SIGTYP 2026

The 8th Workshop on Research in Computational Linguistic Typology and Multilingual NLP

Proceedings of the Workshop

March 29, 2026

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-374-6

Introduction

We are very pleased to welcome you to the 8th edition of the workshop for typology-related research and its integration into multilingual Natural Language Processing (SIGTYP 2026). The workshop is co-located with the 19th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2026), which takes place in Rabat, Morocco.

Encouraged by the 2019 – 2025 workshops, the aim of the eigth edition of SIGTYP workshop is to act as a platform and a forum for the exchange of information between typology-related research, multilingual NLP, and other research areas that can lead to the development of truly multilingual NLP methods. The workshop is specifically aimed at raising awareness of linguistic typology and its potential in supporting and widening the global reach of multilingual NLP, as well as at introducing computational approaches to linguistic typology. It fosters research and discussion on open problems, not only within the active community working on cross- and multilingual NLP but also inviting input from leading researchers in linguistic typology.

The workshop provides focused discussions on a range of topics, including the following:

1. Integration of typological features in language transfer and joint multilingual learning. In addition to established techniques such as "selective sharing", are there alternative ways to encoding heterogeneous external knowledge in machine learning algorithms?

2. Development of unified taxonomy and resources. Building universal databases and models to facilitate understanding and processing of diverse languages.

3. Automatic inference of typological features. The pros and cons of existing techniques (e.g. heuristics derived from morphosyntactic annotation, propagation from features of other languages, supervised Bayesian and neural models) and discussion on emerging ones.

4. Typology and interpretability. The use of typological knowledge for interpretation of hidden representations of multilingual neural models, multilingual data generation and selection, and typological annotation of texts.

5. Improvement and completion of typological databases. Combining linguistic knowledge and automatic data-driven methods towards the joint goal of improving the knowledge on cross-linguistic variation and universals.

6. Linguistic diversity and universals. Challenges of cross-lingual annotation. Which linguistic phenomena or categories should be considered universal? How should they be annotated?

In addition, this year we have introduced a novel theme: using LLMs for typological studies. Can LLMs be utilised to formulate or prove typological hypotheses? Are they capable of making useful cross-linguistic generalisations?

The final program of SIGTYP contains 2 keynote talks, 6 archival papers, 2 EACL Findings and 2 extended abstracts, intertwined with linguistic trivia elements. This workshop would not have been possible without the contribution of its program committee, to whom we would like to express our gratitude! We also thank Terry Regier and Jennifer Culbertson for kindly accepting our invitation to be keynote speakers. Finally, we are also very thankful to Vilém Zouhar who offered his help in running the in-person part of the workshop.

Wishing you a wonderful time at SIGTYP 2026!

Ekaterina Vylomova, Priya Rani, Andrei Shcherbakov on behalf of the SIGTYP 2026 Program Co-Chairs

Organizing Committee

Program Chairs

Priya Rani, University of Galway
Michael Hahn, Saarland University
Alexey Sorokin, Lomonosov Moscow State University
Vilém Zouhar, ETH Zurich
Oleg Serikov, King Abdullah University of Science and Technology
Andrei Shcherbakov, CTI
Ryan Cotterell, ETH Zurich
Ekaterina Vylomova, The University of Melbourne

Local Chair

Vilém Zouhar, ETH Zurich

Publication Chairs

Ekaterina Vylomova, The University of Melbourne
Andrei Shcherbakov, CTI

SIGTYP Officers

President: Ekaterina Vylomova, The University of Melbourne
Secretary: Ryan Cotterell, ETH Zurich

Program Committee

Program Committee

Giuseppe G. A. Celano, Universität Leipzig
Elisabetta Jezek, University of Pavia
Esther Ploeger, Aalborg University
Leonie Weissweiler, Uppsala University
Barend Beekhuizen, University of Toronto
Jannic Alexander Cutura, DSTI School of Engineering
Ioana-Madalina Silai, Paris Nanterre University
Priya Rani, University of Galway
Michael Hahn, Saarland University
Alexey Sorokin, Lomonosov Moscow State University
Andrei Shcherbakov, CTI
Ekaterina Vylomova, The University of Melbourne

Invited Speakers

Terry Regier, University of California, Berkeley
Jennifer Culbertson, University of Edinburgh

Keynote Talk

A Keynote talk by Terry Regier

Terry Regier
University of California, Berkeley

2026-03-29 16:00:00 – Room: **SALLE LE LIXUS**

Bio: Terry Regier is a cognitive scientist and linguist whose research investigates language, meaning, and cognition. He is best known for his work exploring word meanings across languages, examining how word meanings reflect and sometimes shape thought and perception. Regier's lab integrates computational modeling, cross-linguistic data, and behavioral experiments to study universals and variation in semantic domains such as color, kinship, number, and spatial relations. His research contributes to understanding how languages evolve, and how language and cognition are influenced by cultural diversity.

Keynote Talk

A Keynote talk by Jennifer Culbertson

Jennifer Culbertson
University of Edinburgh

2026-03-29 09:30:00 – Room: **SALLE LE LIXUS**

Bio: Jennifer Culbertson is Professor in the Department of Linguistics and English Language at the University of Edinburgh. She is a founding member of the Centre for Language Evolution, with her research focusing on how typological universals are shaped by properties of human cognition. She is best known for her work investigating universals of word order and morphological categories using the experimental method of Artificial Language Learning.

Table of Contents

Program

Sunday, March 29, 2026

09:00 - 09:10 *Opening Remarks*

09:10 - 09:30 *Session 1*

Automatic Grammatical Case Prediction for Template Filling in Case-Marking Languages: Implementation and Evaluation for Finnish
Johannes Laurmaa

09:30 - 10:30 *Keynote talk by Jennifer Culbertson*

10:30 - 11:00 *Coffee/Tea Break*

11:00 - 11:50 *Session 2*

Beyond Multilinguality: Typological Limitations in Multilingual Models for Meitei Language
Badal Nyalang

A Valency Lexicon for Universal Dependencies
Petr Kocharov and Lilit Kharatyan

Modality Matching Matters: Calibrating Language Distances for Cross-Lingual Transfer in URIEL+
York Hay Ng, Aditya Khan, Xiang Lu, Matteo Salloum, Michael Zhou, Phuong H. Hoang, Seza Dogruoz and En-Shiun Annie Lee

11:50 - 12:30 *Session 3: Findings*

ParsTranslit: Truly Versatile Tajik-Farsi Transliteration
Rayyan Merchant and Kevin Tang

SrcMix: Mixing of Related Source Languages Benefits Extremely Low-resource Machine Translation
Sanjeev Kumar, Preethi Jyothi and Pushpak Bhattacharyya

12:30 - 14:00 *Lunch*

Sunday, March 29, 2026 (continued)

14:00 - 14:30	*Linguistic Trivia*
14:30 - 15:30	*Session 4*
	Evaluating the Interplay of Information Status and Information Content in a Multilingual Parallel Corpus Julius Steuer, Toshiki Nakai, Andrew Thomas Dyer, Luigi Talamo and Annemarie Verkerk
	What does Surprisal have to do with Information Status? Andrew Thomas Dyer
	A RAG Approach for Typological Database Completion Jonathan Hus and Antonios Anastasopoulos
	Translation as Efficient Meaning Compression: An Information Bottleneck Analysis of Spatial Terms Antoine Taroni, Ludovic Moncla and Frederique Laforest
15:30 - 16:00	*Coffee/Tea Break*
16:00 - 17:00	*Keynote talk by Terry Regier*
17:00 - 17:15	*Closing*

Automatic Grammatical Case Prediction for Template Filling in Case-Marking Languages: Implementation and Evaluation for Finnish

Johannes Laurmaa
johlaurmaa@gmail.com

Abstract

Automatically generating grammatically correct sentences in case-marking languages is hard because nominal case inflection depends on context. In template-based generation, placeholders must be inflected to the *right case* before insertion, otherwise the result is ungrammatical. We formalise this *case selection* problem for template slots and present a practical, data-driven solution designed for morphologically rich, case-marking languages, and apply it to Finnish. We automatically derive training instances from raw text via morphological analysis, and fine-tune transformer encoders to predict a distribution over 14 grammatical cases, with and without lemma conditioning. The predicted case is then realized by a morphological generator at deployment. On a held-out test set in the lemma-conditioned setting, our model attains 89.1% precision, 81.1% recall, and 84.2% F1, with recall@3 of 93.3% (macro averages). The probability outputs support abstention and top-k suggestion User Interfaces, enabling robust, lightweight template filling for production use in multiple domains, such as customer messaging. The pipeline assumes only access to raw text plus a morphological analyzer and generator, and can be applied to other languages with productive case systems.

1 Introduction

Generating natural-sounding text in highly inflected languages poses challenges that are often overlooked in languages like English. For example, Finnish relies on a rich system of grammatical cases, so that a word like `Helsinki` takes different endings depending on its role in the sentence. In English, an email template such as

`Your trip to [CITY] is starting`

allows any city name to be dropped in directly—no changes needed. But in Finnish, you can't simply insert `Helsinki` into the same sentence structure, because the word ending will depend on the grammatical case. The correct form is

`Matkasi Helsinkiin alkaa`

, where `Helsinki` has been inflected to `Helsinkiin`, the illative case, to indicate movement towards the city.

While it is technically possible for template authors to hand-select the case for each placeholder (slow and error-prone), in practice templates are often rewritten to keep the placeholder nominative—at the cost of fluency and ongoing content maintenance (e.g. reworded as `Matkasi kohteeseen Helsinki alkaa`[1], which would be translated as `Your trip to the destination Helsinki is starting`).

This is a simple example, but the problem is widespread in template-based text generation, which is common in applications ranging from automatic email messages to travel apps and customer notifications.

These workarounds are inefficient and underscore the need for automated solutions for inflected languages that can select the correct word form in context.

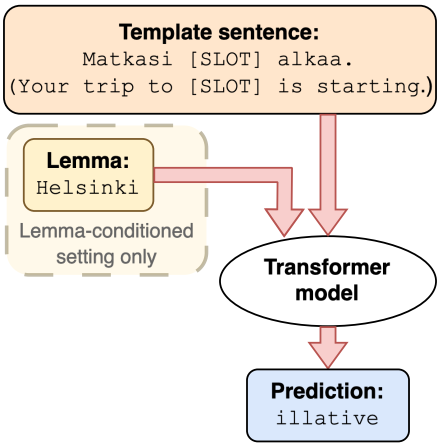


Figure 1: Our approach of grammatical case prediction.

