Indic-MULAN: A Study of Fact Mutability in Language Models for Low-resource Indian languages

Anonymous ACL submission

Abstract

Large language models(LLMs) open up new possibilities for Low-resource languages(LRLs) by showing impressive results in various classification and generation tasks with zero/few-shot inference. Due to pre-training from large datasets, including web-based corpora, these LLMs have the knowledge of factual information, and factual knowledge can both be time-dependent and independent. Having said that, LLM's capability of extracting factual information from LLMs involving LRLs is an interesting task, though not much explored. In this work, we present Indic-MULAN, a benchmark dataset to evaluate LLM's capability to extract time-aware factual knowledge involving one-to-one and one-to-many relations. Our dataset comprises 34 relations and ∼30K queries covering 11 Indian languages. We experimented with two LLMs, GPT-4(proprietary) and Llama-3(open-source). We find performance is poor when queried with native languages but improves when translated to English. Then, we do a brief analysis of the embedding space using t-SNE plots, which leads to some interesting observations. We hope Indic-MULAN will help future studies of LLMs involving time-aware factual knowledge in Indian languages.

1 Introduction

Large language models like GPT (OpenAI et al., 2023) and Llama (Touvron et al., 2023) families are showing great promise for the advancement of NLP research in Low-resource languages(LRLs). As these models are pre-trained with large amounts of open-sourced worldwide web data, they have embedded factual knowledge (Petroni et al., 2019; Jiang et al., 2020; Liu et al., 2021b). Although LRL's performance is evaluated using LLMs for various classification and generation tasks (Ahuja et al., 2023, 2024), their effectiveness in extracting factual information has not been explored much. Things get more interesting when the factual knowledge has a time dimension attached to it. For example, the 'place of birth' of a person and the 'head of a country' both fall into the category of factual information, but when the latter depends upon time, the former is time-independent.

Recent studies (Dhingra et al., 2022; Jain et al., 2023; Qiu et al., 2023) show that LLMs find it difficult to provide correct responses for time-aware queries, and also below per reasoning capability using temporal information. In response to the previous observation, MULAN (Fierro et al., 2024) shows although the LLMs struggle with time-aware tasks, LLMs encode time-mutable information differently using various time-independent/dependent queries from Wikidata triplets(subject, relation, object). Unlike previous studies, their benchmark dataset for time mutability includes relations from both time-independent and dependent classes with balance. Following their footprint, we present Indic-MULAN, a dataset to evaluate LLM's time-contingency in 11 Indian languages that includes 34 Wikidata relations covering around ∼30K queries.

Using this resource, we explore the time-awareness capabilities of two recent LLMs, GPT-4(proprietary) and Llama-3(open-source) for 11 Indian languages. Our study shows poor performance across relation types and languages. To confirm if the performance drops are related to language understanding or the knowledge gap of the LLM, we perform the same experiment with corresponding English translation, which improves the performance across relation type and language, confirming that the poor performance in the Indian language mostly comes from a lack of language understanding of the LLMs. We then plot the t-SNE (van der Maaten and Hinton, 2008) using the query embedding of LLama-3 for both the English translation and the corresponding language script

and find the sentence cluster of different relation types is more separable in the English translation compared to the native script. This might be one of the reasons for performance improvement in English translation. Then, we compare the error cases in the t-SNE plot and study the overlapping relation type clusters to find relations involving the same target object type that fall closely in the clusters. Along with that, we show relations that are doing good and bad are more or less the same across Indian languages, which makes the problem more interesting.

2 Related Work

Due to pre-training with huge amounts of data, including the World Wide Web, LLMs have the ability to encode factual knowledge (Petroni et al., 2019; Jiang et al., 2020; Liu et al., 2021a). Different studies (Meng et al., 2022; Yin et al., 2022; Chalkidis et al., 2023) using cloze-test or next-token prediction have shown that retrieval of fact can be possible from LLMs. Although when the factual knowledge changes over time, studies (Dhingra et al., 2022; Jain et al., 2023; Qiu et al., 2023) showed the limits of LLMs to provide correct responses in such scenarios. Existing studies like LAMA (Petroni et al., 2019) and pLAMA (Dhingra et al., 2022) try to benchmark the time awareness ability of the language models using Immutable and Mutable relations. Whereas recent MULAN (Fierro et al., 2024), creates a more balanced benchmark and studies contemporary LLM's ability to extract factual knowledge for English. Our work primarily follows MULAN and extends the benchmark to 11 Indian languages.

