity Recognition through Distant Supervision framework", and its utilization to create 24 FgNER datasets, for six low-resource Indian languages in four different taxonomies.
- Chapter 4 illustrates the "Fine-grained Named Entity Recognition through Annotation Projection" for creating datasets for three vulnerable languages.
- Chapter 5 demonstrates the proposed framework "Cross-Lingual Annotation Projection Enhancement through Script Similarity for Fine-grained Named Entity Recognition" and the corresponding generated dataset in five low-resource Indian languages.
- Chapter 6 elaborates on the highly efficient framework "Entity-Anchored Machine Translation for Fine-grained Named Entity Recognition" and its application in creating high-quality FgNER datasets in 26 Indian languages, including four vulnerable languages.
- Chapter 7 concludes the thesis and discusses future work.

2

Background and Literature Survey

This chapter presents the foundational concepts underlying Named Entity Recognition (NER) and Fine-grained Named Entity Recognition (FgNER), along with a comprehensive survey of relevant literature. It begins by formalizing the NER task, its key components, labeling schemes, and historical evolution, followed by an overview of multilingual NER with a particular focus on Indian languages. The chapter then discusses the progression from coarse-grained to fine-grained entity typing, existing datasets, methodologies for dataset construction, and noise-aware learning strategies. Finally, it outlines standard evaluation metrics and provides linguistic and demographic context for the Indian languages considered in this research, thereby establishing the background necessary for the subsequent chapters.

2.1 Named Entity Recognition (NER)

Named Entity Recognition (NER) is a core NLP task that involves identifying and classifying text spans into specific, predefined semantic categories [2]. The text spans are known as Named Entities. Using the illustration in the text box, several important concepts related to NER are explained below:

Michael Phelps from New York is diagnosed with Marfan Syndrome.

Entity refers to an object that exists within a defined universe. For example, Marfan Syndrome is a *Disease* category within the biomedical domain.

Named Entity is a span of text that represents the name of a predefined type, such as person, organization, or location. For instance, Michael Phelps denotes the name of a *Person*, and Marfan Syndrome is the name of a *Disease*.

Entity Detection focuses on locating spans that correspond to predefined categories or types. For example, New York is treated as a *Location* in this setting, while New as a standalone noun is not classified. Moreover, detecting only York would be erroneous as it is the name of a different valid *Location*. Thus, detecting New York is appropriate, but detecting New or York alone would be incorrect.

Entity Typing assigns specific types or categories to the detected entity mentions. Depending on the requirements or definitions, New York may be typed as *Location*.

Entity Recognition involves performing Entity Detection and Entity Typing together.

NER is commonly formulated as a sequence labeling task, similar to Part-of-Speech (POS) tagging and Chunking [34]. Nonetheless, some approaches reformulate NER as a question-answering task, where the unstructured text and the predefined entity categories serve as questions, and the system-predicted entity tags act as answers [25].

In practice, the most widely used labeling formats include IO, BIO, BIOES, etc [35]. Here, B indicates the Beginning of a span, I the Inside of a span, E the End of a span, S a Single-word span, and O denotes tokens outside these categories. The following example illustrates how these schemes apply:

Example:	Michael	Clarke	Duncan	starred	in	Armageddon	.
IO labeling	I-PER	I-PER	I-PER	O	O	I-MISC	O
BIO labeling	B-PER	I-PER	I-PER	O	O	B-MISC	O
BIOES labeling	B-PER	I-PER	E-PER	O	O	S-MISC	O

Here, PER stands for *Person* and MISC stands for *Miscellaneous* entity types. Thus, Michael, Clarke, and Duncan receive Inside-*Person*, Inside-*Person*, Inside-*Person* tags under IO labeling, Begin-*Person*, Inside-*Person*, Inside-*Person* under BIO labeling, and Begin-*Person*, Inside-*Person*, End-*Person* under BIOES labeling, respectively. Armageddon is a movie title, labeled as Inside-*Miscellaneous* in IO labeling, Begin-*Miscellaneous* in BIO labeling, and Single-*Miscellaneous* under BIOES labeling, respectively. All other tokens are labeled as O. In this thesis, BIO labeling is used.

2.1.1 Named Entity Recognition: Overview of Existing Works

Research on Named Entity Recognition (NER) began in 1991 with early rule-based systems that relied on handcrafted patterns and domain knowledge to identify specific categories of entities, such as company names [6]. The field expanded rapidly after the Message Understanding Conference-6 (MUC-6) in 1996, where the term "Named Entity" was popularized and task definitions were standardized [7]. Subsequent shared tasks and evaluations: HUB-4 [36], IREX [37], CoNLL 2002/2003 [38], ACE [39], HAREM [40] etc., catalyzed progress by providing common benchmarks and broadening coverage in terms of domains, entity types, and languages. These initiatives helped move NER from a narrow, English-centric task on newswire text toward a more general technology applicable to email, biomedical text, social media, and e-commerce [8].

Early NER systems were mostly rule-based and feature-intensive supervised methods [8]. Rule-based approaches exploited lexical, syntactic, and orthographic regularities as well as handcrafted domain-specific rules, often achieving strong performance in restricted domains but at the cost of extensive manual effort and limited portability across domains and languages [4, 41]. With the advent of machine learning, supervised models such as Hidden Markov Models (HMM) [42], Conditional Random Fields (CRF) [43], Support Vector Machines (SVM) [44], and Maximum Entropy Markov Models (MEMM) [45] emerged as the promising paradigm. These systems leveraged rich feature sets, including lexicon-based, corpus-level, and word-level features, but remained heavily dependent on manually annotated corpora, making large-scale and multilingual development expensive.

To mitigate annotation costs, semi-supervised [14, 15] and unsupervised [12, 13] approaches were explored. Semi-supervised methods started from a small set of seed examples and iteratively

expanded coverage by learning contextual patterns and annotating new instances, thereby reducing manual labeling requirements. Unsupervised methods drew on association measures and clustering techniques, exploiting co-occurrence statistics and distributional regularities across corpora to identify entity mentions without labeled data. These approaches demonstrated the feasibility of NER in settings where annotated resources are scarce [5].

Collobert et. al, [46] introduced an early deep learning-based NER architecture using word embeddings, convolutional context encoding, and a CRF tagging layer. Later approaches incorporated character-level encoding and Long Short Term Memory (LSTM)-based sequence modeling that improved contextual understanding, handling of misspellings, and overall NER performance [47].

2.2 Multilingual Named Entity Recognition

Although early research on entity extraction initially focused on English, the scope of this research soon broadened to include many other languages, either through independent efforts or within multilingual paradigms. From 1992 onward, a significant body of work began to concentrate on Chinese [48]. Subsequently, research activity expanded to Portuguese [49] in 1997, followed by Japanese [10], Italian [50], and Swedish [51] in 1998, and then to Greek, Hindi, Romanian, and Turkish [15] in 1999. Further progress was observed with studies on French [9] in 2001, Spanish and Dutch [38] in 2002. During the period of 2003–2004, several additional languages attracted research attention, including German [52], Basque [11], Bulgarian [53], Catalan [54], Cebuano [55], Danish [56], Korean [11], Polish [57], Russian [58], and so on. In recent years, researchers have employed distant supervision techniques for NER, enabling the development of Polyglot NER [16] for 40 languages in 2015 and further extending it to 282 languages through WikiANN [17] in 2017. Although a few Indian languages were covered in these resources, the size of such datasets was extremely small. Early efforts specifically for Indian languages included the IJCNLP-2008 shared task, which provided coarse-grained NER datasets for Hindi, Bengali, Oriya, Telugu, and Urdu [59]. This dataset became a key benchmark for early Hindi NER research [60, 61, 62, 63, 64]. FIRE-2014 [65] subsequently introduced datasets for Tamil and Malayalam, further enhancing the resource landscape for Indian languages.

Several methodologies have been adopted to generate NER datasets in various Indian languages. Some of the significant and widely adopted methodologies are:

2.2.1 Distant Supervision for Multilingual NER

At a broader scale, multilingual silver-standard resources such as Polyglot NER [16] and WikiANN [17] substantially increased language coverage. Polyglot NER covered 40 major languages, including some Indian languages, by leveraging Wikipedia and heuristics [16]. WikiANN extended this coverage to 282 languages, including 16 Indian languages, using article titles and heuristic rules to derive span-level annotations [17]. While these resources enabled large-scale multilingual NER experimentation, their automatic construction led to issues with noise, domain mismatch, and inconsistent label quality. Moreover, the number of samples in these datasets in Indian languages was extremely limited.

2.2.2 Annotation Projection for Multilingual NER

Annotation projection using parallel corpora and word alignment is a long-standing technique. Pioneering work by Yarowsky and Ngai [66] demonstrated how annotations in a source language

(typically English) can be projected to a target language using alignment links in parallel texts. This idea has since been applied to a wide range of tasks, including dependency parsing [67], relation extraction [68], and NER [69, 70, 71, 72, 73, 74, 75, 76]. Projection-based initiatives have also been instrumental. Naamapadam [75] is one of the largest NER datasets for 11 Indian languages by projecting English NER annotations onto target languages using Samanantar [77] parallel corpora and alignment techniques [78, 79]. However, cross-lingual transfer in NER faces additional linguistic and typological challenges. Zero-shot transfer performance generally degrades for structurally distant language pairs and for languages with scripts that diverge from those of high-resource source languages [80, 81]. Multilingual pre-trained encoders such as mBERT [82], XLM-RoBERTa [83], MuRIL [84], and IndicBERT variants [85] provide strong baselines, but they do not fully resolve disparities, particularly for languages that are absent or severely underrepresented in pretraining corpora. In practice, best-performing systems combine multilingual PLMs with language-specific resources (e.g., parallel corpora, projected annotations, or domain-specific corpora) and architectural choices that explicitly handle label noise and transfer issues.

2.2.3 Machine Translation for Multilingual NER

Machine Translation (MT) provides a complementary route to multilingual data creation. Instead of word-by-word projection, entire English sentences and their annotations are translated into target languages, with span boundaries and labels transferred accordingly. MultiCoNER-1 [24] and MultiCoNER2 [3] relied heavily on translation-based methods to construct datasets in German, Chinese, Hindi, and Bengali. Translation-based approaches are relatively straightforward to implement given strong MT systems, but they introduce risks related to translation error, translationese, and drift in span boundaries, especially for morphologically rich or syntactically divergent languages.

2.2.4 Manual Annotation for Multilingual NER

To address quality and domain issues, several manually annotated datasets have been developed for individual Indian languages. Examples include HiNER for Hindi [34], AsNER for Assamese [86], MahaNER for Marathi [87], EverestNER for Nepali [88], and language-specific NER corpora for Bengali [89], and Telugu [90]. Additional efforts target under-resourced languages such as Bishnupriya Manipuri [91, 92], Bodo [93], Bhojpuri [94], and Maithili [95]. These manually curated datasets provide higher-quality annotations and are crucial for fair evaluation and model development in low-resource settings.

2.3 Fine-grained Named Entity Recognition

The most commonly used entity types in coarse-grained NER are *Person*, *Location*, and *Organization*, also known as *Enamex* [7, 8]. In 2002, for the first time, a multi-level entity type hierarchy was defined by Sekine and Nobata [96]. In this work, many levels of hierarchy were considered, such as *Fish* is a type of *Vertebrate*, which is a type of *Animal*, which is under the hierarchy of *Natural_Object*, and finally under the top-level hierarchy *Name*. According to the recently introduced MultiCoNER2 taxonomy [3], which is illustrated in Figure 2.1, the first-level entity types include *Person*, *Location*, and *Product*, among others. The second-level fine-grained entity types include *Athlete*, *Artist*, *Scientist*, etc. under the *Person* type hierarchy, *Food*, *Clothing*, *Vehicle*, etc. under the *Product* type hierarchy, and so on.

Example:	Michael Phelps	won 8	gold medals	at the	2008 Summer Olympics
Coarse-grained NER:	*person*				*miscellaneous*
Fine-grained NER:	*person/athlete*		*award*		*event/sports_event*

Based on the context of a sentence, the fine-grained type assigned to an entity mention may vary. For example, according to the FIGER taxonomy [2], the fine-grained type of New York is *location/city*. However, in the example sentence "Michael Phelps from New York is diagnosed with Marfan Syndrome.", presented in section 2.1, Michael Phelps is annotated as *person* only, and not as *person/athlete*, because the sentence does not discuss his profession but focuses solely on his health condition. In contrast, in the example sentence "Michael Phelps won 8 gold medals at the 2008 Summer Olympics.", the entity mention Michael Phelps is typed as *person/athlete*, as the context of the sentence pertains to sports.

2.3.1 Fine-grained Named Entity Recognition: Overview of Existing Works

In parallel with the methodological developments in NER research, the conceptualization of hierarchical entity types evolved from coarse-grained categories (Person, Location, Organization) to fine-grained taxonomies. Sekine and Nobata [96] introduced one of the earliest multi-level taxonomies, defining 150 entity types organized in a hierarchical structure. Ling and Weld [2] further advanced fine-grained NER (FgNER) by proposing a two-level hierarchy with 113 entity types. Subsequent datasets expanded both the number and granularity of types: ACE with 52 types [39], BBN with 93 types [97], OntoNotes with 88 types [27], and HYENA with 505 types [98]. Large-scale resources such as WikiSense [99], FINET [100], TypeNet [101], and UFET [23] extended coverage to thousands of types, enabling ultra fine-grained entity typing. Abhishek et. al. [22] introduced 118 entity types and improved the annotation quality by applying language-specific heuristics and sentence filtering methodologies. Manually annotated resources like FewNERD [33], with 66 entity types, further enriched the space of fine-grained named entity recognition datasets. In the SemEval2023 task, MultiCoNER2 [3] introduced a dataset with 33 entity types, extending fine-grained named entity recognition to 12 languages (Figure 2.1).

2.4 Multilingual Fine-grained Named Entity Recognition

Multilingual fine-grained NER is comparatively less explored, but recently it has gained attention through the SemEval2023 shared task [102]. MultiCoNER2 [3] introduced multilingual FgNER benchmarks with 33 fine-grained entity types across 12 languages. It is the first FgNER dataset in any Indian language, which was created by translating the English fine-grained NER dataset to Hindi and Bengali. The leading teams in the shared task, such as USTC-NELSLIP [103], PAI [104], DAMO-NLP [105], and IXA/Cogcomp [106], further advanced the dataset generation techniques and resources for these languages. Apart from these, many low-resource and vulnerable languages still lack fine-grained, high-quality annotations, leading to the interest in distant supervision, annotation projection methodologies, and translation-based approaches as key strategies to bridge the resource gap.

2.5 Challenges in Fine-grained Named Entity Recognition

The challenges for Fine-grained Named Entity Recognition (FgNER) have diversified along two major dimensions: strategies for dataset construction and noise-aware learning methods.

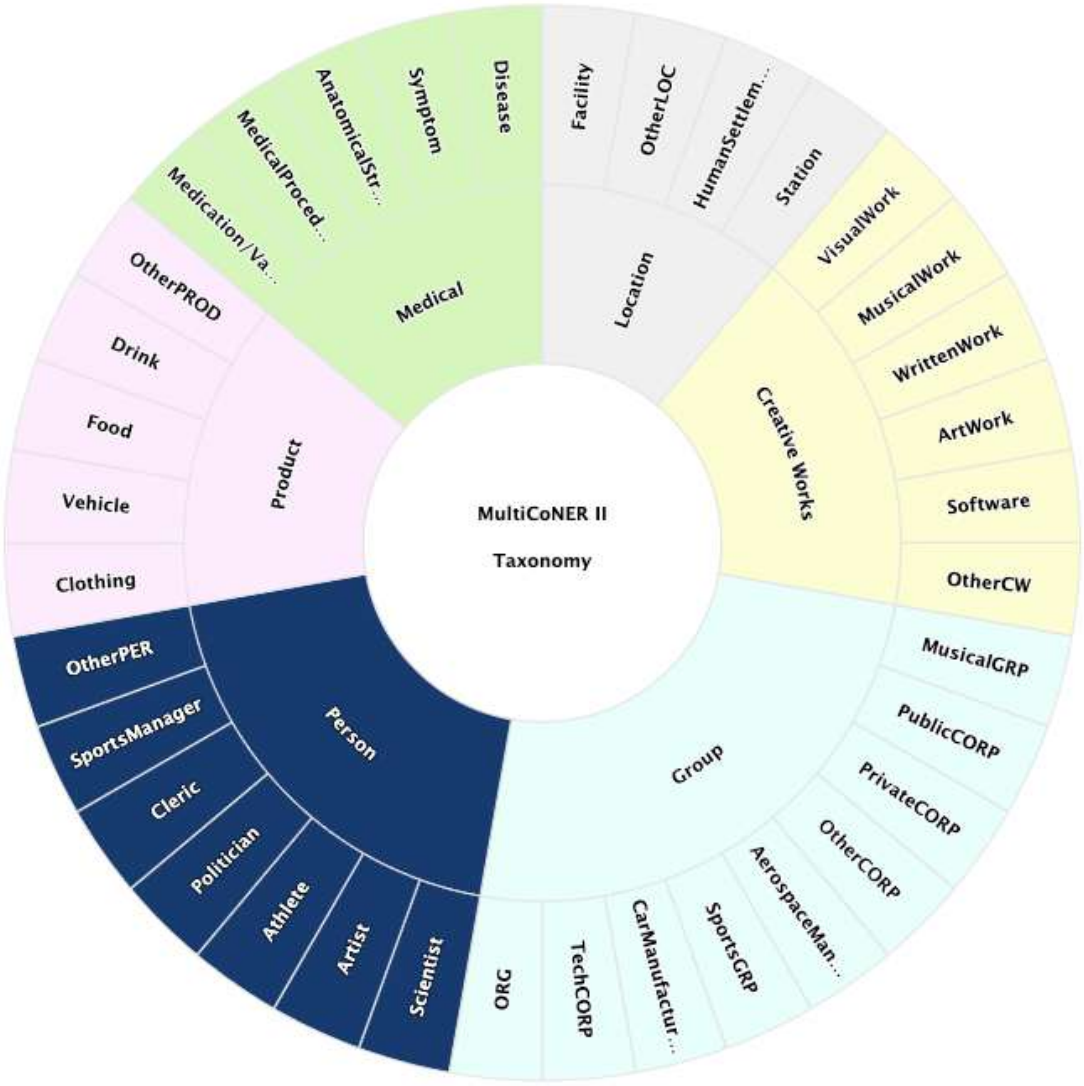

Figure 2.1: Fine-grained Entity Types in MultiCoNER2 Taxonomy [3]. The inner circle depicts the coarse-grained type & outer circle demonstrates the corresponding fine-grained types

2.5.1 FgNER Dataset Creation

A central distinction in dataset construction is between manually annotated corpora and distantly supervised resources. Manually annotated datasets, such as FewNERD [33], offer relatively clean labels but require substantial annotation effort. Distant supervision, as utilized in FIGER [2], OntoNotes [27], HYENA [98], UFET [23], and HAnDS [22] scales more readily by linking textual mentions in linked corpora (e.g, Wikipedia) to knowledge-base (e.g., Freebase or WikiData) entries and inheriting corresponding type information. However, knowledge-base (KB) based labeling often produces noisy, multi-type supervision: a mention may be associated with multiple KB types, only a subset of which are contextually appropriate. This mismatch between distant labels and context-salient types introduces label noise that must be explicitly addressed during training.

2.5.2 Noise Aware Models for FgNER

To cope with the noise induced by distant supervision methods, a rich line of work has developed noise-aware models for fine-grained and ultra-fine entity typing [107, 108]. These approaches include attention-based denoising [109, 110], multi-instance learning [111, 112], label refinement[113, 114], and curriculum learning [115, 116, 117]. Some models use hierarchical type structures to constrain predictions and regularize learning; others generate or reweight training instances based on estimated label reliability [118, 119]. Overall, noise-aware modeling has become a key component in leveraging large-scale distantly supervised corpora for FgNER.

Another axis of variation lies in the formulation of the prediction problem. Early FgNER models treated typing as multi-label classification over a fixed type taxonomy, often incorporating hierarchical regularization to respect the taxonomy structure. More recently, state-of-the-art approaches have reframed entity typing as a Natural Language Inference (NLI) problem. In this formulation, each candidate entity type is expressed as a textual hypothesis, and the model predicts whether the entity type is entailed by the entity mention's context. Ultra-fine typing models such as "Language Inference based Typing of Entities (LITE)" [120] and "Decoupled Encoding and Cross-Attention for Efficient Named Entity Typing (DECENT)" [121] exemplify this strategy, combining strong encoders with NLI-style objectives and systematic generation of negative samples to sharpen decision boundaries. These methods have achieved leading performance on English benchmarks, including "Ultra Fine-grained Entity Typing (UFET)" [23], "Fine-grained Entity Recognition (FIGER)" [2], and OntoNotes [27].

2.6 Evaluation Metrics for FgNER

The correctness of both the entity boundary and entity type is considered for evaluation. If the system prediction matches the actual ground truth, then it is considered a True Positive. In the example shown in the section 2.3, 2008 Summer Olympics belonging to entity type *event/sports_event* is a True Positive. If the system prediction does not exist in the ground truth, then it is a False Positive. For example, recognition of 8 as an entity mention is a False Positive, as the entity type *number* is not predefined in the FIGER taxonomy [2]. If the entities present in the ground truth cannot be predicted by the system, then such mentions are considered False Negatives. For example, if gold medals could not be categorized as *award* entity type by the system, then it is a False Negative. **Precision**, **Recall** and **F1-score** are calculated as follows:

$$\textbf{Precision} = \frac{True\,Positive}{True\,Positive + False\,Positive}$$

$$\textbf{Recall} = \frac{True\,Positive}{True\,Positive + False\,Negative}$$

$$\textbf{F1-Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Micro-Average: Micro-averaging computes the overall Precision, Recall, and F1-score by aggregating the total True Positives, False Positives, and False Negatives across all entity types before applying the formulas. For example, the entity mentions Michael Phelps, gold medals, and 2008 Summer Olympics will be considered together to calculate the Micro-Average F1-score for the entire system. This approach gives equal weight to every individual instance, making it more influenced by the performance on frequent entity types.

Table 2.1: Details of multilingual encoder models used: no. of parameters, no. of languages pretrained on, and coverage of Indian languages.

Model	**Parameters**	**No. of Languages**	**Coverage of Indian languages**
BERT-base-multilingual-cased [82]	110M	104	Bengali, Gujarati, Hindi, Kannada, Malayalam, Marathi, Nepali, Punjabi, Tamil, Telugu
IndicBERTv2-MLM-Sam-TLM [85]	278M	26	Assamese, Bengali, Bodo, Dogri, Gujarati, Hindi, Kannada, Kashmiri, Konkani, Maithili, Malayalam, Marathi, Manipuri, Nepali, Odia, Punjabi, Sanskrit, Santali, Sindhi, Tamil, Telugu, Urdu
MuRIL-large-cased [84]	340M	17	Assamese, Bengali, Gujarati, Hindi, Kannada, Malayalam, Marathi, Nepali, Odia, Punjabi, Sanskrit, Sindhi, Tamil, Telugu, Urdu
XLM-RoBERTa-large [83]	355M	100	Assamese, Bengali, Gujarati, Hindi, Kannada, Malayalam, Marathi, Nepali, Odia, Punjabi, Tamil, Telugu, Urdu

Macro-Average: Macro-averaging calculates Precision, Recall, and F1-score independently for each entity type and then takes the unweighted mean across all types. For example, F1-score for each entity type *person/athlete*, *award*, and *event/sports_event* will be calculated separately, and their unweighted mean will be the Macro-Average F1-score of the entire system. As a result, every entity type contributes equally to the final score, regardless of how many instances it contains, providing a more balanced view of performance across both common and rare categories.

Evaluations for types of errors: Several forms of errors may be made by the system while recognizing the entities.

- **Boundary Mismatch** errors occur when the entire entity mention cannot be detected correctly (e.g. 2008 Summer is detected instead of 2008 Summer Olympics).
- **Type Mismatch** errors occur when an entity mention is categorized to a wrong entity type (e.g. 2008 Summer Olympics is categorized as *organization/sports_league* instead of *event/sports_event*).
- **Spurious** entities are False Positive entity mentions, which are detected by the system for undefined entity types (e.g. 8 is recognized, although *number* entity type is undefined).

2.6.1 State-of-the-Art Multilingual NER Evaluation Methodology

In the sequence labeling paradigm, the most prominent and widely adopted methodology for evaluation is to fine-tune multilingual or language-specific Pre-trained Language Models (PLMs), such as mBERT [82], XLM-RoBERTa [83], MuRIL [84], IndicBERTv2 [85], and MizBERT[122]. PLM fine-tuning on annotated NER datasets provides a strong baseline and often achieves state-of-the-art

performance in supervised settings [24, 34, 75, 86, 87, 88, 95, 123, 124]. The details of multilingual encoder models used in this work, i.e. mBERT [82][1], MuRIL [84][2], IndicBERTv2 [85][3], and XLM-RoBERTa-large [83][4] are shown in Table 2.1.

2.7 Information on the Languages Considered in this Work

Worldwide, more than two billion people speak various Indian languages. Yet, the resources for such languages are very scarce, especially for fine-grained named entity recognition. This scarcity motivated this research work to propose various methods to develop FgNER resources in multiple Indian languages. The 26 Indian languages considered in this research work are briefly discussed in this section and summarized in Table 2.2. Figure 2.2 shows the official languages in Indian states and neighboring countries. Similarly, Figure 2.3 shows additional official languages, Figure 2.4 shows language families, and Figure 2.5 shows the scripts used for various languages in Indian states and neighboring countries.

1. **Assamese (ISO 639-1 code: as)**: With more than 15 million native speakers [125], Assamese is an official language of the Indian state of Assam (Figure 2.2), and a scheduled language in India [126]. This Indo-European language, having more than 8.3 million second-language speakers, is also spoken in various other Indian states, such as Arunachal Pradesh, Meghalaya, Nagaland etc. Assamese is written using the Bengali-Assamese script.

2. **Bengali (ISO 639-1 code: bn)**: The sixth most spoken first language, Bengali (also known as Bangla), is an official language of Bangladesh, and the Indian states of West Bengal, Tripura (Figure 2.2), and the Barak Valley region of Assam. This Indo-European language is included in the The Eighth Schedule to the Constitution of India [126], and is written using the Bengali-Assamese script. Having more than 285 million speakers, Bengali is also widely spoken in the Indian states of Odisha, Bihar, Andaman and Nicobar Islands, etc.

3. **Bhojpuri (ISO 639-2 code: bho)**: Bhojpuri is considered within the group of Hindi language [125] which is spoken in some regions of Indian states of Bihar, Uttar Pradesh, and Jharkhand, along with Nepal, Fiji, Mauritius, Suriname, South Africa, etc. This Indo-European language has more than 51 million speakers and is written using the Devanagari script.

4. **Bishnupriya Manipuri (ISO 639-3 code: bpy)**: Bishnupriya Manipuri, also known as Bishnupriya, is a creole of Bengali and Manipuri/Meitei languages that is written using the Bengali-Assamese script. Bishnupriya is spoken in some regions of Bangladesh and the Indian states of Assam, Tripura, and Manipur. This vulnerable language [127] is spoken by more than 120 thousand speakers.

5. **Bodo (ISO 639-3 code: brx)**: Bodo, also known as Boro, is an additional official language of Assam (Figure 2.3), primarily spoken in the Bodoland Territorial Region, and in some regions in Bangladesh and Nepal. This Sino-Tibetan language, included as a scheduled language in India [126], is written using the Devanagari script. Bodo is recognized as a vulnerable language [127] due to its approximately 1.5 million native speakers in India [125].

[1] https://huggingface.co/google-bert/bert-base-multilingual-cased
[2] https://huggingface.co/google/muril-large-cased
[3] https://huggingface.co/ai4bharat/IndicBERTv2-MLM-Sam-TLM
[4] https://huggingface.co/FacebookAI/xlm-roberta-large

Figure 2.2: Official languages in Indian states & neighboring countries.

6. **Chhattisgarhi (ISO 639-3 code: hne)**: This Indo-European language, written using the Devanagari script, is considered a co-official language in the Indian state Chhattisgarh (Figure 2.3). More than 17 million people from the Indian states of Chhattisgarh, Odisha, Madhya Pradesh, and Maharashtra speak this language [125].

7. **Dogri (ISO 639-2 code: doi)**: This scheduled Indian language [126] is spoken by around 2.6 million people in some regions of Pakistan and the Indian states of Punjab, Himachal Pradesh, and the union territory Jammu and Kashmir, [125]. This Indo-European language is written using different scripts in different regions, including the Devanagari, Dogra, and Nastaliq scripts.

8. **Gujarati (ISO 639-1 code: gu)**: Gujarati, an Indo-European language (Figure 2.4) written using the Gujarati script (Figure 2.5), is an official language of the Indian states of Gujarat and the union territories of Dadra and Nagar Haveli, and Daman and Diu. This scheduled Indian language [126] is spoken by around 56 million people across various countries.

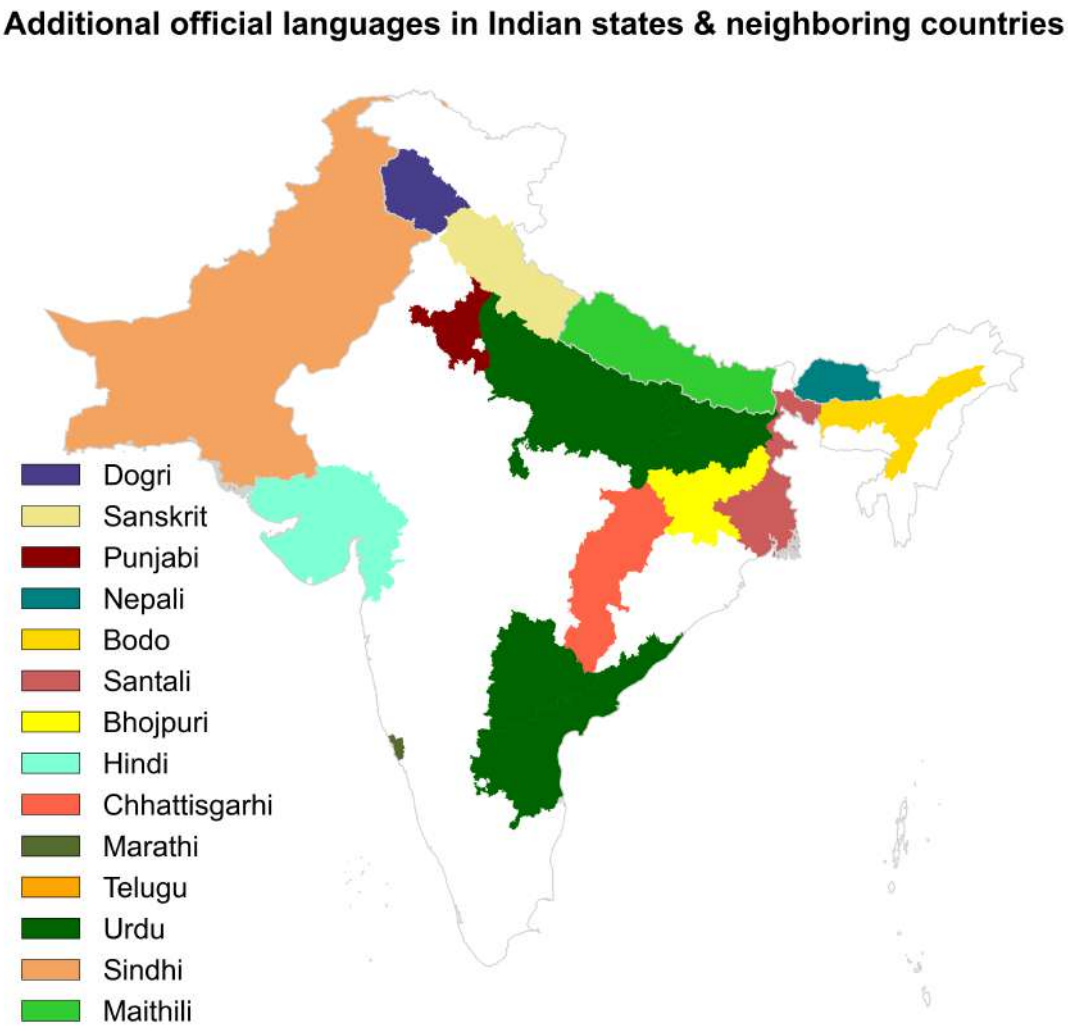

Figure 2.3: Additional official languages in Indian states & neighboring countries.

9. **Hindi (ISO 639-1 code: hi)**: The fourth most-spoken first language, Hindi, is an official language of the Government of India, Fiji, and ten states and six union territories of India [126]. This Indo-European language, written using the Devanagari script, is widely accepted across various nations with more than 610 million speakers.

10. **Kannada (ISO 639-1 code: kn)**: The Dravidian language, Kannada, is an official language of the Indian state Karnataka. This scheduled Indian language [126] is written using Kannada script (Figure 2.5) and spoken by more than 60 million people [125].

11. **Kashmiri (ISO 639-1 code: ks)**: This scheduled Indian language [126] is one of the official languages in Jammu and Kashmir in India. This Indo-European language is written using the Perso-Arabic script and the Devanagari script in different areas, with more than 7.1 million speakers, including in Pakistan.

12. **Konkani (ISO 639-2 code: gom)**: The official language of the Indian state of Goa, Konkani, is an Indo-European language written using the Devanagari script. With more

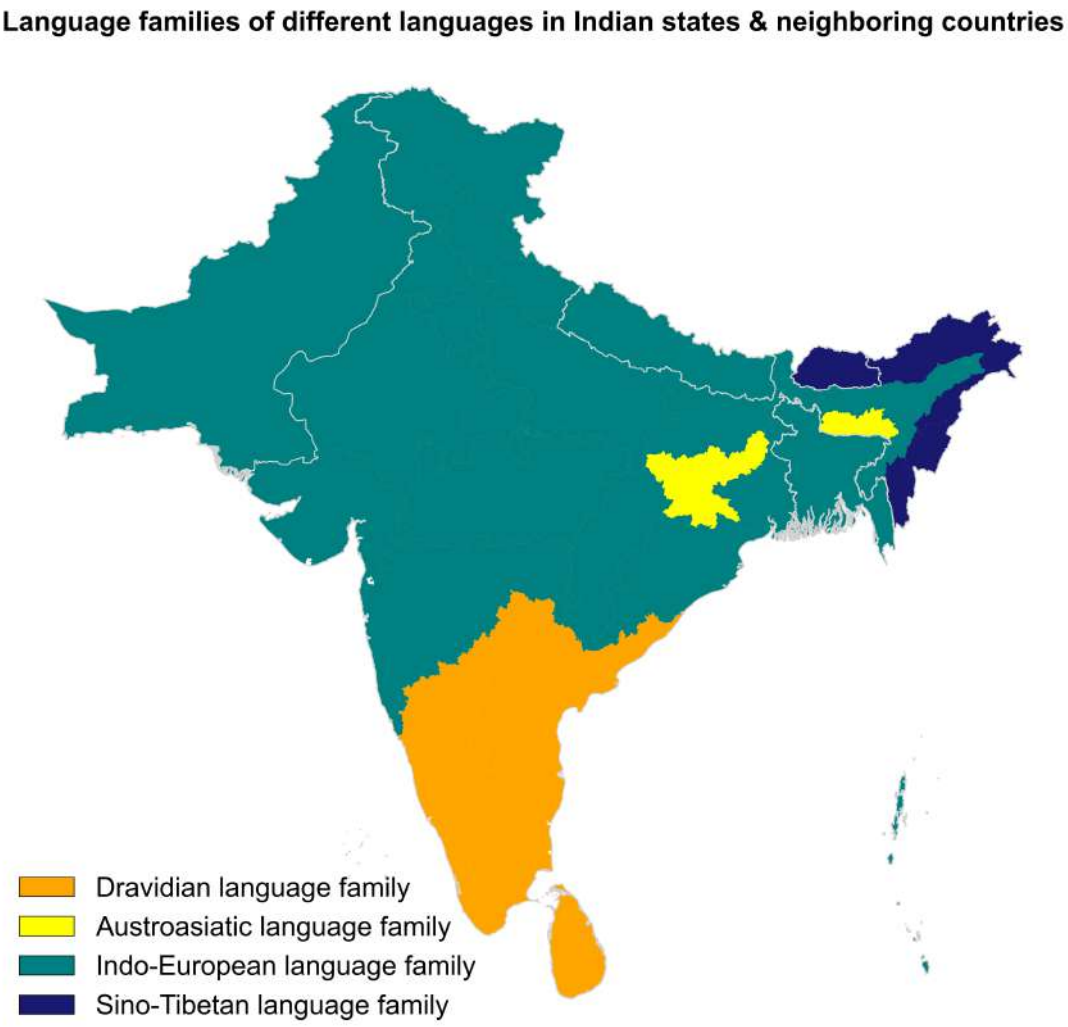

Figure 2.4: Language families of different languages in Indian states & neighboring countries.

than 2.3 million speakers, this scheduled Indian language [126] is also spoken in Karnataka, Maharashtra, Kerala, Gujarat, etc.