[1] Real-world example of an email

Proceedings of the 8th Workshop on Research in Computational Linguistic Typology and Multilingual NLP, pages 1–11
March 29, 2026 ©2026 Association for Computational Linguistics

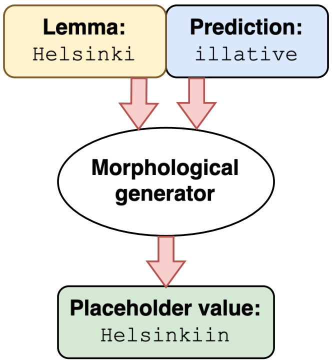


Figure 2: Surface realisation left to the morphological generator.

In this paper, we present a machine learning approach for predicting the required grammatical case when filling placeholders in Finnish sentence templates. Our approach creates training data from any Finnish text and uses transformer-based models to predict the correct grammatical case for incomplete template sentences (cf. Figure 1). We publish a high-performing model for this task and discuss both the challenges and the remaining ambiguities involved. Our goal is a deployable solution: predict the correct case with confidence scores, expose top-k when needed, and offload surface realisation to a morphological generator (cf. Figure 2).

Contributions.

- Formalize Finnish *case selection* with slot-only and lemma-conditioned settings.
- Automatic *dataset construction* from raw Finnish text via morphological analysis.
- Trained transformer encoder model predicting grammatical case probabilities.

2 Related work

2.1 Template-based NLG and morphology

Template-based Natural Language Generation (NLG) typically relies on sentence skeletons with slots filled at runtime (Van Deemter et al., 2005). For morphologically rich languages, template systems need to offload morphology to separate components that realise surface forms given a lemma and the sentence context. Dušek and Jurcícek (2019) already suggest approaches to this problem: using language models to select correct inflected forms, and using sequence-to-sequence models generating sequences of lemmas and morphological tags before passing them to a morphological generator. We adapt a similar approach to the Finnish language, but using language models to select correct grammatical cases and leaving the surface realisation to a morphological generator, also touching on the concept of aleatoric uncertainty and comparing prediction accuracy with and without lemma conditioning.

2.2 Finnish morphology: analysis and generation

Two-level finite-state morphology and HFST-based tools provide one established way to model Finnish morphology (Hämäläinen and Alnajjar, 2021). Omorfi provides a finite-state lexicon and analyzer for Finnish, supporting both morphological analysis and generation (Pirinen, 2015). UralicNLP exposes these analyzers and generators via a Python API, returning lemmas and rich feature bundles (including case) and generating inflected forms from lemma+features (Hämäläinen, 2019). Morphological generators solve the *realisation* problem: given lemma and features, output the correct surface form. They do not decide which case to use in a particular sentence, which is the task we address in this paper.

2.3 Finnish transformer models and LLMs

Virtanen et al. (2019) introduced FinBERT, a BERT-style model trained on large Finnish corpora, which outperforms multilingual BERT on core Finnish NLP tasks. We build directly on this line by fine-tuning FinBERT as a case classifier and comparing it to a multilingual model (XLM-R).

2.4 Positioning

Our setup sits between morphological tagging and morphological generation. In tagging, the model sees fully inflected tokens and assigns case labels to them in context. Here, the surface form is absent: we see a sentence with a marked slot, optionally with a lemma, and predict the case that should be used for that slot.

Finnish morphological generators, in turn, take lemma+features (including case) as input and realise the surface form. Our method is intended as an upstream component: it decides which case to request from the generator for a given slot in context.

To our knowledge, automatic case selection for template slots has not been studied directly. This work applies the idea to Finnish and integrates it with existing morphology tools in a template-filling setting.

Case	Example	Rough meaning
Core cases		
Nominative	talo	house
Genitive	talon	of the house
Partitive	taloa	(some) house
Accusative	talo(n)	the house (object)
Internal locative cases		
Inessive	talossa	in the house
Elative	talosta	out of the house
Illative	taloon	into the house
External locative cases		
Adessive	talolla	at the house
Ablative	talolta	from the house
Allative	talolle	to the house
Other cases		
Essive	talona	as a house
Translative	taloksi	into a house
Abessive	talotta	without a house
Instructive	taloin	by means of houses
Comitative	taloineen	with houses

Table 1: Finnish grammatical cases with the word talo (house) as an example.

3 Methodology

3.1 Grammatical cases

It is generally agreed that the Finnish language has 15 grammatical cases. Table 1 shows the grammatical cases in Finnish with the singular form of the word talo (house) used as an example.

3.2 Problem formulation

We cast Finnish template filling as a *case selection* task. The input is a sentence template s containing an annotated *slot* for a head noun (a noun in its baseform). The model outputs a categorical distribution over grammatical cases for that slot. In training, we expose both renderings (slot-only and lemma-conditioned) so the model learns to operate under either input condition.

Scope (what we predict). We predict a case label $y \in \mathcal{Y}$ for the head noun. The label set consists of 14 cases:

```
Nom, Gen, Par, Ine, Ela,
Ill, Ade, Abl, All, Ess,
Tra, Abe, Ins, Com
```

We left out the accusative class, as in contemporary Finnish, object marking is often realized as nominative or genitive. Surface-form realisation (inflecting a lemma into the final form of the word) is delegated to a morphological generator at deployment time. We also do not predict number, possessive suffixes or clitics, as these can also be handled by the morphological generator.

Supervision (single label). Training and test instances carry a *single* case $y^\star$ derived from morphological analysis of observed text. Although multiple cases can be pragmatically plausible for the same context (aleatoric uncertainty), supervision remains single-label.

Inference settings. We consider two settings that differ in what is provided about the slot content:

3.2.1 Slot-only (no-lemma) setting

Only the template with a slot marker is given; the model must infer the case from context alone.

> **Input:** Haluatko muuttaa [SLOT]?
> **Translation:** *Do you want to move [SLOT]?*
> **Label:** Ill

3.2.2 Lemma-conditioned setting

The lemma ℓ of the noun to be inserted (e.g., a city name) is also provided, which can inform case preferences.

> **Input:** Haluatko muuttaa [SLOT: Helsinki]?
> **Translation:** *Do you want to move [SLOT: Helsinki]?*
> **Label:** Ill

Output and evaluation. The model estimates $p(y \mid s)$ or $p(y \mid s, \ell)$. We report top-1 accuracy (via $\arg\max_y p(y \mid \cdot)$) and top-$k$ accuracy. For applications, systems may surface top-k candidates or abstain under low confidence.

3.3 Word inflection

Once the correct case has been predicted, the word can be inflected to the selected case using morphological generation tools (Hämäläinen, 2019; Alnajjar and Hämäläinen, 2023), cf. Figure 2.

3.4 Inherent uncertainty

Note that the setups above do not always admit a unique solution, as the correct grammatical case will depend on the intent of the writer. For the example shown above, there are other possible solutions besides the illative case:

Label: `Ill`
Output: Haluatko muuttaa Helsinkiin?
Translation: *Do you want to move to Helsinki?*

Label: `Ela`
Output: Haluatko muuttaa Helsingistä?
Translation: *Do you want to move out of Helsinki?*

Label: `Par`
Output: Haluatko muuttaa Helsinkiä?
Translation: *Do you want to change Helsinki?*

The multiplicity of solutions must be taken into account in application design. The model must be able to predict multiple classes with varying probability. This is a probabilistic single-label classification task with aleatoric uncertainty, so the expected optimal output is a probability distribution over classes rather than a single hard prediction.

While it cannot fully clear the uncertainty, the lemma-conditioned formulation will help alleviate it by bringing extra information about the user's intent. In the example above, knowing that the placeholder will contain a city name makes locative cases (elative / illative) more likely.

3.5 Dataset creation

We outline a simple, corpus-agnostic recipe to build supervision for case selection:

1. **Collect Finnish text.** Any raw Finnish text from the target domain is suitable.

2. **Run morphological analysis.** Using a Finnish morphological analyser, obtain for each token, its lemma (baseform) and grammatical case (when applicable), along with sentence boundaries.

3. **Construct instances.** For randomly selected target nouns in each sentence, create inputs in two views:

    - *Slot-only*: replace the surface token with a special marker (e.g., `Helsinkiin` replaced by `[MASK]`).
    - *Lemma-conditioned*: replace the surface token with its lemma (e.g., `Helsinkiin` replaced by `Helsinki`).

    The target label is the noun's grammatical case extracted from the morphological analysis; supervision is single-label.

Concrete choices (corpus, splits, and any training specifics) are detailed in Section 4.1.

3.6 Model selection

We considered:

- The problem highly depends on the context, relations among words and writer intent.
- The task exhibits aleatoric ambiguity: even with full context, multiple labels may be plausible. The model should therefore output a categorical distribution $p(y|x)$ rather than a hard label, and expose top-k classes with their probabilities.
- We aim for a lightweight model that can be easily deployed in production.

Given these specifications, we opted to use a Transformer encoder trained on Finnish or multilingual data (e.g., FinBERT or XLM-R) fine-tuned for K-way single-label classification with a softmax head. Top-k probabilities are returned at inference, with an optional post-hoc probability calibration.

4 Experiments

4.1 Experimental setup

As source data, we use Finnish news articles from Yleisradio (Yle) released via Kielipankki (Yleisradio, 2022). We utilise the sentence boundaries and morphological annotations already provided within the corpus's VRT format.

Split by year to prevent leakage. We train and validate on the **2019** portion of the corpus and evaluate on the **2021** portion only. This ensures no document- or sentence-level overlap across train and test. The 2019 slice contains **2,183,722** sentences; the 2021 slice contains **2,110,654** sentences.

During training, a held-out subset of 2019 is reserved for validation (used for early stopping and model selection). We used a max sequence length of 128 tokens with truncation and padding, a cross-entropy objective, AdamW as optimizer, trained on 1 epoch with a batch size of 16, a learning rate of $2 \cdot 10^{-5}$ and weight decay 0.01. We marked template placeholders in the slot-only setting with the '[MASK]' token.