3 Indic-MULAN

3.1 Construction

We created Indic-MULAN inspired by MULAN (Fierro et al., 2024), where they gathered 35 relations from 3 relation types(Immutable-1, Immutable-n, and Mutable). Immutable-1 and Immutable-n are time-independent relation types, but the only difference is that Immutable-1 is a one-to-one relation like 'father', while Immutable-n is a one-to-many relation like 'award received'. On the other hand, Mutable is a time-dependent relation type like 'head of government'. We started with MULAN's 35 relations and queried Wikidata using SPARQL to get the <subject, relation, object> triplets for each Indian languages. To make the data more practical, for each of the 11 Indian languages, we only selected the subject with an entry in Wikipedia of that particular language. Also, the object should have a label in that language. As we assume that all these LLMs are trained with a large amount of web data, including Wikipedia, there is a high chance that they have seen these triplets in different Indian language scripts while training. After this process, we are left with 34 relations (Immutable-1:11, Immutable-n:10, Mutable:13), as one of the relations does not satisfy our criteria. We collected around ∼30K triplets across 11 languages. Then, we used the MULAN English templates of each of the 34 relations and translated them into 11 Indian languages, and rectified the translated templates where necessary. The full details of the dataset are in Appendix Table 3 and 4.

3.2 Inference

To query the LLMs, we provide the task description(details in Appendix Figure 6) with the particular relation template to fill the [blank] spot. As an example, for a 'continent' relation in Bengali, the template can be "ভারত [blank] মহাদেশের অধীনে রয়েছে।"([India] belongs to the continent of [blank].), where India(ভারত) is the subject, and Asia is the object, which is our expected outcome for the [blank] space.

3.3 Models & metric

We use two LLMs for our analysis, GPT-4 and Llama-3 8b parameter model(hyperparameter details in Appendix Table 5). For evaluation, we use accuracy as a metric. Exact matches are sometimes not possible due to the inherent probabilistic nature of these LLMs, keep that in mind, we use the partial match above 80% as a correct answer.

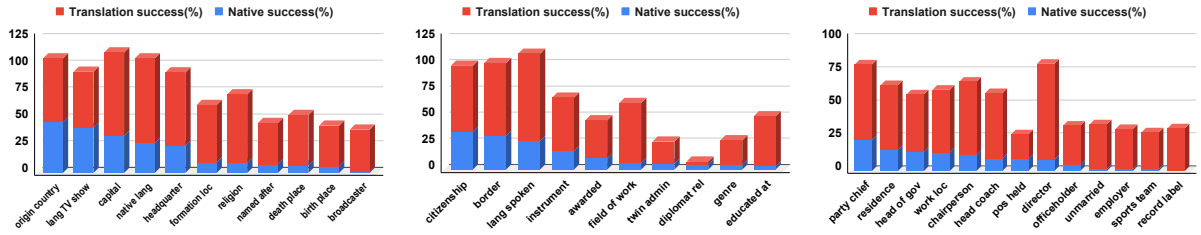
4 Results

4.1 Not all relation types perform similarly

In Table 1, we show the result of GPT-4 and Llama-3 performance for all the languages across relation types. For both the LLMs performance of the native language, means when we are passing the query in native language script, give poor results. Although the performance of the GPT-4 is slightly better than Llama-3, which is somewhat expected as the latter is a smaller model compared to GPT-4. Another interesting observation is performance is worse in the Mutable relation class throughout