13. **Maithili (ISO 639-2 code: mai)**: Being an additional official language in Nepal and in the Indian state of Jharkhand, Maithili is spoken by more than 22 million people (Figure 2.3). This Indo-European language is a scheduled Indian language [126], and mostly written using the Devanagari script.

14. **Malayalam (ISO 639-1 code: ml)**: The Dravidian language, Malayalam, is an official language of the Indian state Kerala and the union territories Lakshadweep and Puducherry. This scheduled Indian language [126] is written using the Malayalam script, and spoken by more than 38 million people.

15. **Manipuri (ISO 639-2 code: mni)**: Manipuri, also known as Meitei, is an official language of the Indian state of Manipur. Along with Bodo, Manipuri is the only other vulnerable language [127], included as a scheduled language of India [126]. This Sino-Tibetan language

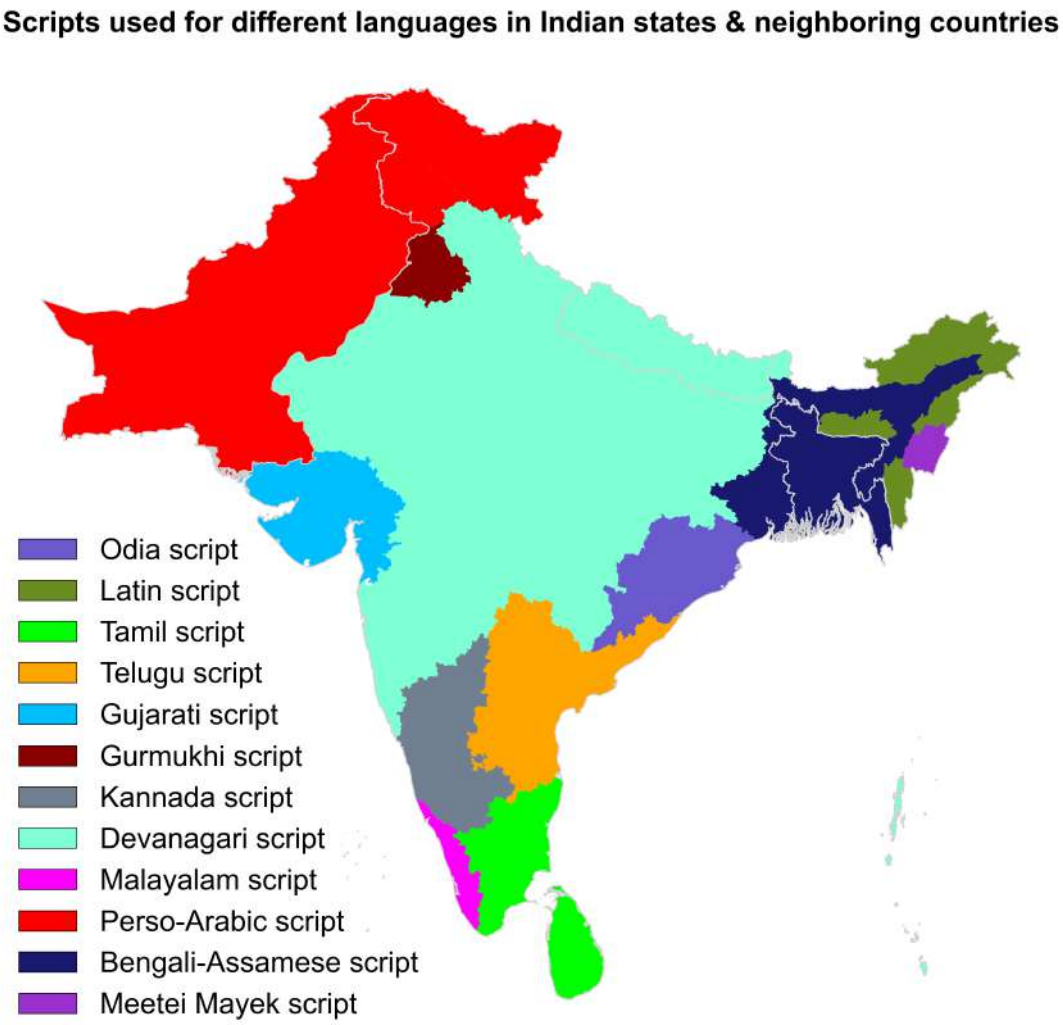

Figure 2.5: Scripts used for different languages in Indian states & neighboring countries.

is written using the Meetei Mayek script and spoken by around 1.8 million people, including in Myanmar and Bangladesh.

16. **Marathi (ISO 639-1 code: mr)**: The official language of the Indian states Maharashtra and Goa, the scheduled Indian language Marathi is the 15th most spoken language in the world [126]. This Indo-European language (Figure 2.4) is written using the Devanagari script (Figure 2.5), and spoken by more than 100 million people in more than ten states and union territories in India.

17. **Mizo (ISO 639-2 code: lus)**: The Sino-Tibetan language Mizo (Figure 2.4), also known as Lushai, is written using the Latin script (Figure 2.5) and considered the official language of the Indian state Mizoram. This vulnerable language [127] with only one million speakers [125] is yet to be included as a scheduled language of India [126].

18. **Nepali (ISO 639-1 code: ne)**: The official language of Nepal and the Indian state Sikkim, Nepali, is an Indo-European language written using the Devanagari script. This scheduled

Indian language [126] is spoken by more than 32 million people across various countries, including Bhutan, Myanmar, Brunei, Australia etc.

19. **Oriya (ISO 639-1 code: or)**: Oriya, also known as Odia, is a scheduled Indian language [126] spoken by more than 38 million people [125]. This Indo-European language (Figure 2.4) is written using the Odia script (Figure 2.5). Oriya (or Odia) is the official language of the Indian state of Odisha.

20. **Punjabi (ISO 639-1 code: pa)**: Punjabi, also known as Panjabi, is an Indo-European language (Figure 2.4) spoken by more than 150 million speakers across India, Pakistan, Canada, etc. This scheduled Indian language [126] is written using the Gurmukhi and Shahmukhi scripts in different regions (Figure 2.5).

21. **Sanskrit (ISO 639-1 code: sa)**: Sanskrit is a classical Indo-European language, often found in the ancient texts of multiple religions. Although only around 25 thousand native speakers speak Sanskrit [125], it is considered a scheduled Indian language [126] and an additional official language in the Indian states of Himachal Pradesh and Uttarakhand.

22. **Santali (ISO 639-2 code: sat)**: Santali is an Austroasiatic language spoken by around 8 million people across India, Bangladesh, Bhutan, and Nepal. Officially, Santali is written using Ol Chiki script. This scheduled Indian language [126] is an additional official language in the Indian states of Jharkhand and West Bengal.

23. **Sindhi (ISO 639-1 code: sd)**: The Indo-European language Sindhi is an official language in India and Pakistan with over 37 million speakers [126]. Although Sindhi is written using the Perso-Arabic script, in some regions in India, the Devanagari script is also used.

24. **Tamil (ISO 639-1 code: ta)**: The Dravidian language Tamil is one of the longest-surviving classical languages in the world. Tamil is an official language in India [126], Sri Lanka, and Singapore, and a recognized minority language in Malaysia and South Africa. It is written using the Tamil script and spoken by more than 86 million people worldwide.

25. **Telugu (ISO 639-1 code: te)**: The Dravidian language Telugu is the official language in the Indian states of Telangana, Andhra Pradesh, and the union territory of Puducherry. This scheduled Indian language [126] is written using the Telugu script and spoken by more than 100 million people globally.

26. **Urdu (ISO 639-1 code: ur)**: The Indo-European language Urdu is a scheduled Indian language [126], official language in Pakistan and the Indian union territories of Jammu and Kashmir, and Ladakh, and additional official language in seven Indian states. Urdu is spoken by more than 253 million people across various nations, and written using the Perso-Arabic script.

Table 2.2: Summary of 26 Indian Languages and Their Status

Language (ISO Code)	**Speakers (Estimate)**	**Language Family**	**Scripts Used**	**Official/Scheduled Status**
Assamese (as)	> 24 Million	Indo-European	Bengali-Assamese	Official in **Assam**; Scheduled in India.
Bengali (bn)	> 285 Million	Indo-European	Bengali-Assamese	Official in **Bangladesh** and Indian states of **West Bengal, Tripura,** and **Barak Valley of Assam**; Scheduled in India.
Bhojpuri (bho)	> 51 Million	Indo-European	Devanagari	Additional official in **Jharkhand**; Spoken in parts of **Bihar, Uttar Pradesh, Jharkhand, Nepal, Fiji, Mauritius,** etc.
Bishnupriya Manipuri (bpy)	> 120 Thousand	Creole	Bengali-Assamese	Vulnerable [127]; Spoken in parts of **Bangladesh, Assam, Tripura, Manipur.**
Bodo (brx)	≈ 1.5 Million native (India)	Sino-Tibetan	Devanagari	Official in **Assam** (primarily Bodoland Territorial Region); Scheduled in India; Vulnerable [127].
Chhattisgarhi (hne)	> 17 Million	Indo-European	Devanagari	Co-official in **Chhattisgarh**; Spoken in **Odisha, Madhya Pradesh, Maharashtra.**
Dogri (doi)	≈ 2.6 Million	Indo-European	Devanagari, Dogra, Nastaliq	Scheduled in India; Spoken in parts of **Pakistan, Punjab, Himachal Pradesh, Jammu and Kashmir (UT).**
Gujarati (gu)	≈ 56 Million	Indo-European	Gujarati	Official in **Gujarat** and UTs of **Dadra and Nagar Haveli, Daman and Diu**; Scheduled in India.
Hindi (hi)	> 610 Million	Indo-European	Devanagari	Official language of **Government of India, Fiji,** and **10 Indian states and 6 UTs**; Scheduled in India.
Kannada (kn)	> 60 Million	Dravidian	Kannada	Official in **Karnataka**; Scheduled in India.
Kashmiri (ks)	> 7.1 Million	Indo-European	Perso-Arabic, Devanagari	Official in **Jammu and Kashmir (UT)**; Scheduled in India.
Konkani (gom)	> 2.3 Million	Indo-European	Devanagari	Official in **Goa**; Scheduled in India; Also spoken in **Karnataka, Maharashtra, Kerala.**
Maithili (mai)	> 22 Million	Indo-European	Devanagari	Official in **Jharkhand** and **Nepal**; Scheduled in India.

Table 2.2: Summary of 26 Indian Languages and Their Status (Continued)

Language (ISO Code)	**Speakers (Estimate)**	**Language Family**	**Scripts Used**	**Official/Scheduled Status**
Malayalam (ml)	> 38 Million	Dravidian	Malayalam	Official in **Kerala** and UTs of **Lakshadweep, Puducherry**; Scheduled in India.
Manipuri (Meitei) (mni)	≈ 1.8 Million	Sino-Tibetan	Meetei Mayek	Official in **Manipur**; Scheduled in India; Vulnerable [127]; Also spoken in **Myanmar, Bangladesh.**
Marathi (mr)	> 100 Million	Indo-European	Devanagari	Official in **Maharashtra** and **Goa**; Scheduled in India.
Mizo (Lushai) (lus)	≈ 1 Million	Sino-Tibetan	Latin	Official in **Mizoram**; Vulnerable [127].
Nepali (ne)	> 32 Million	Indo-European	Devanagari	Official in **Nepal** and Indian state **Sikkim**; Scheduled in India.
Odia (Oriya) (or)	> 38 Million	Indo-European	Odia	Official in **Odisha**; Scheduled in India.
Punjabi (Panjabi) (pa)	> 150 Million	Indo-European	Gurmukhi, Shahmukhi	Scheduled in India; Spoken across **India, Pakistan, Canada**, etc.
Sanskrit (sa)	≈ 25 Thousand native	Indo-European (Classical)	Devanagari	Additional official in **Himachal Pradesh** and **Uttarakhand**; Scheduled in India.
Santali (sat)	≈ 8 Million	Austro-asiatic	Ol Chiki	Additional official in **Jharkhand** and **West Bengal**; Scheduled in India.
Sindhi (sd)	> 37 Million	Indo-European	Perso-Arabic, Devanagari	Official in **India** and **Pakistan**; Scheduled in India.
Tamil (ta)	> 86 Million	Dravidian	Tamil	Official in **India, Sri Lanka, Singapore**; Scheduled in India; Recognized minority in **Malaysia, South Africa.**
Telugu (te)	> 100 Million	Dravidian	Telugu	Official in **Telangana, Andhra Pradesh,** and UT **Puducherry**; Scheduled in India.
Urdu (ur)	> 253 Million	Indo-European	Perso-Arabic	Official in **Pakistan** and UTs of **Jammu and Kashmir, Ladakh**; Additional official in **7 Indian states**; Scheduled in India.

3

Taxonomy Adaptable Fine-grained Named Entity Recognition through Distant Supervision

Chapter Highlights

- The objective of this chapter is to build a taxonomy-adaptable framework for Fine-grained Named Entity Recognition (FgNER) in Indian languages.
- The framework efficiently generates high-quality annotations by leveraging the deep interlink between WikiData and Wikipedia through multi-stage heuristics and quality selection methods.
- The highly scalable framework successfully created a massive resource of approximately three million FgNER samples across four taxonomies for six languages, which belong to different language families and are written using four different scripts and in two different directions.
- The generated datasets proved robust across four different taxonomies, leading to an 83% average relative F1 score improvement over zero-shot baselines.
- This chapter is based on the publication "TAFSIL: Taxonomy Adaptable Fine-grained Entity Recognition through Distant Supervision for Indian Languages" published in "Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'25)".
- The TAFSIL dataset is publicly available at hf.co/datasets/prachuryyaIITG/TAFSIL.

3.1 Introduction

Recognizing the names of identifiable entities such as locations, buildings, and products from unstructured text is an important task for many natural language processing (NLP) applications such as recommendation systems, knowledge-base construction, relation extraction, etc. The types of identifiable objects or entities may vary across different domains, applications, and requirements. Although the most common entity types considered for Named Entity Recognition (NER) tasks include *Person*, *Location*, and *Organization*, efficient extraction of detailed information from unstructured texts often requires fine-grained types such as *Artist*, *Athlete*, *Company*, *Sports Team*,

etc. Many significant efforts have been made over the years to solve such fine-grained named entity recognition (FgNER) problems [2, 96, 27, 98, 101]. Since manual annotation for FgNER task is expensive [33], most of such resources are developed based on distant supervision [2] and improved the quality of annotations by applying language-specific heuristics [22].

Although the supervised models trained on such FgNER datasets can detect more than a hundred fine-grained entity types, there is a bottleneck in creating solutions to detect unseen and new entity types [128]. Since FgNER datasets are developed based on a defined taxonomy with a fixed set of entity types, the mentions detected as an entity as per one taxonomy may not be detected as per another taxonomy. For example, mentions detected as */biomolecules* entity type as per *HAnDS* [22] taxonomy will not be considered as an entity in *MultiCoNER2* [3] taxonomy. Similarly, the *CarManufacturer* entity type as per *MultiCoNER2* [3] taxonomy can be considered as a generic */organization/company* entity type as per *HAnDS* [22] or *FIGER* [2] taxonomy, which may lead to loss of information learned by the supervised models.

Moreover, the progress of such FgNER resources is often limited to high-resource languages. MultiCoNER2 [102][1], as a task of SemEval-23[2], can be considered as one of the first attempts to create FgNER resources for two Indian languages, *Hindi (hi)* and *Bengali (bn)*. Therefore, we present **TAFSIL**[3]: the **t**axonomy **a**daptable **f**ine-grained entity recognition through distant **s**upervision for **I**ndian **l**anguages. TAFSIL framework utilizes the interlink of knowledge base WikiData with the linked corpora Wikipedia and applies multiple stages of processing, heuristics-based matching and quality sentence selection to detect fine-grained entity mentions. Utilizing the taxonomy and language adaptability of TAFSIL framework, we have generated FgNER datasets in four taxonomies *FIGER*, *OntoNotes*, *HAnDS* and *MultiCoNER2*. The six different languages considered are *Hindi (hi)*, *Marathi (mr)*, *Sanskrit (sa)*, *Tamil (ta)*, *Telugu (te)* and *Urdu (ur)*, which are spoken by more than one billion people across various South Asian and South-east Asian countries. These languages belong to two different language families *Indo-European* and *Dravidian*, written using four different scripts *Devanagari*, *Tamil*, *Telugu* and *Nastaliq* in two different directions *left-to-right* and *right-to-left*. To the best of our knowledge, this is the first attempt to create fine-grained named entity recognition datasets for the languages *Marathi*, *Sanskrit*, *Tamil*, *Telugu*, and *Urdu*. Our contributions can be summarized as follows:

1. Introduction of a distant supervision based framework TAFSIL for fine-grained named entity recognition which is adaptable across different taxonomies and languages.

2. 24 FgNER datasets of a total size of around three million samples for six Indian languages (Hindi, Marathi, Sanskrit, Tamil, Telugu, and Urdu) in each of four taxonomies (FIGER, OntoNotes, HAnDS, and MultiCoNER2).

3. 24 FgNER human annotated datasets for the six mentioned Indian languages in the four mentioned taxonomies. Each of these Gold datasets contains more than 2700 samples in the development set and more than 1500 samples in the test set.

4. Our extensive experiments with state-of-the-art (SOTA) FgNER models suggest the good quality of these generated datasets. The trained models with our datasets improve the relative performance of 83% in average F1 score over zero-shot performances of various SOTA models.

[1] https://multiconer.github.io/
[2] https://semeval.github.io/SemEval2023/
[3] "*Tafsil*": تفصیل in Urdu and तफसील in Hindi means details or description

3.2 Related Works

Sekine and Nobata [96] introduced entity type hierarchy in 2002 in which they defined 150 entity types in multi-level hierarchies. One of the pioneering works in Fine-grained entity typing was proposed by Ling and Weld [2] in 2012, in which 113 types were defined in two hierarchical depths. Some other significant works in FgNER tasks include HYENA [98], OntoNotes [27], TypeNet [101], etc. To improve the quality of the datasets, language-specific heuristics and quality sentence selection methods were adopted by HAnDS [22], MultiCoNER2 [3], etc. On the other hand, FewNERD [33] is one of the largest datasets created for FgNER task through manual annotations. Subsection 2.3.1 in Chapter 2 demonstrates a few fine-grained entity typing datasets in English and the details of these datasets, such as the number of entity types and the hierarchical depth of entity types.

Creation of FgNER datasets in Indian languages have been very limited. A shared task in SemEval-2023, MultiCoNER2 [102] can be considered as one of the first attempt to create FgNER datasets in two Indian languages Hindi and Bengali. The best performing teams in MultiCoNER2 such as USTC-NELSLIP [103], PAI [104], DAMO-NLP [105], IXA/Cogcomp [106] etc. enhanced the resources in these two languages. The significant attempts to create resources for coarse-grained NER tasks include Polyglot NER [16], WikiANN [17] etc. in which some Indian languages were considered among more than hundreds of languages. HiNER [34], AsNER [86], MahaNER [87], NER in Bengali [89], NER in Telugu [90], EverestNER [88] etc. are a few examples of manually annotated datasets in different Indian languages. Naamapadam [75] is one of the largest NER datasets in 11 Indian languages created through the projection method from English NER. After preparing coarse-grained and fine-grained datasets in English, translation is used to create datasets in Hindi and Bengali in MultiCoNER-1 [24] and MultiCoNER2 [3].

3.3 TAFSIL Framework

As shown in Figure 3.1, the TAFSIL framework has three inputs: a linked corpus (Wikipedia[4]), knowledge base (Wikidata[5]), and the mapping of knowledge base property values to taxonomy-specific entity types. Wikidata contains structured knowledge in the form of property-key values, such as *San Francisco (qid: Q62)* is an *instance of (pid: P31)* a *city (qid: Q515)*. Such keys in Wikidata are mapped to the taxonomy-specific entity types. For example, the key *film (qid: Q11424)* in Wikidata is mapped to *VisualWork* entity type as per *MultiCoNER2* taxonomy and */art/film* entity type as per *FIGER* taxonomy. So, *the instances* of *film* will be considered valid entity mentions as per the defined taxonomy. Wikidata also contains the list of aliases in multiple languages. For example, the name गौहाटी (*Gauhati*) is an alias of the city गुवाहाटी (*Guwahati*). Leveraging such information from Wikidata and Wikipedia, a master dictionary is created in the preprocessing step which works as the backbone of the TAFSIL framework.

3.3.1 Stage 1 [Removal of Non-Entity Links]

As per the Algorithm 1, the hyperlinks in Wikipedia are checked against the master dictionary entries, and the matching mentions are selected to update a trie. In contrast, non-matching mentions are discarded to reduce false positives. For instance, in this Wikipedia sentence,"The state was the first site for oil drilling in Asia.", although oil drilling is a valid hyperlink, it does not belong to a defined entity type and is thus removed. But Asia is a valid entity mention as per

[4]https://www.wikipedia.org/
[5]https://www.wikidata.org/wiki/Wikidata:Main_Page

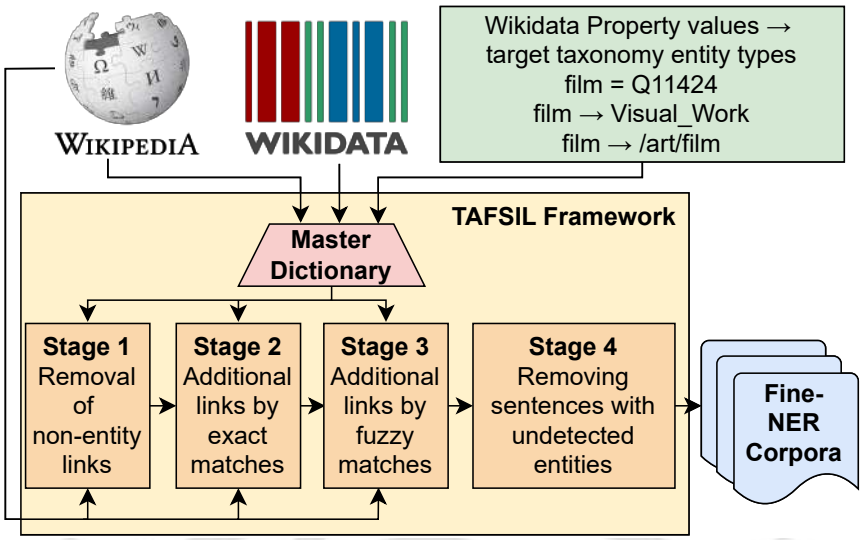

Figure 3.1: Overview of TAFSIL framework. TAFSIL utilizes the high interlink between the knowledge base WikiData and the linked corpora Wikipedia through multi-stage heuristics and improves annotation through fuzzy match and quality sentence selection.

the defined taxonomy and hence is updated in the trie. Wikipedia dumps are processed using the WikiExtractor [129].

3.3.2 Stage 2 [Additional Links through Exact Match]

As per Wikipedia guidelines, only the first mention of an entity within an article must be hyperlinked to the respective Wikipedia page. The rest of the mentions must be plain text, not anchor links. Therefore, detecting all such mentions is necessary to reduce false negatives. The trie created in stage 1 is utilized in this stage to find the exact matches in Wikipedia sentences. For example, the valid mention *Assam* is not hyperlinked in the sentence "*Assam and adjoining regions have evidences of human settlement from the beginning of the Stone Age.*" in the Wikipedia article on 'Assam' as *Assam* was hyperlinked previously in the same article. In this stage of TAFSIL framework, such entity mentions are detected through exact match utilizing the trie created in the stage 1.

3.3.3 Stage 3 [Additional Links through Fuzzy Match]

In many Indian languages, multiple valid spellings for the same word exist, including variations due to agglutinations. For example, both मुम्बई and मुंबई are valid Hindi spellings of the city 'Mumbai'. Similarly, the agglutinated word in Telugu రామానగరం is the same as the separated words రామా నగరం which means "Rama Nagaram", i.e. "Rama's City". To handle such variations, an empirical fuzzy matching heuristic is applied, which accepts all tokens within a maximum edit distance. The formula for the maximum edit distance includes one error for all tokens, one error for every white space, and one error for every six characters of the token. The white space error is taken into consideration to accommodate agglutination:

$$d(j) = 1 + white\_space(j) + length(j)/6$$

Algorithm 1: TAFSIL Framework

Input:

- C: Linked corpus (Wikipedia)
- W: Knowledge base (Wikidata)
- V: Mapping of property values to entity types

Symbols:

- h: Hyperlink in a sentence, H: Set of hyperlinks
- e: Entity type, E: Set of entity types
- p_i: Property in Wikidata(W)
- k_i: Key in Wikidata(W)
- PK: Set of property-key values
- MD: Master dictionary of mentions and entity types
- T: Trie for mention lookup, t: Term in Trie
- s: Sentence
- j: Individual token of a sentence s
- J: Set of tokens of a sentence s
- m: Entity mention detected in sentence s
- M_s: Set of entity mentions in sentence s
- $d(j)$: Maximum edit distance for token j
- e_d(j, t): Edit distance between j and t
- POS(j): Part-of-speech tag of j
- NN, NNP: Noun, proper noun POS tags

Master Dictionary Creation

1. Extract property-value pairs: $PK = \{(p_i, k_i)\}$
2. Map k_i to entity types: $E = V(k_i)$
3. Build master dictionary: $MD = \{(m, e)\}$ using aliases

Stage 1: Removal of Non-Entity Links

1. Initialize Trie: $T \leftarrow \emptyset$
2. **for all** $s \in C$ **do**
3. Extract hyperlinks: $H \leftarrow$ hyperlinks h in s
4. Update Trie: $T \leftarrow T \cup \{h \mid h \in MD\}$
5. Remove non-matching hyperlinks: $H \leftarrow H \setminus T$
6. **end for**

Stage 2: Additional Links through Exact Match
1. **for all** $s \in C$ **do**
2. Create set of tokens from sentence s: $J = \{j\}$
3. Exact match using Trie: $M_s \leftarrow \{j \mid j \in T\}$
4. Annotate M_s as entities
5. **end for**

Stage 3: Additional Links through Fuzzy Match
1. **for all** $s \in C$ **do**
2. Calculate: $d(j) = 1 + white\_space(j) + \frac{length(j)}{6}$
3. Fuzzy match: $M_s \leftarrow \{j \mid \exists t \in T,\ \text{e\_d}(j,t) \leq d(j)\}$
4. Annotate M_s as entities
5. **end for**

Stage 4: Removing Sentences with Undetected Entities
1. **for all** $s \in C$ **do**
2. Discard s if: $\forall m \in M_s$ and $\text{POS}(m) \notin \{NN, NNP\}$
3. Or discard s if: $\text{POS}(j) \in \{NN, NNP\}$ and $j \notin M_s$
4. **end for**
5. Update corpus: $C \leftarrow C \setminus \{s\}$

Output: Refined dataset with entity annotations

Where, d is the maximum edit distance, j is the token in a sentence s. Only word-based tokenization is used in this framework. So, in this stage, the heuristics-based fuzzy matches are detected utilizing the trie created in stage 1, and the datasets are updated with such entity mentions.

3.3.4 Stage 4 [Removing Sentences with Undetected Entities]

In this stage, a parts-of-speech (POS) tagger [130][6] is used to match entity mentions with their respective POS tags. If a sentence has a token with POS tag as noun (NN) or a proper noun (NNP) but is not detected as an entity, or if an entity mention in a sentence is not tagged as a noun (NN) or a proper noun (NNP), then such a sentence is removed from the dataset. For instance, in the given sentence, "On the right hand?", *hand* is tagged as a noun (NN) but not an entity (e.g., /body_part), and hence such a sentence is discarded to enhance the quality of the dataset. The percent of sentences excluded based on POS tagging varies from around 3.1% in Hindi to 3.93% in Sanskrit. This quality sentence selection step results in the exclusion of around 3.5% sentences across all the languages in all the taxonomies.

3.4 TAFSIL Dataset

We have used Wikidata and Wikipedia dumps released on 1st June, 2024 to create large-scale FgNER datasets through TAFSIL framework. Table 3.1 summarizes the statistics of the 24 datasets created in six languages for each of the four taxonomies.

[6] https://github.com/stanfordnlp/stanza

Table 3.1: TAFSIL datasets statistics.

Taxonomy	Language	Sentences	Entities	Tokens
FIGER	Hindi	697,585	928,941	25,015,025
	Marathi	138,114	193,953	4,010,213
	Sanskrit	18,946	25,515	520,970
	Tamil	523,264	825,287	17,376,040
	Telugu	419,933	616,143	10,103,255
	Urdu	641,235	1,219,517	43,556,208
OntoNotes	Hindi	699,557	1,177,512	32,234,646
	Marathi	138,535	194,317	4,021,660
	Sanskrit	19,201	25,459	520,836
	Tamil	522,768	796,498	16,817,316
	Telugu	420,700	608,041	9,982,017
	Urdu	646,713	1,197,187	42,869,948
HAnDS	Hindi	702,941	1,108,130	30,249,166
	Marathi	139,163	195,251	4,039,536
	Sanskrit	19,294	25,581	523,540
	Tamil	526,275	804,132	16,988,161
	Telugu	422,736	611,226	10,034,077
	Urdu	647,595	1,200,111	42,988,031
MultiCoNER2	Hindi	643,880	799,987	21,492,069
	Marathi	125,981	174,861	3,628,450
	Sanskrit	17,740	23,372	479,185
	Tamil	464,293	690,035	14,583,842
	Telugu	392,444	561,088	9,266,603
	Urdu	603,682	1,069,354	38,216,564

Table 3.2: TAFSIL Gold dataset statistics in FIGER Taxonomy. IAA (κ) gives the inter-annotator agreement. Urdu is annotated by a single annotator.

Dataset	Development		Test		IAA
Splits	Entities	Tokens	Entities	Tokens	(κ)
Hindi	2,910	20,362	1,578	11,967	0.8334
Marathi	2,905	16,863	1,567	10,527	0.8067
Sanskrit	2,891	14,571	1,570	8,354	0.7834
Tamil	2,885	15,334	1,562	8,961	0.7653
Telugu	2,895	15,359	1,568	9,090	0.8204
Urdu	2,710	21,148	1,511	12,505	-

3.4.1 Comparison with Public Datasets

As shown in Table 3.4 the datasets created using the TAFSIL framework are larger than many other publicly available datasets in Hindi, Marathi, Sanskrit, Tamil, Telugu, and Urdu. Moreover, to the best of our knowledge, these are the first fine-grained named entity recognition datasets created in five Indian languages: Marathi, Sanskrit, Tamil, Telugu, and Urdu. In this work, whenever comparative analysis is done, the **highest value** or the **best result** in the tables are shown in **bold** and the second highest value or the second best result are shown as underlined.

Table 3.3: TAFSIL Gold datasets Inter-annotator Agreement (κ) scores across different taxonomies.

Taxonomy	Hindi	Marathi	Sanskrit	Tamil	Telugu
FIGER	0.8334	0.8067	0.7834	0.7653	0.8204
OntoNotes	0.8141	0.7888	0.7698	0.7523	0.8024
HAnDS	0.8365	0.8003	0.7712	0.7563	0.8121
MultiCoNER2	0.8466	0.8184	0.7982	0.7789	0.8386

Table 3.4: Comparison of TAFSIL datasets with some publicly available NER datasets in Indian languages.

	Dataset	Sentences	Entities	Types
Hindi	TAFSIL(HAnDS)	702,941	1,108,130	**118**
	MultiCoNER2 [131]	9,632	12,870	33
	Naamapadam [75]	**985,757**	**2,184,600**	3
	MultiCoNER [24]	15,300	19,733	6
	HiNER [34]	108,608	224,148	11
Marathi	TAFSIL(HAnDS)	139,163	195,251	**118**
	Naamapadam [75]	**455,201**	**529,600**	3
	MahaNER [87]	21,500	27,300	7
	WikiANN [132]	14,977	18,756	3
Sanskrit	TAFSIL(HAnDS)	**19,294**	**25,581**	**118**
	WikiANN [132]	101	115	3
Tamil	TAFSIL(HAnDS)	**526,275**	**804,132**	**118**
	Naamapadam [75]	497,900	741,100	3
	WikiANN [132]	25,662	31,814	3
Telugu	TAFSIL(HAnDS)	422,736	611,226	**118**
	Naamapadam [75]	**507,700**	**774,800**	3
	WikiANN [132]	9,928	11,964	3
Urdu	TAFSIL(HAnDS)	**647,595**	**1,200,111**	**118**
	WikiANN [132]	74,840	77,132	3

3.4.2 Entity Types Frequency Distribution

As Wikipedia articles mostly consist of famous personalities, monuments, locations, organizations etc., entity mentions belonging to such entity types are quite large in counts compared to very rare entity types. Therefore, the frequency of entity mentions often follows long tail distribution, as shown in Figure 3.2. In fact, the least 50 counts entity types also follow long tail distribution as shown in the sub-figure in Figure 3.2.

3.4.3 Gold Dataset

The same set of sentences have been considered in each language to be manually annotated for the mentioned four taxonomies using BRAT annotation tool [134]. The volunteer annotators are chosen based on their mother tongue and have minimum education as an undergraduate degree. For each taxonomy, the same set of 800 sentences is considered for the development set and 503 sentences for the test set. These sentences are extracted from Wikipedia, from the articles released after 1st June, 2024. Since the TAFSIL dataset is created using the Wikidata and Wikipedia dumps updated

Table 3.5: Adaptation of FgNER Models for Indian Languages. *Abbr.* column gives the abbreviation of the model name used in our discussion.