4.1.1 Preprocessing and instance sampling

We convert running text into classification instances as follows:

Candidate head nouns. We run an external morphological analyser over each sentence and *pre-select only nouns* as candidate heads. Tokens analyzed as accusative objects are *excluded* (we do not model accusative; cf. *Problem formulation* section).

Class-aware sampling. To control class imbalance in downstream training, each noun token is assigned a probability of being selected as the supervised slot. The proportion of samples in each class in the evaluation set is shown in Table 6. By default, we set nouns to have a **20%** probability to be selected as prediction sample. For rare cases, we up-weight selection to guarantee sufficient coverage: **100%** for *comitative* and *abessive*, and **30%** for *instructive*. This concentrates supervision on long-tail labels without altering the target label distribution for common cases and avoids severely undertrained heads for rare labels.

Slot rendering (two settings). For each selected noun we create *two* training views:

- **slot-only** by replacing the token with a special marker `[MASK]` and
- **lemma-conditioned** by replacing the surface form with its *lemma*.

We chose to sample these views with a **50/50** proportion so the model learns both settings jointly.

Target labels. The target case label for the slot is taken directly from the morphological analysis provided with the Yle corpus. We treat the corpus analysis as the reference and *do not* attempt to resolve alternative parses; supervision is single-label as mentioned in section 3.4.

Baselines. As baselines, we include two lightweight heuristics. First, we use a pure prior baseline that always outputs the global class probability found in the training data, close to the one mentioned in Table 6. Second, we use an adposition-based rule baseline. In Finnish, as in several other languages, the case of a word can be influenced by a pre- or postposition. For example, the postposition *lähellä* (*close to* in English) generally follows a word inflected to the genitive case (e.g. *talon lähellä*, meaning *close to the house*). We collected a list of adpositions and their governing cases in *Iso suomen kielioppi* (§692-720) (Hakulinen, 2004), see Table 7. At inference, this baseline scans for an adposition adjacent to the slot and predicts the governed case using the appropriate pre/post rule; if no governing adposition is present, we default to the prior baseline.

Our baselines are slot-only by design, focusing on identifying surface-level markers from the adjacent context. Incorporating the lemma into these baselines would require complex rule-based logic to map specific nouns to their most probable cases. The low performance of these baselines reflects the task's inherent complexity, which demands models capable of capturing deeper linguistic dependencies.

5 Results

We evaluate two encoder models (FinBERT, XLM-R) under two settings: (i) *No-lemma* (slot-only) and (ii) *Lemma-conditioned*. Unless noted, metrics are computed over the 14-case label set (no separate accusative).

5.1 Overall performance

All results use the instance construction procedure described in Sections 3.5 and 4.1.

Table 2 summarizes main metrics. FinBERT in the lemma-conditioned setting attains the best accuracy and F1. Providing the lemma yields a substantial gain over the slot-only setting.

5.2 Per-class and confusion analysis

Tables 3 and 4 show confusion matrices for the FinBERT model in the slot-only and lemma-conditioned settings, respectively. The dominant confusions cluster within the locative system: most confused classes are adessive and inessive, particularly in the slot-only prediction (see example predictions in table 8 in the appendix). This is understandable, as the Finnish language treats place names either as something you go "into" (internal locative cases, e.g. `Helsinki` → `Helsinkiin`) or "at" (external locative cases, e.g.

	Lemma-conditioned		Slot-only		Baselines	
Metric	FinBERT	XLM-R	FinBERT	XLM-R	Majority class	Adpositions
Top-1						
Precision	89.1%	85.2%	73.3%	51.9%	2.5%	19.4%
Recall	81.1%	72.2%	63.0%	41.6%	7.1%	7.7%
F1 score	84.2%	75.3%	66.7%	44.7%	3.7%	5.0%
Accuracy	91.4%	87.8%	82.6%	80.8%	34.9%	35.7%
Top-3						
Recall	93.3%	90.8%	84.4%	73.9%	21.4%	21.7%
Accuracy	96.0%	94.1%	90.0%	89.6%	72.4%	72.6%

Table 2: Model performance in lemma-conditioned and slot-only settings. Baselines are slot-only heuristics. Precision, recall, and F1 are macro-averaged over the 14 classes.

		Predicted													
		Abe	Abl	Ade	All	Com	Ela	Ess	Gen	Ill	Ine	Ins	Nom	Par	Tra
Actual	Abe	38	0	3	3	0	2	1	8	9	13	0	7	14	1
	Abl	0	48	4	2	0	17	2	6	3	9	1	4	3	0
	Ade	0	0	45	1	0	2	5	6	2	31	0	4	3	0
	All	0	0	2	56	0	3	1	6	19	6	0	3	2	5
	Com	0	1	4	1	26	2	2	14	2	19	1	23	3	0
	Ela	0	2	1	1	0	77	1	5	2	3	0	4	4	0
	Ess	0	0	3	1	0	2	62	6	1	15	0	7	2	0
	Gen	0	0	0	0	0	0	0	91	1	1	0	4	2	0
	Ill	0	0	1	3	0	2	1	5	80	3	0	2	2	1
	Ine	0	1	5	1	0	3	3	7	2	69	0	6	3	0
	Ins	0	1	5	1	0	2	2	8	3	7	63	6	2	0
	Nom	0	0	0	0	0	0	0	4	0	1	0	92	2	0
	Par	0	0	0	0	0	1	0	3	1	1	0	7	86	0
	Tra	1	1	2	2	0	4	3	7	17	6	0	4	4	50

Table 3: Confusion matrix for FinBERT in slot-only setting

`Tampere` → `Tampereelle`). This pattern also accounts for the confusion of ablative with elative, and illative with allative. Since the choice often depends on the lemma itself, this confusion is significantly alleviated in the lemma-conditioned setting, where the model is able to use the lemma for disambiguation.

Beyond locatives, the model often misclassifies long-tail cases (e.g., abessive, comitative, translative) as their higher-frequency counterparts (e.g., inessive, genitive, nominative). These errors likely stem from shared syntactic roles; for instance, the translative and illative both frequently denote a terminal state or destination in a sentence. In such cases, the model defaults to the most statistically probable category when the specific semantic marker of the rare case is absent from the context.

5.3 Top-3 performance

Table 5 shows the top-3 performance metrics by case in the lemma-conditioned setting. Top-3 performance could be relevant for applications where the user can select from a couple of options from a dropdown menu, for example. The model achieves excellent recall@3 values of 93.3% (macro average) and 96.0% (micro average).

6 Discussion

6.1 What about LLMs?

In recent years, LLMs have become increasingly popular and they allow far greater flexibility in generating text. However, flexibility is not always

		Predicted													
		Abe	Abl	Ade	All	Com	Ela	Ess	Gen	Ill	Ine	Ins	Nom	Par	Tra
	Abe	81	0	1	0	0	0	0	1	0	1	0	7	7	0
	Abl	0	78	4	2	0	5	1	3	1	2	0	3	0	0
	Ade	0	1	85	0	0	1	0	3	1	5	0	3	1	0
	All	0	1	3	80	0	1	1	3	6	2	0	3	1	0
	Com	0	0	2	2	35	5	4	9	1	3	0	39	2	0
	Ela	0	1	0	0	0	85	0	2	1	2	0	4	3	0
Actual	Ess	0	0	0	0	0	1	84	2	1	1	0	9	1	0
	Gen	0	0	0	0	0	0	0	94	0	1	0	3	1	0
	Ill	0	0	1	2	0	1	1	2	86	3	0	3	2	0
	Ine	0	0	2	0	0	2	0	3	2	86	0	4	1	0
	Ins	0	0	5	0	0	0	1	5	2	1	79	6	0	0
	Nom	0	0	0	0	0	0	0	1	0	0	0	97	1	0
	Par	0	0	0	0	0	1	0	2	0	0	0	8	88	0
	Tra	0	0	1	1	0	2	4	2	5	1	0	5	1	77

Table 4: Confusion matrix for FinBERT in lemma-conditioned setting

Table 5: Top-3 performance metrics by case (lemma-conditioned setting)

Case	Recall @ 3
Abe	97%
Abl	95%
Ade	93%
All	94%
Com	71%
Ela	95%
Ess	94%
Gen	96%
Ill	95%
Ine	93%
Ins	95%
Nom	98%
Par	96%
Tra	97%
Micro avg	96%
Macro avg	93%

Table 6: Number of examples per class in the evaluation dataset

Case	Number of examples	Percentage
Nom	75305	33.4%
Gen	54045	23.9%
Par	31256	13.9%
Ine	17407	7.7%
Ill	11045	4.9%
Ela	10907	4.8%
Ade	10284	4.6%
Ess	4910	2.2%
All	4885	2.2%
Abl	2132	0.9%
Tra	1498	0.7%
Ins	1203	0.5%
Abe	289	0.1%
Com	230	0.1%
Total	225396	100.0%

desired in all applications. Template filling is still a simple and effective method when more control is needed over what is being sent to the recipient.

LLMs could also be used for inflection of placeholder values in templates. They are also easily accessible and usable by anyone through APIs. However, we argue that this would not be as lightweight as our approach. Another drawback is that off-the-shelf commercial LLMs often only predict the top-1 result. They would be less flexible in producing probabilities of multiple possible grammatical cases, as we do in our approach.

We still performed an experiment comparing our approach with the performance of a state-of-the-art LLM on this task. Details of the experiment are in Appendix A.