		Immutable-1		Immutable-n		Mutable	
	Language	Native	Translation	Native	Translation	Native	Translation
GPT-4	as	27.27	61.82	34.78	58.70	24.21	43.16
	bn	34.55	68.18	27.55	53.06	19.51	33.33
	gu	39.22	75.49	33.00	53.00	26.58	44.30
	hi	36.36	72.73	32.65	54.08	19.67	37.70
	kn	32.73	73.64	28.28	46.46	33.64	45.79
	ml	30.00	69.09	18.37	46.94	21.05	49.12
	mr	28.18	68.18	20.21	51.06	28.07	30.70
	or	37.04	51.85	35.71	51.02	22.47	33.71
	pa	33.64	67.27	37.11	42.27	20.00	40.00
	ta	26.36	72.73	14.58	51.04	14.68	36.70
	te	44.55	75.45	18.09	41.49	25.00	33.65
	Avg.	33.63	68.77	27.30	49.92	23.17	38.92
Llama-3	as	19.93	55.62	24.61	61.34	8.73	37.55
	bn	14.84	53.96	12.23	48.92	7.60	33.37
	gu	18.25	55.27	29.42	51.73	11.27	38.31
	hi	17.85	63.16	16.99	47.37	11.29	42.84
	kn	27.20	57.07	18.88	53.95	11.97	45.30
	ml	24.52	61.80	7.27	43.34	4.90	44.28
	mr	14.82	61.45	10.62	53.94	12.72	41.35
	or	45.17	52.11	38.59	59.34	7.67	33.24
	pa	23.49	59.86	22.08	49.92	7.10	40.65
	ta	24.77	61.57	16.02	47.48	4.57	40.70
	te	32.87	58.66	17.77	47.65	17.47	37.68
	Avg.	23.97	58.23	19.50	51.36	9.57	39.57

Table 1: Performance of different relation types in native script and English translation across languages. The last column shows the average performance across languages.



(a) Immutable-1 (b) Immutable-n (c) Mutable

Figure 1: Relation-wise average success percentage of 3 relation types using native script and English translation across different Indian languages.

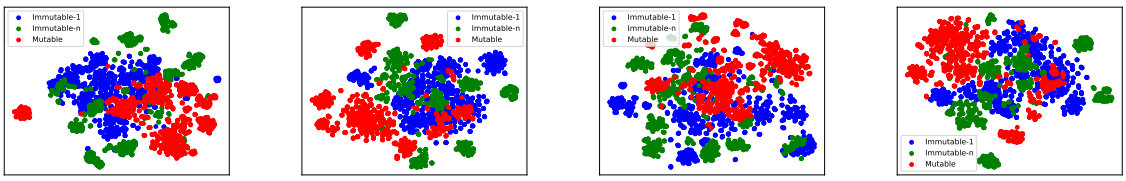
languages, improves a bit to Immutable-n, and further improvement can be seen in the Immutable-1 relation type. The observation aligns with the MULAN works, where they show similar observations for the English language. Immutable-1 relation types seem easier for the model to identify as it is a one-to-one relation and not changeable over time, whereas Immutanle-n relation also does not change over time, but it is a one-to-many relation, so it can confuse the model. A further drop in Mutable relation types may come from the time dependency.

4.2 Translation improves performance

We also experimented with English translation for each query to check if the model lacks knowledge irrespective of language or if the poor performance is due to a lack of language understanding in Indian languages. In Table 1, we show the English translation result corresponding to each native script result. Although the inherent pattern remains the same(performance of Mutable<Immutable-n<Immutable-1), individual performance improves after translation. For GPT-4, the improvement is from 68% to as high as 104%, and for Llama-3, the improvement ranges between 143% to 313%. From this observation, it's evident that LLMs perform poorly due to language understanding in Indian languages.