FgNER Model	Adapted encoder model	Abbr.
LITE [120]	XLM-RoBERTa-large-xnli [83]	L_XLM
	mDeBERTa-v3-base-mnli-xnli [133]	L_mD
DECENT [121]	XLM-RoBERTa-large [83]	D_XLM
	BERT-base-multilingual-cased [82]	D_mB
	IndicBERTv2-MLM-Sam-TLM [85]	D_IB
	MuRIL [84]	D_MR

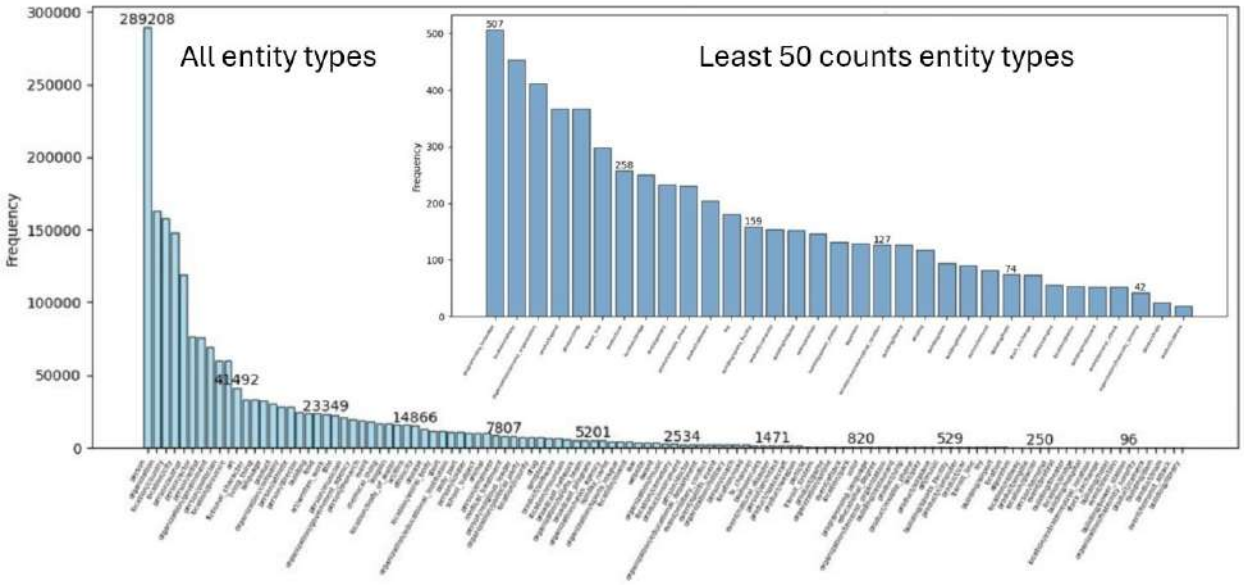

Figure 3.2: Long Tail distribution of entity mentions. As Wikipedia articles mostly consist of famous personalities, monuments, locations, organizations etc., entity mentions belonging to such entity types are quite large in counts compared to very rare entity types.

till 1st June, 2024, the sentences are considered from the articles published after that date, so that there is no leakage of testing data. For checking the robustness of the TAFSIL datasets, we made sure that both development and test sets contain a minimum of five entity mentions belonging to every fine-grained entity type in any considered taxonomy. Depending on the language and agreement between the annotators, the number of entities and total tokens vary marginally, as shown in Table 3.2. The good quality of these manually annotated datasets can ascertained based on the inter-annotator agreement (IAA), which is above 0.75 for each language [Table 3.3]. Due to the lack of annotators, the dataset for Urdu was annotated by a single annotator only.

3.5 Experimental Setup

The entity mentions in the datasets created using the distant supervision paradigm may be linked to one or more entity types depending on the associated knowledge base properties. But, depending on the context of an entity mention, only one entity type may be the most suitable. Therefore, to efficiently learn from such noisy datasets, numerous noise-aware models have been developed

for fine-grained and ultra fine-grained named entity recognition [107, 108, 109, 110, 135, 136, 137, 138, 139, 140, 111, 112, 113, 114, 118, 119, 115, 116, 117]. The current SOTA models for ultra-fine entity typing (UFET [23]) are LITE [120] and DECENT [121], which perform superior in fine-grained entity typing across different English FgNER datasets (FIGER [2] and OntoNotes [27]). These models address the entity typing problem as a Natural Language Inference (NLI) problem and improve the entity typing correctness by creating several negative samples during training. We have adapted these FgNER SOTA models to accommodate Indian languages as shown in Table 3.5. Four variations of the DECENT model are created by changing its base encoder from RoBERTa-large [141], to XLM-RoBERTa-large [83] (D_XLM), bert-base-multilingual-cased [82] (D_mB), IndicBERTv2-MLM-Sam-TLM [85] (D_IB) and MuRIL [84] (D_MR) respectively. The hyperparameters for DECENT-based models are set as follows: learning rate for encoder = 5e-6, learning rate for head = 5e-4, dropout probability for head = 0.5, epoch = 2, batch size = 16, negative oversampling rate = 31, and prediction threshold = 0.9. Similar to the DECENT model, two variations are checked for the LITE model by changing its base encoder from roberta-large-mnli [141] to xlm-roberta-large-xnli [83] (L_XLM) and mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 [133] (L_mD) respectively. The hyperparameters set for LITE-based models are batch size = 16, pre-training learning rate = 1e-6, margin = 0.1, epoch = 2, batch size = 16, margin for the margin ranking loss (λ) = 0.05. All the experiments have been conducted on one Nvidia A100 GPU. For the Hindi language, we performed extensive experiments in all of these six model variations and finally considered the best-performing two model variations for other Indian languages.

3.6 Results & Analysis

Although the TAFSIL framework is designed to be a highly robust framework through which fine-grained entity typing datasets can be created in any taxonomies, only four taxonomies are considered for experimental purposes. Numerous experiments are conducted to evaluate the robustness of the TAFSIL method in creating good-quality datasets in multiple languages across various taxonomies. Since this work is one of the first attempts to create FgNER datasets in five Indian languages, there are no such datasets available in these languages for comparative analysis.

3.6.1 Is a Distantly Supervised Noisy Dataset equivalent to a Clean Dataset?

MultiCoNER2, a SemEval'23 task on multi-lingual complex fine-grained named entity recognition, can be considered one of the pioneering works in FgNER in Hindi and Bengali. To encourage the development of larger datasets and high-performing models in FgNER tasks, the organizers kept the training datasets size smaller than the testing datasets in BIO format, where each entity mention belongs to only one predefined entity type. We have performed several experiments to check whether various FgNER models trained on our noisy datasets created through the TAFSIL framework learn as efficiently as compared to the clean MultiCoNER2 dataset. We trained all six models with different train sets and evaluated on the test set released by the task organizers. Table 3.6 shows the results of all six models. The results suggest that with the same-sized training datasets (9,632 sentences having 12,870 entity mentions), the models' performances trained on our dataset are poor compared to those trained on the MultiCoNER2 training dataset. However, when a large training dataset of 643,880 sentences having 799,987 entities is used, all six models performed better. The results shown in Table 3.6 confirm that a large dataset created by the TAFSIL framework can be equivalent to a clean small FgNER dataset such as MultiCoNER2. Hence, the TAFSIL framework is further applied across multiple taxonomies and multiple languages to create FgNER datasets.

Table 3.6: Performance of different models trained on TAFSIL Hindi dataset and tested on MultiCoNER2 test set. Compared to equal sized dataset with MultiCoNER2, when the entire TAFSIL training dataset having 799,987 entities (800k) is used, all six models performed better.

D_XLM Model

Train Set(No. of entities)	Micro P	Micro R	Micro F1(Δ%)	Macro P	Macro R	Macro F1(Δ%)
MultiCoNER2(12.8k)	72.18	70.78	71.47(-)	73.85	70.79	72.28(-)
TAFSIL(12.8k)	43.63	48.19	45.79(↓35.9)	44.67	48.86	46.36(↓35.9)
TAFSIL(800k)	63.37	82.51	**71.69**(↑0.3)	74.84	70.86	**72.80**(↑0.7)

D_IB Model

Train Set	Micro P	Micro R	Micro F1(Δ%)	Macro P	Macro R	Macro F1(Δ%)
MultiCoNER2(12.8k)	66.44	76.24	71.04(-)	66.56	76.33	71.11(-)
TAFSIL(12.8k)	42.53	56.56	46.00(↓35.2)	42.71	56.70	46.70(↓34.3)
TAFSIL(800k)	62.95	81.82	71.35(↑0.2)	62.35	84.13	71.62(↑0.7)

D_mB Model

Train Set(No. of entities)	Micro P	Micro R	Micro F1(Δ%)	Macro P	Macro R	Macro F1(Δ%)
MultiCoNER2(12.8k)	63.16	69.23	66.06(-)	61.88	79.69	69.66(-)
TAFSIL(12.8k)	40.60	48.04	44.00(↓33.3)	40.98	48.21	44.47(↓36.1)
TAFSIL(800k)	62.97	71.76	67.08(↑1.6)	62.42	79.61	69.98(↑0.5)

D_MR Model

Train Set(No. of entities)	Micro P	Micro R	Micro F1(Δ%)	Macro P	Macro R	Macro F1(Δ%)
MultiCoNER2(12.8k)	65.69	78.51	69.88(-)	63.57	80.18	70.92(-)
TAFSIL(12.8k)	42.14	50.43	45.71(↓34.5)	42.43	52.51	46.52(↓34.4)
TAFSIL(800k)	65.18	78.80	71.18(↑2.1)	63.18	82.62	71.60(↑1.0)

L_XLM Model

Train Set(No. of entities)	Micro P	Micro R	Micro F1(Δ%)	Macro P	Macro R	Macro F1(Δ%)
MultiCoNER2(12.8k)	62.79	69.01	65.95(-)	61.58	78.63	67.54(-)
TAFSIL(12.8k)	39.78	47.87	43.92(↓33.4)	40.76	48.03	43.06(↓36.2)
TAFSIL(800k)	62.76	71.06	66.11(↑0.2)	62.03	78.91	67.87(↑0.5)

L_mD Model

Train Set(No. of entities)	Micro P	Micro R	Micro F1(Δ%)	Macro P	Macro R	Macro F1(Δ%)
MultiCoNER2(12.8k)	61.13	66.93	64.20(-)	60.31	75.50	65.21(-)
TAFSIL(12.8k)	38.95	46.60	41.19(↓35.8)	39.58	46.50	41.99(↓36.4)
TAFSIL(800k)	61.36	70.04	64.37(↑0.3)	60.91	76.28	65.98 (↑1.2)

3.6.2 Evaluations on Hindi

All the six models as stated in Table 3.5 are trained on the TAFSIL Hindi datasets. As per the results shown in the Table 3.7 the quality of the datasets can be verified based on the relative improvement of F1 scores over the zero-shot performance of the models.

Based on the experiments on Hindi datasets, the top-performing models D_XLM and D_IB

Table 3.7: Performance of the different models trained on TAFSIL Hindi dataset in multiple taxonomies. DECENT model with IndicBERTv2 and XLM-RoBERTa base encoders performed the best.

TAFSIL Hindi dataset in FIGER Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	59.84	80.85	**67.28**(69.5)	64.52	78.82	**67.95**(69.9)
D_mB	57.67	79.44	61.65(74.9)	55.67	72.44	62.69(76.1)
D_IB	59.70	80.62	67.18(75.5)	61.21	74.76	67.78(74.7)
D_MR	58.27	73.26	65.18(75.4)	59.51	75.65	65.93(75.3)
L_XLM	56.61	73.13	62.34(82.6)	55.82	72.26	62.99(83.5)
L_mD	55.81	71.31	60.82(80.9)	54.32	71.10	61.31(79.9)

TAFSIL Hindi dataset in OntoNotes Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	52.15	81.58	**63.41**(84.7)	53.13	79.70	**63.70**(83.1)
D_mB	50.67	77.37	60.05(87.8)	51.17	77.50	60.60(89.1)
D_IB	52.83	79.11	63.04(86.6)	52.02	80.89	63.32(85.3)
D_MR	51.18	79.15	62.28(84.9)	59.16	80.40	62.71(84.3)
L_XLM	54.89	65.24	59.62(97.8)	54.88	65.85	59.87(95.9)
L_mD	52.27	63.06	57.61(97.5)	53.72	64.03	58.37(98.4)

TAFSIL Hindi dataset in HAnDS Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	62.06	85.79	**72.02**(73.5)	63.67	85.25	72.90(75.3)
D_mB	61.39	83.59	70.95(74.1)	62.59	85.27	71.25(73.6)
D_IB	61.03	88.17	71.38(63.5)	61.25	90.71	**73.13**(67.9)
D_MR	63.82	85.29	71.70(69.3)	61.39	86.89	71.95(65.4)
L_XLM	60.89	81.29	69.36(75.6)	61.33	83.94	70.06(76.0)
L_mD	58.93	80.02	68.14(76.1)	60.01	82.68	68.80(76.6)

TAFSIL Hindi dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	64.52	76.19	69.97(70.1)	64.63	76.18	70.04(68.9)
D_mB	58.51	79.68	67.47(73.9)	59.45	79.86	68.10(71.0)
D_IB	67.22	75.56	**71.15**(76.3)	67.35	75.59	**71.22**(74.4)
D_MR	65.17	82.89	70.84(73.9)	65.69	74.51	71.15(73.5)
L_XLM	56.91	78.13	65.95(77.4)	58.05	78.25	66.21(76.0)
L_mD	55.81	76.91	64.05(79.3)	56.28	77.31	65.01(80.9)

are considered for evaluating datasets in other languages.

3.6.3 Different Languages & Multiple Taxonomies

Similar to Hindi, the tables 3.8, 3.9, 3.10, 3.11, 3.12 show the impact of TAFSIL datasets in all the languages trained on different FgNER models across different taxonomies. As shown in the tables,

Table 3.8: Performance of the different models trained on Marathi datasets in multiple taxonomies.

TAFSIL Marathi dataset in FIGER Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	43.47	69.68	**53.97**(58.9)	43.97	71.88	**54.56**(63.5)
D_IB	42.69	68.81	52.71(64.1)	44.13	68.87	53.79(64.9)

TAFSIL Marathi dataset in OntoNotes Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	41.88	61.66	**50.70**(56.4)	40.34	69.57	**51.07**(55.9)
D_IB	39.17	71.87	49.88(57.9)	38.47	74.37	50.71(60.6)

TAFSIL Marathi dataset in HAnDS Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	52.32	79.03	**62.95**(78.0)	54.55	82.87	**65.79**(83.3)
D_IB	51.51	79.17	62.41(76.3)	54.01	81.32	64.91(82.2)

TAFSIL Marathi dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	49.36	83.73	**62.10**(77.9)	50.89	81.93	**62.28**(77.6)
D_IB	48.57	82.81	61.23(84.9)	49.33	81.49	61.46(82.9)

Table 3.9: Performance of the different models trained on Sanskrit datasets in multiple taxonomies.

TAFSIL Sanskrit dataset in FIGER Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	27.68	53.21	36.18(99.4)	27.79	55.18	36.85(97.9)
D_IB	28.24	54.11	**37.16**(101.3)	28.42	55.24	**37.53**(94.3)

TAFSIL Sanskrit dataset in OntoNotes Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	20.05	50.56	28.07(69.3)	20.08	51.99	28.95(70.7)
D_IB	20.89	53.46	**30.50**(79.3)	22.00	53.63	**31.23**(79.6)

TAFSIL Sanskrit dataset in HAnDS Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	35.45	54.71	42.05(104.2)	35.70	54.71	43.09(104)
D_IB	36.78	56.04	**44.36**(100.4)	37.57	56.84	**45.24**(94.2)

TAFSIL Sanskrit dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	31.35	53.40	40.20(100.4)	32.74	53.29	40.56(95.5)
D_IB	32.74	56.03	**41.03**(97.3)	32.94	55.96	**41.47**(96.8)

significant improvement in F1 scores is observable over the zero-shot performances of such models.

Table 3.10: Performance of the different models trained on Tamil datasets in multiple taxonomies.

TAFSIL Tamil dataset in FIGER Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	41.31	79.29	**55.23**(95.5)	41.24	80.38	**55.83**(87.1)
D_IB	42.16	78.73	54.25(84.4)	41.53	79.35	55.21(79.7)

TAFSIL Tamil dataset in OntoNotes Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	41.79	71.07	52.63(84.4)	43.45	71.67	**53.93**(89.8)
D_IB	40.48	76.58	**52.96**(94.1)	40.53	76.88	53.00(83.1)

TAFSIL Tamil dataset in HAnDS Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	55.27	81.40	**66.87**(105.5)	55.98	82.94	**67.13**(106)
D_IB	55.15	81.21	66.11(98.0)	55.23	81.38	66.34(93.1)

TAFSIL Tamil dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	50.67	81.01	**63.41**(105.9)	51.67	81.12	**64.02**(107)
D_IB	50.12	80.13	62.91(100.8)	51.03	80.86	63.12(97.7)

Table 3.11: Performance of the different models trained on Telugu datasets in multiple taxonomies.

TAFSIL Telugu dataset in FIGER Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	45.59	77.74	**57.48**(86.1)	47.26	77.33	**58.67**(83.8)
D_IB	45.45	77.35	57.26(89.0)	46.52	76.33	57.80(84.5)

TAFSIL Telugu dataset in OntoNotes Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	40.69	80.24	**54.00**(83.6)	41.93	80.99	**54.86**(84.1)
D_IB	40.97	79.17	53.32(85.4)	41.71	79.57	54.49(86.9)

TAFSIL Telugu dataset in HAnDS Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	55.97	82.40	**67.83**(94.2)	56.28	83.14	**68.10**(90.0)
D_IB	55.44	82.21	67.11(91.9)	55.73	83.12	67.91(91.1)

TAFSIL Telugu dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	52.06	81.34	**64.22**(105)	52.47	82.82	**64.94**(103.1)
D_IB	50.96	80.93	63.92(106)	51.03	81.55	64.32(104.5)

Table 3.12: Performance of the different models trained on Urdu datasets in multiple taxonomies.

TAFSIL Urdu dataset in FIGER Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	60.60	73.98	66.61(95.4)	61.93	74.19	**67.51**(93.8)
D_IB	59.58	75.53	**66.62**(96.3)	60.40	75.88	67.25(94.0)

TAFSIL Urdu dataset in OntoNotes Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	50.08	79.78	**60.94**(92.9)	49.30	82.86	**61.82**(91.4)
D_IB	49.05	79.00	60.52(93.1)	49.08	82.19	61.46(91.3)

TAFSIL Urdu dataset in HAnDS Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	62.40	84.83	**71.90**(91.8)	67.62	79.78	**73.20**(94.6)
D_IB	62.71	83.15	71.50(93.0)	66.82	79.21	72.49(95.1)

TAFSIL Urdu dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Model	P	R	F1(↑Δ%)	P	R	F1(↑Δ%)
D_XLM	60.17	78.34	**68.07**(90.8)	63.04	77.07	**69.35**(92.8)
D_IB	56.97	83.41	67.70(92.3)	58.74	83.89	69.09(96.2)

Table 3.13: Performance of the D_XLM model trained on English datasets in various taxonomies.

TAFSIL English dataset in FIGER Taxonomy

	Micro			Macro		
Dataset	P	R	F1(↓Δ%)	P	R	F1(↓Δ%)
Public	65.66	95.59	77.84(-)	72.18	95.79	82.33(-)
TAFSIL	73.33	80.90	76.93(1.2)	79.83	84.40	82.05(0.3)

TAFSIL English dataset in OntoNotes Taxonomy

	Micro			Macro		
Dataset	P	R	F1(↓Δ%)	P	R	F1(↓Δ%)
Public	62.15	65.69	63.87(-)	65.73	69.00	67.33(-)
TAFSIL	62.24	62.32	62.28(2.5)	65.77	66.53	66.15(1.8)

TAFSIL English dataset in HAnDS Taxonomy

	Micro			Macro		
Dataset	P	R	F1(↓Δ%)	P	R	F1(↓Δ%)
Public	74.46	91.30	82.02(-)	78.71	91.18	84.49(-)
TAFSIL	73.43	91.96	81.66(0.5)	75.11	91.75	82.60(2.2)

TAFSIL English dataset in MultiCoNER2 Taxonomy

	Micro			Macro		
Dataset	P	R	F1(↓Δ%)	P	R	F1(↓Δ%)
Public	75.73	80.20	77.90(-)	76.88	84.86	80.67(-)
TAFSIL≡	57.05	63.50	60.10(22.8)	57.26	63.54	60.22(25.3)
TAFSIL†	72.95	82.86	77.28(0.8)	75.73	80.82	79.18(1.8)

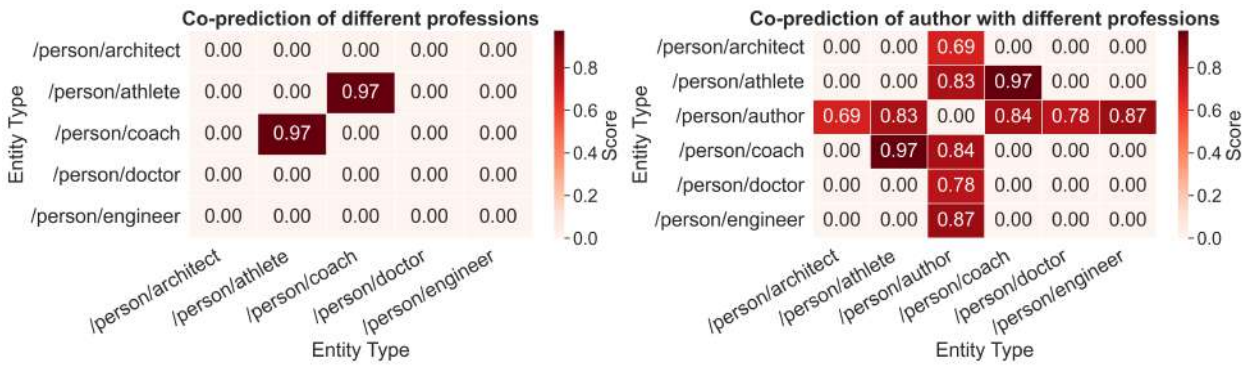

Figure 3.3: Co-prediction of different profession (/person fine-entity types). Confusion between */person/athlete* and */person/coach* is inevitable, as most coaches were once players, so Wikipedia-derived datasets often contain overlapping information. Similarly, because Wikipedia mainly covers notable personalities, many are also authors of books, blogs, or other media, leading to */person-/author* mappings for such entities.

The pattern of taxonomy is evident in all languages. OntoNotes' three-level hierarchy makes learning harder, resulting in lower F1 scores. Dataset size also plays a key role as larger Urdu datasets [Table 3.12] train FgNER models more effectively than smaller Sanskrit datasets [Table 3.9]. Overall, TAFSIL datasets trained models achieve an 83% improvement in average F1 over zero-shot performance across taxonomies.

3.6.4 TAFSIL on English

Although the TAFSIL framework is designed for Indian languages only, we have generated datasets in English language and performed rigorous experiments (Table 3.13). When compared to equal-sized noisy public datasets (FIGER [2], OntoNotes [27], and HAnDS [22]), our datasets are almost equivalent, with a maximum of 2.5% reduction in F1 scores. This establishes the language adaptability of the TAFSIL framework.

3.6.5 Importance of Dataset Size

Similar to the analysis on Hindi language as described in Table 3.6, we extend our analysis to the English dataset in MultiCoNER2 taxonomy to evaluate the importance of the large size of datasets. The TAFSIL≡ train set in MultiCoNER2 taxonomy is of the same size (16,778 sentences) as the MultiCoNER2 train set in English. Whereas TAFSIL† train set has one million sentences. By comparing models trained on different dataset sizes (Tables 3.6 and 3.13), we demonstrate that a large, noisy dataset, created through distant supervision method and trained with noise-aware models, can be as effective as a smaller clean dataset. This highlights the robustness of our approach and validates the quality of datasets created for other languages.

3.6.6 Error Analysis

Since these datasets are created using distant supervision methods, there remain some issues. One of the major problems is that these entity mentions are often tagged with multiple entity types instead of just one. Therefore, while training such noisy datasets, special FgNER models such as LITE or DECENT are required which are computationally heavy. Moreover, although the context

Table 3.14: Ablation study: Effect of Sentence selection (Stage 4) on the performance of the entity detection models on Hindi, Marathi & Sanskrit languages in multiple taxonomies. ¬**S4** means datasets created without the stage 4.

TAFSIL Hindi datasets

Taxonomy	Model	Micro F1	Micro ¬**S4**-F1(↓Δ%)	Macro F1	Macro ¬**S4**-F1(↓Δ%)
FIGER	D_XLM	67.28	66.18(1.66)	67.95	65.92(3.08)
	D_IB	67.18	65.88(1.97)	67.78	65.45(3.56)
OntoNotes	D_XLM	63.41	61.98(2.31)	63.70	62.17(2.46)
	D_IB	63.04	61.65(2.27)	63.32	61.91(2.28)
HAnDS	D_XLM	72.02	70.18(2.61)	72.90	72.79(2.98)
	D_IB	71.38	70.08(1.86)	73.13	71.30(2.56)
MultiCoNER2	D_XLM	69.97	68.19(2.61)	70.04	69.20(1.21)
	D_IB	71.15	69.36(2.58)	71.22	69.94(1.83)

TAFSIL Marathi datasets

Taxonomy	Model	Micro F1	Micro ¬**S4**-F1(↓Δ%)	Macro F1	Macro ¬**S4**-F1(↓Δ%)
FIGER	D_XLM	53.97	52.15(3.49)	54.56	53.39(2.19)
	D_IB	52.71	51.65(2.04)	53.79	52.42(2.61)
OntoNotes	D_XLM	50.70	49.46(2.53)	51.07	49.67(2.81)
	D_IB	49.88	48.77(2.28)	50.71	49.23(3.00)
HAnDS	D_XLM	62.95	61.71(2.00)	65.79	64.25(2.40)
	D_IB	62.41	60.19(3.69)	64.91	63.73(1.86)
MultiCoNER2	D_XLM	62.10	60.70(2.30)	62.28	60.89(2.29)
	D_IB	61.23	60.02(2.02)	61.46	60.27(1.98)

TAFSIL Sanskrit datasets

Taxonomy	Model	Micro F1	Micro ¬**S4**-F1(↓Δ%)	Macro F1	Macro ¬**S4**-F1(↓Δ%)
FIGER	D_XLM	36.18	35.13(3.00)	36.85	36.22(1.74)
	D_IB	37.16	36.40(2.07)	37.53	36.56(2.67)
OntoNotes	D_XLM	28.07	27.26(2.96)	28.95	27.47(5.39)
	D_IB	30.50	29.27(4.20)	31.23	29.95(4.29)
HAnDS	D_XLM	42.05	41.60(1.08)	43.09	41.85(2.97)
	D_IB	44.36	42.70(3.88)	45.24	43.26(4.57)
MultiCoNER2	D_XLM	40.20	38.66(3.98)	40.56	39.46(2.81)
	D_IB	41.03	39.20(4.68)	41.47	40.20(3.17)

dependency is taken care of by such models, some entity mentions are co-learned and co-predicted along with other entity types.

As shown in the correlation graph in Figure 3.3, most the entity types describing different professions are not confused except the */person/player* and */person/coach*. Such confusion is quite imminent as every coach had been a player before and hence the dataset created from Wikipedia will contain such information. Depending on the entity mention, multiple entity types may get mapped to the same entity mention.

Similarly, when the */person/author* entity type is included in the profession list, the co-prediction pattern changes as most of the fine entity types of */person* coarse type are co-predicted

Table 3.15: Ablation study: Effect of Sentence selection (Stage 4) on the performance of the entity detection models on Tamil, Telugu & Urdu languages in multiple taxonomies. ¬**S4** means datasets created without the stage 4.

TAFSIL Tamil datasets

Taxonomy	Model	Micro F1	Micro ¬**S4**-F1(↓Δ%)	Macro F1	Macro ¬**S4**-F1(↓Δ%)
FIGER	D_XLM	55.23	53.86(2.55)	55.83	54.03(3.33)
	D_IB	54.25	52.94(2.47)	55.21	53.61(2.99)
OntoNotes	D_XLM	52.63	51.49(2.21)	53.93	52.16(3.38)
	D_IB	52.96	51.20(3.45)	53.00	51.43(3.05)
HAnDS	D_XLM	66.87	65.12(2.69)	67.13	65.28(2.83)
	D_IB	66.11	64.32(2.79)	66.34	64.14(3.43)
MultiCoNER2	D_XLM	63.41	61.35(3.35)	64.02	62.17(2.98)
	D_IB	62.91	61.13(2.91)	63.12	61.27(3.02)

TAFSIL Telugu datasets

Taxonomy	Model	Micro F1	Micro ¬**S4**-F1(↓Δ%)	Macro F1	Macro ¬**S4**-F1(↓Δ%)
FIGER	D_XLM	57.48	56.32(2.05)	58.67	56.65(3.55)
	D_IB	57.26	55.75(2.71)	57.80	56.21(2.83)
OntoNotes	D_XLM	54.00	52.27(3.30)	54.86	53.61(2.33)
	D_IB	53.32	52.14(2.26)	54.49	53.17(2.48)
HAnDS	D_XLM	67.83	65.87(2.98)	68.10	66.22(2.84)
	D_IB	67.11	65.27(2.82)	67.91	66.02(2.87)
MultiCoNER2	D_XLM	64.22	62.18(3.28)	64.94	63.12(2.88)
	D_IB	63.92	61.29(4.29)	64.32	62.25(3.32)

TAFSIL Urdu datasets

Taxonomy	Model	Micro F1	Micro ¬**S4**-F1(↓Δ%)	Macro F1	Macro ¬**S4**-F1(↓Δ%)
FIGER	D_XLM	66.62	64.35(3.53)	67.51	65.71(2.74)
	D_IB	66.61	64.13(3.87)	67.25	65.89(2.07)
OntoNotes	D_XLM	60.94	59.56(2.32)	61.82	60.18(2.71)
	D_IB	60.52	58.75(3.02)	61.46	59.96(2.50)
HAnDS	D_XLM	71.90	70.19(2.44)	73.20	71.39(2.53)
	D_IB	71.50	69.97(2.18)	72.49	70.54(2.78)
MultiCoNER2	D_XLM	68.07	66.54(2.29)	69.35	67.94(2.08)
	D_IB	67.70	65.94(2.67)	69.09	67.73(2.01)

with the */person/author* (Figure 3.3). Since Wikipedia articles are mostly about famous personalities, many of such personalities must have been authors of books, blogs, or other media, which leads to */person/author* type mapping to such entity mentions. Apart from such closely related fine entity types, most of the models learn very efficiently when trained on TAFSIL datasets to recognize the fine entity types distinctly without much confusions.

3.6.7 Ablation Study

The stage 4 of the TAFSIL framework is extremely important as the removal of sentences having undetected entities improves the FgNER model training. Table 5.5 suggests the significance of stage 4 in all the languages across all the taxonomies. Without stage 4 (shown as ¬**S4** in the table 5.5), the datasets contain some undetected entity mentions, leading to deterioration in model training. Stage 4 ensures that the final datasets must contain sentences having all the mentions detected and mapped to the correct entity types.

3.7 Discussion

Since the TAFSIL datasets are constructed using Wikipedia, it may limit their representation of diverse text genres, potentially impacting the performance of machine learning models in cross-domain training and evaluation. Additionally, the size of the dataset is inherently constrained by the volume of Wikipedia content. For many languages, Wikipedia does not exist, which limits the applicability of the proposed TAFSIL framework. As entity types are mapped to Wikidata key values, the TAFSIL framework may not be adaptable to taxonomies where entity types cannot be aligned with Wikidata. Furthermore, the effectiveness of Stage 4 is contingent on the availability of high-quality POS taggers, posing a limitation for languages lacking reliable POS tagging resources.

3.8 Conclusion

In this work, we present TAFSIL, a robust and efficient framework for generating large-scale, fine-grained entity typing datasets across multiple Indian languages and taxonomies. We have successfully created 24 different datasets, demonstrating the adaptability and effectiveness of our approach. We also created 24 high-quality manually annotated gold datasets corresponding to the languages and taxonomies considered. Our experiments on SOTA models validate the quality and utility of these datasets, paving the way for advancements in fine-grained named entity recognition in Indian languages.

The TAFSIL dataset is publicly available at hf.co/datasets/prachuryyaIITG/TAFSIL

4

Fine-grained Named Entity Recognition through Annotation Projection

Chapter Highlights

- This chapter focuses on solving the critical lack of fine-grained Named Entity Recognition (FgNER) resources for the vulnerable languages of India's North Eastern Region.
- Since the taxonomy-adaptable distant supervision framework discussed in the last chapter is limited to the unavailability of Wikipedia in vulnerable low-resource languages, this work details the creation of essential FgNER datasets for the low-resource languages Bodo, Manipuri, and Mizo, spoken by over six million people across South and Southeast Asia.
- This chapter presents a methodology that utilizes annotation projection from high-resource FgNER datasets via parallel corpora and a multilingual encoder tool to generate new FgNER data.
- The work includes the validation of the new, high-quality dataset, and an exploration of zero-shot and cross-lingual FgNER performance, examining the impact of script similarity and multilingualism.
- This chapter is based on the publication "FiNERVINER: Fine-grained Named Entity Recognition for Vulnerable languages of India's North Eastern Region" published in "Proceedings of the Fifteenth biennial Language Resources and Evaluation Conference (LREC) 2026".
- The FiNERVINER dataset, associated fine-tuned models and interactive web application are publicly available at hf.co/collections/prachuryyaIITG/finerviner.

4.1 Introduction

Despite considerable progress in coarse-grained NER for Indian languages [34, 75, 86, 87], the development of FgNER resources for these languages remains in its early stages. Recent initiatives, such as the MultiCoNER2 shared task at SemEval-2023 [131], have introduced FgNER datasets for Hindi and Bengali through translated English annotations [3]. Similarly, the TAFSIL initiative generated noisy FgNER data for six other Indian languages by mining Wikipedia links and Wikidata [29]. Since Wikipedia is unavailable for many vulnerable languages such as Bodo, Manipuri, and

Mizo, the TAFSIL framework described in the Chapter 3 could not be applied to create datasets in such low-resource languages. Therefore, comprehensive and high-quality fine-grained resources remain scarce for most low-resource Indian languages.

To address this gap, we create FiNERVINER: Fine-grained Named Entity Recognition dataset for Vulnerable languages of India's North Eastern Region. We have projected the FgNER annotations of the MultiCoNER2 English dataset to the target languages, utilizing the parallel corpora and a multilingual encoder-based annotation projection and word alignment tool [142, 143]. The vulnerable languages [127] considered are: *Bodo/Boro (brx)*, *Manipuri/Meitei (mni)*, and *Mizo/Lushai (lus)*, and are regarded as one of the official languages in the Indian states of Assam, Manipur, and Mizoram, respectively. These Sino-Tibetan languages are spoken by more than six million people across five countries in South and Southeast Asia. To the best of our knowledge, this is the first FgNER dataset created for these three vulnerable languages. Our contributions can be summarized as follows:

1. Construction of a large-scale FgNER dataset FiNERVINER comprising over a total of 700k sentences, 963k entity mentions, and 11M tokens for three low-resource and vulnerable Indian languages: Bodo/Boro (brx), Manipuri/Meitei (mni), and Mizo/Lushai (lus).
2. Creation of high-quality human-annotated test sets consisting of 1000 sentences for each language with inter-annotator agreement (κ) above 0.81.
3. Our rigorous analyses establish the good quality of the generated dataset.
4. Zero-shot and cross-lingual analysis to examine the influence of multilingualism and script similarities on cross-lingual FgNER performance.

4.2 Related Works

Sekine and Nobata (2002) [96] pioneered fine-grained entity classification with a hierarchy of 150 types, and Ling and Weld (2012) [2] advanced fine-grained named entity recognition (FgNER) by defining 113 types in a two-level hierarchy. Since then, FgNER datasets have varied in entity type count and complexity: e.g. ACE has 52 types [39], BBN has 93 [97], OntoNotes has 88 [27], and HYENA has 505 [98]. Meanwhile, large-scale resources like WikiSense [99], FINET [100], UFET [23], and TypeNet [101] offer thousands of categories. [22] improved quality with language-specific heuristics and refined selection, whereas [33] provides a large, manually annotated dataset covering 66 fine-grained types.

Yarowsky et. al [66] pioneered entity projection using parallel corpora and word alignment, yet zero-shot transfer remains limited for linguistically distant languages [81], especially when structural and word order differences are significant [80]. Annotation projection has been widely used for dependency parsing [67], relation extraction [68], and NER [74, 123]. To address the need for larger, more diverse datasets in Indian languages, Naamapadam [75] was developed for 11 Indian languages via annotation projection from English NER data, utilizing Samanantar parallel corpora [77] and word alignment techniques [78, 79].