Table 7: Adpositions and their case government, used as our baseline (from §692-710, *Iso suomen kielioppi* (Hakulinen, 2004))

§	Adpositions	Neighbour case(s)	Position(s)
692	luona, luota, asemesta, takia, vuoksi	genitive	postposition
	varten	partitive	postposition
	vastoin	genitive	preposition
	vasten	partitive	preposition or postposition
696	lukuun ottamatta	partitive	preposition or postposition
	huolimatta, riippumatta	elative	preposition or postposition
	katsomatta	illative	postposition
697	mennessä	illative	postposition
		allative	
	kuluessa, kuluttua	genitive	postposition
	verrattuna, suhteutettuna	illative	preposition or postposition
698	alkaen, lähtien	elative	preposition or postposition
		ablative	
	lukien, laskien, pitäen	elative	postposition
	riippuen, johtuen	elative	preposition or postposition
	katsoen, nähden, perustuen, liittyen	illative	preposition or postposition
	koskien	partitive	preposition or postposition
	mukaan lukien, huomioon ottaen, pois lukien	nominative	preposition or postposition
702	halki, poikki, läpi, ali, yli, alla, alle, alta, yllä, ylle, yltä	genitive	preposition or postposition
703	lähellä, edellä, vastapäätä	partitive	preposition or postposition
		genitive	postposition
	ympäri	partitive	preposition or postposition
		genitive	
704	keskellä, keskelle, keskeltä	partitive	preposition
		genitive	postposition
	kesken	partitive	preposition or postposition
		genitive	
705	pitkin, kohti, kohden, vastaan, vailla, vaille	partitive	preposition or postposition
710	päin	partitive	preposition or postposition
708–709	paitsi	partitive	preposition or postposition
	kautta	genitive	preposition or postposition
693	suhteessa	illative	preposition

7 Conclusion

In this paper we framed template filling in morphologically rich languages as a grammatical case selection task for a head-noun slots and proposed a practical encoder-based solution. Instead of handwritten inflection rules, our system predicts a probability distribution over 14 grammatical cases from raw text context and delegates surface realisation to an external morphological generator. The supervision pipeline is corpus-agnostic: it harvests training instances from morphologically analyzed texts with minimal manual effort.

On held-out Finnish news data, a fine-tuned FinBERT model in the lemma-conditioned setting achieves 89.1% precision, 81.1% recall, 84.2% macro F1 and 91.4% accuracy, with macro recall@3 of 93.3%. These results show that accurate case selection is feasible with a lightweight model, and that providing the lemma substantially reduces confusions within the Finnish locative system. The strong top-3 performance makes the approach particularly suitable for interfaces that can present a small set of alternatives or abstain under low confidence.

A small comparison with a modern LLM indicates that a specialised encoder is competitive on this focused task. Overall, the pipeline we describe—automatic extraction of case-labelled slots, lemma-aware slot rendering, and probabilistic case prediction—provides a reusable recipe for Finnish, but also for other morphologically rich languages such as Estonian, Hungarian, or Czech. By decoupling high-level case selection from surface realization, we offer a robust pathway for improving grammaticality in typologically diverse languages with comparable resources.

Limitations

While we proposed a practical solution for grammatical case detection for Finnish templates, several limitations remain.

Our formulation as a single-label prediction task inherits inherent aleatoric uncertainty. Often, several predictions are plausible depending on the writer's intent. In this work, we do not estimate an upper bound on achievable performance under such ambiguity, so it remains unclear how close the reported scores are to the best possible performance without access to user intent.

Second, our experiments are run on news texts. The grammatical case distribution and typical syntactic patterns learned by our model may differ from those typical in the application domain (e.g. in marketing emails or app notifications). We do not quantify how much performance would degrade or shift when moving from news to such domains.

Third, our study is currently restricted to Finnish, and we do not present cross-lingual experiments. Our pipeline could be run on other languages where nouns are regularly inflected to different grammatical cases depending on context (e.g. other Uralic languages, Czech). The effectiveness of the approach in these languages remains to be demonstrated.

Finally, we treat each slot independently, predicting a case for a single head noun at a time. In real templates, multiple placeholders may appear in the same sentence, and the preferred grammatical case for one slot can depend on the choice made for another. Our current model does not enforce global consistency: selecting top-1 predictions for different slots may not jointly produce a coherent sentence.

References

Khalid Alnajjar and Mika Hämäläinen. 2023. Pyhfst: A pure python implementation of hfst. pages 32–35.

Ondřej Dušek and Filip Jurcícek. 2019. Neural generation for czech: Data and baselines. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 563–574.

Auli Hakulinen. 2004. Iso suomen kielioppi.

Mika Hämäläinen and Khalid Alnajjar. 2021. The current state of finnish nlp. *arXiv preprint arXiv:2109.11326*.

Mika Hämäläinen. 2019. UralicNLP: An NLP library for Uralic languages. *Journal of Open Source Software*, 4(37):1345.

Tommi A Pirinen. 2015. Development and use of computational morphology of finnish in the open source and open science era: Notes on experiences with omorfi development. *Finnish Journal of Linguistics*, (28):381–393.

Kees Van Deemter, Emiel Krahmer, and Mariët Theune. 2005. Squibs and discussions: Real versus template-based natural language generation: A false opposition? *Computational linguistics*, 31(1):15–24.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: Bert for finnish. *arXiv preprint arXiv:1912.07076*.

Yleisradio. 2022. Ylen suomenkielinen uutisarkisto 2019-2021, Korp.

Input sentence	Input lemma	Predicted case	Correct?
Posti sulkee ovensa Turun [SLOT] *(The post office closes its doors Turku [SLOT])*	N/A	Inessive (-ssa)	✗
	Eerikinkatu *(Eric Street)*	Adessive (-lla)	✓
Koronavirus on koetellut Etelä-Afrikkaa eniten Afrikan [SLOT] *(Coronavirus has hit South Africa the hardest Africa [SLOT])*	N/A	Inessive (-ssa)	✗
	mantere *(continent)*	Adessive (-lla)	✓
Ensimmäiset merkit taudista havaittiin [SLOT] *(The first signs of the illness were detected [SLOT])*	N/A	Inessive (-ssa)	✗
	viikonloppu *(weekend)*	Essive (-na)	✓
Työttömien mielenosoitus laajeni [SLOT] 1990-luvulla *(The unemployed worker's protest spread [SLOT] in the 1990s)*	N/A	Illative (-een)	✗
	Mannerheimintie *(Mannerheim Street)*	Allative (-lle)	✓
Kläbo hiihti [SLOT] ja joukkuetoveri tuli toisena maaliin *(Kläbo skied [SLOT] and his teammate finished second)*	N/A	Illative (-een)	✗
	ykkönen *(first place)*	Translative (-ksi)	✓

Table 8: Examples of predictions from the most confused predicted classes in the slot-only setting. In these examples, all the predicted cases can be grammatically correct, however only the lemma-conditioned predictions match the groundtruth.

Prompt	Precision	Recall	F1	Accuracy
#1	48.2%	35.8%	34.9%	35.8%
#2	77.1%	71.6%	71.3%	71.9%
#3	81.1%	76.6%	76.7%	77.3%

Table 9: GPT-5 performance (macro averages) on the lemma-conditioned task.

A LLM Experiment Details

We evaluated OpenAI's GPT-5 model on the lemma-conditioned task using three different prompting strategies (Figures 3, 4 and 5). We sampled 1400 instances from the evaluation set, with 100 examples per case. We ran predictions using the `gpt-5-2025-08-07` model with default parameters. For prompts where the model outputs an inflected word or a full sentence, we use UralicNLP's morphological analyzer (Hämäläinen, 2019) to recover the predicted grammatical case.

Table 9 reports macro-averaged top-1 performance. The LLM performs worst when asked to predict the case label directly (Prompt 1), better when asked to output only the inflected word (Prompt 2), and best when rewriting the whole sentence with the inflected form (Prompt 3). Even in the strongest setting, GPT-5 remains slightly below our FinBERT lemma-conditioned model.

These results suggest that a heavier LLM does not automatically outperform a specialised encoder on this focused case selection task. The encoder models remain an attractive option: they are relatively lightweight, achieve higher accuracy on our data, and naturally provide a full probability distribution over cases. In contrast, most off-the-shelf LLM APIs expose only top-1 outputs and do not directly return class probabilities.

```
Your task is to detect the matching
    grammatical case in Finnish
    sentences. You will be given a
    sentence with a placeholder. Your
    output should be grammatical case
    that best fits words that are
    inserted in the placeholder.
The possible cases are Nom, Gen, Par,
    Ine, Ela, Ill, Ade, Abl, All, Ess,
    Tra, Abe, Ins, Com.

# Example

Input: `Soitin eilen [kaveri].`
Output: `All`

# Task
Input: `{input_sentence}`
```

Figure 3: Prompt 1: Predict grammatical case name directly.

```
You will be given a sentence in Finnish.
     Your task is to inflect the word in
     the brackets to the correct
    grammatical case. Your output should
     contain the inflected word.

# Example
Input: `Soitin eilen [kaveri].`
Output: `kaverille`

# Task
Input: `{input_sentence}`
```

Figure 4: Prompt 2: Inflect word only.

```
You will be given a sentence in Finnish.
     Your task is to inflect the word in
     the brackets to the correct
    grammatical case. Your output should
     contain the corrected sentence.

# Example
Input: `Soitin eilen [kaveri].`
Output: `Soitin eilen kaverille.`

# Task
Input: `{input_sentence}`
```

Figure 5: Prompt 3: Rewrite whole sentence with word inflected.

A Valency Lexicon for Universal Dependencies

Petr Kocharov
University of Würzburg
petr.kocharov@uni-wuerzburg.de

Lilit Kharatyan
University of Würzburg
lilit.kharatyan@uni-wuerzburg.de

Abstract

The paper presents a prototype of a web-app designed to automatically generate verb valency lexica based on the Universal Dependencies (UD) treebanks. It offers an overview of the structure of the app, its core functionality, and functional extensions designed to handle treebank-specific features. Besides, the paper highlights the limitations of the prototype and the potential of its further development.

1 Introduction

A prototype of the web-app “UDVaL: Valency Lexicon for Universal Dependencies” (see supplementary demo files[1]) is designed to build valency lexica of languages provided with treebanks, the morphosyntactic annotation of which is stored in the CoNNL-U format and follows the guidelines of the Universal Dependency (UD) project[2]. UDVaL is instrumental for research on a wide range of topics related to verbal morphosyntax, since it provides corpus data on valency and valency alternations of individual verbs and verb classes of a given language within a standardized, typologically oriented annotation framework of UD (De Marneffe et al., 2021).

The app is a derivative of the “CAVaL: Classical Armenian Valency Lexicon”[3], which in turn closely follows the structure and functionality of other corpus-driven valency lexica, in particular, “IT-VaLex” for Latin (Passarotti et al., 2016) and “HoDeL: Homeric Dependency Lexicon” for Ancient Greek (Zanchi, 2021), which in turn rely on the model of “PDT-Vallex” for Czech (Hajic et al., 2003).