4.3 Embedding space analysis

To investigate the performance improvement in English translation further, we use the Llama-3 model to encode each query(both in the native script and English translation) coming from different relation types and use t-SNE to plot them. In Figure 2, we show the t-SNE plots of the native script and English translation side by side for Hindi(Indo-Aryan family) and Malayalam(Dravidian family)(all language results in Appendix Figure 4). From the plots, it's evident that the sentence clusters of different relation types overlap more in the native script than in the English translation, where the clusters are more separable. This can be one of

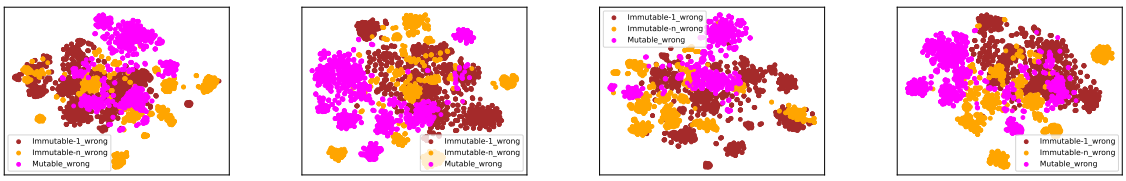


(a) Hindi (b) Hindi(Translation) (c) Malayalam (d) Malayalam(Translation)

Figure 2: t-SNE plots of Llama-3 sentence embeddings for 3 relation types across different Indian languages in native script and English translation.

Target object	Relation	Relation type	Example sentence
Location	work location	Mutable	**Hermann Hesse** took up work located in [Blank].
	place of death	Immutable-1	**Emil Abderhalden** passed away in the location of [Blank].
	residence	Mutable	**Jim Parsons** takes up residence in [Blank].
	headquarters location	Immutable-1	**Cloudflare** has its headquarters in [Blank].
Language	native language	Immutable-1	The primary language of **Paul Dirac** is [Blank].
	original language of film or TV show	Immutable-1	The original language of work of **Frost** is [Blank].
	languages spoken, written or signed	Immutable-n	The language that **William John Macquorn Rankine** would normally communicate in is [Blank].

Table 2: Error instances: Different relation types for a target object class falling in close-proximity in t-SNE plot. Subject of the sentences are **Bold**.



(a) Hindi (b) Hindi(Translation) (c) Malayalam (d) Malayalam(Translation)

Figure 3: t-SNE plots of Llama-3 sentence embeddings for error cases in 3 relation types across different Indian languages in native script and English translation.

the reasons behind the performance improvement in English translation.

4.4 Error case analysis

Next, we investigate the error cases using the t-SNE plots derived from Llama-3 sentence embedding. In Figure 3, we plot the t-SNE of the native script and English translation side by side for Hindi and Malayalam(all language results in Appendix Figure 5). Besides the relation type clusters being more separable in English translation, we find that the Mutable relation type intersects more with Immutable-1 in English translation. While investigating the overlapped clusters, we find some interesting patterns, like all the relations involving a particular target object class, are in close proximity irrespective of their relation types. As shown in Table 2, for a target object type 'Location', relations like 'work location'(Mutable), 'place of death'(Immutable-1), 'residence'(Mutable), and 'headquarters location'(Immutable-1), falls close in the t-SNE plot. As the target object type is the same, models can get confused due to similar wordings in the query, although the relations are different.

5 Conclusion

We develop Indic-MULAN, a factual knowledge-checking dataset for 11 Indian languages covering 34 relations and ∼30k queries. Experimenting with the two recent LLMs, GPT-4 and Llama-3, we see that LLMs perform poorly for all Indian languages, but their performance improves when using English translation. Then, after analysing the sentence embeddings using t-SNE plots, we found that the clusters of different relation types are more separable in English translation than in native script. Also, by studying the embeddings of error cases, we find relations with the same target object classes in close proximity, irrespective of their relation types. Through this effort, we aim to initiate future fact-checking research on LLMs for Indian languages and focus on improving their performance in this task.

6 Limitations

Although we collected triplets for 34 relations, some of the relations may not be well suited for Indian languages due to social and cultural boundaries. Relations that are more culturally related to Indian languages may give a better understanding of the factual knowledge. To make the benchmark more reliable, we need to add more query templates designed by native speakers for particular languages. Some of the triplets that are extracted for mutable relation types may be updated in Wikidata after the LLM's release data, and it can underestimate the LLM's performance. We will consider all these points to improve the benchmark in future versions.