An early step in developing NER resources for Indian languages was the IJCNLP-2008 dataset [59], which provided data for Hindi, Bengali, Oriya, Telugu, and Urdu and became a key resource in early Hindi NER research [60, 61, 62, 63, 64]. The FIRE-2014 dataset [65] expanded coverage to Tamil and Malayalam using sources like Wikipedia, blogs, and forums. Polyglot NER further contributed by covering over a hundred languages [16]. WikiANN spans 282 languages, including 16 Indian languages [17]. Subsequent manual annotation efforts produced datasets such as HiNER [34], AsNER [86], MahaNER [87], EverestNER [88], OurNepali [144], NER in Bengali [89], NER in Telugu [90], Maithili [94], Bishnupriya Manipuri [91, 92], Bodo [93], etc. MultiCoNER-1 [24] and

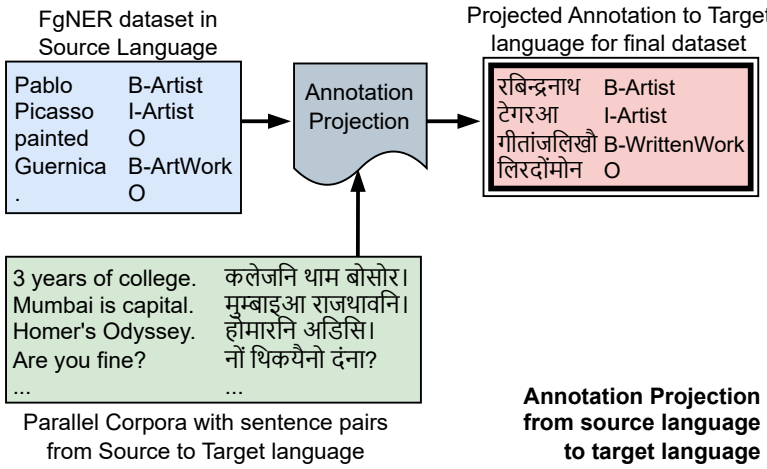

Figure 4.1: Annotation projection from FgNER dataset in source language to target language utilizing parallel corpora with annotation projection and word alignment tool.

MultiCoNER-2 [3] created datasets in Hindi and Bengali, by translating English coarse-grained and fine-grained NER datasets, respectively. In the SemEval-2023 shared task, multiple teams [103, 104, 105, 106] further enhanced these resources in Hindi and Bengali. The TAFSIL framework [29] as described in Chapter 3 Recentlyutilized Wikipedia links and Wikidata to produce noisy FgNER datasets covering six Indian languages (Hindi, Marathi, Sankrit, Tamil, Telugu, and Urdu) across four taxonomies. Despite these advancements, resources in vulnerable Indian languages remain scarce, underscoring the need for more multilingual FgNER datasets.

4.3 Annotation Projection

As illustrated in the Figure 4.1 and Algorithm 2, the annotation projection method is a three-component process that leverages a high-resource source language (src). This language provides both an annotated FgNER dataset $D_{src} = \{(s_{src}^{(i)}, y_{src}^{(i)})\}_{i=1}^{N}$ and a parallel corpus $PC = \{(s_{src}^{(i)}, s_{tgt}^{(i)})\}_{i=1}^{N}$ that comprises sentence pairs aligned with the target language (tgt).

The projection process is executed for each parallel pair $(s_{src}, s_{tgt}) \in PC$ and involves the following steps:

1. **Embedding Computation**: A pretrained multilingual encoder M is fine-tuned on the parallel corpus PC to produce contextual embeddings $\mathbf{EB}_{src} = ME(s_{src})$ and $\mathbf{EB}_{tgt} = ME(s_{tgt})$ for the source and target sentences, respectively.

2. **Alignment Mapping**: A word alignment service, $\mathcal{A}(s_{src}, s_{tgt})$, leverages these embeddings to generate a mapping, $mapping$, from source-sentence tokens to target-sentence tokens:
$mapping = \mathcal{A}(s_{src}, s_{tgt})$ using $\mathbf{EB}_{src}$ & $\mathbf{EB}_{tgt}$.

3. **Annotation Projection**: The source annotation sequence y_{src} is projected to the target sentence by setting the annotation for each token j in s_{tgt} as $y_{tgt}^{(j)} = y_{src}^{(mapping^{-1}(j))}$. This operation is performed iteratively for all tokens in the target sentence, yielding a complete annotation sequence y_{tgt}.

4. **Dataset creation for target language**: The newly created pair (s_{tgt}, y_{tgt}) is included to create the final FgNER dataset in the target language, D_{tgt}.

Algorithm 2: Annotation Projection

Symbols:

- ME: Pretrained multilingual encoder.
- N: Number of sentences per dataset.
- y: List containing FgNER annotations for each token of sentence s.
- $D_l = \{(s_l^{(i)}, y_l^{(i)})\}_{i=1}^{N}$: FgNER dataset of size N in language l.
- src, tgt: Source and Target languages respectively.
- $PC = \{(s_{src}^{(i)}, s_{tgt}^{(i)})\}_{i=1}^{N}$: Parallel corpora.
- $\mathcal{A}(s_{src}, s_{tgt})$: Annotation alignment service produces a *mapping*.
- y_{tgt}: Annotation sequence for s_{tgt}.
- $D_{tgt} = \{(s_{tgt}^{(i)}, y_{tgt}^{(i)})\}_{i=1}^{N}$: Final target FgNER dataset of size N.

Algorithm:

1. **for** each $(s_{src}, s_{tgt}) \in PC$ **do**
2. **Compute embeddings:** $\mathbf{EB}_{src} = ME(s_{src})$, $\mathbf{EB}_{tgt} = ME(s_{tgt})$
3. **Obtain alignment mapping:** $mapping = \mathcal{A}(s_{src}, s_{tgt})$ using $\mathbf{EB}_{src}$ & $\mathbf{EB}_{tgt}$
4. **for** each token j in s_{tgt} **do**
5. **Project annotation:** $y_{tgt}^{(j)} = y_{src}^{(mapping^{-1}(j))}$
6. **end for**
7. Update (s_{tgt}, y_{tgt}) to D_{tgt}
8. **end for**

4.3.1 Implementation Details

We use the MultiCoNER2 [3] dataset, which provides 33 fine-grained entity types across 12 languages. For our setup, English served as the source language (src), while the target languages (tgt) are Bodo (brx), Manipuri (mni), and Mizo (lus). To obtain the necessary parallel corpora (PC), we used the BPCC corpora [145] for Manipuri and Bodo, which is augmented with additional resources from [146, 147]. The parallel corpus used for Mizo is the Mizo–English parallel corpus [148]. Following Naamapadam [75], we adopted AWESoME align [142] as $\mathcal{A}$, fine-tuned on the English MultiCoNER2 dataset (D_{src}). Finally, we used different multilingual encoders (ME) for our target languages. For Bodo and Manipuri, we fine-tuned IndicBERTv2 [149], as it is the only encoder pretrained on all three languages (English, Bodo, and Manipuri). Since the Mizo language is written using the Latin script (similar to English), the MizBERT [122] utilized the BERT-base [82] tokenizer on English and then pre-trained on a large-scale Mizo corpus [122]. Therefore, we have used MizBERT as multilingual encoders (ME) for our target language Mizo. With the implementation as mentioned above, the FiNERVINER dataset is generated.

Table 4.1: FiNERVINER dataset statistics. **Sent**, **Ent** and **Tkn** means number of Sentences, Entities and Tokens respectively.

	Train Set			Development Set			Test Set		
Language	**Sent**	**Ent**	**Tkn**	**Sent**	**Ent**	**Tkn**	**Sent**	**Ent**	**Tkn**
Bodo (brx)	212,835	302,713	2,958,455	23,649	33,808	329,145	1,000	1,423	14,082
Manipuri (mni)	177,224	252,767	2,515,386	19,692	28,143	279,681	1,000	1,384	14,330
Mizo (lus)	239,813	302,713	4,422,373	26,646	38,330	484,212	1,000	1,426	18,765

Table 4.2: FiNERVINER Gold Test Set. **IAA** (κ) gives the inter-annotator agreement.

Language	Bodo	Manipuri	Mizo
IAA (κ)	0.875	0.811	0.821

4.4 FiNERVINER Dataset

As shown in Table 5.1, the created FiNERVINER dataset consists of more than 198 thousand sentences, 282 thousand entity mentions, and 2.8 million tokens in each of the three low-resource vulnerable Indian languages. We randomly pick 1000 sentences from the dataset for manual annotation. From the rest of the dataset, 10% is considered as the development set and the remaining as the training set.

4.4.1 Gold Test Set

Volunteer annotators, who have a minimum education of an undergraduate degree, are chosen based on their mother tongue. We compute inter-annotator agreement (IAA) on the 1000 sentences annotated by two annotators for each language. As shown in the Table 4.2, the good quality of these gold datasets can be ascertained based on the Cohen's kappa coefficient (κ) [150], which is above 0.81 for each language.

4.4.2 Entity Type Frequency Distribution

As shown in Figure 4.2, more number of entity mentions are detected for the fine types of Person (e.g. Artist) and Location (e.g. HumanSettlement). Since Artist type includes the mentions of musicians, actors, directors, authors, etc, the count is the highest among all the fine entity types. Similarly, very specific fine types such as AerospaceManufacturer, Drink, AnatomicalStructure, etc., are very scarce. Similar trends can be observed across all three languages.

4.5 Experimental Setup

The state-of-the-art approach for sequence labeling tasks involves fine-tuning pre-trained language models (PLM) with the NER datasets [24, 34, 75, 86, 87, 88, 95, 123, 124]. Similarly, we have fine-tuned mBERT (bert-base-multilingual-cased) [82], IndicBERTv2 (IndicBERTv2-MLM-Sam-TLM) [149], MuRIL (muril-large-cased) [84], XLM-RoBERTa (XLM-RoBERTa-large) [83], and MizBERT [122] for fine-grained NER using the Hugging Face Transformers[1] library. The models were trained for six epochs with a batch size of 32, utilizing AdamW optimization (learning rate: 5e-5, weight

[1]https://huggingface.co/docs/transformers

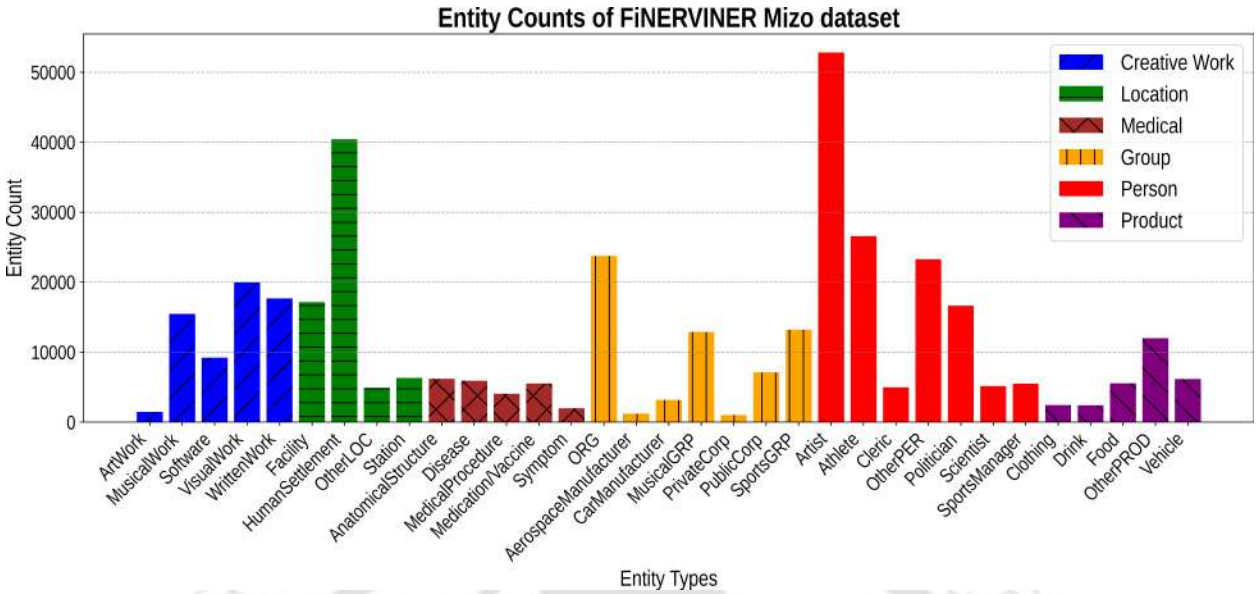

Figure 4.2: Entity type frequency distribution of the FiNERVINER Mizo dataset. Each color depicts the coarse-grained type, and each bar depicts the corresponding fine-grained types.

decay: 0.01). Training was performed on an NVIDIA A100 GPU, with evaluation based on SeqEval metrics, and the best performance determined by the F1-score.

4.6 Results & Analysis

The following subsections cover the analyses of PLM's performance fine-tuned on the FiNERVINER dataset, cross-lingual zero-shot evaluation, the Impact of multilingualism and script similarity, and entity-specific error analyses. The best model performance values (F1 scores) for every language are in **bold**, and the second-best values are underlined.

4.6.1 Performance of PLMs Fine-tuned on FiNERVINER Dataset

Multilingual encoder model IndicBERTv2 is the only PLM pre-trained on Bodo (brx) and Manipuri (mni). Table 4.3 shows the performance of IndicBERTv2 fine-tuned on FiNERVINER is superior. Similarly, the MizBERT's performance is superior when fine-tuned on Mizo (lus) as it is pre-trained on the same language. Although the multilingual PLMs mBERT, XLM-RoBERTa and MuRIL were not pre-trained on any of these three language, their performance is good after fine-tuned with the high-quality FiNERVINER dataset. In fact, an interesting result is observed in the case of the Mizo language. The Mizo language is written using the Latin script (similar to English, German, Spanish, French, Italian, Portuguese, etc.), and Bodo and Manipuri are written using Devanagari and Meitei scripts, respectively. Although the Mizo language was unseen during pre-training of mBERT, XLM-RoBERTa, MuRIL, and IndicBERTv2, the models could perform significantly better than other languages in the FgNER task due to the massive pre-training on the languages written using the Latin script. Also, since the MizBERT utilized BERT-base tokenizer on English and then pre-trained on only the Mizo language, its performance on Mizo is superior but quite inferior on the languages written with other scripts than the Latin script [Table 4.3]. Based on these observations, we have conducted cross-lingual zero-shot analysis, which is described

Table 4.3: Performance of different models fine-tuned on FiNERVINER dataset. The best model performance values (in terms of F1 scores) for every language are in **bold**, and the second-best values are underlined.

Language	Model	Micro			Macro		
		P	R	F1	P	R	F1
Bodo (brx)	mBERT	59.04	61.20	60.17	59.33	62.11	60.61
	XLM-RoBERTa	61.02	62.17	61.82	60.04	61.73	60.89
	MuRIL	60.96	63.31	61.98	60.22	63.08	61.46
	IndicBERTv2	63.85	67.57	**65.66**	62.37	65.57	**64.02**
	MizBERT	41.31	38.96	40.10	39.81	37.61	38.68
Manipuri (mni)	mBERT	56.63	57.14	56.89	56.40	56.77	56.59
	XLM-RoBERTa	57.41	58.72	58.06	56.93	57.77	57.34
	MuRIL	58.32	60.04	59.17	57.16	59.28	58.11
	IndicBERTv2	60.49	64.42	**62.39**	61.39	63.34	**62.02**
	MizBERT	39.37	37.07	38.19	38.76	35.97	37.32
Mizo (lus)	mBERT	79.73	80.99	80.36	79.53	79.43	79.25
	XLM-RoBERTa	80.33	81.83	**81.07**	81.30	80.92	**80.89**
	MuRIL	78.51	81.36	79.91	77.19	77.60	77.02
	IndicBERTv2	76.66	78.62	77.63	77.84	76.81	76.61
	MizBERT	79.51	81.11	80.30	78.29	80.28	79.04

in the following section.

4.6.2 Entity Type Specific Performance of PLMs Fine-tuned on FiNERVINER Dataset

Table 4.4 shows the entity type-specific F1 scores of the best fine-tuned model for each language. Performance generally improved for entity types with more training samples (e.g., *HumanSettlement*). Interestingly, the low performance for the well-sampled *OtherPER* entity type is likely due to confusion with finer-grained entity types like Artist, Athlete, and Politician, as detailed in the 4.6.6 Error Analysis section.

4.6.3 Cross-Lingual Zero-Shot Analysis

We have performed cross-lingual zero-shot analysis for every single language pair. As shown in Figure 4.3, the models are fine-tuned on datasets of respective languages and tested on the test set of other languages. As expected, the best performance of a model is found when it is fine-tuned on a particular language and tested on the same language. A model's performance is quite inferior on an unseen language. For example, when mBERT, XLM-RoBERTa, and MuRIL are fine-tuned only on Bodo (brx) or Mizo (lus), their zero-shot performance is very poor on the Manipuri (mni) test set. Similarly, when these models are models are fine-tuned on Manipuri (mni), their zero-shot performance on Bodo (brx) is quite inferior. But, interestingly, due to the pre-training on languages written using the Latin script, the zero-shot performance on Mizo (lus) is comparatively better. Moreover, if the PLM is pre-trained on a language, then its zero-shot performance improves. This is observed in the case of IndicBERTv2. Since IndicBERTv2 is pre-trained on 26 languages, including Manipuri, Bodo, and English, its zero-shot performance is better than the other three multilingual PLMs.

Table 4.4: Entity-type specific performance of the best fine-tuned models fine-tuned on each language in the FiNERVINER dataset.

	Bodo (brx)			Manipuri (mni)			Mizo (lus)		
Entity Type	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
AerospaceManufacturer	69.61	51.74	58.23	69.23	56.25	62.07	94.12	84.21	88.89
AnatomicalStructure	80.77	75.01	77.78	75.00	78.95	76.92	95.83	98.12	97.23
ArtWork	60.71	36.42	44.74	65.02	35.14	45.61	90.91	66.67	76.93
Artist	72.40	80.51	76.24	71.90	80.32	75.88	82.89	85.16	84.01
Athlete	74.60	77.99	76.26	75.18	78.28	76.70	76.54	81.58	78.98
CarManufacturer	56.72	67.86	61.79	44.23	67.65	53.49	88.89	94.12	92.39
Cleric	65.52	51.82	57.87	52.94	43.90	48.02	61.54	53.33	57.14
Clothing	61.36	61.36	61.36	59.46	52.38	55.70	98.89	94.44	97.14
Disease	61.26	60.18	60.71	66.67	64.10	65.36	83.33	75.00	78.95
Drink	63.04	53.70	58.02	65.51	50.02	56.76	98.00	98.00	98.00
Facility	69.96	73.24	71.56	70.52	75.31	72.84	87.04	85.45	86.24
Food	67.74	61.32	64.37	52.54	57.41	54.87	84.62	95.65	89.80
HumanSettlement	86.25	86.91	86.58	83.77	85.25	84.25	91.20	95.80	93.44
MedicalProcedure	68.37	69.79	69.07	64.44	70.73	67.44	95.00	98.04	97.34
Medication/Vaccine	76.82	84.67	80.56	65.22	75.00	69.77	94.35	95.90	95.12
MusicalGRP	61.45	70.34	65.60	63.64	71.80	67.47	82.50	89.19	85.71
MusicalWork	73.11	70.78	71.93	69.18	69.59	69.36	79.17	77.55	78.35
ORG	63.73	66.39	65.03	64.85	64.58	64.35	73.86	87.50	80.03
OtherLOC	55.00	48.53	51.56	46.90	50.75	48.75	58.51	60.44	59.46
OtherPER	46.83	32.74	38.54	36.00	30.08	32.83	42.85	35.29	38.71
OtherPROD	49.62	48.53	49.07	51.14	45.03	47.97	93.18	85.17	89.13
Politician	52.61	50.68	51.63	60.13	53.80	56.79	63.16	69.23	66.06
PrivateCorp	46.66	50.72	48.53	47.19	48.80	47.98	84.62	73.33	78.57
PublicCorp	45.86	52.75	49.48	49.27	53.97	51.55	80.77	72.41	76.36
Scientist	49.10	53.37	51.14	46.90	50.75	48.75	58.51	60.44	59.46
Software	61.86	67.42	64.52	58.82	66.67	62.50	83.33	96.15	89.29
SportsGRP	86.49	86.49	86.49	88.28	86.92	87.60	90.00	90.00	90.00
SportsManager	75.44	58.90	66.15	77.27	62.97	69.39	84.62	68.75	75.86
Station	79.76	79.76	79.76	72.31	82.46	77.05	94.74	90.00	92.31
Symptom	58.33	57.14	57.73	71.05	60.04	65.26	93.75	93.75	93.75
Vehicle	54.42	54.42	54.42	45.61	60.47	52.05	81.82	85.71	83.72
VisualWork	70.08	72.25	71.15	68.57	67.80	68.19	78.33	78.33	78.33
WrittenWork	77.29	75.32	76.29	75.72	80.87	78.22	83.93	88.68	86.24

We have extended the zero-shot analysis to Bengali (bn), English (en), and Hindi (hi) languages, utilizing the publicly available MultiCoNER2 dataset [3]. Since all these multilingual PLMs are pre-trained on these three high-resource languages, their zero-shot performances are superior compared to the zero-shot performances of the low-resource vulnerable languages.

4.6.4 The Impact of Script Similarity

We further investigate the impact of script similarity for the FgNER task. As already discussed previously, the influence of the Latin script is imminent on the Mizo language. Similar observations

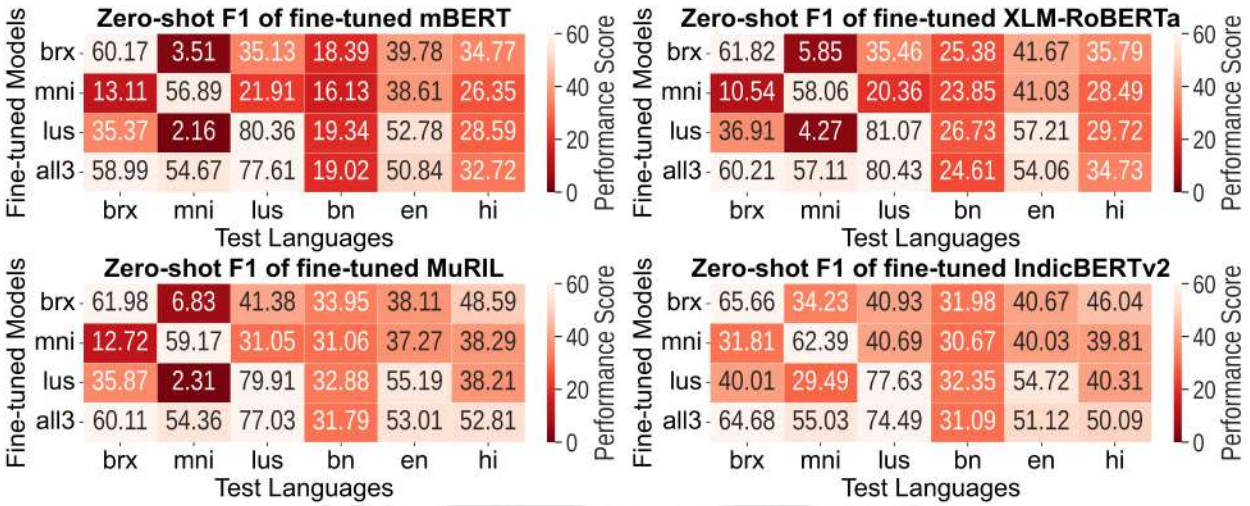

Figure 4.3: Zero-shot performance (micro F1) of fine-tuned mBERT, XLM-RoBERTa, MuRIL, and IndicBERT models on different test languages.

are shown in Fig. 4.3 in the case of English (en) and Hindi (hi). As Bodo (brx) and Hindi (hi) are written using the Devanagari script, the zero-shot performance of Hindi improves on models fine-tuned on Bodo due to the script similarity. Similarly, we observe that the zero-shot performance of English is superior in the case of Mizo compared to other fine-tuned models due to the Latin script similarity. Named entities are nouns that mostly do not change spellings based on the language. Therefore, if two languages are written using the same script, then even though there may be grammatical and linguistic divergences, the detection of the entities is facilitated by the unchanged spelling. This is the reason why the zero-shot performance on languages written with a similar script improves. This is further established by the zero-shot performance of Bengali (bn), which is nearly consistent in any individual PLMs across all the fine-tuned variations. The effect of fine-tuned models is negligible as Bengali is written using a completely different script (Bengali-Assamese script), and its zero-shot performances are completely governed by the inclusion of Bengali during the pre-training of such PLMs.

4.6.5 Multilingualism

Our analysis extends to evaluating multilingualism. We constructed a balanced set (*all3*) comprising all three languages Bodo, Manipuri, and Mizo, ensuring an equal number of samples per language. As seen in the Figure 4.3, there is significant improvement over zero-shot performances in every encoder model when fine-tuned with all the languages and tested on individual languages. Since the *all3* train set has samples of Bodo and Mizo, the zero-shot performance of Hindi and English improved due to the script similarity as discussed in the previous section. These results suggest the necessity of language-specific pre-training and task-specific fine-tuning.

4.6.6 Error Analysis

Fine-grained named entity recognition is crucial, as entity types may vary by context. Therefore, we have analyzed the errors in two different approaches. Table 4.5 shows the details of entity errors in terms of the percentage of predicted entities. The common errors that occur include the boundary error (such as “Incredibles” is marked as *VisualWork* instead of “The Incredibles”),

Table 4.5: Entity errors in terms of the percentage of predicted entities for different languages fine-tuned on IndicBERTv2.

Entity Error Type	Bodo	Manipuri	Mizo
Boundary mismatch	9.27	11.46	4.43
Type mismatch	14.41	15.23	11.75
Spurious entity	2.67	2.21	0.84

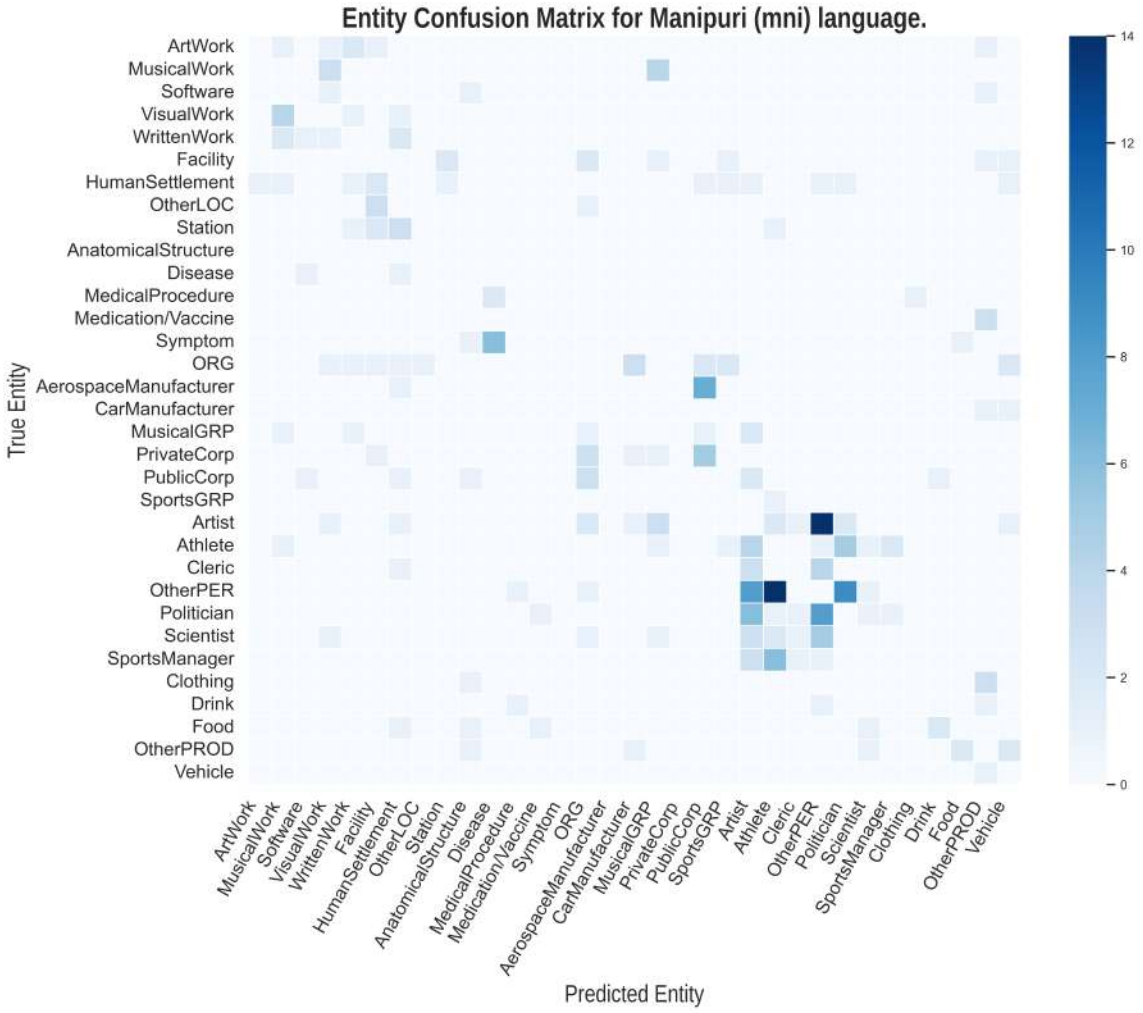

Figure 4.4: Entity type confusion matrix of Manipuri (mni) language. Darker squares depict the entity type pairs often confused by the fine-tuned model.

entity type mismatch error (e.g. "Nissan Cherry" is categorized as *Food* instead of *Vehicle*) and spurious errors (such a "blue" is marked as an entity whereas the entity type *color* is not defined in MultiCoNER2 taxonomy). Entity boundary mismatch errors and entity type mismatch erros are highest in Manipuri (mni), whereas spurious entity errors occur the most in Bodo (brx).

Moreover, we have analyzed the often co-predicted fine-grained types. From Table 4.5, we have selected Manipuri (mni) for this analysis as this language has the highest percentage of mismatch entity types. As shown in Figure 5.6 the fine types of *Artist*, *Athlete*, *Politician* are sometimes

confused with *OtherPER*. Similarly, *Symptom* is sometimes confused with *Disease*. Apart from such closely related fine entity types, most of the other fine entity types are learned by the models without much confusion.

4.7 Discussion

Despite the encouraging results of this study, it is important to acknowledge certain limitations that require further investigation. It is essential to recognize that the characteristics of the generated dataset are directly dependent on the characteristics of the source FgNER dataset; any inherent biases or specificities within this dataset have the potential to influence the generated dataset. Second, the volume and quality of the generated dataset are inherently linked to the availability of parallel corpora and the specific selection of the annotation projection tool and multilingual encoder models. Therefore, further investigation of the impact of different combinations of these tools and models remains a crucial direction for future research. Finally, a comprehensive analysis of the performance of Large Language Models (LLMs) on the FgNER task, both in a zero-shot setting and after fine-tuning with the FiNERVINER dataset, represents a crucial avenue for future work to benchmark our approach against state-of-the-art techniques.

4.8 Conclusion

The generated FiNERVINER dataset by projecting English MultiCoNER2 annotations to the target language, utilizing the parallel corpora and a multilingual encoder-based tool for annotation projection and word alignment. The dataset comprises over 198k sentences, 282k entities, and 2.8M tokens in each low-resource vulnerable language: Bodo, Manipuri, and Mizo. As the first FgNER dataset for these languages created via an annotation projection method, extensive experiments validate its quality. Cross-lingual zero-shot analyses underscore the need for language-specific pre-training and task-specific fine-tuning. Moreover, the impact of the script used for writing a language is imminent in these analyses, which paves the way for future research.

The FiNERVINER dataset, associated fine-tuned models and interactive web application are publicly available at hf.co/collections/prachuryyaIITG/finerviner

5

Cross-Lingual Annotation Projection Enhancement through Script Similarity for Fine-grained Named Entity Recognition

Chapter Highlights

- The chapter introduces CLASSER, a cross-lingual annotation projection framework enhanced through script similarity to create a high-quality Fine-grained Named Entity Recognition (FgNER) dataset for low-resource languages.
- Based on the observations from the previous chapter, the CLASSER framework employs a two-stage process that specifically leverages datasets from script-similar languages to refine and improve the projection of annotations from high-resource sources.
- The framework is successfully applied to generate a substantial dataset of over 1.8 million sentences for five low-resource Indian languages, including Assamese, Bodo, Marathi, Nepali, and Sanskrit.
- Rigorous analysis confirms the dataset's quality, demonstrating significant F1 score improvements (up to 26% in Marathi and 46% in Sanskrit) and providing insights into the impact of script similarity on cross-lingual FgNER performance.
- This chapter is based on the publication "CLASSER: Cross-lingual Annotation Projection enhancement through Script Similarity for Fine-grained Named Entity Recognition" published in "Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL 2025)".
- The CLASSER dataset, associated fine-tuned models and interactive web application are publicly available at hf.co/collections/prachuryyaIITG/CLASSER.

5.1 Introduction

While coarse-grained NER for Indian languages has seen considerable progress, fine-grained NER (FgNER) only began to emerge recently. The MultiCoNER2 [3] shared task at SemEval-2023 introduced FgNER datasets for Hindi and Bengali via translated English annotations [131]. As

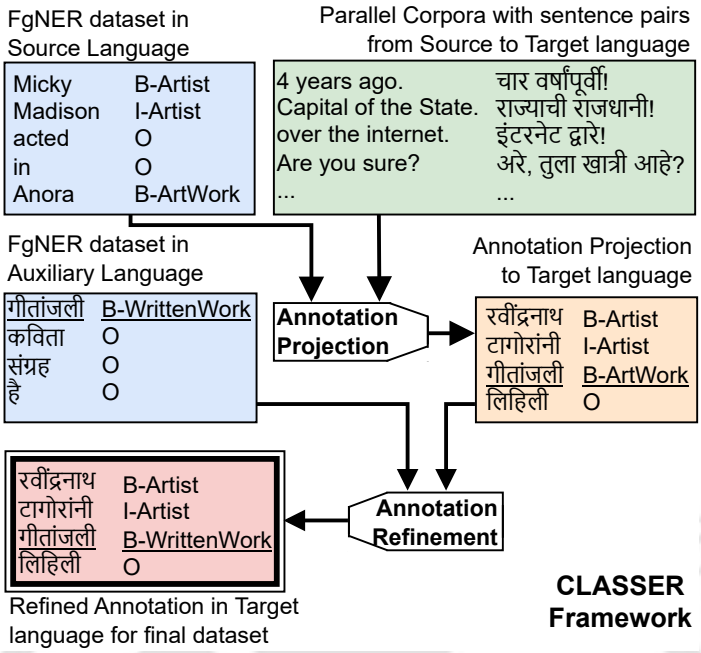

Figure 5.1: Illustration of CLASSER Framework. First, the annotations are projected from the FgNER dataset in the source language to the target language, utilizing parallel corpora with annotation projection and a word alignment tool. Second, the projected annotations are enhanced through FgNER datasets in script-similar languages.

described in Chapter 3, the distant supervision framework TAFSIL was used to create large-scale FgNER datasets in various Indian languages utilizing Wikipedia and Wikidata. But such datasets are often noisy, their size depends on the size of Wikipedia, and they require computationally expensive noise-aware models. To mitigate the lack of datasets in vulnerable languages, in Chapter 4, the annotation projection method was used. Although the data scarcity issue could be alleviated, the existence of noise due to cross-lingual annotation projection impacts the quality of the generated dataset. Moreover, despite these advances, comprehensive and high-quality fine-grained resources still remain scarce for most low-resource Indian languages.

To address this gap, we present CLASSER: Cross-lingual Annotation Projection framework enhanced through Script Similarity for Fine-grained Named Entity Recognition. As shown in Figure 5.1, in Stage 1, we project English MultiCoNER2 annotations to the target language using BPCC parallel corpora [145] and a multilingual encoder-based annotation projection and word alignment tool [142, 143]. In stage 2, we introduce a confidence-score-based cross-lingual refinement method that utilizes an auxiliary NER model fine-tuned on script-similar languages. For example, although Hindi (Indo-European language) and Bodo (Sino-Tibetan language) are typologically distinct, their shared Devanagari script allows us to significantly enhance the Bodo FgNER annotations using the available MultiCoNER2 FgNER dataset in Hindi. Figure 5.1 shows an example of annotation refinement from projected entity type *B-ArtWork* to *B-WrittenWork* based on the FgNER dataset in auxiliary language. We apply the CLASSER framework to generate FgNER dataset in five low-resource languages, including *Assamese (as)*, *Marathi (mr)*, *Nepali (ne)*, *Sanskrit (sa)*, along with a vulnerable language *Bodo (brx)* [127].