UDVaL offers a search engine that supports flexible queries on verb frames based on the combinations of morphological, syntactic and lexical properties of verbs and their dependencies, as specified in section 2. This functionality significantly facilitates research on a wide array of topical linguistic issues, such as valency classes of verbs, valency alternations, alignment, coding of verbal dependencies, etc. (see applications in section 2.3), and contributes to the inventory of digital tools for the linguistic analysis of the rich comparative data accumulated within the UD project (the latest UD release v2.17 includes 339 treebanks of 186 languages).

Unlike general-purpose online tools designed to query dependency treebanks such as PML-TQ[4] and TüNDRA[5], UDVaL is customised for queries on verb frames, and does not require knowledge of formal query languages. The presented prototype version of UDVaL offers a package that can be used for deriving, with more or less adaptations indicated below, a valency lexicon of any language included in the UD database.

The app automatically retrieves verb frames for all verbs of a treebank. A verb frame consists of a verb together with its dependents carrying the grammatical relations of core arguments (subject, direct object, indirect object) and oblique nominals (all other modifiers), and their clausal equivalents (see section 2.1). In line with the UD principles, these types of relations aim at grasping cross-linguistic similarities in abstraction from language-specific overt coding properties of verb frames. With that, a wide-spread distinction between oblique arguments (obligatory or selected by the predicate) and adjuncts (facultative) is abandoned, both types being covered by the same tag “obl”, since it is difficult to consistently apply it to the annotation of one language and across languages (see Haspelmath, 2014 on the controversies of this distinction in a cross-linguistic perspective). In some treebanks,

[1] https://github.com/caval-project/ud_val
[2] https://universaldependencies.org
[3] https://github.com/caval-project
[4] https://lindat.mff.cuni.cz/services/pmltq/#!/treebanks
[5] https://weblicht.sfs.uni-tuebingen.de/Tundra

Proceedings of the 8th Workshop on Research in Computational Linguistic Typology and Multilingual NLP, pages 12–17
March 29, 2026 ©2026 Association for Computational Linguistics

the distinction is partly formalized by the use of specialized tags “obl:arg” (oblique nominal) and “obl:agent” (oblique agent in passive construction), corresponding to arguments, not adjuncts. Regarding UDVaL, the aforementioned annotation principles have the advantage of providing corpus-driven data, less prone to annotators’ bias on whether or not a given oblique dependent is selected by the predicate, while clearly indicating core arguments, which constitute the backbone of verb frames. The distinction between arguments and adjuncts can then be assessed within different theoretical frameworks based on the relative frequency of dependents of specific verbs insofar as the size of a treebank allows it.

2 The UDVaL app

2.1 Backend

UDVaL has a multi-tier architecture. Its backend employs a MariaDB relational database, which is automatically populated with data extracted from a UD treebank of a selected language stored in the CoNNL-U format using parameterised SQL queries and a Python code for dynamic data fetching. The application layer is developed using the Python-based Flask micro web framework to handle application logic and query processing. The frontend layer utilises HTML, CSS, and JavaScript to support a responsive and user-friendly interface. Advanced database indexing and query optimisation techniques have been implemented to facilitate complex queries and ensure high performance.

The interface supports search queries to verb frames and generates complete lists of their occurrences in the database.

The core of the interface functionality is to query constructions with a verbal head and a subset of its immediate syntactic dependents as well as second-order coordinated dependents linked by the “conj” relation.

The presented app prototype only processes predicative constructions headed by verbs (UPOS tag “VERB”). The backend database indexes and stores all verbs of a treebank together with their dependents, carrying the following grammatical relations (and their subtypes):

- nominals: “nsubj” (nominal subject), “obj” (direct object), “iobj” (indirect object), “obl” (oblique modifier), as well as “nmod” (nominal modifier) in the case of dependents of nominalised heads;
- clausal dependents fulfilling the function of core arguments: “csubj” (clausal subject), “ccomp” (clausal complement without an obligatorily controlled subject), “xcomp” (clausal complement with an obligatorily controlled subject);
- auxiliaries of analytic verb forms carrying the relation tag “aux”.

Another feature of the app, determined by the UD annotation guidelines, concerns the morphological expression of nominal dependents. To support the cross-linguistically consistent comparison of grammatical functions of nominals, all adpositions are considered as part of case marking along with inflectional case morphology in UD (see Haspelmath, 2019 on the comparable grammatical status of case markers and adpositions). To account for this annotation feature, nominals are stored in the UDVaL database together with adpositions. This allows making queries to the morphological expression of nominal dependents in terms of attested combinations of the inflectional case (morphological tag “Case”) and adpositions linked to the nominals by a grammatical relation “case”. In case of multiword adpositions (such as English “according to”, cf. the UD English-GUM treebank of the UD database under fn. 2), the constituents linked by the “fixed” relation are included in the encoding pattern.

2.2 Frontend

Some key functionalities of the search engine and the user interface are summarised below.

By default, the interface lists all the verbs which are attested in a treebank. The verbs are linked to pages with complete sets of their occurrences in the treebank. By modifying search parameters, the user restricts the list of verbs and occurrences. Besides selecting a verb from the list, it can be found via a free input field “Search by verb”. A string of characters in the utf-8 format matching a verb lemma (or several homonymous lemmas) restricts the verb list. The default and restricted lists can be arranged in alphabetical order or by token frequency (ascending and descending).

A valency frame can be configured by morphological, syntactic and lexical properties of an open number of dependents of a verbal head insofar as they co-occur in the treebank. These search parameters can be applied to a list of verbs as well as to a specific verb.

By default, the verb frame query contains a set of the following three parameters for one dependent (see Figure 1):

- "Select relation": the grammatical relation tag from the list specified in section 2.1;
- "Select encoding": the inflectional and/or adpositional case marking;
- "Select lemma": a lemma from the list of lemmas filling the dependent in a given treebank.

In the second of these parameters, encoding patterns are given as a closed list of values, automatically generated from the treebank data, in the format "Case + Adp(s)", regardless of the linear order of constituents, to facilitate the value selection. For clausal dependents, the value of this parameter is conventionally set to "Clause" for tokens with the relations "csubj" and "ccomp" and to "Nominal" for the tokens with the relation "xcomp".

Each selected parameter or its reversal to the default value dynamically updates the verb list and lists of occurrences. When at least one of the parameters is selected, a new set can be added for the next dependent. Sets can be added or removed and their parameters specified until the output list of verbs or occurrences is empty.

The flexibility of interface allows to configure queries of increasing complexity suitable for specific research tasks. Disentangling the morphological, syntactic, and lexical tiers enables queries, in which, for example, a syntactic feature is specified for one dependent and a morphological one for another dependent. The dynamic update of the output with frequency data allows to instantly determine the availability and relative frequency of frames or their specific features in the underlying treebank.

A page with occurrences of a verb includes a complete list of its attestations in the treebank. Every occurrence consists of a sentence, its reference id in the treebank (based on the compulsory "text" and "sent_id" comment fields of the CoNNL-U format), and a morphosyntactic BRAT-based[6] visualisation of the verb frame. The visualisation includes part-of-speech attributes of all tokens and syntactic relations that constitute a verb frame specified in a search query. Besides, the visualisation is provided with the mouse-over glossing of all words in the sentence. The UDVaL engine supports conversion of the UD annotation tags to the Leipzig glossing conventions[7].

By default, the user interface utilises an alphabetic selector, which is automatically generated based on the initial characters of verb lemmas. This functionality only applies to treebanks of languages with alphabetic or syllabic writing systems, the inventory of initial characters of which is manageable within the selector field of the user interface and can therefore be seamlessly integrated into the app.

2.3 Applications

UDVaL inherits all key functionalities of the online CAVaL app[8], has been approbated while assembling corpus data for the ongoing research on Classical Armenian morphosyntax, in particular, as part of the "PaVeDa: Pavia Verbs Database" project[9].

In particular, it provides corpus data on the encoding of core arguments and alignment of Classical Armenian, which involves the split marking of all core arguments (Kölligan, 2013; Müth, 2014). For example, it provides instant access to all the occurrences of perfect tenses with the genitive subject (33x, incl. 16x in a transitive construction) and the nominative subject (189x, incl. 3x in a transitive construction) in the Gospels. The emerging mismatches between transitivity and genitive flagging contribute to the ongoing discussion on the role of animacy and affectedness of the subject in this morphosyntactic split.

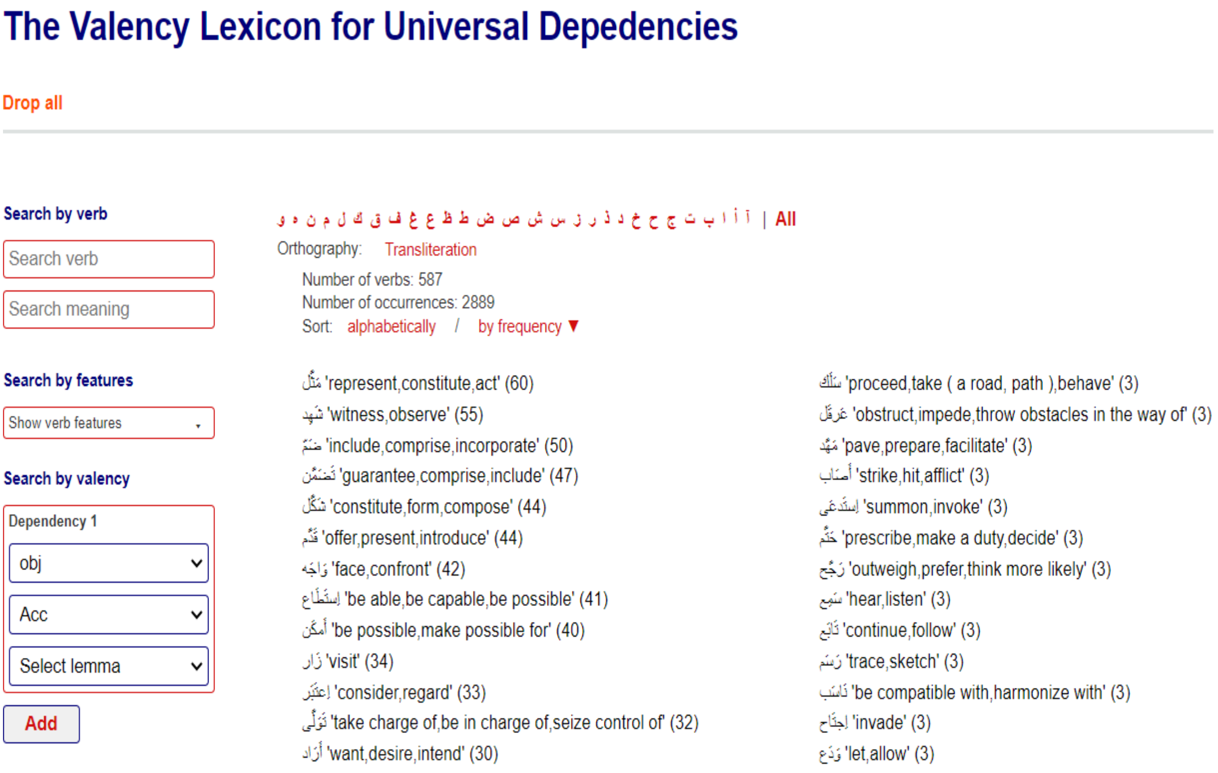
Similarly, the app provides corpus data on the split marking of direct object by a bare accusative and a prepositional phrase built with proclitic *z* plus the accusative, with 1289 and 3124 occurrences in the Gospels, respectively. This data suggests that a more frequent marking pattern, associated with a referentially prominent direct object, is morphosyntactically more complex, again a cross-linguistic generalization suggested in (Haspelmath, 2021). Within the same subcorpus, the ditransitive construction occurs 181 times with the prepositional marking of direct object, and 299 times with an accusative marking. This data points to the tripartite ditransitive alignment: P (*z* plus Acc), T (Acc), and R (Dat) are expressed by different majority encoding types. Such ditransitive is typologically rare (Haspelmath, 2005).