References

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Maxamed Axmed, Kalika Bali, and Sunayana Sitaram. 2023. Mega: Multilingual evaluation of generative ai.

Sanchit Ahuja, Divyanshu Aggarwal, Varun Gumma, Ishaan Watts, Ashutosh Sathe, Millicent Ochieng, Rishav Hada, Prachi Jain, Maxamed Axmed, Kalika Bali, and Sunayana Sitaram. 2024. Megaverse: Benchmarking large language models across languages, modalities, models and tasks.

Ilias Chalkidis, Nicolas Garneau, Catalina Goanta, Daniel Katz, and Anders Søgaard. 2023. LeXFiles and LegalLAMA: Facilitating English multinational legal language model development. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15513–15535, Toronto, Canada. Association for Computational Linguistics.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Constanza Fierro, Nicolas Garneau, Emanuele Bugliarello, Yova Kementchedjhieva, and Anders Søgaard. 2024. Mulan: A study of fact mutability in language models.

Raghav Jain, Daivik Sojitra, Arkadeep Acharya, Sriparna Saha, Adam Jatowt, and Sandipan Dandapat. 2023. Do language models have a common sense regarding time? revisiting temporal commonsense reasoning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6750–6774, Singapore. Association for Computational Linguistics.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing.

Xin Liu, Baosong Yang, Dayiheng Liu, Haibo Zhang, Weihua Luo, Min Zhang, Haiying Zhang, and Jinsong Su. 2021b. Bridging subword gaps in pretrain-finetune paradigm for natural language generation. 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 6001–6011, Online. Association for Computational Linguistics.

Zaiqiao Meng, Fangyu Liu, Ehsan Shareghi, Yixuan Su, Charlotte Collins, and Nigel Collier. 2022. Rewire-then-probe: A contrastive recipe for probing biomedical knowledge of pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4798–4810, Dublin, Ireland. Association for Computational Linguistics.

OpenAI et al. 2023. Gpt-4 technical report.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo M. Ponti, and Shay B. Cohen. 2023. Are large language models temporally grounded?

Hugo Touvron et al. 2023. LLaMA 2: Open foundation and fine-tuned chat models.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Da Yin, Hritik Bansal, Masoud Monajatipoor, Liunian Harold Li, and Kai-Wei Chang. 2022. GeoMLAMA: Geo-diverse commonsense probing on multilingual pre-trained language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2039–2055, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Indic-MULAN: A Study of Fact Mutability in Language Models for Low-resource Indian languages
(Appendix)

A Supplementary results

	Immutable-1	Immutable-n	Mutable
as	1114	386	229
bn	4239	695	987
gu	1337	520	355
hi	1692	665	691
kn	1500	519	351
ml	1513	743	551
mr	1471	584	503
or	1304	482	352
pa	1328	643	465
ta	1288	674	656
te	1506	574	475