Our contributions can be summarized as follows:

1. Development of CLASSER framework for cross-lingual annotation projection with script-

similarity-based refinement to create high-quality FgNER datasets.

2. Construction of a large-scale FgNER dataset comprising of 1.8M sentences, 2.6M entity mentions, and 24.7M tokens for five low-resource Indian languages: Assamese (as), Bodo (brx), Marathi (mr), Nepali (ne), and Sanskrit (sa).

3. Creation of a high-quality human-annotated test set consisting of 1000 sentences for each language with inter-annotator agreement (κ) above 0.86.

4. Our rigorous analyses establish the effectiveness of the proposed method and the good quality of the generated dataset, which has achieved 26% and 46% improvement in F1 scores over equal-sized TAFSIL [29] datasets in Marathi and Sanskrit, respectively.

5. Zero-shot and cross-lingual analysis to examine the influence of multilingualism and script similarities on cross-lingual FgNER performance.

5.2 Related Works

[96] first introduced fine-grained entity classification with 150 entity types in a multi-level hierarchy. Subsequent FgNER resources vary widely in entity type granularity: ACE (52 types) [39], BBN (93 types) [97], HYENA (505 types) [98], FIGER (113 types) [2], and OntoNotes (88 types) [27]. Large-scale resources like WikiSense [99], FINET [100], TypeNet [101], and UFET [23] proposed thousands of entity types. [22] improved quality with language-specific heuristics and refined selections, whereas [33] provides a large, manually annotated dataset covering 66 fine-grained types.

Early Indian-language NER began with IJCNLP-2008 for Hindi, Bengali, Oriya, Telugu, and Urdu [59], and further enhanced by [60, 61, 62, 63, 65]. Polyglot NER [16] and WikiANN [17] extended coverage to many languages, including a few Indian languages. Manually annotated corpora include NER in Bengali [89], Telugu [90], Maithili [95], Hindi [34], Assamese [86], Marathi [87], Nepali [88], Bishnupriya Manipuri [92], Bodo [93] etc.

Yarowsky et. al [66] pioneered projection via parallel corpora, but zero-shot transfer struggles with typological distance [80, 81], and cross-lingual encoders [83, 151] have not closed the gap yet [78]. [75] used Samanantar corpora [77] and word alignment [78, 79] to project English NER to eleven Indian languages. More broadly, projection-based translation was used to generate NER resources across various languages [69, 70, 71, 72, 73, 76], and capitalization of script similarity boosted low-resource translation, Parts-of-Speech tagging, and NER [152, 153, 154, 155].

MultiCoNER-1 [24] and MultiCoNER2 [102] produced Hindi and Bengali datasets via translated English annotations, which were further improved by SemEval-2023 shared-task participating teams [103, 104, 105, 106]. The TAFSIL framework [29] as described in Chapter 3 utilized Wikipedia links and Wikidata to produce noisy FgNER datasets covering six Indian languages (Hindi, Marathi, Sankrit, Tamil, Telugu, and Urdu) across four taxonomies. Despite these advances, high-quality multilingual FgNER resources for most Indian languages remain scarce.

5.3 Methodology

5.3.1 CLASSER Framework

The proposed framework CLASSER adopts a two-stage approach that incorporates three distinct categories of languages, as illustrated in Figure 5.1. The high-resource source language (*src*) provides both an annotated FgNER dataset and parallel corpora comprising sentence pairs aligned with the target language (*tgt*). The source language, however, differs notably from the target language in terms of grammatical structure and script. In contrast, an auxiliary language (*aux*)

is characterized by the availability of an annotated FgNER dataset written in a script similar to that of the target language, but it lacks corresponding parallel sentence pairs with the target language. For a language l, the FgNER dataset $D_l = \{(s_l^{(i)}, y_l^{(i)})\}_{i=1}^{N}$ consists of N samples where each sentence s is paired with its corresponding annotation sequence y. For the source language src, we possess both FgNER dataset D_{src} and a parallel corpus having source-to-target language aligned-sentence pairs, $PC = \{(s_{src}^{(i)}, s_{tgt}^{(i)})\}_{i=1}^{N}$. The auxiliary language provides only the FgNER dataset D_{aux}, sharing its script with tgt but offering no additional parallel data.

Stage 1 (Annotation Projection):

As per the Algorithm 3, we first fine-tune a pretrained multilingual encoder ME on parallel corpora P so that it learns to produce contextual embeddings $\mathbf{EB}_{src} = ME(s_{src})$ and $\mathbf{EB}_{tgt} = ME(s_{tgt})$ for source and target languages, respectively. For each parallel pair $(s_{src}, s_{tgt}) \in PC$, a word alignment service $\mathcal{A}(s_{src}, s_{tgt})$ leverages $\mathbf{EB}_{src}$ and $\mathbf{EB}_{tgt}$ to yield a *mapping* from source-sentence tokens to target-sentence tokens based on the FgNER dataset in the source language (D_{src}). We then transfer each source annotation sequence y_{src} to the target sentence by setting $\hat{y}_{tgt}^{(j)} = y_{src}^{(mapping^{-1}(j))}$ for every token j in the target sentence s_{tgt}. This produces an **initial annotation** sequence $\hat{y}_{tgt}$ for each target sentence, which may be noisy due to alignment errors, script mismatches, or syntactic divergences.

Stage 2 (Annotation Refinement):

To correct such projection noise, we exploit the observation that many named entities retain nearly identical orthographic forms when expressed in the same script, even across grammatically distinct languages. The semantics of entities are already captured by a pre-trained multilingual encoder during fine-tuning on a language written using a similar script. Although other factors, such as typological differences between languages, play a role in various NLP tasks, we have found that leveraging shared script characteristics is highly effective for the FgNER task. We therefore introduce a refinement stage driven by an auxiliary language (aux) whose writing system matches that of the target language. A multilingual encoder model ME_{aux} is fine-tuned on the auxiliary FgNER dataset D_{aux}. When applied to each target sentence s_{tgt}, for every token j, ME_{aux} outputs a discrete probability distribution $prob_{aux}(j) = ME_{aux}(s_{tgt})[j]$ and an auxiliary annotation sequence $\ddot{y}_{tgt}$. The maximum probability of the auxiliary annotation $\ddot{y}_{tgt}^{(j)}$ for the jth token is denoted as $prob_{aux}^{max}(j)$.

We then apply a refinement function:

$$f\big(\hat{y}_{tgt}^{(j)}, \ddot{y}_{tgt}^{(j)}\big) = \begin{cases} \ddot{y}_{tgt}^{(j)}, & \text{if } prob_{aux}^{max}(j) \geq \tau \text{ and } \ddot{y}_{tgt}^{(j)} \neq \hat{y}_{tgt}^{(j)}, \\ \hat{y}_{tgt}^{(j)}, & \text{otherwise.} \end{cases}$$

where τ is a confidence threshold. For every token j, the refinement function selectively replaces the initial projected label $\hat{y}_{tgt}^{(j)}$ with the auxiliary label $\ddot{y}_{tgt}^{(j)}$ only when the auxiliary annotation is both confident (i.e. $prob_{aux}^{max}(j) \geq \tau$) and disagrees with the initially projected annotation (i.e. $\ddot{y}_{tgt}^{(j)} \neq \hat{y}_{tgt}^{(j)}$). The result is the **final refined annotation** $\bar{y}_{tgt}^{(j)}$ for each token j, i.e. $\bar{y}_{tgt}^{(j)} = f(\hat{y}_{tgt}^{(j)}, \ddot{y}_{tgt}^{(j)})$.

Finally, aggregation of all sentence–annotation pairs $\big(s_{tgt}^{(i)}, \bar{y}_{tgt}^{(i)}\big)$ produces the fully refined target-language dataset of size N:

$$D_{tgt}^{ref} = \big\{(s_{tgt}^{(i)}, \bar{y}_{tgt}^{(i)})\big\}_{i=1}^{N}$$

Algorithm 3: CLASSER Framework

Symbols:

- ME: Pretrained multilingual encoder.
- N: Number of sentences per dataset.
- y: List containing FgNER annotations for each token of sentence s.
- $D_l = \{(s_l^{(i)}, y_l^{(i)})\}_{i=1}^{N}$: FgNER dataset of size N in language l.
- src, tgt, aux: Source, Target and Auxiliary languages respectively.
- $PC = \{(s_{src}^{(i)}, s_{tgt}^{(i)})\}_{i=1}^{N}$: Parallel corpora.
- $\mathcal{A}(s_{src}, s_{tgt})$: Annotation alignment service produces a $mapping$.
- $\hat{y}_{tgt}$: Initial annotation sequence for s_{tgt}.
- ME_{aux}: Multilingual encoder fine-tuned on D_{aux}.
- $prob_{aux}(j) = ME_{aux}(s_{tgt})[j]$: The annotation probability distribution for jth token in s_{tgt}th sentence after refinement using ME_{aux}
- $\ddot{y}_{tgt}$: Auxiliary annotation sequence s_{tgt}.
- $prob_{aux}^{max}(j)$: Maximum probability of $\ddot{y}_{tgt}^{(j)}$ for jth token.
- τ: Confidence Threshold.
- Refinement function:
  $f(\hat{y}_{tgt}^{(j)}, \ddot{y}_{tgt}^{(j)}) = \begin{cases} \ddot{y}_{tgt}^{(j)}, & \text{if } prob_{aux}^{max}(j) \geq \tau \text{ and } \ddot{y}_{tgt}^{(j)} \neq \hat{y}_{tgt}^{(j)}, \\ \hat{y}_{tgt}^{(j)}, & \text{otherwise.} \end{cases}$
- Final refined annotation: $\bar{y}_{tgt}^{(j)} = f(\hat{y}_{tgt}^{(j)}, \ddot{y}_{tgt}^{(j)})$
- $D_{tgt}^{ref} = \{(s_{tgt}^{(i)}, \bar{y}_{tgt}^{(i)})\}_{i=1}^{N}$: Final refined target FgNER dataset of size N.

Algorithm:

for each $(s_{src}, s_{tgt}) \in PC$ **do**
  Compute embeddings: $\mathbf{EB}_{src} = ME(s_{src})$, $\mathbf{EB}_{tgt} = ME(s_{tgt})$
  Obtain alignment mapping: $mapping = \mathcal{A}(s_{src}, s_{tgt})$ using $\mathbf{EB}_{src}$ & $\mathbf{EB}_{tgt}$
  for each token j in s_{tgt} **do**
    Project annotation: $\hat{y}_{tgt}^{(j)} = y_{src}^{(mapping^{-1}(j))}$
  end for
  Obtain auxiliary predictions: $\forall j$ in s_{tgt}: $prob_{aux}(j) = ME_{aux}(s_{tgt})[j]$
  for each token j in s_{tgt} **do**
    Refine annotation: $\bar{y}_{tgt}^{(j)} = f(\hat{y}_{tgt}^{(j)}, \ddot{y}_{tgt}^{(j)})$
  end for
  Update $(s_{tgt}, \bar{y}_{tgt})$ to D_{tgt}^{ref}
end for

Table 5.1: CLASSER dataset statistics. **Lng** means language, **Sent**, **Ent** and **Tkn** means number of Sentences, Entities and Tokens respectively. **IAA**(κ) gives the inter-annotator agreement.

	Train set			Development set			Test set			
Lng	**Sent**	**Ent**	**Tkn**	**Sent**	**Ent**	**Tkn**	**Sent**	**Ent**	**Tkn**	**IAA**(κ)
as	140,257	204,611	1,972,697	15,585	15,763	219,114	1000	1,407	14,270	0.901
brx	212,835	302,713	2,958,455	23,649	33,808	329,145	1000	1,423	14,082	0.875
mr	611,902	889,217	8,135,813	67,990	97,943	948,020	1000	1,443	13,996	0.887
ne	414,561	617,957	5,531,683	46,062	64,098	642,489	1000	1,436	14,142	0.882
sa	265,114	378,287	3,488,871	29,458	40,589	377,306	1000	1,412	12,925	0.861

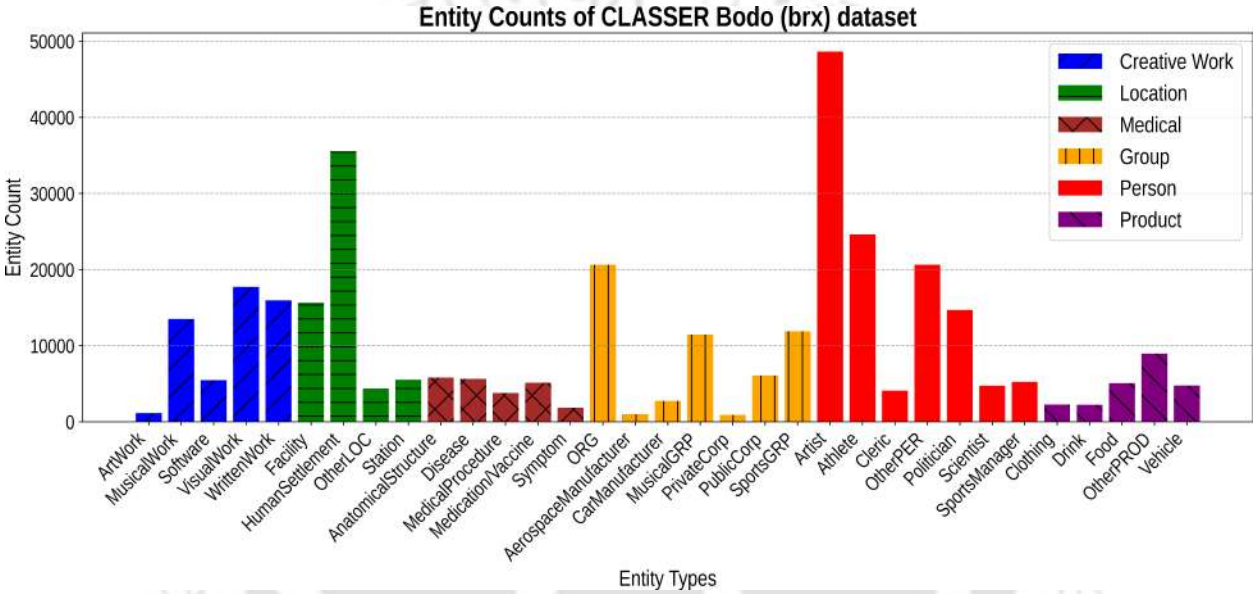

Figure 5.2: Entity counts of CLASSER Bodo (brx) dataset. Each color depicts the coarse-grained type, and each bar depicts the corresponding fine-grained types.

5.3.2 Implementation of CLASSER Framework

MultiCoNER2 [3], a SemEval-2023 shared task [102], provides 33 fine-grained entity types across 12 languages (including English, Hindi, Bengali). In our setup, the source language (*src*) is English; target languages (*tgt*) are Assamese (as), Bodo (brx), Marathi (mr), Nepali (ne), and Sanskrit (sa); auxiliary languages (*aux*) are Bengali (bn) and Hindi (hi). Assamese and Bengali use the Bengali-Assamese script, whereas Hindi, Bodo, Marathi, Nepali, and Sanskrit use Devanagari. We employ BPCC [145] as parallel corpora (PC), augmenting the smaller Bodo data with [147, 146]. Following [143], we adopted AWESoME align [142] as $\mathcal{A}$, fine-tuned on the English MultiCoNER2 dataset (D_{src}). The Hindi and Bengali MultiCoNER2 datasets serve as D_{aux}, and we fine-tune IndicBERTv2 [85] as ME and ME_{aux}. IndicBERTv2 is the only encoder pre-trained on all *src*, *tgt*, and *aux* languages. Based on language-specific evaluations (Figure 5.5, we set the confidence threshold τ=0.85. Following the Algorithm 3, the CLASSER framework is implemented with the mentioned details, and the dataset is created.

Table 5.2: Comparison of CLASSER dataset with some publicly available NER datasets in low-resource Indian languages. Abbreviations: MCN2: MultiCoNER2 taxonomy, **Entities**: number of entity mentions, **Types**: number of entity types.

Language	Dataset	Tokens	Entities	Types
Assamese (as)	CLASSER	**2,206,081**	**221,781**	**33**
	Naamapadam [75]	122,413	5,045	3
	AsNER [86]	98,623	34,963	5
	WikiANN [132]	7,632	1,418	3
Bodo (brx)	CLASSER	**3,301,682**	337,944	**33**
	Bodo NER [93]	2,797,101	**641,604**	5
Marathi (mr)	CLASSER	**9,097,829**	**988,603**	**33**
	TAFSIL(MCN2) [29]	3,628,450	174,861	**33**
	Naamapadam [75]	6,086,136	529,000	3
	MahaNER [87]	231,959	27,300	7
	WikiANN [132]	123,556	18,756	3
Nepali (ne)	CLASSER	**6,188,314**	**683,491**	**33**
	EverestNER [88]	616,706	24,587	5
	OurNepali [144]	16,225	11,183	4
	WikiANN [132]	14,535	2,326	3
Sanskrit (sa)	CLASSER	**3,879,102**	**420,288**	**33**
	TAFSIL(MCN2) [29]	479,185	23,372	**33**
	WikiANN [132]	2,255	115	3

5.3.3 CLASSER Dataset

As shown in Table 5.1, the created CLASSER dataset consists of more than 157 thousand sentences, 222 thousand entity mentions, and 2.2 million tokens in each of the five low-resource Indian languages. After the creation of the dataset through the proposed method, 1000 sentences are randomly selected for human annotation. From the rest of the dataset, 10% is considered as the development set and the remaining as the training set.

5.3.4 Gold Test Set

Two volunteer annotators per language, having a minimum education of an undergraduate degree, were chosen based on their mother tongue. For Sanskrit, the annotations were done by professional Sanskrit teachers. Annotators were first briefed on entity types with examples, then instructed to perform the task on 1000 sentences in two stages: detecting relevant entity mentions and assigning types from a given list using the BRAT tool [134]. With two annotators per language, one annotator's work was treated as the gold standard, and inter-annotator agreement (IAA) was measured against it. The quality of these gold datasets can be ascertained based on a high Cohen's kappa coefficient (κ) [150], which is above 0.86 for each language (Table 5.1).

5.3.5 Comparison with Public Datasets

As shown in Table 5.2, CLASSER is the largest NER dataset across all languages in terms of the tokens compared to the publicly available datasets. In fact, except for Bodo (brx), it is the largest dataset in terms of the number of entities as well. To the best of our knowledge, CLASSER is

the only FgNER dataset created through the entity projection method for these five low-resource languages. In this work, whenever comparative analysis is done, the **highest value** or the **best result** in the tables are shown in **bold** and the second highest value or the second best result are shown as underlined.

5.3.6 Entity Type Frequency Distribution

As shown in Figure 5.2, a larger number of entity mentions are detected for the fine types of Location (e.g. HumanSettlement) and Person (e.g. Artist) because the HumanSettlement includes the mentions of cities, provinces and countries, and the Artist type includes the mentions of musicians, actors, directors, authors, etc. Whereas, very specific fine types such as AerospaceManufacturer, Drink, AnatomicalStructure, etc., are very scarce. Similar trends can be observed across all five languages.

5.4 Analysis & Results

5.4.1 Experimental Setup

The state-of-the-art approach for sequence labeling tasks involves fine-tuning pre-trained language models (PLM) with the NER datasets [24, 34, 75, 86, 87, 88, 95, 123, 124]. Similarly, we have fine-tuned mBERT (bert-base-multilingual-cased) [82], IndicBERTv2 (IndicBERTv2-MLM-Sam-TLM) [85], MuRIL (muril-large-cased) [84] and XLM-RoBERTa (XLM-RoBERTa-large) [83] for fine-grained NER using the Hugging Face Transformers library [156]. The models were trained for six epochs with a batch size of 64, utilizing AdamW optimization (learning rate: 5e-5, weight decay: 0.01). Training was performed on an NVIDIA A100 GPU, with evaluation based on SeqEval metrics, and the best performance determined by the F1-score following [157, 158].

To compare with the state-of-the-art noisy FgNER dataset, following [29], we adopted the best-performing two variations of DECENT [121] to accommodate Indian languages by changing its base encoder from RoBERTa-large [141], to XLM-RoBERTa-large [83], and IndicBERTv2-MLM-Sam-TLM [85]. The hyperparameters for DECENT-based models are: learning rate for encoder = 5e-6, learning rate for head = 5e-4, dropout probability for head = 0.5, epochs = 2, batch size = 16, negative oversampling rate = 31, and prediction threshold = 0.9.

5.4.2 Comparison with the State-of-the-Art baseline

To the best of our knowledge, the only existing FgNER datasets in Indian languages are MultiCoNER2 [3] for Hindi and Bengali, and TAFSIL [29] for Hindi, Marathi, Sanskrit, Tamil, Telugu, and Urdu. Accordingly, we used the Marathi and Sanskrit TAFSIL datasets in the MultiCoNER2 taxonomy and fine-tuned DECENT model variants as described in the previous section. As shown in Table 5.3, when tested on the TAFSIL test set, the models fine-tuned on CLASSER outperform models fine-tuned on TAFSIL by a large margin. With the subset of CLASSER dataset having an equal number of entities as TAFSIL, the F1 scores improve by about 22% for Marathi and 38% for Sanskrit, respectively. Similarly, with the subset of CLASSER dataset having an equal number of sentences as TAFSIL, the F1 scores improve by about 26% for Marathi and 40% for Sanskrit, and using the entire CLASSER dataset, gains rise to roughly 40% and 95%, respectively. These results demonstrate both the effectiveness of our method and the high quality of the generated CLASSER dataset.

Table 5.3: Performance of different DECENT models fine-tuned on TAFSIL and CLASSER datasets and tested on the TAFSIL test set. Abbreviations: **Sent**: Number of sentences, **Ent**: Number of entities.

DECENT model with XLM-RoBERTa-large base encoder

				Micro			Macro		
Lang	**Dataset**	**Sent**	**Ent**	**P**	**R**	**F1**(↑Δ%)	**P**	**R**	**F1**(↑Δ%)
mr	TAFSIL	126k	175k	49.36	83.73	62.10	50.89	81.93	62.28
	CLASSER	120k	175k	72.85	80.12	76.32(**23**)	73.45	79.13	75.61(**21**)
	CLASSER	126k	183k	76.12	80.97	78.28(**26**)	73.42	82.19	77.56(**25**)
	CLASSER	612k	989k	82.66	91.74	86.98(**40**)	84.40	91.78	87.94(**41**)
sa	TAFSIL	18k	23k	31.35	53.40	40.20	32.74	53.29	40.56
	CLASSER	16k	23k	52.54	59.03	55.04(**37**)	53.29	58.66	55.84(**38**)
	CLASSER	18k	25k	56.14	60.03	58.02(**44**)	57.25	60.99	59.06(**46**)
	CLASSER	265k	378k	77.21	81.69	79.38(**97**)	78.60	80.89	79.93(**97**)

DECENT model with IndicBERTv2-MLM-Sam-TLM base encoder

				Micro			Macro		
Lang	**Dataset**	**Sent**	**Ent**	**P**	**R**	**F1**(↑Δ%)	**P**	**R**	**F1**(↑Δ%)
mr	TAFSIL	126k	175k	48.57	82.81	61.23	49.33	81.49	61.46
	CLASSER	120k	175k	72.11	78.10	75.04(**23**)	72.62	76.03	74.34(**21**)
	CLASSER	126k	183k	76.65	80.12	77.29(**26**)	75.38	79.27	76.36(**24**)
	CLASSER	612k	989k	83.07	89.44	86.14(**41**)	81.75	89.37	85.43(**39**)
sa	TAFSIL	18k	23k	32.74	56.03	41.03	32.94	55.96	41.47
	CLASSER	16k	23k	56.29	60.66	57.82(**40**)	56.27	60.71	57.33(**38**)
	CLASSER	18k	25k	59.70	61.60	60.64(**48**)	59.34	61.50	60.36(**46**)
	CLASSER	265k	378k	79.52	78.94	79.01(**93**)	78.09	81.49	80.81(**95**)

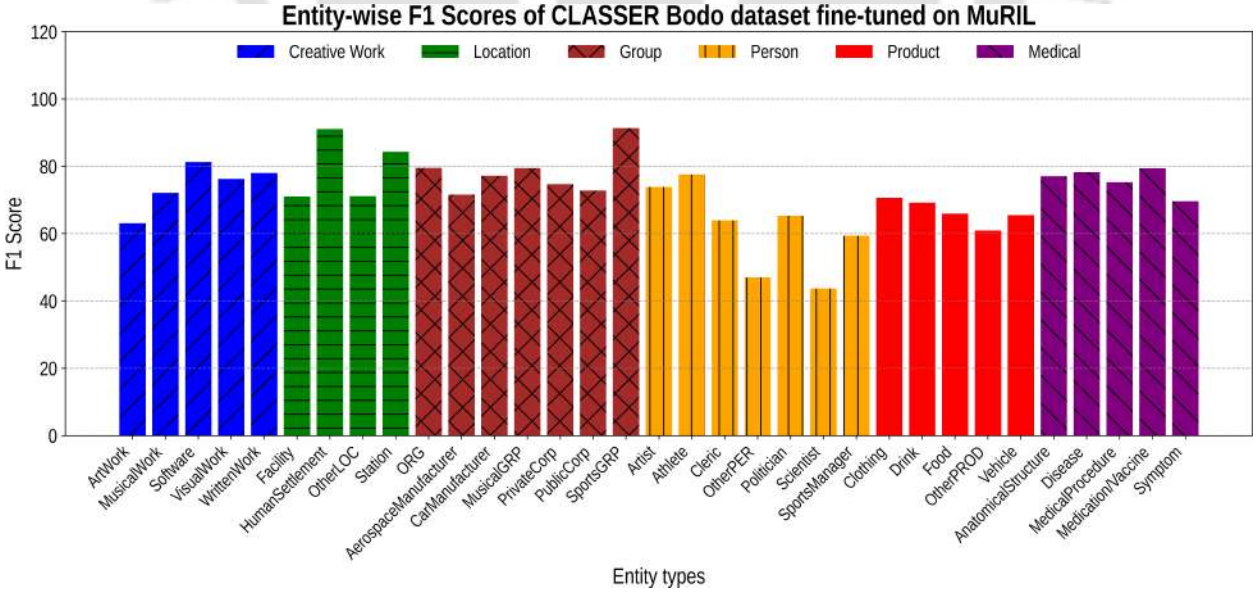

Figure 5.3: Entity-wise F1 score of MuRIL model fine-tuned on Bodo. Each color depicts the coarse-grained type, and each bar depicts the corresponding fine-grained types.

Table 5.4: Performance of different models fine-tuned on CLASSER dataset. Abbreviations: IB: IndicBERTv2, mB: mBERT, MR: MuRIL, XL: XLM-RoBERTa.

Assamese (as)

	Micro			Macro				Micro			Macro		
	P	**R**	**F1**	**P**	**R**	**F1**		**P**	**R**	**F1**	**P**	**R**	**F1**
IB	71.10	71.50	71.30	62.33	63.67	63.11	mB	66.09	68.30	67.18	64.12	61.48	63.13
MR	74.88	75.62	**75.25**	73.91	70.54	**72.44**	XL	71.35	73.63	72.47	69.50	69.42	69.46

Bodo (brx)

	Micro			Macro				Micro			Macro		
	P	**R**	**F1**	**P**	**R**	**F1**		**P**	**R**	**F1**	**P**	**R**	**F1**
IB	70.61	72.64	71.61	69.53	68.58	68.98	mB	68.75	71.03	69.87	68.65	68.23	68.59
MR	73.83	76.37	**75.08**	74.76	73.73	**74.08**	XL	71.60	73.77	72.67	73.28	71.66	72.60

Marathi (mr)

	Micro			Macro				Micro			Macro		
	P	**R**	**F1**	**P**	**R**	**F1**		**P**	**R**	**F1**	**P**	**R**	**F1**
IB	74.38	76.49	75.42	70.67	71.89	71.22	mB	74.83	76.28	75.55	69.81	71.57	70.82
MR	79.24	81.00	**80.11**	75.94	77.59	**76.83**	XL	78.58	78.85	78.71	73.55	75.36	74.41

Nepali (ne)

	Micro			Macro				Micro			Macro		
	P	**R**	**F1**	**P**	**R**	**F1**		**P**	**R**	**F1**	**P**	**R**	**F1**
IB	73.80	75.24	74.52	71.52	70.50	71.02	mB	74.33	75.52	74.92	73.40	72.40	72.91
MR	76.92	79.50	**78.19**	75.34	76.37	**75.88**	XL	74.93	78.80	76.82	73.32	75.95	74.14

Sanskrit (sa)

	Micro			Macro				Micro			Macro		
	P	**R**	**F1**	**P**	**R**	**F1**		**P**	**R**	**F1**	**P**	**R**	**F1**
IB	73.45	75.02	74.23	72.96	72.17	72.58	mB	70.26	72.25	71.24	71.41	69.24	70.35
MR	77.62	78.99	**78.30**	77.61	76.57	**77.04**	XL	75.41	77.50	76.44	74.79	74.17	74.49

5.4.3 Performance of PLMs on Unseen Languages

Marathi (mr) and Nepali (ne) are the only languages among the five languages on which all four PLMs (IndicBERTv2, mBERT, MuRIL, and XLM-RoBERTa) are pre-trained (Table 2.1 in Chapter 2). Therefore, the performance of all the fine-tuned models in Nepali and Marathi is superior (Table 5.4). Although mBERT was not pre-trained on Assamese (as) and Sanskrit (sa), it performs well after fine-tuning with CLASSER dataset on these languages. This is due to the script similarity between Assamese with Bengali (Bengali-Assamese script) and between Sanskrit with Hindi (Devanagari script). Similarly, although mBERT, MuRIL, and XLM-RoBERTa are not pre-trained in Bodo, the fine-tuned models could perform better due to their pre-training on the script-similar language Hindi. MuRIL, pre-trained exclusively on 16 Indian languages, outperforms other PLMs. These results emphasize the importance of language-specific pre-training and the effect of script-similarity in fine-tuning, which are discussed further in the following sections.

5.4.4 Entity type specific performance

As shown in the Figure 5.3, we have analyzed the F1 scores per entity type for the MuRIL model fine-tuned on Marathi. Notably, although the entity counts of fine types such as *ArtWork*, *AerospaceManufacturer*, *Clothing*, *AnatomicalStructure* etc. are fewer in number, the model could

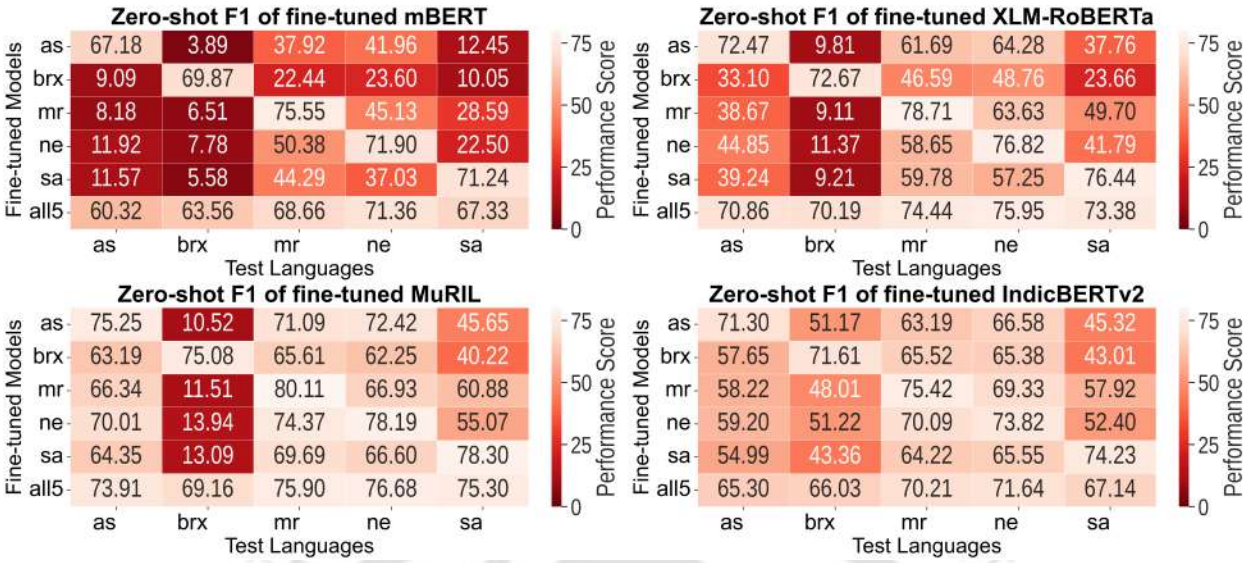

Figure 5.4: Zero-shot performance (micro F1) of fine-tuned mBERT, MuRIL, XLM-RoBERTa and IndicBERTv2 models on different test languages. The **all5** set includes equal number of samples from all five languages.

still perform well through fine-tuning with high-quality data. But the performance of the model is quite low for the fine-type *OtherPER* even after having a significant number of entity mentions in the training set. Details of this performance are discussed in the 5.4.8 Error Analysis section.

5.4.5 Cross-Lingual Zero-Shot Analysis

We have performed cross-lingual zero-shot analysis for every single language pair. As shown in Figure 5.4, the models are fine-tuned on datasets of respective languages and tested on the test set of other languages. Zero-shot performance of mBERT model is quite poor across all the languages. A similar trend is observed in the case of XLM-RoBERTa on unseen languages during its pre-training. However, there is an improvement in the case of MuRIL because of its pre-training on 16 Indian languages. The impact of pre-training of an encoder is imminent through the zero-shot performance of IndicBERTv2. Since IndicBERTv2 is pre-trained on all five languages, its zero-shot performance is superior to other PLMs. Whereas, fine-tuning on Bodo (brx) significantly improved the performance of mBERT, XLM-RoBERTa, and MuRIL over their zero-shot performances. These emphasize that due to script similarity, PLMs can perform better after fine-tuning on an unseen language. But, without fine-tuning with language-specific datasets, the pre-trained knowledge cannot capture the intricacies of an unseen language.

5.4.6 Multilingualism

We have extended our analysis to evaluate the multilingual aspect of the FgNER task. We constructed a balanced **all5** set, comprising an equal number of samples from all five languages. As seen in Figure 5.4, there are significant improvements in every encoder model when fine-tuned with all the languages and tested on test sets of individual languages. In fact, for a vulnerable language like Bodo (brx), multilingual fine-tuning can be very beneficial. These results also suggest the necessity of language-specific pre-training and task-specific fine-tuning.

Table 5.5: Ablation study: The impact of refinement using Bengali (⊕ bn) and Hindi (⊕ hi) on the initial annotation projection (Stg 1). The performance of fine-tuned MuRIL models in terms of micro-F1 scores are shown with **percentage improvements**.

	Stg 1	**⊕ bn**	**⊕ hi**	**⊕ hi+bn**
as	63.11	74.48(**18**)	67.43(**7**)	75.25(**19**)
brx	61.82	63.90(**3**)	73.84(**19**)	75.08(**21**)
mr	71.08	73.26(**3**)	78.85(**11**)	80.11(**13**)
ne	70.01	73.10(**4**)	77.05(**10**)	78.19(**12**)
sa	65.14	67.12(**3**)	76.87(**18**)	78.30(**20**)

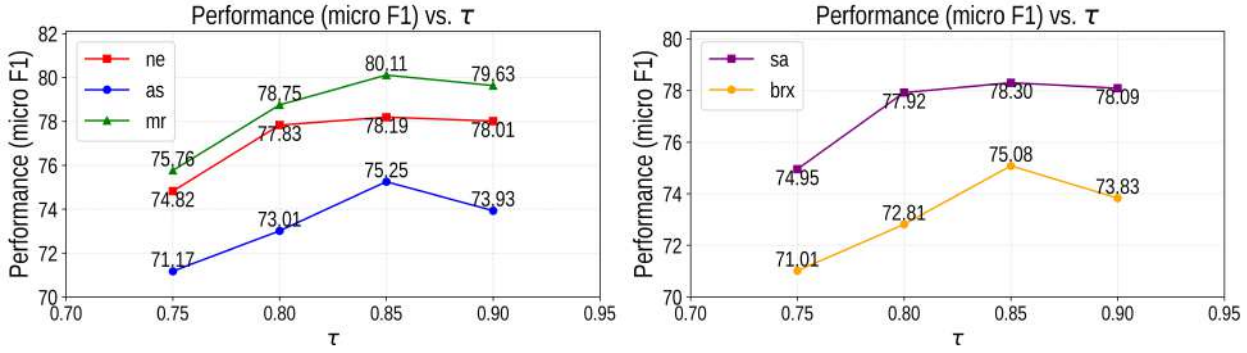

Figure 5.5: Impact of confidence score threshold (τ) on the performance of MuRIL fine-tuned on CLASSER dataset. With increasing τ values until the optimum, micro F1 score keeps increasing, beyond which it declines.