[6] https://brat.nlplab.org/embed.html

[7] https://www.eva.mpg.de/lingua/pdf/Glossing-Rules.pdf

[8] https://caval.dch.phil-fak.uni-koeln.de

[9] https://paveda.unipv.it/contributions/clas1256



The Valency Lexicon for Universal Depedencies

Drop all

Search by verb

Search verb

Search meaning

Search by features

Show verb features

Search by valency

Dependency 1

obj

Acc

Select lemma

Add

آ أ إ ا ب ت ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و | All

Orthography: Transliteration

Number of verbs: 587
Number of occurrences: 2889
Sort: alphabetically / by frequency ▼

مَثَّل 'represent,constitute,act' (60)
شَهِد 'witness,observe' (55)
ضَمَّ 'include,comprise,incorporate' (50)
تَضَمَّن 'guarantee,comprise,include' (47)
شَكَّل 'constitute,form,compose' (44)
قَدَّم 'offer,present,introduce' (44)
وَاجَه 'face,confront' (42)
إستَطَاع 'be able,be capable,be possible' (41)
أَمكَن 'be possible,make possible for' (40)
زَار 'visit' (34)
إعتَبَر 'consider,regard' (33)
تَوَلَّى 'take charge of,be in charge of,seize control of' (32)
أَرَاد 'want,desire,intend' (30)

سَلَك 'proceed,take (a road, path),behave' (3)
عَرقَل 'obstruct,impede,throw obstacles in the way of' (3)
مَهَّد 'pave,prepare,facilitate' (3)
أَصَاب 'strike,hit,afflict' (3)
إستَدعَى 'summon,invoke' (3)
حَتَّم 'prescribe,make a duty,decide' (3)
رَجَّح 'outweigh,prefer,think more likely' (3)
سَمِع 'hear,listen' (3)
تَابَع 'continue,follow' (3)
رَسَم 'trace,sketch' (3)
نَاسَب 'be compatible with,harmonize with' (3)
إجتَاح 'invade' (3)
وَدَع 'let,allow' (3)

Figure 1: Adaptation of the UDVaL app for the Arabic-PADT UD treebank (v2.17).

2.4 Functional extensions

The core functionality of UDVaL, summarised above, can be supplemented by secondary functions, which were developed for the UD treebank of Classical Armenian and can be applied with due modifications to subsets of UD treebanks. By default, these extensions are included in the supplemented demo codes as customised for Classical Armenian. Adjustments of the app to other treebanks with respective features are facilitated by internal comments of the demo codes.

Latinized input: In order to facilitate the search of verbs via the free input field "Search by verb", the code allows setting up latinized input conventions. This functionality requires extending the code with a chart of equations between original and input characters or groups of characters.

Transliteration: Based on the "Translit" and "LTranslit" (wordform and lemma transliteration, respectively) attributes of the MISC field for miscellaneous information of the CoNNL-U format, the UDVaL search engine supports representation of verb lists and occurrences in the transliterated form. The implementation of this feature requires adding a chart of correspondences between the original and transliterated characters, and integrating these correspondences with the alphabetic selector.

Translation: The UDVaL engine allows searching verbs by meaning provided that a treebank has the "Gloss" attribute in the MISC field; the attribute contains an approximate translation of the word form or lemma to another language, typically but not exclusively English.

A translation of sentences can be switched on/off in the lists of occurrences for treebanks with sentential translations. In the demo code, this feature is based on the "# translated_text" comment field of the CoNNL-U format, and can be adjusted to the metadata properties of a given treebank.

External lexicographic sources: Given the necessarily approximate lexicographic information stored in the "Gloss" attribute (when available), the UDVaL engine allows linking verb lemmas to external online lexicographic sources. The code enriches the database with external links based on a txt-file with a list of verb lemmas and corresponding URLs of the external lexeme entries.

Subcorpus selector: If relevant for a given treebank, a selector of subcorpora or individual texts can be integrated into the search interface to facilitate research of morphosyntactic variation across texts. This feature has proven to be useful for the study of Classical Armenian verb frames in the CAVaL implementation of the app. For example, while 14,5% of verbs are attested in the Gospels with a dative argument, a post-classical text, “History of the Armenians” by Movses Khorenatsi, the precise dating of which is debated, has 18,2% of such verbs. This data is relevant for the study of the gradual increase of dative marking towards Modern Armenian (Daniel and Khurshudian, 2015). In the demo code, this feature is based on the “# sent_id” comment field of the CoNNL-U format, and can be adjusted to the metadata of a given treebank.

Verb features: For treebanks with annotated morphological features, the verb frame query can be further extended with verbal features insofar as they are attested in the treebank for a specified verb frame. This functionality requires adjusting the code to integrate the language-specific list of verbal features into the search interface.

Limitations

Insofar as the CAVaL implementation allows to judge, the retrieval accuracy of the UDVaL engine, based on deterministic database queries, is exact (cf. the frequency data on the co-occurrence of tags in the Classical Armenian treebank of the release UD v2.17 and in the online app under the link in fn. 3). Any perceived errors must be attributed to the quality of the underlying treebank annotations and evaluated as such.

The presented prototype of the UDVaL app requires manual adjustments of the code to tailor functional extensions, mentioned in section 2.4, to available features of a given treebank. Subsequent iterations of the app envisage automated detection of treebank features and integration of respective search functions into the user interface.

The current version of the app is limited to the queries of verb frames with a verbal head and a restricted subset of types of syntactic dependents, listed in section 2.1. This limitation excludes queries to nominal predicates, polipredicative constructions, clausal adverbial modifiers, etc., that are relevant for the study of valency. Thus, the app does not allow the comparison of case frames and their alternations for verbal and nominal predicates.

Being oriented towards typological comparative studies, the app does not allow integrating multiple treebanks into one interface. This limitation points to the potential of further development of the app to implement functionality that would support instant comparison of verb frames across treebanks of different languages within one search interface.

The response time of the app is currently coupled with corpus size: while interaction remains efficient for smaller datasets, integration of larger corpora yields a noticeable increase in latency. Future iterations will address this scalability bottleneck by refactoring the data-access layer around an Object–Relational Mapping (ORM) approach, to reduce response time, especially for complex frame queries, and enable seamless interaction with substantially larger corpora.

Ethical Considerations

The publication complies with the ACL Ethics Policy.[10] In particular, neither part of the presented technology violates the license permissions of reuse for non-commercial purposes.

Acknowledgments

This research is part of the Classical Armenian Valency Lexicon project, funded by the Deutsche Forschungsgemeinschaft (DFG), project number 518003859. We are grateful to two anonymous reviewers for their valuable comments.

References

Michael Daniel and Victoria Khurshudian. 2015. *14. Valency classes in Eastern Armenian*, pages 483–540. De Gruyter Mouton, Berlin, Boston.

Marie-Catherine De Marneffe, Christopher D Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational linguistics*, 47(2):255–308.

Jan Hajic, Jarmila Panevová, Zdenka Urešová, Alevtina Bémová, Veronika Kolárová, and Petr Pajas. 2003. Pdt-vallex: Creating a large-coverage valency lexicon for treebank annotation. In *Proceedings of the second workshop on treebanks and linguistic theories*, volume 9, pages 57–68.

Martin Haspelmath. 2005. Argument marking in ditransitive alignment types. *Linguistic Discovery*, 3:1–21.

Martin Haspelmath. 2014. Arguments and adjuncts as language-particular syntactic categories and as comparative concepts. *Linguistic Discovery*, 12:3–11.

Martin Haspelmath. 2019. Indexing and flagging, and head and dependent marking. volume 62, pages 93–115.

[10] https://www.aclweb.org/portal/content/acl-code-ethics

Martin Haspelmath. 2021. Role-reference associations and the explanation of argument coding splits. *Linguistics*, 59(1):123–174.

Daniel Kölligan. 2013. *Non-canonical subject marking: Genitive subjects in Classical Armenian*, pages 73–90. John Benjamins Publishing Company.

Angelika Müth. 2014. *Indefiniteness, animacy and object marking: A quantitative study based on the Classical Armenian Gospel translation. PhD Thesis*. University of Oslo, Oslo.

Marco Passarotti, Berta González Saavedra, and Christophe Onambele. 2016. Latin vallex. a treebank-based semantic valency lexicon for latin. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2599–2606.

Chiara Zanchi. 2021. The homeric dependency lexicon: What it is and how to use it. *Journal of Greek Linguistics*, 21(2):263–297.