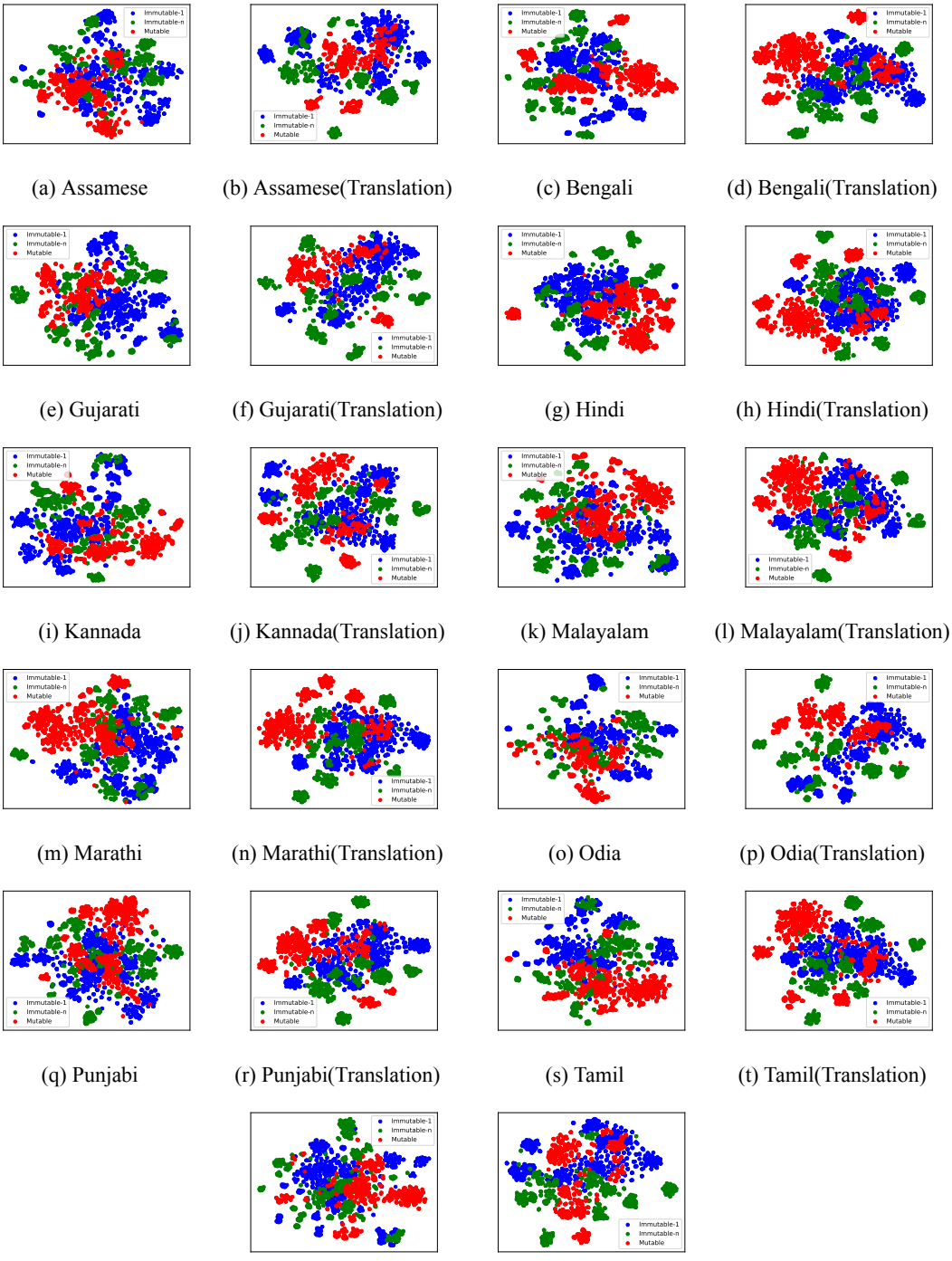
Table 3: Triplets across relation types for 11 Indian languages in Indic-MULAN.