5.4.7 Ablation Study

As already seen in different analyses, the script similarity of the language plays a major role in FgNER task. The intuition of cross-lingual refinement after annotation projection is based on this property. The performance of fine-tuned MuRIL models in terms of micro-F1 scores are shown in Table 5.5. For Assamese (as), refinement using Bengali (bn) is the most effective, since both of these languages use the Bengali-Assamese script. Similarly, for Bodo (brx), Marathi (mr), Nepali (ne) and Sanskrit (sa), refinement using Hindi (hi) is the most effective due to the shared Devanagari script. But refinement using both hi+bn gives the best result across all the languages. These improvements are highest for extremely low-resource languages Assamese (as), Bodo (brx), and Sanskrit (as).

For the selection of the best confidence threshold τ in the cross-lingual refinement stage of CLASSER method, we conducted experiments with four empirically selected values 0.75, 0.80, 0.85, 0.90 for all the languages. In Figure 5.5, the performances of fine-tuned MuRIL are shown. Hence, based on the experiments, finally τ is set to be 0.85 for all the languages.

5.4.8 Error Analysis

FgNER is very crucial because an entity mention's type may vary significantly depending on the context within a sentence. Therefore, we have analyzed the errors on the test set in two different approaches.

First, the details of entity errors in terms of the percentage of predicted entities are shown in

Table 5.6: Entity errors on test set in terms of the percentage of predicted entities for different languages fine-tuned on MuRIL.

Error Types	**as**	**brx**	**mr**	**ne**	**sa**
Boundary Mismatch	13.67	9.27	10.37	8.23	9.82
Entity Type Mismatch	13.75	14.40	11.44	12.35	11.32
Spurious Entity	2.41	2.67	2.42	2.44	4.27

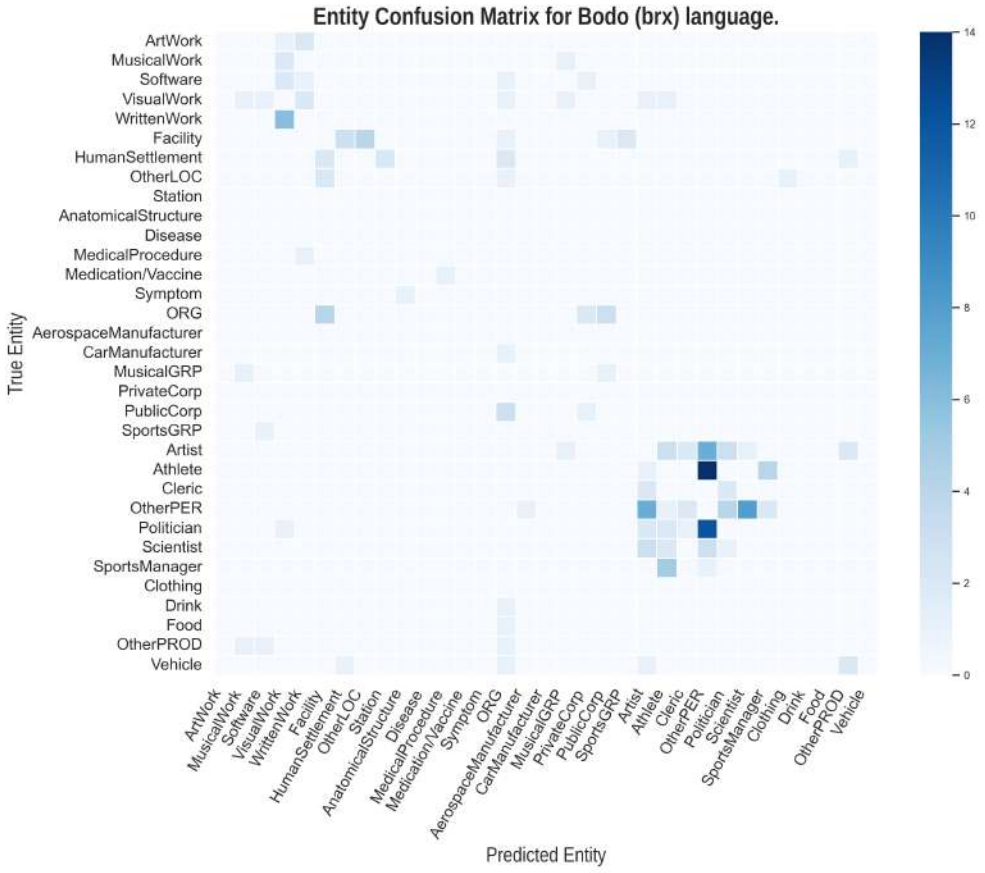

Figure 5.6: Entity Confusion matrix of Bodo (brx) language. Darker squares depict the entity type pairs often confused by the fine-tuned model.

Table 5.6. The common errors that occur include the boundary error (such as "Little Mermaid" is marked as *VisualWork* instead of "The Little Mermaid"), entity type mismatch error (e.g. "Sneezing" is categorized as *Disease* instead of *Symptom*) and spurious errors (such as "purple" is marked as an entity whereas the entity type *color* is not defined in MultiCoNER2 taxonomy). Entity boundary mismatch errors are highest in Assamese (as), entity type mismatch errors and spurious entity errors occur the most in Bodo (brx) and Sanskrit (sa), respectively.

Moreover, we have analyzed the often co-predicted fine-grained types. From Table 5.6, we have selected Bodo (brx) for this analysis as this language has the highest percentage of mismatch entity types. As shown in Figure 5.6, the fine types of *Artist*, *Athlete*, *Politician*, and *Scientist* are sometimes confused with *OtherPER*. Similarly, *WrittenWork* is sometimes confused with *VisualWork*. However, a significant reduction in entity type confusion is observed in the refined dataset

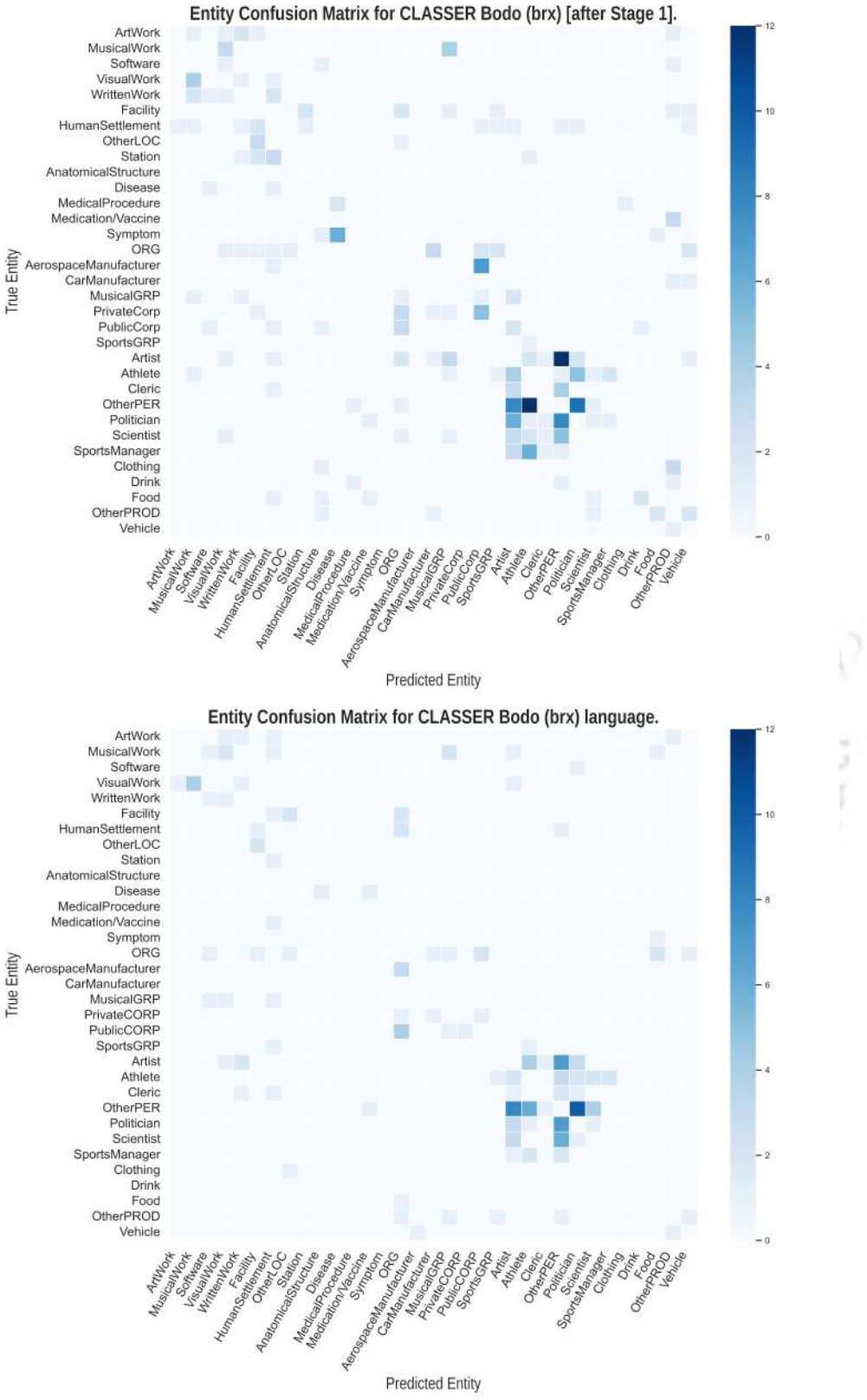

Figure 5.7: Entity Confusion matrix of Bodo (brx) language: only projected dataset (Stage 1: FiNERVINER) and final refined dataset (Stage 2: CLASSER). Darker squares depict the entity type pairs often confused by the fine-tuned model.

after Stage 2, compared to the dataset created through the annotation projection method in Stage 1 (Figure 5.7). These analyses show that script-similarity-based annotation refinement improves the quality of the FgNER dataset, and hence most of the fine entity types are learned by the models without much confusion.

5.5 Discussion

Despite encouraging initial results, this study has limitations that require further exploration. First, the proposed cross-lingual refinement stage relies primarily on resources in languages with similar scripts, for which its direct applicability and scalability to languages with significantly different syntactic structures remain an open question for future research. Second, the generated dataset may inherit biases or specificities from the source FgNER dataset. Third, the dataset's volume and quality depend on the availability of parallel corpora and the choice of annotation projection tools and multilingual encoders. Assessment of different combinations of these resources remains essential. Moreover, while a confidence score of 0.85 yielded the best results among the four tested empirical values (0.75, 0.80, 0.85, 0.90), a more systematic and theoretically grounded analysis is needed. Finally, evaluating Large Language Models (LLMs) on the FgNER task, both in a zero-shot setting and fine-tuned on the CLASSER dataset, remains under-explored.

5.6 Conclusion

We introduce CLASSER, a cross-lingual annotation-projection framework that leverages script similarity to create high-quality FgNER dataset. The generated CLASSER dataset comprises of 1.8M sentences, 2.6M entity mentions, and 24.7M tokens covering five low-resource languages (Assamese, Bodo, Marathi, Nepali, and Sanskrit). Extensive experiments suggest the high quality of the CLASSER dataset, which achieves significant improvement in terms of F1-scores over the state-of-the-art TAFSIL dataset, as introduced in the Chapter 3. The annotation refinement in stage 2 of the CLASSER framework improves the dataset quality manifold (up to 20% in F1-score) compared to only the annotation projection method as discussed in the Chapter 4. Cross-lingual zero-shot analyses reveal the importance of language-specific pre-training and task-specific fine-tuning. Given the availability of MultiCoNER2 FgNER resources in Bengali, Hindi, and Farsi, CLASSER can be readily extended to other Indian languages (e.g., Maithili, Konkani, Dogri, Bhojpuri, Chhattisgarhi, Bishnupriya Manipuri, Urdu, Kashmiri, Sindhi etc.). We expect the CLASSER framework, the generated dataset, and fine-tuned models will significantly advance FgNER across Indian languages and facilitate further developments in multilingual research.

The CLASSER dataset, associated fine-tuned models and interactive web application are publicly available at hf.co/collections/prachuryyaIITG/CLASSER

6

Entity-Anchored Machine Translation for Fine-grained Named Entity Recognition

Chapter Highlights

- The objective of this chapter is to propose an Entity-anchored Machine Translation (EaMaTa) framework for Fine-grained Named Entity Recognition.
- The dependency on the availability of Wikipedia or on the resources in languages written with a similar script for the creation of a high-quality dataset is overcome with the highly scalable and efficient EaMaTa framework.
- Utilizing the EaMaTa framework, a high-quality FgNER dataset is created for 26 Indian languages, spoken by more than two billion people worldwide.
- The dataset's high quality is confirmed through extensive human and statistical analyses, demonstrating the framework's effectiveness with up to a 9% F1-score increase over current state-of-the-art models.
- This chapter is based on the publication "SampurNER: Fine-grained Named Entity Recognition Dataset for 22 Indian Languages" published in "Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-26)" and "FiNE-MiBBiC: Fine-grained Named Entity Recognition for Mizo, Bhojpuri, Bishnupriya Manipuri and Chhattisgarhi", submitted to The ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP).
- The SampurNER dataset, associated fine-tuned models and interactive web application are publicly available at: hf.co/collections/prachuryyaIITG/SampurNER[1].

6.1 Introduction

Although coarse-grained NER for Indian languages has made notable progress, fine-grained NER (FgNER) is a relatively recent area of focus. The MultiCoNER2 [3] shared task at SemEval 2023 [131] provided FgNER datasets for Hindi and Bengali by translating the annotated dataset from English. As outlined in Chapter 3, the distant supervision framework TAFSIL was employed to construct large-scale FgNER datasets for multiple Indian languages using Wikipedia and Wikidata.

[1] '*Sampurna*' means 'entire' in many Indian languages

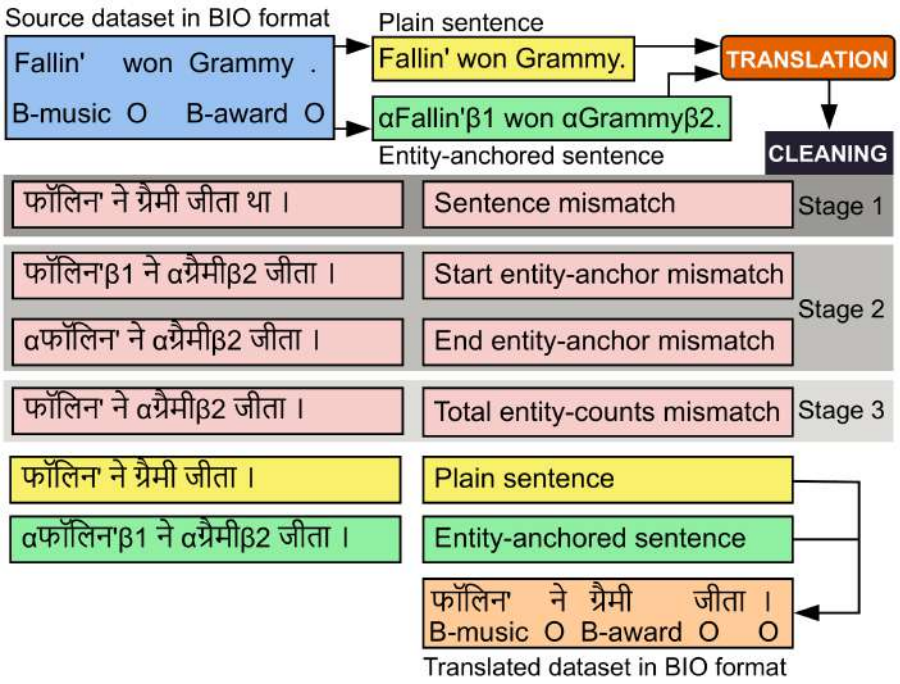

Figure 6.1: Entity-anchored machine translation: The source dataset is translated both as 'Plain sentence' and 'Entity-anchored sentence' to the target languages. The cleaning process includes the removal of sentences with sentence mismatch, entity-anchor boundaries mismatch (both start and end), and total entity counts mismatch between the source and translated sentences.

However, such datasets are often noisy, their scale is constrained by the coverage of Wikipedia, and their effective use requires computationally intensive noise-aware models. To address the absence of datasets for vulnerable languages, Chapter 4 adopts an annotation projection approach. While this method helps mitigate data scarcity, noise introduced through cross-lingual annotation projection adversely affects the quality of the resulting datasets. Based on the observation on the impact of script-similarity in FgNER, the CLASSER framework was proposed in the Chapter 5. The CLASSER framework enables the efficient creation of high-quality FgNER datasets for Assamese, Bodo, Marathi, Nepali, and Sanskrit, but its applicability is still restricted by the availability of FgNER resources in script-similar languages. Hence, comprehensive and high-quality fine-grained resources remain limited for most low-resource Indian languages.

To bridge this gap, we introduce a new initiative aimed at expanding FgNER datasets for Indian languages. Our approach leverages an **Entity-Anchored Machine Translation** (EaMaTa) framework to efficiently translate the high-quality FewNERD dataset into 26 Indian languages. FewNERD is the only large-scale, human-annotated FgNER dataset comprising 188,238 sentences with 4,601,160 words, annotated with a two-level hierarchy of 8 coarse-grained and 66 fine-grained entity types by 70 linguistically trained annotators having 0.76 Inter-Annotator Agreement (κ).
Our contributions can be summarized as follows:

1. Development of an entity-anchored machine translation (EaMaTa) framework to create high-quality multilingual FgNER datasets from English annotations.

2. Construction of a large-scale dataset, comprising over 153,000 sentences, 354,000 entities, and 3.3 million tokens on average for each of the 26 Indian languages spoken by more than two billion people worldwide. The languages included are *Assamese (as)*, *Bengali (bn)*, *Bhojpuri (bho)*,

Bishnupriya (bpy), *Bodo (brx)*, *Chhattisgarhi (hne)*, *Dogri (doi)*, *Gujarati (gu)*, *Hindi (hi)*, *Kannada (kn)*, *Kashmiri (ks)*, *Konkani (gom)*, *Maithili (mai)*, *Malayalam (ml)*, *Manipuri (mni)*, *Marathi (mr)*, *Mizo (lus)*, *Nepali (ne)*, *Odia (or)*, *Punjabi (pa)*, *Sanskrit (sa)*, *Santali (sat)*, *Sindhi (sd)*, *Tamil (ta)*, *Telugu (te)*, and *Urdu (ur)*. Among these languages, *Bishnupriya (bpy)*, *Bodo*, *Manipuri*, *Mizo (lus)* are considered to be vulnerable languages [127].

3. Manually annotated gold test set having 1000 sentences for twelve languages: *as*, *bpy*, *bn*, *brx*, *hi*, *lus*, *mr*, *ne*, *sa*, *ta*, *te*, and *ur*.

4. Rigorous human evaluations and extensive experiments with state-of-the-art suggest the good quality of the datasets created through the proposed EaMaTa framework.

5. Cross-lingual zero-shot analysis to examine the influence of multilingualism, language families, and script similarities on cross-lingual FgNER performance.

6.2 Related Works

Annotation projection-based and annotation preserving translation methods have been extensively used for NER across multiple languages [69, 70, 71, 159], leading to the creation of several NER resources [72, 73, 76, 123]. Similarly, in MultiCoNER-1 [24] and MultiCoNER2 [3], machine translation was used to develop coarse-grained and fine-grained NER datasets for multiple languages, including Hindi and Bengali, which were further enriched by SemEval [131] top-performing teams [103, 104, 105, 106].

6.3 Entity-Anchored Machine Translation

We propose an entity-anchored machine translation (EaMaTa) framework to preserve NER information during translation. As shown in Figure 6.1, we convert the source NER dataset from the BIO format into both plain sentence and entity-anchored sentence formats. In the entity-anchored format, each entity mention m_j for the jth token is marked with a unique start symbol (α_j) and a distinct end symbol (β_{je}) which varies based on the entity type e. The design of these special anchor symbols is guided by evaluations to ensure that the anchors remain intact at entity boundaries after the completion of the translation process without the loss of crucial information.

As per Algorithm 4, both the plain sentence and entity-anchored sentence are independently translated from the source language into the target language. In Stage 1, we first consider the entity-anchored sentence without the anchor symbols and compare it with the corresponding plain translated sentence. If discrepancies exist between these two versions, then the sentence is discarded.

Next, in Stage 2, the entity boundary errors are identified: sentences in which an entity mention contains a start anchor but lacks a corresponding closing anchor (or vice versa) are removed. Finally, in Stage 3, if the total count of entity types in the translated sentence does not match that of the original source sentence, then the translated sentence is discarded, as such discrepancies indicate instances where a valid entity mention has been incorrectly omitted. After filtering out erroneous sentences, we obtain a clean, FgNER dataset for the target language.

6.4 Experimental Setup

The state-of-the-art approach for sequence labeling tasks involves fine-tuning pre-trained language models (PLM) with the NER datasets [24, 34, 75, 86, 87, 88, 95, 123, 124]. Similarly, we have fine-tuned mBERT (bert-base-multilingual-cased) [82] and IndicBERTv2 (IndicBERTv2-MLM-Sam-TLM) [149] for FgNER using the Hugging Face Transformers [156]. The models were fine-tuned

Algorithm 4: Entity-Anchored Machine Translation (EaMaTa) Framework

Input: S_{src}: Source dataset sentences in BIO format
Required: $\mathcal{TLT}$: Translation service
Output: S_{tgt}^{clean}: FgNER dataset in target language T

Preprocessing:

1. **for** each tokenized sentence $s \in S_{src}$ **do**
2. Convert to plain format: s_{src}^{plain}
3. Convert to entity-anchored format: Create s_{src}^{anchor}, by replacing entity mention m_{je} for the jth token belonging to entity type e with $\alpha_j m_{je} \beta_{je}$; where α_j: start anchor, β_{je}: end anchor.
4. **end for**

Translation:

1. **for** each $s \in S_{src}$ **do**
2. $s_{tgt}^{plain} = \mathcal{TLT}(s_{src}^{plain}), \quad s_{tgt}^{anchor} = \mathcal{TLT}(s_{src}^{anchor})$
3. **end for**

Cleaning:

1. **for** each s_{tgt}^{anchor} **do**
2. **Stage 1: Sentence mismatch detection:** Convert (s_{tgt}^{anchor}) to $\overline{s}_{tgt}^{\textbf{plain}}$ by ignoring the anchors
3. **if** $\overline{s}_{tgt}^{\textbf{plain}} \neq s_{tgt}^{plain}$ **then**
4. Discard s_{tgt}^{anchor}
5. **end if**
6. **Stage 2: Entity-anchors mismatch detection:**
7. **if** For all m in s_{tgt}^{anchor}, there exists m_{je} such that $\alpha_j \notin s_{tgt}^{anchor}$ OR $\beta_{je} \notin s_{tgt}^{anchor}$ **then**
8. Discard s_{tgt}^{anchor}
9. **end if**
10. **Stage 3: Total entity counts mismatch detection:**
11. **if** $\sum (m \in s_{src}^{anchor}) \neq \sum (\ m \in s_{tgt}^{anchor})$ **then**
12. Discard s_{tgt}^{anchor}
13. **end if**
14. **end for**
15. **return** $S_{tgt}^{clean} = \{s_{tgt}^{anchor} \mid s_{tgt}^{anchor}$ is not discarded$\}$

as per the following hyperparameters: batch size: 64, epochs: 6, optimizer: AdamW, learning rate: 5e-5, and weight decay: 0.01. Fine-tuning was performed on an NVIDIA A100 GPU, with evaluation based on SeqEval metrics, and the best performance determined by the F1-score.

To compare with the noisy FgNER dataset TAFSIL, we considered their best-performing two variations of the DECENT model [121], where the base encoder RoBERTa-large [141] is changed to XLM-RoBERTa-large [83], and IndicBERTv2-MLM-Sam-TLM to accommodate fine-tuning with Indian languages. The hyperparameters for DECENT-based models are: learning rate for encoder: 5e-6, learning rate for head: 5e-4, dropout probability for head: 0.5, epoch: 2, batch size: 16,

Table 6.1: Translation metrics (**BLEU** and **TER**) for comparison of Google, Bing, and IndicTrans2 translators (**best values** are in **bold**).

	BLEU			TER		
Language	**Google**	**Bing**	**IndicTrans2**	**Google**	**Bing**	**IndicTrans2**
Assamese (as)	**43.35**	39.89	36.68	**42.62**	44.95	50.45
Bengali (bn)	**47.50**	45.34	42.40	**37.97**	38.46	41.53
Bhojpuri (bho)	**46.09**	43.75	–	**35.36**	41.11	–
Bishnupriya (bpy)	–	–	**31.15**	–	–	**54.40**
Bodo (brx)	–	**18.72**	18.14	–	**64.06**	64.19
Chhattisgarhi (hne)	–	**35.33**	–	–	**47.41**	–
Dogri (doi)	**56.76**	55.89	53.23	**28.57**	28.48	29.12
Gujarati (gu)	**36.49**	34.33	33.60	**46.08**	48.98	50.15
Hindi (hi)	**63.88**	58.90	50.94	24.03	**28.64**	34.65
Kannada (kn)	**27.67**	27.40	26.66	**56.61**	57.29	59.89
Kashmiri (ks)	–	**30.90**	30.60	–	50.44	**50.40**
Konkani (gom)	**35.81**	35.44	33.41	48.30	**47.03**	48.86
Maithili (mai)	**25.02**	22.02	21.27	**55.48**	59.15	60.67
Malayalam (ml)	**41.26**	37.98	36.33	**45.52**	48.53	50.80
Manipuri (mni)	**47.09**	44.74	43.45	**36.36**	42.10	42.76
Marathi (mr)	**32.40**	28.53	27.72	**52.41**	55.05	58.98
Mizo (lus)	**33.22**	–	–	**51.94**	–	–
Nepali (ne)	**35.76**	33.71	31.84	**47.25**	51.45	50.24
Oriya (or)	**21.49**	21.33	20.56	**59.36**	59.52	61.57
Punjabi (pa)	**49.63**	47.29	46.01	**34.22**	36.09	37.36
Sanskrit (sa)	**14.54**	12.87	13.75	**76.15**	79.05	77.09
Santali (sat)	24.69	–	**27.49**	57.46	–	**56.04**
Sindhi (sd)	**32.22**	31.95	31.13	**52.94**	52.96	54.41
Tamil (ta)	**29.89**	29.37	27.18	**56.96**	57.23	60.43
Telugu (te)	**31.83**	30.73	29.80	**52.31**	54.46	54.85
Urdu (ur)	**76.21**	74.69	71.56	**16.03**	17.02	19.89

negative oversampling rate: 31, and prediction threshold: 0.9.

6.4.1 Implementation of Entity-Anchored Machine Translation (EaMaTa) Framework

First, we selected three translation services based on their coverage of Indian languages: [160], [161], and [162]. We randomly chose three sets of 1000 English-to-Indian language sentence pairs from the BPCC corpus [145]. The English sentence from each pair was translated into the respective Indian language sentence. We assessed translation quality by comparing them to the corresponding Indian language sentences using BLEU, TER, chrF, and COMET metrics. The average performance across the three experimental sets is presented in Table 6.2 for all 26 languages. Based on our findings, we selected the best translation service for each language: IndicTrans2 for Santali, Bing Translator for Bodo and Kashmiri, and Google Translate for the remaining 19 languages.

To evaluate the proposed EaMaTa framework against the state-of-the-art, we first applied it to the MultiCoNER2 English train set to generate FgNER datasets in Bengali (bn), Hindi (hi), Marathi (mr), Tamil (ta), Telugu (te), Sanskrit (sa), and Urdu (ur), as shown in Table 6.3. Since

Table 6.2: Translation metrics (**chrF** and **COMET**) for comparison of Google, Bing, and IndicTrans2 translators (**best values** are in **bold**).

	chrF			COMET		
Language	**Google**	**Bing**	**IndicTrans2**	**Google**	**Bing**	**IndicTrans2**
Assamese (as)	69.97	67.99	**70.89**	**0.923**	0.879	0.880
Bengali (bn)	72.11	**74.37**	69.30	**0.924**	0.922	0.853
Bhojpuri (bho)	**72.15**	71.60	–	**0.894**	0.889	–
Bishnupriya (bpy)	–	–	**53.91**	–	–	**0.846**
Bodo (brx)	–	**57.95**	57.49	–	**0.647**	0.645
Chhattisgarhi (hne)	–	**67.91**	–	–	**0.881**	–
Dogri (doi)	**76.85**	76.61	74.57	**0.726**	0.725	0.711
Gujarati (gu)	**63.67**	63.24	62.5	**0.909**	0.901	0.905
Hindi (hi)	**79.87**	75.75	72.93	0.869	**0.890**	0.849
Kannada (kn)	67.06	67.19	**67.80**	**0.926**	0.898	0.901
Kashmiri (ks)	–	**61.77**	61.60	–	**0.803**	0.800
Konkani (gom)	**68.18**	68.12	64.50	**0.881**	0.868	0.876
Maithili (mai)	60.48	57.68	**60.70**	**0.739**	0.715	0.677
Malayalam (ml)	**75.23**	72.44	71.30	**0.933**	0.925	0.928
Manipuri (mni)	**73.18**	72.62	70.60	**0.914**	0.912	0.897
Marathi (mr)	**64.55**	62.31	60.24	**0.801**	0.792	0.767
Mizo (lus)	**55.95**	–	–	**0.865**	–	–
Nepali (ne)	**67.90**	66.61	64.01	**0.879**	0.867	0.872
Oriya (or)	60.50	**60.77**	59.89	**0.883**	0.879	0.876
Punjabi (pa)	**71.92**	71.41	70.78	**0.895**	0.893	0.892
Sanskrit (sa)	**56.42**	54.02	55.62	**0.830**	0.811	0.825
Santali (sat)	59.86	–	**61.57**	0.753	–	**0.777**
Sindhi (sd)	54.95	**55.91**	53.96	**0.855**	0.854	0.848
Tamil (ta)	**69.04**	68.93	59.44	**0.925**	0.921	0.892
Telugu (te)	**67.57**	66.35	65.44	**0.913**	0.908	0.907
Urdu (ur)	**85.47**	85.06	83.87	**0.912**	0.910	0.891

Table 6.3: Statistics of train set created via EaMaTa framework from the MultiCoNER2 English (en) set. Abbreviation: **Sents**: No of sentences.

Language	**Sents**	**Entities**	**Tokens**	**Language**	**Sents**	**Entities**	**Tokens**
English (en)	16,778	25,499	269,788	Bengali (bn)	13,014	30,063	273,510
Hindi (hi)	13,123	30,694	335,798	Marathi (mr)	13,085	30,180	272,732
Sanskrit (sa)	8,314	18,357	146,175	Tamil (ta)	12,024	26,540	206,609
Telugu (te)	12,489	29,686	230,102	Urdu (ur)	13,054	30,101	345,791

MultiCoNER2 was created via distant supervision and contains induced noise, we alternatively used the large-scale, manually annotated FewNERD dataset to build the FgNER dataset in Indian languages.

Table 6.4: Comparison of DECENT model performance with XLM-RoBERTa base encoder fine-tuned on different datasets. Abbreviations: **Size**: No. of train set samples, **Lang**: Language, MCN2: MultiCoNER2 dataset, TFSL-M: TAFSIL dataset in MultiCoNER2 taxonomy, CLSR: CLASSER dataset, EaMaTa: dataset built from the MultiCoNER2 English dataset with the EaMaTa framework. **Best values** in **bold**, second best values are underlined. Percentage **F1** improvements are within (**brackets**).

	Dataset Splits			Macro			Micro		
Lang	**Test**	**Train**	**Size**	**P**	**R**	**F1**(↑%)	**P**	**R**	**F1**(↑%)
Hindi	MCN2	MCN2	9.6k	73.8	70.8	72.3	72.2	70.8	71.5
	MCN2	EaMaTa	13k	71.4	75.9	**73.6**(**1.80**)	70.6	75.1	**72.8**(**1.82**)
Bengali	MCN2	MCN2	9.7k	72.7	69.8	71.2	71.3	69.9	70.6
	MCN2	EaMaTa	13k	69.7	74.1	**71.8**(**0.84**)	69.1	73.4	**71.2**(**0.85**)
Hindi	TFSL-M	TFSL-M	644k	64.6	76.2	70.0	64.5	76.2	70.0
	TFSL-M	EaMaTa	13k	68.7	73.0	**70.8**(**1.14**)	68.9	73.1	**70.9**(**1.29**)
Marathi	TFSL-M	TFSL-M	126k	50.9	81.9	62.3	49.4	83.7	62.1
	TFSL-M	CLSR	13k	62.7	66.8	64.7(**3.85**)	62.1	65.9	64.2(**3.38**)
	TFSL-M	EaMaTa	13k	63.0	66.9	**64.9**(**4.17**)	62.7	66.6	**64.6**(**4.03**)
Tamil	TFSL-M	TFSL-M	464k	51.7	81.1	64.0	50.7	81.0	63.4
	TFSL-M	EaMaTa	12k	63.2	67.2	**65.1**(**1.72**)	63.6	65.6	**64.5**(**1.74**)
Telugu	TFSL-M	TFSL-M	392k	52.5	82.8	64.9	52.1	81.3	64.2
	TFSL-M	EaMaTa	12k	64.8	68.9	**66.8**(**2.93**)	64.1	68.2	**66.1**(**2.96**)
Sanskrit	TFSL-M	TFSL-M	18k	32.7	53.3	40.6	31.4	53.4	40.2
	TFSL-M	CLSR	8.3k	42.8	45.0	43.9(**8.13**)	42.4	44.8	43.5(**8.21**)
	TFSL-M	EaMaTa	8.3k	42.9	45.6	**44.2**(**8.87**)	42.5	45.2	**43.9**(**9.20**)
Urdu	TFSL-M	TFSL-M	604k	63.0	77.1	69.4	60.2	78.3	68.1
	TFSL-M	EaMaTa	13k	68.3	72.6	**70.4**(**1.44**)	67.1	71.3	**69.1**(**1.47**)

6.5 Comparison with State-of-the-Art

Following [29], we performed rigorous analysis against two state-of-the-art datasets: the MultiCoNER2 dataset in Bengali and Hindi, and the TAFSIL dataset in Hindi, Marathi, Tamil, Telugu, Sanskrit, and Urdu. As shown in Table 6.4 and Table 6.5, the dataset generated by the EaMaTa framework is of very high quality with up to 9% increase in F1-score against the current state-of-the-art. In fact, it is superior to the CLASSER framework introduced in the last chapter. Hence, we continued to utilize the proposed EaMaTa framework to create the silver-standard FgNER dataset in 26 Indian languages using the FewNERD dataset.

6.6 Dataset Details

As presented in Table 6.6, the generated FgNER dataset, on average, consists of over 153 thousand sentences, 354 thousand entities, and 3.3 million tokens for each of the 26 Indian languages. The generated dataset is the first FgNER dataset across all 26 Indian languages. This dataset is a silver dataset as all the train, test, and development sets are translated and cleaned from the large-scale manually annotated FewNERD dataset.

Table 6.5: Comparison of DECENT model performance with IndicBERTv2 base encoder fine-tuned on different datasets. Abbreviations: **Size**: No. of train set samples, **Lang**: Language, MCN2: MultiCoNER2 dataset, TFSL-M: TAFSIL dataset in MultiCoNER2 taxonomy, CLSR: CLASSER dataset, EaMaTa: dataset built from the MultiCoNER2 English dataset with the EaMaTa framework. **Best values** in **bold**, second best values are underlined. Percentage **F1** improvements are within (**brackets**).