Evaluating the Interplay of Information Status and Information Content in a Multilingual Parallel Corpus

Julius Steuer[2], Andrew Dyer[1], Toshiki Nakai[1],
Luigi Talamo[1], Annemarie Verkerk[1]
[1]Department of Language Science and Technology Saarland University
[2]Heidelberg Institute for Theoretical Studies
Correspondence: julius.steuer@h-its.org

Abstract

The uniform information density (UID) hypothesis postulates that linguistic units are distributed in a text in such a way that the variance around an average information density is minimized. The relationship between information density and information status (IS) is so far underexplored. In this ongoing work, we project IS annotations on the English section of the CIEP+ corpus (Verkerk and Talamo, 2024) to parallel sections in other languages. We then use the projected annotations to evaluate the relationship between IS and information content in a typologically diverse sample of languages. Our preliminary findings indicate that there is an effect of information status on information density, with the directionality of the effect depending on language and part of speech.

1 Introduction

Research on information status (IS) and information content has the same aim: assessing how information is distributed across words in sentences and larger discourse units; sometimes, special attention is paid to word order and communicative efficiency, e.g., Tsipidi et al. (2024). IS refers to whether a listener or speaker should look for the referents of a word or phrase among a set of already mentioned or given entities, or add a new entity to this set (Chafe, 1976), e.g., in centering theory (Gundel, 1997). IS has an effect on the placement of words: new items are preferentially placed earlier in the utterance than given items (Clark and Clark, 1978). As an example, consider the following sequence of sentences:

1.a The Hobbit, or there and back again.

1.b In a hole in the ground there lived a hobbit.

1.c It was a hobbit-hole, and that means comfort.

Here a new entity *hole* is mentioned at the beginning of **1.b**, and referred to by co-referring pronoun *it* at the beginning of **1.c**. This coreference relationship establishes the entity referred to by *it* as given. *Hobbit* in **1.b** is already given (as it was already mentioned in **1.a**), and thus placed at the end of **1.b**.

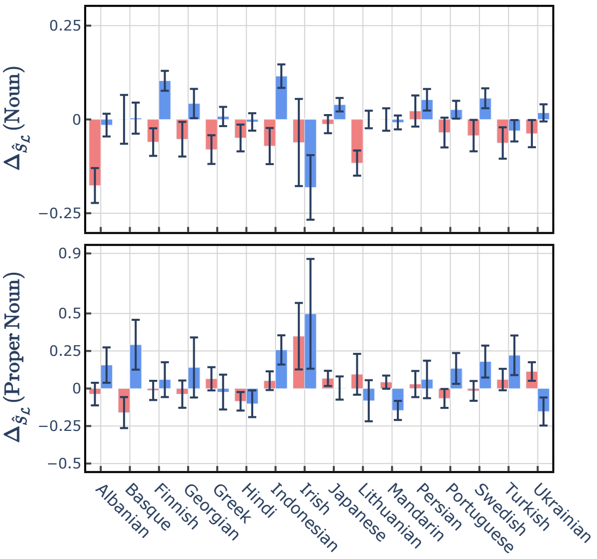


Figure 1: Mean deviation $\Delta_{\hat{S}_\mathcal{L}}$ of surprisal from channel capacity $\hat{S}_\mathcal{L}$ for nouns and proper nouns for new and given items. Error bars show standard error.

While information status refers to the mental state of the speaker or listener *prior* to encountering a new or given word, information content, measured in bits per item (Shannon, 1948), as the basis of surprisal theory (Hale, 2001; Levy, 2008) refers to the amount of information gained by the listener *after* encountering it. Surprisal, like IS, has been shown to impact word and morpheme order (e.g., Hahn et al. 2021 and Cuskley et al. 2021): *Hobbit* may have been placed at the end of **1.b** because (for a first-time reader of *The Hobbit*), this word would have high surprisal.

Thus, surprisal theory and IS use the term "information" in different ways, which becomes clear

Proceedings of the 8th Workshop on Research in Computational Linguistic Typology and Multilingual NLP, pages 18–25
March 29, 2026 ©2026 Association for Computational Linguistics

if we try to reformulate the content of each in the verbiage of the other: Surprisal measures *how* new a word is in context *after* it has been encountered, as the information gained at each word is by definition new information. IS, in contrast, assigns one of two or more labels to a mental list of entities that are part of a discourse, with new entities expected to yield more information (in bits) than given entities. To our knowledge, this conjectured relationship between surprisal and IS has not been evaluated before. In this work, we are taking a first step towards an evaluation of this relationship in a multilingual setting.

Our starting point is the uniform information density hypothesis (UID) (Levy and Jaeger, 2007), which posits that speakers prefer sentences in which the information content of words stays close to their channel capacity (Collins (2014) inter alia), that is, the average information rate at which language transmission occurs. In order to evaluate UID and its interaction with IS in a multilingual setting, we use the English section of miniCIEP+ (Verkerk and Talamo, 2024) annotated for IS by (Dyer et al., 2024) to automatically annotate parallel data in miniCIEP+.

The remainder of the paper is structured as follows: Section 2 introduces a formal definition of the surprisal of new and given items, and briefly reviews existing work on UID and the interaction of information content and word order. Section 3 describes the annotation projection from English to other languages. In Section 4, we evaluate the annotation projection and discuss preliminary results: we find that while surprisal of nouns stays close to channel capacity independent of IS, surprisal of given pronouns consistently falls below the channel capacity. For new proper nouns, we find that in some languages surprisal on the average falls below channel capacity, while in others, channel capacity is usually exceeded.

2 Surprisal, information status and UID

As already laid out in the introduction, from the vantage point of surprisal theory, we expect new and given items to behave in fundamentally different ways: Since new items introduce entities to the discourse, they should be less predictable from context and their surprisal should be higher than that of given items. UID, in contrast, predicts that new and given items are distributed in such a way that their surprisal deviation from channel capacity is minimized. In this section, we will introduce formal definitions of surprisal of new and given items, and a corresponding operationalization of UID.

2.1 Surprisal of new and given entities

Following the nomenclature of Dyer et al. (2024), we express that an entity $\mathbf{e}$ was *mentioned* in the discourse segment or preceding context $\mathbf{c}$ as $\mathcal{M}(\mathbf{c}, \mathbf{e})$, and the opposite case, i.e., if $\mathbf{e}$ was not mentioned in $\mathbf{c}$ and thus has IS new, as $\neg\mathcal{M}(\mathbf{c}, \mathbf{e})$.

We now want to formalize our intuition about the information content of new and given entities. We expect that the information gained by observing a string representation $\mathbf{s}$ of mentions of entity $\mathbf{e}$ given $\mathbf{c}$ will be lower if condition $\mathcal{M}(\mathbf{c}, \mathbf{e})$ holds, i.e., that on average already mentioned entities receive lower surprisal S:

$$S[\mathbf{s}|\mathbf{c}, \mathcal{M}(\mathbf{c}, \mathbf{e})] \leq S[\mathbf{s}|\mathbf{c}, \neg\mathcal{M}(\mathbf{c}, \mathbf{e})] \tag{1}$$

Our intuition is based on the fact that, by definition, there must be *some* information about (a referent of) $\mathbf{e}$ in $\mathbf{c}$ if $\mathbf{e}$ is given, but not necessarily if $\mathbf{e}$ is new. However, this does not mean that the context $\mathbf{c}$ is entirely devoid of information about entity $\mathbf{e}$ if it is new, for example in bridging entities (*I went to the hospital. The doctor took my blood pressure.*). It is this relationship between an entity's IS and the relevance of its context in predictive processing that we want to unravel:

Research Question

Do given items receive lower surprisal than new items?

2.2 UID and information status

As already mentioned in Section 1, UID makes a very different prediction for the average surprisal of new and given entities: All else equal, a speaker will accommodate words in such a way that the variance of surprisal from the channel capacity[1] is minimized, and IS is only one of the factors at play. Thus, from the vantage point of UID, we would expect a different relationship between new and given entities, i.e., that on average the effects of newness and givenness cancel out:

$$S[\mathbf{s}|\mathbf{c}, \mathcal{M}(\mathbf{c}, \mathbf{e})] \approx S[\mathbf{s}|\mathbf{c}, \neg\mathcal{M}(\mathbf{c}, \mathbf{e})] \tag{2}$$

UID has been shown to hold for English both from the perspective of dependency length (Collins,

[1] Of the speaker, but also the listener CITE

2014) and word order (Cuskley et al., 2021). However, both methods calculate the deviation from channel capacity as the word-to-word variance of unigram surprisal. In contrast, we estimate surprisal from a causal transformer language model (see Appendix B for details).

A more comprehensive evaluation of different formalizations of UID is offered by Meister et al. (2021), who systematically vary the scope with which the channel capacity is calculated, finding that the language level aligns better with human data. Based on their work, for a language $\mathcal{L}$, we will calculate channel capacity $\hat{S}_{\mathcal{L}}$ on the language level as average surprisal over all N words in that language's section of miniCIEP+. We refer to this formalization of UID as $\mathrm{UID}_{\mathcal{L}}$:

$$\mathrm{UID}_{\mathcal{L}} = \frac{1}{N-1}\sum_{i=2}^{N}(S(\mathbf{s}_i|\mathbf{c}) - \hat{S}_{\mathcal{L}})^2 \tag{3}$$

Here, $\mathbf{c}$ expands into a prefix of words of fixed length T, $\mathbf{c} = \mathbf{s}_{i-T-1}, ..., \mathbf{s}_{i-1}$. We can now rephrase our initial research question it in terms of $\mathrm{UID}_{\mathcal{L}}$:

Research Question*

Does $\mathrm{UID}_{\mathcal{L}}$ hold for given and new items, i.e., is there a difference in surprisal between new and given items when viewing deviation from channel capacity?

3 Annotation projection on miniCIEP+

3.1 Data

We start out from English CiepInf (Dyer et al., 2024), which comprises a subcorpus of miniCIEP+ (Verkerk and Talamo, 2024). We use the annotations for IS on the English to automatically annotate other languages in miniCIEP+ by projecting English IS labels to word-aligned parallel data. Although there are more fine-grained notions of IS (e.g., Gundel et al. 1993; Markert et al. 2012), we restrict ourselves to a dichotomy between new and given entities for the sake of simplicity. We evaluate the projected annotations by comparing them to hand-aligned gold-standard data from CiepInf.