	as	bn	gu	hi	kn	ml	mr	or	pa	ta	te
capital	74	461	176	192	150	182	146	107	102	47	195
country of origin	172	440	175	168	186	168	186	192	175	152	183
headquarters location	133	398	140	153	155	142	156	122	127	151	162
location of formation	27	196	74	199	164	193	50	24	49	48	50
named after	44	257	74	81	70	87	77	50	75	92	91
native language	160	200	160	177	166	177	173	192	189	181	174
original broadcaster	10	285	2	140	12	14	119	8	87	29	19
original language of film or TV show	186	184	69	188	196	183	179	189	178	174	180
place of birth	75	700	160	126	135	106	121	154	113	138	150
place of death	91	753	168	123	144	125	128	135	114	129	152
religion or worldview	142	365	139	145	122	136	136	131	119	147	150
award received	5	8	18	8	9	8	4	14	7	6	4
country of citizenship	109	152	123	143	128	138	124	121	137	147	125
diplomatic relation	20	60	65	61	62	58	20	56	61	61	62
educated at	11	33	47	32	14	35	27	37	37	39	31
field of work	49	88	42	76	70	69	65	41	65	69	68
genre	24	50	24	51	43	34	38	35	27	48	58
instrument	17	49	23	40	48	129	44	16	45	39	38
languages spoken, written or signed	107	124	123	126	100	134	123	119	123	128	109
shares border with	37	22	42	23	33	32	27	35	28	30	52
twinned administrative body	7	109	13	105	12	106	112	8	113	107	27
chairperson	13	94	9	79	19	73	49	10	53	70	22
director / manager	1	37	60	10	9	28	17	4	8	31	6
employer	25	70	1	61	27	61	51	53	60	69	68
head coach	2	63	8	26	6	5	35	3	2	6	4
head of government	14	53	35	49	13	37	42	18	31	50	11
member of sports team	26	23	8	33	30	28	22	50	31	54	45
officeholder	9	82	1	114	13	30	36	5	28	56	19
party chief representative	1	3	46	2	1	2	3	55	2	2	1
position held	13	38	1	30	32	31	26	4	31	43	41
record label	2	114	103	23	1	7	1	85	8	1	3
residence	61	189	1	82	88	77	83	3	67	90	125
unmarried partner	18	127	82	86	31	98	51	62	62	80	14
work location	44	94	151	96	81	74	87	94	82	104	116

Table 4: Number of triplets for each relation across 11 Indian languages in Indic-MULAN.

B Experimental settings

We use GPT-4-0613 model, which costs $0.03 / 1K tokens for input and $0.06 / 1K tokens for output. We run all the experiments using GPT-4 in a 16GB RAM CPU-based system without any GPU usage. For



(a) Assamese (b) Assamese(Translation) (c) Bengali (d) Bengali(Translation)

(e) Gujarati (f) Gujarati(Translation) (g) Hindi (h) Hindi(Translation)

(i) Kannada (j) Kannada(Translation) (k) Malayalam (l) Malayalam(Translation)

(m) Marathi (n) Marathi(Translation) (o) Odia (p) Odia(Translation)

(q) Punjabi (r) Punjabi(Translation) (s) Tamil (t) Tamil(Translation)