Dataset Splits				Macro			Micro		
Language	**Test**	**Train**	**Size**	**P**	**R**	**F1**(↑%)	**P**	**R**	**F1**(↑%)
Hindi	MCN2	MCN2	9.6k	66.6	76.3	71.1	66.4	76.2	71.0
	MCN2	EaMaTa	13k	70.1	74.5	**72.2**(**1.55**)	70.0	74.4	**72.1**(**1.55**)
Bengali	MCN2	MCN2	9.7k	72.5	69.7	71.1	71.2	69.7	70.2
	MCN2	EaMaTa	13k	69.6	74.0	**71.7**(**0.84**)	68.7	73.1	**70.9**(**1.00**)
Hindi	TFSL-M	TFSL-M	644k	67.4	75.6	71.2	67.2	75.6	71.1
	TFSL-M	EaMaTa	13k	70.0	74.4	**72.1**(**1.26**)	69.9	74.3	**72.0**(**1.27**)
Marathi	TFSL-M	TFSL-M	126k	49.3	81.5	61.5	48.6	82.8	61.2
	TFSL-M	CLSR	13k	61.7	65.7	63.7(**3.58**)	61.5	66.2	63.2(**3.27**)
	TFSL-M	EaMaTa	13k	62.0	65.9	**63.9**(**3.9**)	61.7	66.6	**63.6**(**3.92**)
Tamil	TFSL-M	TFSL-M	464k	51.0	80.9	63.1	50.1	80.1	62.9
	TFSL-M	EaMaTa	12k	62.1	66.0	**64.0**(**1.43**)	61.9	65.8	**63.8**(**1.43**)
Telugu	TFSL-M	TFSL-M	392k	51.0	81.6	64.3	51.0	80.9	63.9
	TFSL-M	EaMaTa	12k	63.9	68.0	**65.9**(**2.49**)	63.6	67.6	**65.5**(**2.50**)
Sanskrit	TFSL-M	TFSL-M	18k	32.9	56.0	41.5	32.7	56.0	41.0
	TFSL-M	CLSR	8.3k	43.8	46.6	45.1(**8.67**)	43.1	46.0	44.4(**8.29**)
	TFSL-M	EaMaTa	8.3k	44.1	46.8	**45.4**(**9.40**)	43.6	46.3	**44.9**(**9.5**)
Urdu	TFSL-M	TFSL-M	604k	58.7	83.9	69.1	57.0	83.4	67.7
	TFSL-M	EaMaTa	13k	67.9	72.2	**70.0**(**1.30**)	66.6	70.8	**68.6**(**1.33**)

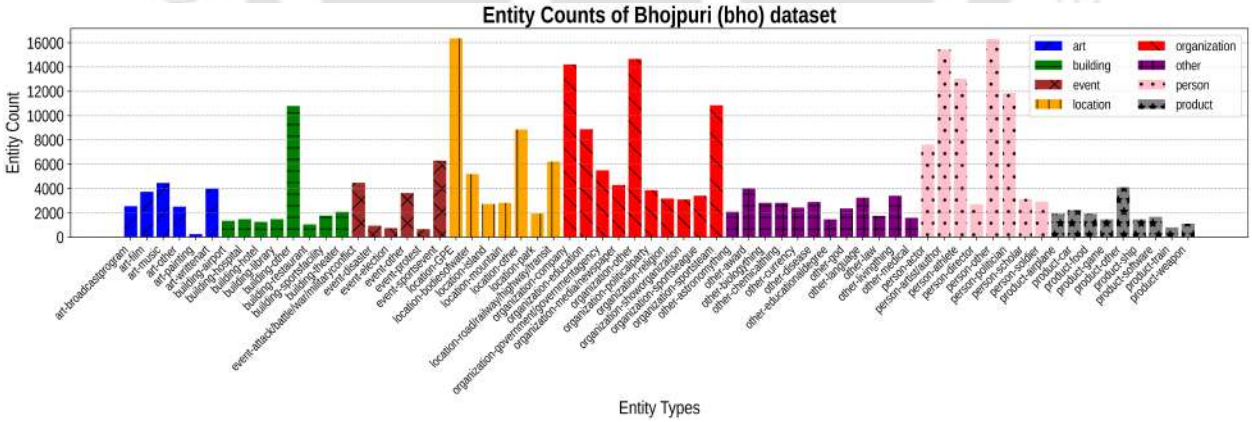

Figure 6.2: Entity type frequency distribution of the Bhojpuri (bho) dataset. Each color depicts the coarse-grained type, and each bar depicts the corresponding fine-grained types.

6.6.1 Gold Test Set

As shown in Table 6.7, 1000 sentences have been randomly selected from the silver test set, and annotated by at least two annotators. All volunteer annotators have a minimum undergraduate

Table 6.6: Statistics of FewNERD (en†) and the generated datasets. Abbreviations: **Ln**: Language; **Sent**, **Entity**, **Token** mean the number of sentences, entities, and tokens, respectively.

	Train set			Development set			(Silver) Test set		
Ln	**Sent**	**Entity**	**Token**	**Sent**	**Entity**	**Token**	**Sent**	**Entity**	**Token**
en†	131,767	340,387	3,359,329	18,824	48,770	482,037	37,648	96,902	958,765
as	107,249	237,260	2,194,925	15,438	34,560	318,105	30,658	67,466	625,870
bho	111,276	268,709	2,819,109	15,909	38,518	405,276	31,831	76,469	805,066
bn	119,296	287,264	2,484,304	17,513	42,877	368,063	33,374	79,340	689,690
bpy	79,349	161,191	1,646,195	11,700	23,987	244,318	25,296	42,842	517,433
brx	117,659	262,792	2,354,696	16,762	37,496	336,269	33,615	74,576	672,246
doi	112,329	264,154	2,885,149	17,619	42,526	459,537	34,931	82,597	903,796
gom	83,415	182,806	1,637,018	12,276	27,262	243,817	23,759	51,483	463,980
gu	126,581	315,919	2,828,298	18,122	45,431	406,929	28,959	69,207	619,889
hi	124,887	290,192	3,298,116	17,882	41,824	457,573	35,713	82,440	908,513
hne	108,979	242,460	4,422,373	15,633	35,048	484,212	31,108	68,743	751,897
kn	115,565	266,523	2,083,241	16,962	39,781	308,326	26,327	59,365	453,817
ks	123,679	288,544	2,910,937	17,417	40,350	408,053	35,106	81,181	823,040
lus	78,762	183,595	1,983,240	12,232	29,004	313,115	27,826	67,254	717,052
mai	108,826	256,701	2,763,005	10,224	22,706	245,657	19,899	43,530	472,498
ml	91,743	199,485	1,504,839	15,608	35,140	265,049	23,480	50,319	377,213
mni	110,068	246,084	2,264,925	15,561	34,869	321,556	31,463	69,739	644,709
mr	125,543	309,220	2,614,024	17,650	43,407	367,882	36,237	89,295	754,851
ne	125,695	311,439	2,661,064	18,252	45,778	389,382	35,498	87,112	747,802
or	118,633	289,943	2,427,051	18,090	45,247	376,152	32,477	78,893	657,395
pa	96,986	234,436	2,348,393	17,655	44,415	443,788	36,920	92,655	928,798
sa	69,581	152,269	1,214,021	10,043	22,175	176,574	19,729	42,643	341,208
sat	87,650	153,533	2,223,951	12,526	22,159	312,706	24,921	43,264	619,556
sd	90,362	214,371	2,218,078	17,221	42,845	440,340	32,159	78,317	809,085
ta	96,004	216,285	1,711,203	10,702	23,542	183,893	25,160	55,927	441,141
te	85,893	193,425	1,505,321	16,790	39,909	309,345	21,729	47,988	372,946
ur	122,794	298,069	3,229,867	17,570	43,205	465,417	35,198	85,785	929,427

degree and are chosen based on their mother tongue. The good quality of these manually annotated datasets can be ascertained as the inter-annotator agreement (IAA(κ)) is above 0.8 for each language.

6.6.2 Entity Type Frequency Distribution

As shown in Figure 6.2, a larger number of entity mentions are detected for different fine types of person, location, organization, and building-other. On the other hand, very specific fine types, such as painting, disaster, election, protest, and fine types of products, are fewer in number. Similar trends can be observed across all four languages.

Table 6.7: Statistics of the gold test set. **IAA** (κ) gives the inter-annotator agreement.

Language	**Sentences**	**Entities**	**Tokens**	**IAA**(κ)
Assamese (as)	1,000	2,201	20,415	0.812
Bengali (bn)	1,000	2,377	20,666	0.834
Bishnupriya (bpy)	1,000	2,052	20,057	0.807
Bodo (brx)	1,000	2,219	19,999	0.804
Hindi (hi)	1,000	2,308	25,439	0.863
Marathi (mr)	1,000	2,464	20,831	0.855
Mizo (lus)	1,000	2,311	24,824	0.821
Nepali (ne)	1,000	2,454	21,265	0.831
Sanskrit (sa)	1,000	2,162	17,296	0.801
Tamil (ta)	1,000	2,223	17,534	0.829
Telugu (te)	1,000	2,209	17,164	0.825
Urdu (ur)	1,000	2,437	26,406	0.818

Table 6.8: Human evaluation and entity errors of FewNERD (en†) and generated datasets, Abbreviations: **Ln**: Language, **XSTS**: Crosslingual Semantic Text Similarity, **BM**: Boundary Mismatch, **TM**: Type Mismatch, and **SP**: Spurious Entity.

		Entity Errors					**Entity Errors**		
Ln	**XSTS**	**BM**	**TM**	**SP**	**Ln**	**XSTS**	**BM**	**TM**	**SP**
as	3.77	14.3	14.8	4.7	bho	3.72	14.4	10.6	5.7
bn	4.14	13.6	13.9	4.9	bpy	3.45	16.4	13.2	6.0
brx	3.65	13.2	15.7	4.5	doi	3.72	18.8	10.1	7.3
gom	4.04	16.9	13.4	5.9	gu	4.23	13.6	13.7	4.9
hi	4.07	18.0	9.1	8.8	hne	3.63	16.1	9.8	6.6
kn	3.63	14.3	13.5	5.1	ks	3.62	18.2	11.5	6.9
lus	3.88	12.1	11.8	6.0	mai	3.59	17.3	10.9	7.1
ml	3.92	15.1	13.6	5.7	mni	3.42	18.9	17.7	5.6
mr	4.11	13.1	14.0	4.8	ne	4.08	13.1	14.7	3.9
or	4.02	14.8	13.5	5.2	pa	4.16	16.7	11.7	6.3
sa	3.59	14.9	13.3	6.7	sat	3.45	19.9	18.1	9.9
sd	3.58	19.1	10.9	7.3	ta	3.89	15.9	13.6	5.5
te	4.19	15.1	13.4	5.6	ur	4.05	17.8	11.1	6.6

6.7 Results & Analysis

This section covers the analyses performed on the generated dataset, including human evaluations, analysis of PLMs' performance fine-tuned on the generated dataset, cross-lingual zero-shot evaluation, impact of multilingualism and script similarity, entity error analysis, and ablation study. Initially, all the experiments were performed considering both silver and gold test datasets, in which the variations in the results are within ±5% (Table 6.10). Since silver test datasets are available for 26 languages compared to the gold test set for 12 languages only, further analyses are conducted with the silver test set.

Table 6.9: Performance of different models fine-tuned on the generated train sets, evaluated on the gold test set. en† denotes the FewNERD dataset.

	mBERT						IndicBERTv2					
	Macro			Micro			Macro			Micro		
Ln	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
en†	59.1	63.9	61.3	65.2	69.1	67.1	58.2	62.5	60.1	64.0	68.2	66.0
as	53.1	55.6	54.3	59.0	62.5	60.6	54.3	57.4	55.7	60.1	63.6	61.8
bn	56.2	58.4	57.2	61.1	64.3	64.3	56.5	59.7	57.9	61.8	65.0	63.5
bpy	53.8	56.1	54.9	47.8	52.7	50.6	52.2	54.7	53.4	47.9	50.1	49.2
brx	55.3	58.0	56.5	60.0	63.5	61.7	56.1	59.0	57.3	61.1	64.1	62.6
hi	53.8	56.3	54.9	58.6	62.0	60.2	53.4	56.2	54.7	58.7	61.3	60.0
lus	62.6	66.6	64.6	57.2	61.9	59.4	60.4	65.1	62.7	55.8	61.2	58.4
mr	56.4	58.5	57.4	62.2	65.2	63.7	56.4	59.2	57.6	62.2	65.4	63.9
ne	57.6	60.0	58.7	63.0	65.8	64.4	58.1	60.4	59.1	63.1	65.8	64.5
sa	52.4	54.5	53.2	58.5	61.8	60.1	51.1	54.7	52.7	58.3	62.1	60.2
ta	52.5	54.8	53.4	58.8	62.0	60.3	52.6	55.2	53.7	58.9	62.2	60.6
te	52.9	54.3	53.8	58.4	61.6	59.9	54.1	56.4	55.0	59.9	63.2	61.7
ur	54.1	57.0	55.4	59.2	62.6	60.8	52.8	56.2	54.4	58.3	61.9	60.2

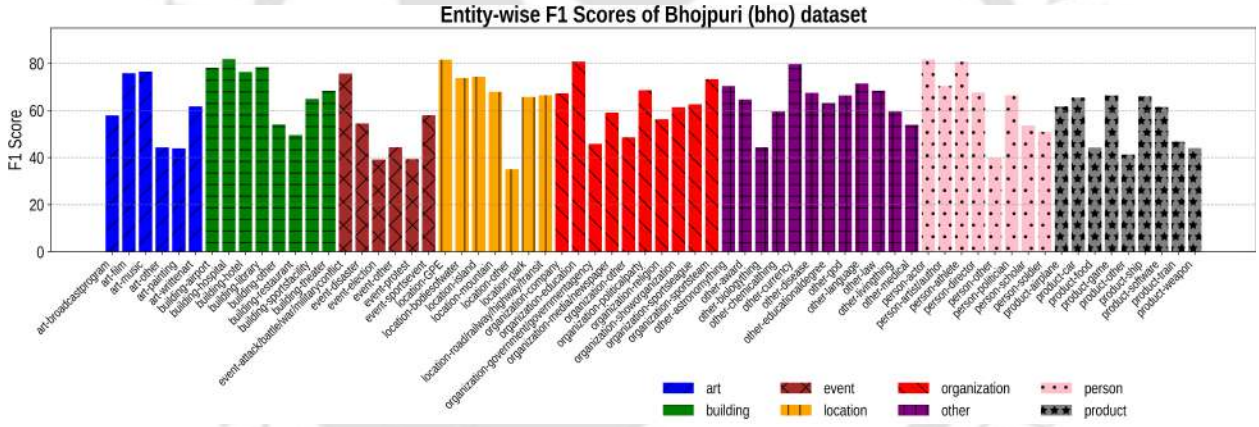

Figure 6.3: Entity-wise F1 Scores of Bhojpuri (bho) dataset. Each color depicts the coarse-grained type, and each bar depicts the corresponding fine-grained types.

6.7.1 Human Evaluation

To ensure the reliability and quality of the generated dataset, human evaluation was conducted as per the XSTS methodology [163], which focuses on meaning preservation by having annotators rate translated sentences from 1 to 5. Following [145, 164], two native-speaker annotators rated 100 sentences per language. All annotators are at least undergraduate degree holders. The consistently high XSTS scores (Table 6.6) confirm the good quality of the generated dataset.

6.7.2 Performance of PLMs on Unseen Languages

As shown in the Table 6.10, the mBERT model fine-tuned on the generated dataset has performed fairly well in the Indian languages it has been pre-trained on: Bengali (bn), Gujarati (gu), Hindi (hi), Kannada (kn), Malayalam (ml), Marathi (mr), Nepali (ne), Punjabi (pa), Tamil (ta), Telugu

Table 6.10: Performance of different models fine-tuned on the generated train sets, evaluated on the silver test set. en† denotes the FewNERD dataset.

	mBERT						IndicBERTv2					
	Macro			Micro			Macro			Micro		
Ln	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
en†	59.1	63.9	61.3	65.2	69.1	67.1	58.2	62.5	60.1	64.0	68.2	66.0
as	54.1	56.7	55.3	60.1	63.7	61.8	56.7	60.0	58.2	62.6	66.2	64.4
bho	60.4	64.2	62.2	53.9	56.9	55.2	62.8	66.9	64.8	56.4	60.4	58.2
bn	57.3	59.5	58.3	62.3	65.5	63.8	59.0	62.4	60.5	64.4	67.7	66.0
bpy	54.8	57.2	56.0	48.7	53.7	51.6	53.2	55.7	54.4	48.7	51.1	50.1
brx	56.4	59.1	57.6	61.2	64.7	62.9	58.6	61.6	59.9	63.6	66.7	65.1
doi	52.5	55.4	53.6	58.1	61.8	59.9	54.1	57.6	55.5	59.3	63.6	61.4
gom	51.2	54.1	52.5	57.7	61.8	59.7	53.8	56.4	54.8	60.0	63.9	61.9
gu	55.9	57.8	56.7	62.2	65.2	63.7	58.7	61.3	59.8	64.8	67.9	66.3
hi	54.8	57.4	56.0	59.7	63.2	61.4	55.8	58.7	57.1	61.1	63.8	62.4
hne	59.6	63.1	61.3	55.8	57.1	56.5	61.4	66.1	63.7	55.9	59.9	58.1
kn	55.2	58.2	56.5	61.4	64.9	63.1	58.0	61.2	59.3	63.8	67.0	65.4
ks	51.6	54.3	52.8	58.4	61.8	60.1	52.6	55.9	54.0	59.3	63.3	61.2
lus	63.8	67.9	65.8	58.3	63.1	60.5	61.6	66.3	63.9	56.9	62.4	59.5
mai	51.4	53.8	52.5	58.4	61.9	60.1	53.5	57.0	55.0	60.2	64.3	62.2
ml	53.5	55.6	54.4	58.8	62.1	60.4	56.5	59.2	57.8	62.0	65.4	63.6
mni	18.6	4.4	6.4	25.3	7.3	11.3	49.7	50.1	49.6	55.6	58.8	57.2
mr	57.5	59.6	58.5	63.4	66.5	64.9	58.9	61.8	60.2	64.8	68.1	66.4
ne	58.7	61.2	59.8	64.2	67.1	65.6	60.7	63.1	61.7	65.7	68.5	67.1
or	24.9	9.1	12.4	32.0	11.4	16.8	57.2	60.6	58.8	63.1	66.4	64.7
pa	51.9	54.2	52.9	58.7	61.9	60.3	54.6	58.1	56.2	61.2	65.1	63.1
sa	53.4	55.5	54.2	59.6	63.0	61.3	53.4	57.1	55.0	60.7	64.7	62.6
sat	38.2	12.0	17.1	43.0	15.5	22.8	41.3	42.3	41.4	50.6	51.8	51.2
sd	35.6	34.9	35.0	43.3	42.7	43.0	50.7	53.8	52.1	57.9	61.6	59.7
ta	53.5	55.9	54.4	59.9	63.2	61.5	54.9	57.7	56.1	61.3	64.8	63.0
te	53.9	55.3	54.4	59.5	62.8	61.1	56.5	58.9	57.4	62.4	65.8	64.1
ur	55.1	58.1	56.5	60.3	63.8	62.0	55.2	58.7	56.8	60.7	64.5	62.6

(te) and Urdu (ur). It is interesting to see that on some unseen languages such as Assamese (as), Bodo (brx), Dogri (doi), Konkani (gom), Maithili (main), Sanskrit (sa), and Kashmiri (ks), mBERT model performed well. This is due to script similarity of Bengali with Assamese, use of Arabic script for Kashmiri and Urdu, and Devanagari script for Hindi and other unseen languages mentioned above. But, for the unseen languages having a different script, such as Manipuri (mni), Oriya (or), Santali (sat) etc., the fine-tuned mBERT model could not perform well in terms of F1-score. However, such variations are not observed in the case of IndicBERTv2 (Table 6.10) as the PLM is pre-trained on all 26 Indian languages. Hence, all the 26 fine-tuned IndicBERTv2 models on the generated dataset performed better than corresponding fine-tuned mBERT models for the respective language. Table 2.1 in Chapter 2 shows the language coverage of the considered multilingual encoder models.

Figure 6.4: Zero-shot performance (micro F1) of fine-tuned mBERT models on different test languages.

6.7.3 Cross-Lingual Zero-Shot Analyses

We have performed cross-lingual zero-shot analyses for every language pair, including the English dataset from FewNERD. As shown in Figure 6.4, and Figure 6.5, the models are fine-tuned on the dataset of respective languages and tested on the test set of other languages. The pre-trained knowledge of mBERT in English is very proficient, due to which the zero-shot performance is very high across all languages. In fact, it is sometimes higher than the language-specific fine-tuned models for unseen languages such as Manipuri (mni), Santali (sat), Sindhi (sd) etc. The impact of script similarity depends on the languages covered during pre-training of the PLM. For example, when mBERT is fine-tuned with Assamese (as), on which it was not pre-trained, the zero-shot performance on Bengali (bn) is 49.5 micro-F1 score. Whereas, when it is fine-tuned with Bengali, on which it was pre-trained, the zero-shot performance in Assamese drops down to a 28.3 micro-F1 score, although both languages are written using Bengali-Assamese script. Whereas, such variations are not observed when the similar tests are performed in fine-tuned IndicBERTv2 due to its pre-training on all the 26 languages (Figure 6.5).

6.7.4 Multilingualism & Script Similarity

Our analysis extends to evaluating multilingualism, script similarity, and the influence of language families. We constructed balanced datasets comprising multiple languages, ensuring an equal num-

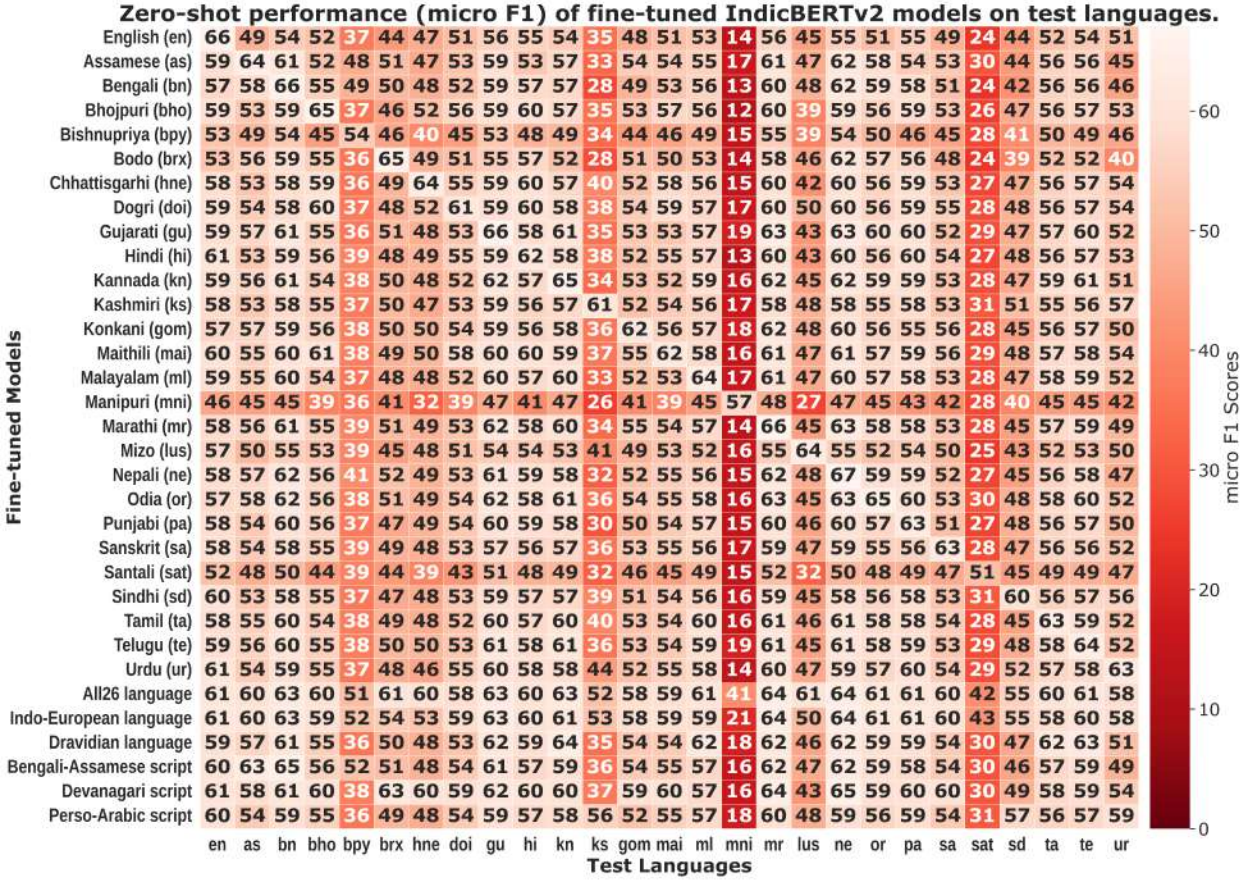

Figure 6.5: Zero-shot performance (micro F1) of fine-tuned IndicBERTv2 models on different test languages.

ber of samples per language. The **All26** set includes all 26 Indian languages, while the **Indo-European language** set consists of Indo-European languages: Assamese (as), Bengali (bn), Dogri (doi), Konkani (gom), Gujarati (gu), Hindi (hi), Kashmiri (ks), Maithili (mai), Marathi (mr), Nepali (ne), Odia (or), Punjabi (pa), Sanskrit (sa), Sindhi (sd), and Urdu (ur). Similarly, the **Dravidian language** set includes Kannada (kn), Malayalam (ml), Tamil (ta), and Telugu (te).

Based on script similarity, we categorized languages as follows: Assamese and Bengali under **Bengali-Assamese script**, Bodo (brx), Dogri, Konkani, Hindi, Maithili, Marathi, Nepali, and Sanskrit under **Devanagari script**, and Kashmiri, Sindhi, and Urdu under **Perso-Arabic script**. As expected, the best model performance for each language is achieved when fine-tuned exclusively on that language (Figure 6.4 and Figure 6.5). Performance also improves when languages share the same script or belong to the same family, while it typically degrades when the language differs in script or linguistic lineage.

However, two intriguing observations emerge. First, when IndicBERTv2 is fine-tuned exclusively on Arabic-script languages, its zero-shot performance on Santali (sat) is markedly improved, even exceeding that of the model fine-tuned on all 26 languages despite the inclusion of Santali samples in the latter. Second, Manipuri (mni), a Sino-Tibetan language, attains superior zero-shot results when the model is fine-tuned on Dravidian languages rather than on the complete 26-language set.

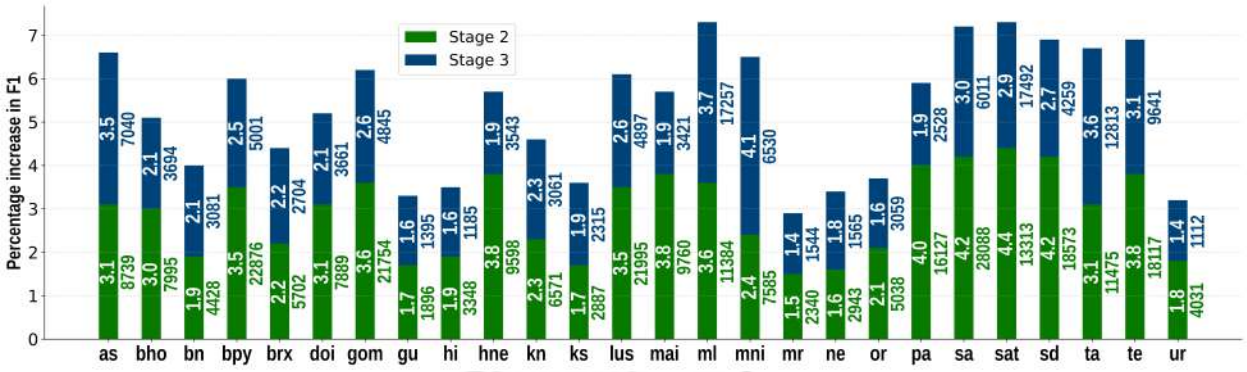

Figure 6.6: Impact of Stage 2 and Stage 3 on percentage increase in F1 scores. Values inside bars indicate percentage increase of F1-scores after Stage 2 and Stage 3. Values outside indicate the number of sentences removed at each stage.

6.7.5 Error Analysis

FgNER is very crucial, as depending on the context, an entity's mention type may vary significantly. Therefore, we have analyzed the errors in two different approaches. First, the details of entity errors in terms of the percentage of predicted entities are shown in Table 6.6. The common errors that occur include the Boundary Mention error (such as 'Good Dinosaur' is marked as *film* instead of 'The Good Dinosaur'), Entity Type error (e.g. 'Fainting' is categorized as *Disease* instead of *Symptom*) and Spurious Entity errors (such as 'red' is marked as an entity although the entity type *color* is not defined in FewNERD taxonomy).

Moreover, we have analyzed the often co-predicted fine-grained types. From Table 6.10, we have selected Santali (sat) for this analysis, as this language has the highest percentage of mismatched entity types. As shown in Figure 6.7, the fine types *actor*, *artist/author*, and *politician* are confused with *person-other*. Similarly, *location-GPE* is sometimes confused with *location-other*. Apart from such closely related entity types, most of the other fine entity types are learned by the models without much confusion.

6.7.6 Ablation Study

The impact of Stage 2 and Stage 3 are discussed through the Figure 6.6. The removal of erroneous sentences due to entity-anchors mismatch in Stage 2 improves the F1 scores by 2.8% on average. Similarly, the effect of the removal of sentences having a mismatch of total entity counts in Stage 3 is imminent, with an average improvement of F1 scores by 2.4%. Both stages improve the recall and eventually improve the F1 scores.

Our analyses confirm the high quality of the generated dataset and the effectiveness of the EaMaTa framework. Cross-lingual zero-shot analyses underscore the importance of language-specific pre-training and task-specific fine-tuning of PLMs. Fine-tuning IndicBERTv2 on the generated dataset yields highly effective FgNER performance, while the All26 model further demonstrates robust cross-lingual capabilities, making it well suited for broad multilingual applications.

6.8 Discussion

Translation-based dataset creation risks introducing errors and is limited by the source dataset's size and diversity. Although the FewNERD dataset spans multiple domains, the generated dataset still

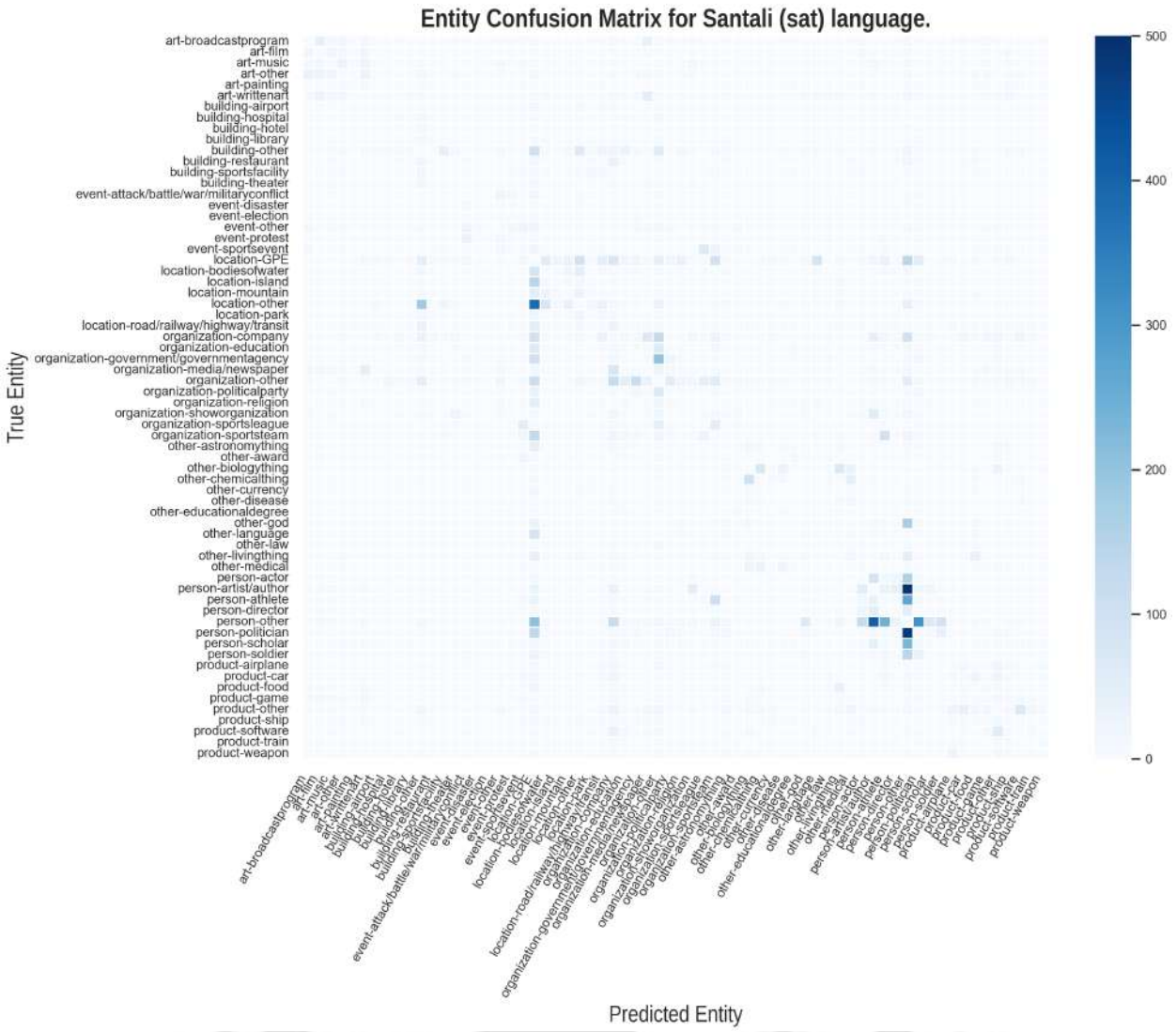

Figure 6.7: Entity Confusion matrix of Santali (sat) language. Darker squares depict the entity type pairs often confused by the fine-tuned model.

lacks certain linguistic, geographical, and cultural nuances specific to the Indian subcontinent. In the future, we aim to develop culturally enriched datasets spanning all 26 languages. Furthermore, we plan to systematically evaluate the FgNER capabilities of diverse large language models across these languages.

6.9 Conclusion

We introduced the entity-anchored machine translation (EaMaTa) framework to create a high-quality FgNER dataset, on average comprising over 153,000 annotated sentences for each of the 26 Indian languages. This proposed method mitigates the need for a language-specific Wikipedia corpus for dataset generation, as discussed in Chapter 3, and dependence on the availability of FgNER resources in a script-similar language, as described in Chapter 5. To the best of our knowledge, it is the first FgNER dataset covering all these languages. Extensive experiments validate the

high quality of the dataset created through the EaMaTa framework, compared to various dataset generation methods described in the previous chapters. Cross-lingual zero-shot analyses highlight the importance of language-specific pre-training and task-specific fine-tuning. We believe the generated dataset and the associated fine-tuned models will advance NER across Indian languages and support broader multilingual research.