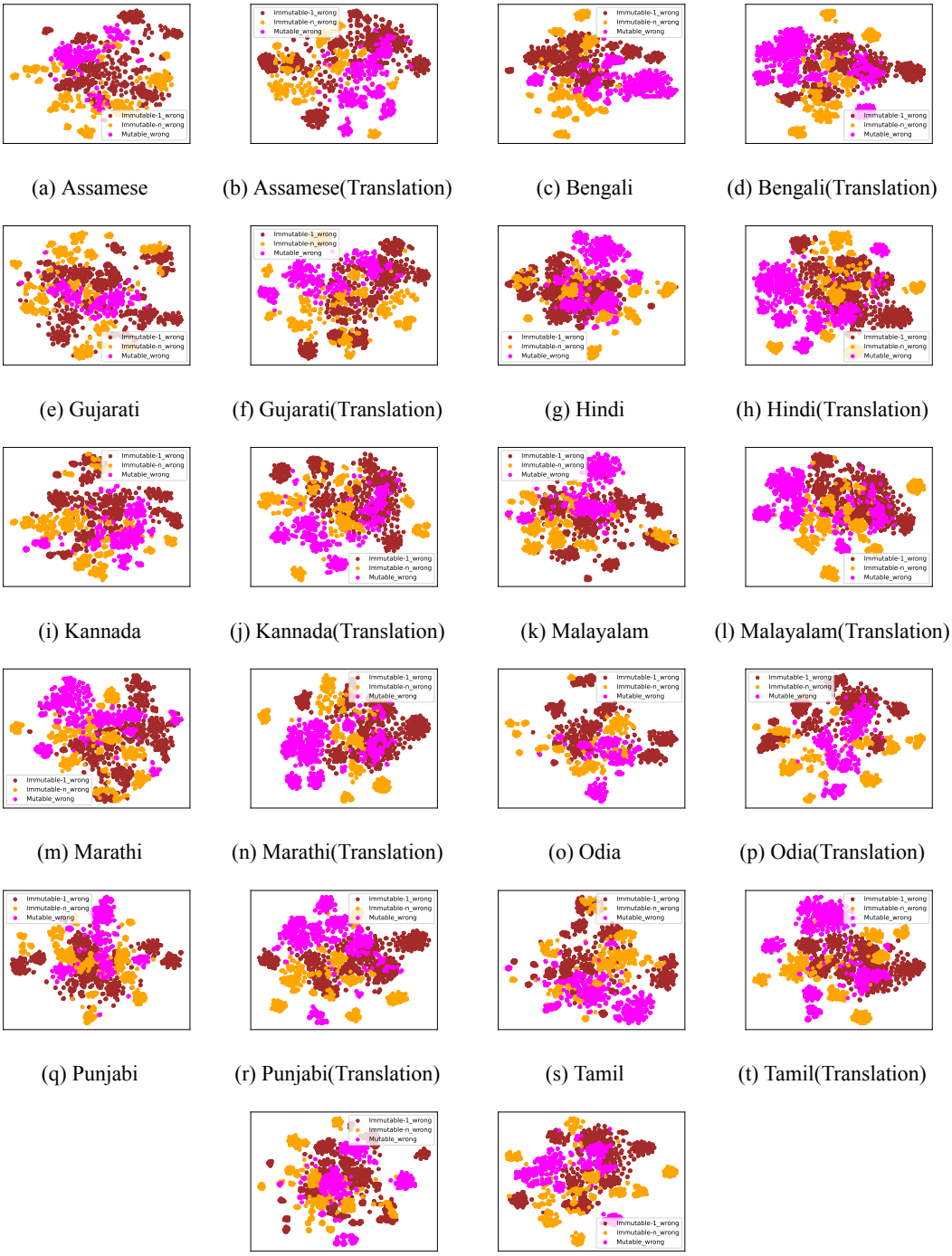
(u) Telugu (v) Telugu(Translation)

Figure 4: t-SNE plots of Llama-3 sentence embeddings for 3 relation types across different Indian languages in native script and English translation.

Llama-3, we downloaded its 8B parameter variant and used the Transformers library from Huggingface to load it into a machine having an NVIDIA RTX A6000(48GB) GPU. It used ∼32GB of GPU memory while inference. To preserve cost, we do all the experiments one time, and to make them reproducible, we fix the seed value to 42 and set the temperature close to zero of the GPT-4 API.

Hyperparameter	Value
LLM	GPT-4-0613, Llama-3-8b
temperature	0.01
max token	64
GPT4 Seed	42

Table 5: Details of GPT-4 and Llama-3 hyperparameters.



(a) Assamese (b) Assamese(Translation) (c) Bengali (d) Bengali(Translation)

(e) Gujarati (f) Gujarati(Translation) (g) Hindi (h) Hindi(Translation)

(i) Kannada (j) Kannada(Translation) (k) Malayalam (l) Malayalam(Translation)

(m) Marathi (n) Marathi(Translation) (o) Odia (p) Odia(Translation)

(q) Punjabi (r) Punjabi(Translation) (s) Tamil (t) Tamil(Translation)

(u) Telugu (v) Telugu(Translation)

Figure 5: t-SNE plots of Llama-3 sentence embeddings for error cases in 3 relation types across different Indian languages in native script and English translation.

Native Script

Prompt description: Output the correct answer which can fill up the [Blank] tag in the sentence in Hindi. Output only the answer, if the answer is not known, output 'Not known'.
Sentence: पोलैंड की राजधानी शहर [Blank] है।
Answer:

Translated instance

Prompt description: Output the correct answer which can fill up the [Blank] tag in the sentence in English. Output only the answer, if the answer is not known, output 'Not known'.
Sentence: The capital city of Poland is [Blank].
Answer:

Figure 6: Prompts used to query LLMs.