The SampurNER dataset, associated fine-tuned models and interactive web application are publicly available at: hf.co/collections/prachuryyaIITG/SampurNER

7

Conclusion

This thesis presents four contributions that represent a substantial and concerted effort to overcome the significant resource scarcity for Fine-grained Named Entity Recognition (FgNER) in Indian languages. Three distinct, yet powerful, frameworks have been successfully introduced and validated: TAFSIL, for distant supervision and taxonomy adaptation; CLASSER, a cross-lingual projection method enhanced by script similarity; and EaMaTa, an entity-anchored machine translation approach. Moreover, annotation projection is utilized to create the FgNER dataset in extremely resource-scarce vulnerable languages. These contributions have resulted in the creation of high-quality FgNER datasets in five different taxonomies that collectively cover 26 Indian languages, including four vulnerable languages (Bishnupriya, Bodo, Manipuri, and Mizo), spoken by over two billion people worldwide. Rigorous experiments on State-of-the-Art (SOTA) demonstrated the robustness and utility of generated datasets, yielding remarkable performance improvements, such as an average relative F1 score increase of 83% over zero-shot baselines with TAFSIL, and up to 46% and 9% improvements with CLASSER and EaMaTa, respectively. The comprehensive cross-lingual zero-shot analyses performed across three contributions consistently emphasize the crucial roles of both language-specific pre-training and task-specific fine-tuning for achieving effective FgNER in this challenging multilingual setting. Overall, this research has provided the necessary foundational data resources: TAFSIL, FiNERVINER, CLASSER, SampurNER, and FiNE-MiBBiC, which are essential to spur advancements in FgNER and broader multilingual research for Indian languages.

Limitations

Despite the encouraging and substantial results achieved, certain limitations inherent to the resource creation methodologies must be acknowledged. A primary constraint is the dependence on source data characteristics, as the projection and translation-based contributions (e.g., CLASSER, EaMaTa, etc.) risk inheriting biases or specificities from the source FgNER datasets (e.g., MultiCoNER2, FewNERD, etc.). Similarly, TAFSIL's reliance on Wikipedia confines the generated corpus to a specific text genre, potentially limiting the models' performance in cross-domain applications. Moreover, TAFSIL requires reliable linguistic tools, such as POS taggers for all target languages. Another limitation arises from the applicability of the alignment strategies: TAFSIL requires entity types to align with Wikidata key values, while the cross-lingual refinement in CLASSER relies heavily on the availability of resources in languages with similar scripts, limiting its scalability to languages with significantly different syntactic structures. Furthermore, the quality and volume of the datasets are intrinsically linked to the availability of external resources.

This includes high-quality parallel corpora, the selection of the annotation projection tool, and the multilingual encoder in the case of FiNERVINER and CLASSER. Finally, the translation-based methods, while effective, still risk lacking certain linguistic, geographical, and cultural nuances specific to the Indian subcontinent that are not captured by the original English source dataset.

Future Works

The successful creation and validation of these extensive FgNER resources naturally pave the way for several critical avenues of future research. A priority is the development of methodologies to generate culturally enriched and domain-diverse datasets across all 26 languages, moving beyond the current text genres to fully capture the linguistic and cultural heterogeneity of the Indian subcontinent. TAFSIL and EaMaTa framework can be combined to create Fine-grained NER datasets with evolving entity types, even for extremely low-resource languages. First, the evolving entity type mapping to Wikidata QIDs will be required to generate a distant supervision dataset through the TAFSIL framework. Since Wikipedia is most frequently updated in English compared to other languages, such a dataset in English can be translated and cleaned to other languages through the EaMaTa framework. However, the noise present in the distantly supervised dataset, the impact of the quality of the translation service, and computationally expensive training of the distantly supervised dataset will create further challenges.

A systematic and thorough evaluation of the performance of Large Language Models (LLMs) on the FgNER task is necessary for benchmarking in both zero-shot settings and after fine-tuning with the newly generated datasets (TAFSIL, FiNERVINER, CLASSER, SampurNER, and FiNE-MiBBiC). This will provide crucial insights into state-of-the-art performance. Furthermore, future work will involve enhancing the generalizability and robustness of the developed frameworks, including research into generalizing the CLASSER refinement stage to languages with varied syntactic structures and systematically investigating the impact of different combinations of parallel corpora and multilingual tools on the resulting dataset quality.

Bibliography

[1] Lora Aroyo, Matthew Lease, Praveen Paritosh, and Mike Schaekermann. Data excellence for ai: why should you care? *Interactions*, 29(2):66–69, 2022.

[2] Xiao Ling and Daniel S Weld. Fine-grained entity recognition. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[3] Besnik Fetahu, Zhiyu Chen, Sudipta Kar, Oleg Rokhlenko, and Shervin Malmasi. Multiconer v2: a large multilingual dataset for fine-grained and noisy named entity recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2027–2051, 2023.

[4] Archana Goyal, Vishal Gupta, and Manish Kumar. Recent named entity recognition and classification techniques: a systematic review. *Computer Science Review*, 29:21–43, 2018.

[5] Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70, 2020.

[6] Lisa F Rau. Extracting company names from text. In *Proceedings the Seventh IEEE Conference on Artificial Intelligence Application*, pages 29–30. IEEE Computer Society, 1991.

[7] Ralph Grishman and Beth M Sundheim. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996.

[8] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1), 2007.

[9] Georgios Petasis, Frantz Vichot, Francis Wolinski, Georgios Paliouras, Vangelis Karkaletsis, and Constantine D Spyropoulos. Using machine learning to maintain rule-based named-entity recognition and classification systems. In *proceedings of the 39th annual meeting of the association for computational linguistics*, pages 426–433, 2001.

[10] Sekine Satoshi. Nyu: Description of the japanese ne system used for met-2. In *Proceedings of the Seventh Message Understanding Conference (MUC-7), Fairfax, Virginia, USA, May 1998*, 1998.

[11] Casey Whitelaw and Jon Patrick. Evaluating corpora for named entity recognition using character-level features. In *Australasian Joint Conference on Artificial Intelligence*, pages 910–921. Springer, 2003.

[12] Enrique Alfonseca and Suresh Manandhar. An unsupervised method for general named entity recognition and automated concept discovery. In *Proceedings of the 1st international conference on general WordNet, Mysore, India*, pages 34–43, 2002.

[13] David Nadeau, Peter D Turney, and Stan Matwin. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In *Conference of the Canadian society for computational studies of intelligence*, pages 266–277. Springer, 2006.

[14] Sergey Brin. Extracting patterns and relations from the world wide web. In *International workshop on the world wide web and databases*, pages 172–183. Springer, 1998.

[15] S CUCERZAN. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 90–99, 1999.

[16] Rami Al-Rfou, Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. Polyglot-ner: Massive multilingual named entity recognition. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 586–594. SIAM, 2015.

[17] Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, 2017.

[18] Daniel Hanisch, Katrin Fundel, Heinz-Theodor Mevissen, Ralf Zimmer, and Juliane Fluck. Prominer: rule-based protein and gene entity recognition. *BMC bioinformatics*, 6(Suppl 1):S14, 2005.

[19] Chris Pal, Gideon Mann, and Richard Minerich. Putting semantic information extraction on the map: Noisy label models for fact extraction. In *Proceedings of the Workshop on Information Integration on the Web at AAAI*, 2007.

[20] Zhehuan Zhao, Zhihao Yang, Ling Luo, Yin Zhang, Lei Wang, Hongfei Lin, and Jian Wang. Ml-cnn: A novel deep learning based disease named entity recognition architecture. In *2016 IEEE international conference on bioinformatics and biomedicine (BIBM)*, pages 794–794. IEEE, 2016.

[21] Zhehuan Zhao, Zhihao Yang, Ling Luo, Lei Wang, Yin Zhang, Hongfei Lin, and Jian Wang. Disease named entity recognition from biomedical literature using a novel convolutional neural network. *BMC medical genomics*, 10(Suppl 5):73, 2017.

[22] Abhishek Abhishek, Sanya Bathla Taneja, Garima Malik, Ashish Anand, and Amit Awekar. Fine-grained entity recognition with reduced false negatives and large type coverage. In *AKBC*, 2019.

[23] Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. Ultra-fine entity typing. *arXiv preprint arXiv:1807.04905*, 2018.

[24] Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. Multiconer: A large-scale multilingual dataset for complex named entity recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3798–3809, 2022.

[25] Anubhav Shrimal, Avi Jain, Kartik Mehta, and Promod Yenigalla. NER-MQMRC: Formulating named entity recognition as multi question machine reading comprehension. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*, pages 230–238, Hybrid: Seattle, Washington + Online, July 2022. Association for Computational Linguistics.

[26] Simone Tedeschi and Roberto Navigli. Multinerd: A multilingual, multi-genre and fine-grained dataset for named entity recognition (and disambiguation). In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 801–812, 2022.

[27] Dan Gillick, Nevena Lazic, Kuzman Ganchev, Jesse Kirchner, and David Huynh. Context-dependent fine-grained entity type tagging. *arXiv preprint arXiv:1412.1820*, 2014.

[28] Abhishek Abhishek, Balaji Ganesan, Ashish Anand, and Amit Awekar. Collective learning from diverse datasets for entity typing in the wild. In *Proceedings of the 2nd International Workshop on Entity Retrieval*, volume 2446, 2019.

[29] Prachuryya Kaushik, Shivansh Mishra, and Ashish Anand. TAFSIL: Taxonomy adaptable fine-grained entity recognition through distant supervision for indian languages. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3753–3763, 2025.

[30] Prachuryya Kaushik and Ashish Anand. FiNERVINER: Fine-grained named entity recognition for vulnerable languages of india's north eastern region. In Stelios Piperidis, Núria Bel, Henk van den Heuvel, Nancy Ide, Simon Krek, and Antonio Toral, editors, *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, pages 7655–7667, Palma, Mallorca, Spain, May 2026. European Language Resources Association (ELRA).

[31] Prachuryya Kaushik and Ashish Anand. CLASSER: Cross-lingual annotation projection enhancement through script similarity for fine-grained named entity recognition. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 1745–1760, Mumbai, India, December 2025. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

[32] Prachuryya Kaushik and Ashish Anand. SampurNER: Fine-grained named entity recognition dataset for 22 indian languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 31410–31418, Mar. 2026.

[33] Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. Few-nerd: A few-shot named entity recognition dataset. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3198–3213, 2021.

[34] Rudra Murthy Venkataramana, Pallab Bhattacharjee, Rahul Sharnagat, Jyotsana Khatri, Diptesh Kanojia, and Pushpak Bhattacharyya. Hiner: A large hindi named entity recognition dataset. In *International Conference on Language Resources and Evaluation*, 2022.

[35] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, with Language Models*. 3rd edition, 2026. Online manuscript released January 6, 2026.

[36] Nancy Chinchor, Patricia Robinson, and Elizabeth Brown. Hub-4 IE-NE Task Definition Version 4.8, 1998.

[37] SEKINE Satoshi. Irex: Ir and ie evaluation-based project in japanese. In *Proceedings of the Language Resource and Evaluation Conference, 2000*, 2000.

[38] Erik F. Tjong Kim Sang. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002.

[39] George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Lrec*, volume 2, pages 837–840. Lisbon, 2004.

[40] Diana Santos, Nuno Seco, Nuno Cardoso, and Rui Vilela. Harem: An advanced ner evaluation contest for portuguese. In *quot; In Nicoletta Calzolari; Khalid Choukri; Aldo Gangemi; Bente Maegaard; Joseph Mariani; Jan Odjik; Daniel Tapias (ed) Proceedings of the 5 th International Conference on Language Resources and Evaluation (LREC'2006)(Genoa Italy 22-28 May 2006)*, 2006.

[41] Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. Named entity recognition: fallacies, challenges and opportunities. *Computer Standards & Interfaces*, 35(5):482–489, 2013.

[42] Daniel M Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. Nymble: a high-performance learning name-finder. In *fifth conference on applied natural language processing*, pages 194–201, 1997.

[43] Andrew Mccallum and Wei Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 188–191, 2003.

[44] Masayuki Asahara and Yuji Matsumoto. Japanese named entity extraction with redundant morphological analysis. In *Proceedings of the 2003 human language technology conference of the North American chapter of the association for computational linguistics*, pages 8–15, 2003.

[45] Andrew Borthwick, John Sterling, Eugene Agichtein, and Ralph Grishman. Nyu: Description of the mene named entity system as used in muc-7. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[46] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537, 2011.

[47] Vikas Yadav and Steven Bethard. A survey on recent advances in named entity recognition from deep learning models. In *27th International Conference on Computational Linguistics, COLING 2018*, pages 2145–2158. Association for Computational Linguistics (ACL), 2018.

[48] Liang-Jyh Wang, Wei-Chuan Li, and Chao-Huang Chang. Recognizing unregistered names for mandarin word identification. In *COLING 1992 Volume 4: The 14th International Conference on Computational Linguistics*, 1992.

[49] David D Palmer and David Day. A statistical profile of the named entity task. In *Fifth Conference on Applied Natural Language Processing*, pages 190–193, 1997.

[50] William J Black, Fabio Rinaldi, and David Mowatt. Facile: Description of the ne system used for muc-7. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[51] Dimitrios Kokkinakis. Aventinus, gate and swedish lingware. In *Proceedings of the 11th Nordic Conference of Computational Linguistics (NODALIDA 1998)*, pages 22–33, 1998.

[52] Erik F Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147, 2003.

[53] Joaquim Ferreira Da Silva, Zornitsa Kozareva, and José Gabriel Pereira Lopes. Cluster analysis and classification of named entities. In *LREC*, 2004.

[54] Xavier Carreras, Lluís Màrquez, and Lluís Padró. Named entity recognition for catalan using only spanish resources and unlabelled data. In *10th Conference of the European Chapter of the Association for Computational Linguistics*, 2003.

[55] Jonathan May, Ada Brunstein, Prem Natarajan, and Ralph Weischedel. Surprise! what's in a cebuano or hindi name? *ACM Transactions on Asian Language Information Processing (TALIP)*, 2(3):169–180, 2003.

[56] Eckhard Bick. A named entity recognizer for danish. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, 2004.

[57] Jakub Piskorski. Extraction of polish named-entities. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, 2004.

[58] Borislav Popov, Angel Kirilov, Diana Maynard, and Dimitar Manov. Creation of reusable components and language resources for named entity recognition in russian. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, 2004.

[59] Anil Kumar Singh. Named entity recognition for south and south east asian languages: taking stock. In *Proceedings of the IJCNLP-08 workshop on named entity recognition for South and South East Asian languages*, 2008.

[60] Shalini Gupta and Pushpak Bhattacharyya. Think globally, apply locally: using distributional characteristics for hindi named entity identification. In *Proceedings of the 2010 Named Entities Workshop*, pages 116–125, 2010.

[61] Mahathi Bhagavatula, GSK Santosh, and Vasudeva Varma. Language independent named entity identification using wikipedia. In *Proceedings of the First Workshop on Multilingual Modeling*, pages 11–17, 2012.

[62] Karthik Gali, Harshit Surana, Ashwini Vaidya, Praneeth M Shishtla, and Dipti Misra Sharma. Aggregating machine learning and rule based heuristics for named entity recognition. In *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*, 2008.

[63] Sujan Kumar Saha, Pabitra Mitra, and Sudeshna Sarkar. Word clustering and word selection based feature reduction for maxent based hindi ner. In *proceedings of ACL-08: HLT*, pages 488–495, 2008.

[64] Sujan Kumar Saha, Sudeshna Sarkar, and Pabitra Mitra. A hybrid feature set based maximum entropy hindi named entity recognition. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*, 2008.

[65] Sobha Lalitha Devi, Pattabhi RK Rao, CS Malarkodi, and R Vijay Sundar Ram. Indian language ner annotated fire 2014 corpus (fire 2014 ner corpus). *Named-Entity Recognition Indian Languages FIRE*, 2014.

[66] David Yarowsky and Grace Ngai. Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*, 2001.

[67] Pooja Rai and Sanjay Chatterji. Annotation projection-based dependency parser development for nepali. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(2):1–19, 2022.

[68] Arijit Nag, Bidisha Samanta, Animesh Mukherjee, Niloy Ganguly, and Soumen Chakrabarti. Transfer learning for low-resource multilingual relation classification. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(2):1–24, 2023.

[69] Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. Cheap translation for cross-lingual named entity recognition. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2536–2545, 2017.

[70] Arata Ugawa, Akihiro Tamura, Takashi Ninomiya, Hiroya Takamura, and Manabu Okumura. Neural machine translation incorporating named entity. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3240–3250, 2018.

[71] Alankar Jain, Bhargavi Paranjape, and Zachary C Lipton. Entity projection via machine translation for cross-lingual ner. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1083–1092, 2019.

[72] Jian Yang, Shaohan Huang, Shuming Ma, Yuwei Yin, Li Dong, Dongdong Zhang, Hongcheng Guo, Zhoujun Li, and Furu Wei. Crop: Zero-shot cross-lingual named entity recognition with multilingual labeled sequence translation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 486–496, 2022.

[73] Linlin Liu, Bosheng Ding, Lidong Bing, Shafiq Joty, Luo Si, and Chunyan Miao. Mulda: A multilingual data augmentation framework for low-resource cross-lingual ner. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5834–5846, 2021.

[74] Akshara Prabhakar, Gouri Sankar Majumder, and Ashish Anand. Cl-neril: A cross-lingual model for ner in indian languages (student abstract). 2022.

[75] Arnav Mhaske, Harshit Kedia, Sumanth Doddapaneni, Mitesh M. Khapra, Pratyush Kumar, Rudra Murthy, and Anoop Kunchukuttan. Naamapadam: A large-scale named entity annotated data for Indic languages. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10441–10456, Toronto, Canada, July 2023. Association for Computational Linguistics.

[76] Brayan Stiven Lancheros, Gloria Corpas Pastor, and Ruslan Mitkov. Data augmentation and transfer learning for cross-lingual named entity recognition in the biomedical domain. *Language Resources and Evaluation*, pages 1–20, 2024.

[77] Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Divyanshu Kakwani, Navneet Kumar, et al. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162, 2022.

[78] Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, et al. Xtreme-r: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10215–10245, 2021.

[79] Franz Josef OCH and Hermann NEY. A systematic comparison of various statistical alignment models. *Computational linguistics-Association for Computational Linguistics (Print)*, 29(1):19–51, 2003.

[80] K Karthikeyan, Zihan Wang, Stephen Mayhew, and Dan Roth. Cross-lingual ability of multilingual bert: An empirical study. In *International Conference on Learning Representations*, 2020.

[81] Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of bert. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, 2019.

[82] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, 2019.

[83] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *ACL*, 2020.

[84] Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*, 2021.

[85] Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M Khapra, Anoop Kunchukuttan, and Pratyush Kumar. Towards leaving no indic language behind: Building monolingual corpora, benchmark and models for indic languages. *arXiv preprint arXiv:2212.05409*, 2022.

[86] Dhrubajyoti Pathak, Sukumar Nandi, and Priyankoo Sarmah. Asner-annotated dataset and baseline for assamese named entity recognition. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6571–6577, 2022.

[87] Onkar Litake, Maithili Ravindra Sabane, Parth Sachin Patil, Aparna Abhijeet Ranade, and Raviraj Joshi. L3cube-mahaner: A marathi named entity recognition dataset and bert models. In *Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, pages 29–34, 2022.

[88] Nobal Niraula and Jeevan Chapagain. Named entity recognition for nepali: data sets and algorithms. In *The International FLAIRS Conference Proceedings*, volume 35, 2022.

[89] Asif Ekbal, Rejwanul Haque, and Sivaji Bandyopadhyay. Named entity recognition in bengali: A conditional random field approach. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*, 2008.

[90] Aniketh Janardhan Reddy, Monica Adusumilli, Sai Kiranmai Gorla, Lalita Bhanu Murthy Neti, and Aruna Malapati. Named entity recognition for telugu using lstm-crf. In *WILDRE4–4th Workshop on Indian Language Data: Resources and Evaluation*, volume 6, 2018.

[91] Jimmy Laishram, Kishorjit Nongmeikapam, and Sudip Kumar Naskar. Deep neural model for manipuri multiword named entity recognition with unsupervised cluster feature. In *Proceedings of the 17th international conference on natural language processing (ICON)*, pages 420–429, 2020.

[92] Laishram Jimmy, Kishorjit Nongmeikappam, and Sudip Kumar Naskar. Bilstm-crf manipuri ner with character-level word representation. *Arabian journal for science and engineering*, 48(2):1715–1734, 2023.

[93] Sanjib Narzary, Anjali Brahma, Sukumar Nandi, and Bidisha Som. Deep learning based named entity recognition for the bodo language. *Procedia Computer Science*, 235:2405–2421, 2024.

[94] Rajesh Mundotiya, Shantanu Kumar, Ajeet Kumar, Umesh Chaudhary, Supriya Chauhan, Swasti Mishra, Praveen Gatla, and Anil Kumar Singh. Development of a dataset and a deep learning baseline named entity recognizer for three low resource languages: Bhojpuri, maithili, and magahi. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(1):1–20, 2023.

[95] Ankur Priyadarshi and Sujan Kumar Saha. The first named entity recognizer in maithili: Resource creation and system development. *Journal of Intelligent & Fuzzy Systems*, 41(1):1083–1095, 2021.

[96] Satoshi Sekine, Kiyoshi Sudo, and Chikashi Nobata. Extended named entity hierarchy. In Manuel González Rodríguez and Carmen Paz Suarez Araujo, editors, *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain, May 2002. European Language Resources Association (ELRA).

[97] Ralph Weischedel and Ada Brunstein. Bbn pronoun coreference and entity type corpus. *Linguistic Data Consortium, Philadelphia*, 112, 2005.

[98] Mohamed Amir Yosef, Sandro Bauer, Johannes Hoffart, Marc Spaniol, and Gerhard Weikum. Hyena: Hierarchical type classification for entity names. In *Proceedings of COLING 2012: Posters*, pages 1361–1370, 2012.

[99] Joseph Z Chang, Richard Tzong-Han Tsai, and Jason S Chang. Wikisense: Supersense tagging of wikipedia named entities based wordnet. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*, pages 72–81, 2009.

[100] Luciano Del Corro, Abdalghani Abujabal, Rainer Gemulla, and Gerhard Weikum. Finet: Context-aware fine-grained named entity typing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 868–878, 2015.

[101] Shikhar Murty, Patrick Verga, Luke Vilnis, and Andrew McCallum. Finer grained entity typing with typenet. *arXiv preprint arXiv:1711.05795*, 2017.

[102] Besnik Fetahu, Sudipta Kar, Zhiyu Chen, Oleg Rokhlenko, and Shervin Malmasi. Semeval-2023 task 2: Fine-grained multilingual named entity recognition (multiconer 2). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2247–2265, 2023.

[103] Jun-Yu Ma, Jia-Chen Gu, Jiajun Qi, Zhenhua Ling, Quan Liu, and Xiaoyi Zhao. Ustc-nelslip at semeval-2023 task 2: Statistical construction and dual adaptation of gazetteer for multilingual complex ner. In *The 61st Annual Meeting Of The Association For Computational Linguistics*, 2023.

[104] Long Ma, Kai Lu, Tianbo Che, Hailong Huang, Weiguo Gao, and Xuan Li. Pai at semeval-2023 task 2: A universal system for named entity recognition with external entity information. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 744–750, 2023.

[105] Zeqi Tan, Shen Huang, Zixia Jia, Jiong Cai, Yinghui Li, Weiming Lu, Yueting Zhuang, Kewei Tu, Pengjun Xie, and Fei Huang. Damo-nlp at semeval-2023 task 2: A unified retrieval-augmented system for multilingual named entity recognition. In *The 61st Annual Meeting Of The Association For Computational Linguistics*, 2023.

[106] Iker García-Ferrero, Jon Ander Campos, Oscar Sainz, Ander Salaberria, and Dan Roth. Ixa/cogcomp at semeval-2023 task 2: Context-enriched multilingual named entity recognition using knowledge bases. In *The 61st Annual Meeting Of The Association For Computational Linguistics*, 2023.

[107] Xiang Ren, Wenqi He, Meng Qu, Lifu Huang, Heng Ji, and Jiawei Han. Afet: Automatic fine-grained entity typing by hierarchical partial-label embedding. In *EMNLP*, 2016.

[108] Abhishek Abhishek, Ashish Anand, and Amit Awekar. Fine-grained entity type classification by jointly learning representations and label embeddings. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 797–807, Valencia, Spain, April 2017. Association for Computational Linguistics.

[109] Sonse Shimaoka, Pontus Stenetorp, Kentaro Inui, and Sebastian Riedel. Neural architectures for fine-grained entity type classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1271–1280, 2017.

[110] Peng Xu and Denilson Barbosa. Neural fine-grained entity type classification with hierarchy-aware loss. In *The 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2018)*. ACL, June 2018.

[111] Weiran Pan, Wei Wei, and Feida Zhu. Automatic noisy label correction for fine-grained entity typing. In *Proceedings of the Thirty-first International Joint Conference on Artificial Intelligence, IJCAI-22*, 2022.

[112] Haoyu Zhang, Dingkun Long, Guangwei Xu, Muhua Zhu, Pengjun Xie, Fei Huang, and Ji Wang. Learning with noise: improving distantly-supervised fine-grained entity typing via automatic relabeling. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3808–3815, 2021.

[113] Qing Liu, Hongyu Lin, Xinyan Xiao, Xianpei Han, Le Sun, and Hua Wu. Fine-grained entity typing via label reasoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4611–4622, 2021.

[114] Hongliang Dai, Yangqiu Song, and Haixun Wang. Ultra-fine entity typing with weak supervision from a masked language model. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1790–1799, 2021.

[115] Yi Chen, Jiayang Cheng, Haiyun Jiang, Lemao Liu, Haisong Zhang, Shuming Shi, and Ruifeng Xu. Learning from sibling mentions with scalable graph inference in fine-grained entity typing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2076–2087, 2022.

[116] Abhinav Nagpal, Riddhiman Dasgupta, and Balaji Ganesan. Fine grained classification of personal data entities with language models. In *Proceedings of the 5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD)*, pages 130–134, 2022.

[117] Jiangshu Du, Wenpeng Yin, Congying Xia, and S Yu Philip. Learning to select from multiple options. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12754–12762, 2023.

[118] Andrew McCallum Greg Durrett Yasumasa Onoe, Michael Boratko. Modeling fine-grained entity types with box embeddings. In *ACL*, 2021.

[119] Weiran PAN, Wei WEI, and Feida ZHU. Automatic noisy label correction for fine-grained entity typing.(2022). In *Proceedings of the 31st International Joint Conference on Artificial Intelligence, Vienna, Austria*, pages 23–29, 2022.

[120] Bangzheng Li, Wenpeng Yin, and Muhao Chen. Ultra-fine entity typing with indirect supervision from natural language inference. *Transactions of the Association for Computational Linguistics*, 10:607–622, 2022.

[121] Alejandro Sierra-Múnera, Jan Westphal, and Ralf Krestel. Efficient ultrafine typing of named entities. In *2023 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 205–214. IEEE, 2023.

[122] Robert Lalramhluna, Sandeep Dash, and Dr Partha Pakray. Mizbert: a mizo bert model. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(7):1–14, 2024.

[123] Paerhati Tulajiang, Yuanyuan Sun, Yuanyu Zhang, Yingying Le, Kelaiti Xiao, and Hongfei Lin. A bilingual legal ner dataset and semantics-aware cross-lingual label transfer method for low-resource languages. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2025.

[124] Rodrigo del Moral-González, Helena Gómez-Adorno, and Orlando Ramos-Flores. Comparative analysis of generative llms for labeling entities in clinical notes. *Genomics & Informatics*, 23(1):1–8, 2025.

[125] Census of India. Statement 1: Abstract of Language Data. Census of India 2011, Language - 2011, 2011. Archived by the Internet Archive Wayback Machine; Accessed: 14 October 2025.

[126] MHA, GoI. The Eighth Schedule to the Constitution of India (List of Official Languages), May 2017. Accessed: 14 October 2025.

[127] Atlas UNESCO. Unesco atlas of the world's languages in danger, 2017.

[128] Isabelle Augenstein, Leon Derczynski, and Kalina Bontcheva. Generalisation in named entity recognition: A quantitative analysis. *Computer Speech & Language*, 44:61–83, 2017.

[129] Giusepppe Attardi. Wikiextractor. https://github.com/attardi/wikiextractor, 2015.

[130] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020.

[131] Besnik Fetahu, Sudipta Kar, Zhiyu Chen, Oleg Rokhlenko, and Shervin Malmasi. SemEval-2023 task 2: Fine-grained multilingual named entity recognition (MultiCoNER 2). In Atul Kr. Ojha, A. Seza Doğruöz, Giovanni Da San Martino, Harish Tayyar Madabushi, Ritesh Kumar, and Elisa Sartori, editors, *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2247–2265, Toronto, Canada, July 2023. Association for Computational Linguistics.

[132] Afshin Rahimi, Yuan Li, and Trevor Cohn. Massively multilingual transfer for ner. *arXiv preprint arXiv:1902.00193*, 2019.

[133] Moritz Laurer, Wouter Van Atteveldt, Andreu Casas, and Kasper Welbers. Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert-nli. *Political Analysis*, 32(1):84–100, 2024.

[134] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. brat: a web-based tool for NLP-assisted text annotation. In Frédérique Segond, editor, *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France, April 2012. Association for Computational Linguistics.

[135] Ying Lin and Heng Ji. An attentive fine-grained entity typing model with latent type representation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6197–6202, 2019.

[136] Ying Lin and Heng Ji. An attentive fine-grained entity typing model with latent type representation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

[137] Hongliang Dai, Donghong Du, Xin Li, and Yangqiu Song. Improving fine-grained entity typing with entity linking. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

[138] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. ERNIE: Enhanced language representation with informative entities. In *Proceedings of ACL 2019*, 2019.

[139] Tongfei Chen, Yunmo Chen, and Benjamin Van Durme. Hierarchical entity typing via multi-level learning to rank. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8465–8475, 2020.

[140] Muhammad Asif Ali, Yifang Sun, Bing Li, and Wei Wang. Fine-grained named entity typing over distantly supervised data based on refined representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7391–7398, 2020.

[141] Yinhan Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364, 2019.

[142] Zi-Yi Dou and Graham Neubig. Word alignment by fine-tuning embeddings on parallel corpora. In *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2021.

[143] Iker García-Ferrero, Rodrigo Agerri, and German Rigau. Model and data transfer for cross-lingual sequence labelling in zero-resource settings. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6403–6416, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[144] Oyesh Mann Singh, Ankur Padia, and Anupam Joshi. Named entity recognition for nepali language. In *2019 IEEE 5th international conference on collaboration and internet computing (cic)*, pages 184–190. IEEE, 2019.

[145] Jay Gala, Pranjal A Chitale, A K Raghavan, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar M, Janki Atul Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M Khapra, Raj Dabre, and Anoop Kunchukuttan. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*, 2023.

[146] Saiful Islam, Abhijit Paul, Bipul Shyam Purkayastha, and Ismail Hussain. Construction of english-bodo parallel text corpus for statistical machine translation. *International Journal on Natural Language Computing (IJNLC) Vol*, 7, 2018.

[147] Saiful Islam. English to bodo statistical machine translation system using multi-domain parallel corpora. In *Proceedings of the 15th International Conference on Natural Language Processing*, pages 75–81, 2018.

[148] Thangkhanhau Haulai and Jamal Hussain. Construction of mizo: English parallel corpus for machine translation. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(8):1–12, 2023.

[149] Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M Khapra, Anoop Kunchukuttan, and Pratyush Kumar. Towards leaving no indic language behind: Building monolingual corpora, benchmark and models for indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12402–12426, 2023.

[150] Louise Deleger, Qi Li, Todd Lingren, Megan Kaiser, Katalin Molnar, Laura Stoutenborough, Michal Kouril, Keith Marsolo, Imre Solti, et al. Building gold standard corpora for medical natural language processing tasks. In *AMIA Annual Symposium Proceedings*, volume 2012, page 144, 2012.

[151] Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual bert? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, 2019.

[152] Noëmi Aepli and Rico Sennrich. Improving zero-shot cross-lingual transfer between closely related languages by injecting character-level noise. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4074–4083, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[153] Vaidehi Patil, Partha Talukdar, and Sunita Sarawagi. Overlap-based vocabulary generation improves cross-lingual transfer among related languages. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 219–233, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[154] Verena Blaschke, Hinrich Schütze, and Barbara Plank. Does manipulating tokenization aid cross-lingual transfer? a study on POS tagging for non-standardized languages. In Yves Scherrer, Tommi Jauhiainen, Nikola Ljubešić, Preslav Nakov, Jörg Tiedemann, and Marcos Zampieri, editors, *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023)*, pages 40–54, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[155] Maharaj Brahma, Kaushal Maurya, and Maunendra Desarkar. Selectnoise: Unsupervised noise injection to enable zero-shot machine translation for extremely low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1615–1629, 2023.

[156] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

[157] Jonas Golde, Patrick Haller, Max Ploner, Fabio Barth, Nicolaas Jedema, and Alan Akbik. Familiarity: Better evaluation of zero-shot named entity recognition by quantifying label shifts in synthetic training data. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 820–834, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.

[158] Zhuojun Ding, Wei Wei, and Chenghao Fan. Selecting and merging: Towards adaptable and scalable named entity recognition with large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9869–9886, Vienna, Austria, July 2025. Association for Computational Linguistics.

[159] Raja Vavekanand, Bhagwan Das, and Teerath Kumar. Daugsindhi: a data augmentation approach for enhancing sindhi language text classification. *Discover Data*, 3(1), 2025.

[160] Google Translate. cloud.google.com/translate, 2025.

[161] Bing Translator. www.bing.com/translator/, 2025.

[162] IndicTrans2. github.com/AI4Bharat/IndicTrans2, 2023.

[163] Daniel Licht, Cynthia Gao, Janice Lam, Francisco Guzman, Mona Diab, and Philipp Koehn. Consistent human evaluation of machine translation across language pairs. In *Proceedings of the 15th biennial conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, September 2022.

[164] Maharaj Brahma, Kaushal Maurya, and Maunendra Desarkar. SelectNoise: Unsupervised noise injection to enable zero-shot machine translation for extremely low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, December 2023.

[165] Prachuryya Kaushik, Adittya Gupta, Ajanta Maurya, Gautam Sharma, V. V. Saradhi, and Ashish Anand. APTFiNER: Annotation preserving translation for fine-grained named entity recognition. In Stelios Piperidis, Núria Bel, Henk van den Heuvel, Nancy Ide, Simon Krek, and Antonio Toral, editors, *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, pages 7668–7680, Palma, Mallorca, Spain, May 2026. European Language Resources Association (ELRA).

Publications

Conference Publications

1. **Prachuryya Kaushik**, Shivansh Mishra, and Ashish Anand, *TAFSIL: Taxonomy Adaptable Fine-grained Entity Recognition through Distant Supervision for Indian Languages*, Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 3753–3763, 2025, Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3726302.3730341

2. **Prachuryya Kaushik** and Ashish Anand, *FiNERVINER: Fine-grained Named Entity Recognition for Vulnerable languages of India's North Eastern Region*, Proceedings of the Fifteenth biennial Language Resources and Evaluation Conference (LREC), 2026. https://lrec.elra.info/lrec2026-main-607

3. **Prachuryya Kaushik** and Ashish Anand, *CLASSER: Cross-lingual Annotation Projection enhancement through Script Similarity for Fine-grained Named Entity Recognition*, Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, 2025, Association for Computational Linguistics. https://aclanthology.org/2025.ijcnlp-long.94/

4. **Prachuryya Kaushik** and Ashish Anand, *SampurNER: Fine-grained Named Entity Recognition Dataset for 22 Indian Languages*, 2026, Proceedings of the AAAI Conference on Artificial Intelligence. https://ojs.aaai.org/index.php/AAAI/article/view/40405

Journal (submitted)

1. **Prachuryya Kaushik**, and Ashish Anand, *FiNE-MiBBiC: Fine-grained Named Entity Recognition for Mizo, Bhojpuri, Bishnupriya Manipuri and Chhattisgarhi*, The ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), 2025, Association for Computing Machinery, New York, NY, USA.

Conference Publication (beyond the thesis)

1. **Prachuryya Kaushik**, Adittya Gupta, Ajanta Maurya, Gautam Sharma, V. Vijaya Saradhi, and Ashish Anand, *APTFiNER: Annotation Preserving Translation for Fine-grained Named Entity Recognition*, Proceedings of the Fifteenth biennial Language Resources and Evaluation Conference (LREC), 2026. https://lrec.elra.info/lrec2026-main-608