3.2 Projection

In CiepInf, entity tags are associated with the spans of noun phrases (NPs), and each entity is annotated with an IS label. Because our projection operates at the token level, we extract only the syntactic head token of the span. We then project IS labels from English to each target language in two stages. We first obtain sentence-level alignments with Bertalign (Liu and Zhu, 2023). Then, for every aligned sentence pair, we compute word alignments with awesome-align (Dou and Neubig, 2021). Whenever an aligned English token carries an IS label, we copy that label to its aligned target token(s).

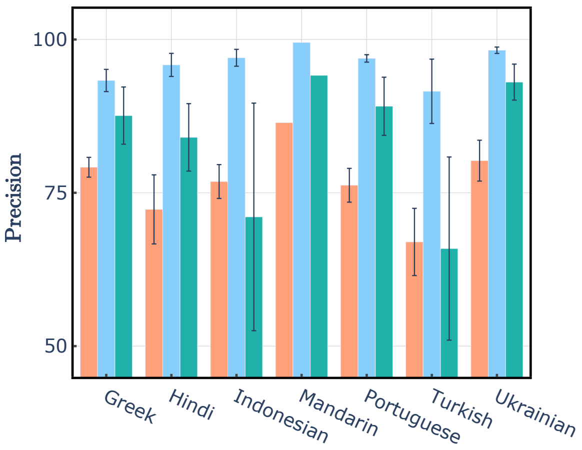


Figure 2: Precision of information status projection on gold data for nouns, pronouns, and **proper nouns** per language. We calculate precision as accuracy on words that have a label in the gold and projected data. Error bars show standard error over books.

We do not fine-tune either aligner for any language. While Bertalign officially supports 25 languages and awesome-align reports evaluation on five languages, both approaches are based on LaBSE (Feng et al., 2022) and mBERT (Devlin et al., 2019) respectively and therefore remain applicable to a wider set of languages, though quality may degrade for typologically distant and/or lower-resource languages. We therefore evaluate projection quality on 7 languages with gold IS labels taken from CiepInf (Chinese, Greek, Hindi, Indonesian, Portuguese, Turkish, Ukrainian).

Evaluation with gold labels To measure intrinsic projection quality, we evaluate token-level agreement on the subset of tokens that have *both* a gold IS label and a projected IS label:

$$\text{Precision} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}[\hat{y}_i = y_i]\,,$$

where y_i is the gold IS label, $\hat{y}_i$ is the projected label, and N is the number of evaluated tokens (intersection of gold-labeled and projected-labeled tokens). $\mathbb{I}$ denotes the indicator function that gives

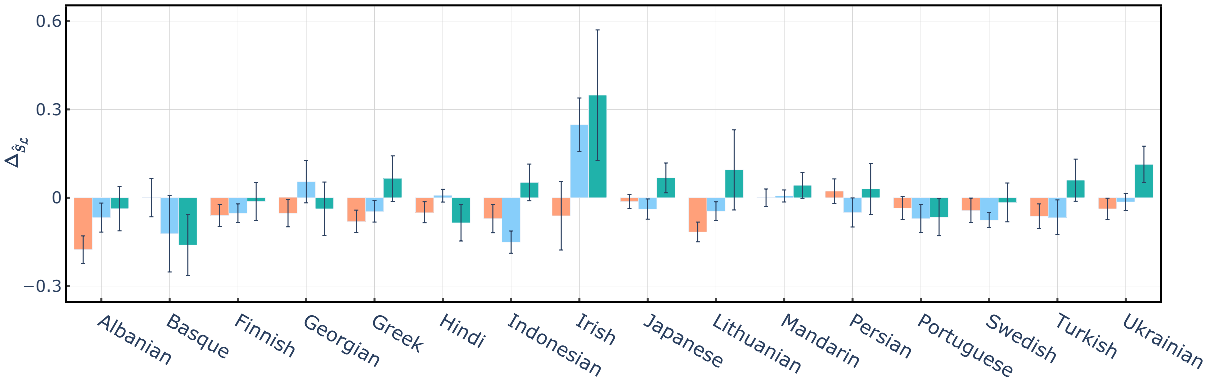


Figure 3: Mean deviation $\Delta_{\hat{S}_\mathcal{L}}$ of surprisal from channel capacity $\hat{S}_\mathcal{L}$ for given nouns, pronouns, and proper nouns per language. Error bars show standard error. Mandarin lacks error bars because the evaluation is based on a single book.

1 if the condition is satisfied, and 0 if not.

Evaluation without gold labels For unlabeled languages, we quantify how frequently projection assigns IS labels by computing (per POS category defined in miniCIEP+) the proportion of tokens that receive any projected IS label:

$$\mathrm{coverage}(\mathrm{pos}) = \frac{N_{\mathrm{projIS}}(\mathrm{pos})}{N(\mathrm{pos})}.$$

where $N(\mathrm{pos})$ is the number of tokens in POS category pos, and $N_{\mathrm{projIS}}(\mathrm{pos})$ is the number of such tokens that have received an IS annotation tag through projection.

Expected label sparsity Since English is the sole source of supervision and projection is one-way (English → target), projected corpora typically contain fewer IS labels than the English gold annotations. Additional label loss arises when sentence/word alignment fails to link an English labeled token to a target token, compounding effects from linguistic distance and aligner quality.

4 Preliminary results

4.1 Projection quality

Figure 2 reports the precision of projected IS labels against gold annotations. Pronouns achieve consistently high precision across all evaluated languages, which is expected because pronouns are strongly biased towards given, and their closed-class nature makes it easier to align them. In contrast, common nouns show substantially lower precision in Hindi and Turkish. A plausible explanation is increased syntactic divergence from English (both are predominantly SOV), which can degrade word-alignment quality and increase noise in projection. **Proper nouns** exhibit the largest cross-linguistic variability, potentially reflecting differences in orthographic conventions, named-entity formation, and language-specific usage patterns that affect both their presence in the target text and their potential alignment. We evaluate the quality of the projected annotations to languages for which we did not collect gold data in Appendix A.

4.2 Surprisal and information status

For our preliminary analysis, we report results for the 7 languages for which we collected gold data, and 9 more languages we selected based on their genealogical diversity: Albanian, Basque, Finnish, Georgian, Irish, Japanese, Lithuanian, Persian, and Swedish. We excluded new pronouns from our analysis because true cataphora, i.e., forward-pointing personal pronouns as in

2. As she was searching for her$_{\mathrm{M}_1}$ hat, Mary$_{\mathrm{M}_2}$ came upon a long-lost letter.

are rare; consequently, new pronouns are rare and almost exclusively equivalents of ‘something’, ‘nobody’, and other such indefinite pronouns.

UID$_\mathcal{L}$ Comparing new and given words in this set of languages, we find that surprisal of nouns usually stays close to channel capacity independently of IS, while there is an apparent effect of IS on the surprisal of **proper nouns**: Figure 1 (on first page) shows that the surprisal of new proper nouns exceeds channel capacity in Basque, Indonesian, Portuguese, Swedish and Turkish, while it falls below channel capacity in Ukrainian. Thus, nouns are more conformant to $\text{UID}_{\mathcal{L}}$ than proper nouns, whose usage may be influenced by other considerations than $\text{UID}_{\mathcal{L}}$.

Givenness Figure 3 shows that for all languages in our sample, surprisal of given nouns and pronouns consistently falls below channel capacity, while that of proper nouns usually exceeds it. Together with our finding for $\text{UID}_{\mathcal{L}}$, this indicates a differing use of proper nouns as per the *mediated* IS in Markert et al. (2012), a distinction which is not reflected in CiepInf.

5 Conclusion

In this work, we laid out our approach to annotate miniCIEP+ for IS by projecting gold annotations from English to parallel data in other languages and presented preliminary results on the interaction of IS and UID. While our results point to an interaction of IS and UID, the quality of the projected annotations is affected by the quality of alignment models, the improvement of which will be a first step to higher-quality projections and further analysis.

Limitations & future work

The work presented in this paper relies heavily on quality of projected annotations, which in turn relies on the quality of BertAlign (Liu and Zhu, 2023) and awesome-align (Dou and Neubig, 2021). We applied these tools to our parallel corpus without any modifications, yielding a lossy projection as evident from Figure 4, e.g., for Arabic, Irish and Latin. We want to address this performance gap by fine-tuning both alignment models.

Secondly, the IS annotations in CiepInf are rather coarse-grained: We only distinguish new and given words, while most IS schemata have intermediate states (e.g., brand-new vs. new and mediated vs. new/given) or distinguish between the hearer/reader and the discourse level (Prince, 1992).

Lastly, the distinction between reader and discourse is only meaningful if the reader has some prior knowledge about the world in which the discourse is situated. While this is arguably the case in a multilingual language model like mGPT (see, e.g., (Li et al., 2021)), it is not possible to know *beforehand*, e.g., if and how the string *Alice* is associated with the main character in *Through the Looking Glass*, i.e., what is part of the discourse the language model is aware of.

References

Wallace L Chafe. 1976. Givenness, contrastiveness, definiteness, subjects, topics, and point of view. *Subject and topic*. Publisher: Academic Press.

Eve Vivienne Clark and Herbert H. Clark. 1978. Universals, relativity, and language processing. In Joseph H. Greenberg, editor, *Universals of human language. 1: Method & theory*. Stanford Univ. Press, Stanford, California. Num Pages: 286.

Michael Xavier Collins. 2014. Information Density and Dependency Length as Complementary Cognitive Models. *Journal of Psycholinguistic Research*, 43(5):651–681.

Christine Cuskley, Rachael Bailes, and Joel Wallenberg. 2021. Noise resistance in communication: Quantifying uniformity and optimality. *Cognition*, 214:104754.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

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

Andrew Dyer, Ruveyda Betul Bahceci, Maryam Rajestari, Andreas Rouvalis, Aarushi Singhal, Yuliya Stodolinska, Syahidah Asma Umniyati, and Helena Rodrigues Menezes de Oliveira Vaz. 2024. A multilingual parallel corpus for coreference resolution and information status in the literary domain. In *Proceedings of the 22nd Workshop on Treebanks and Linguistic Theories (TLT 2024)*, pages 55–64, Hamburg,Germany. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Jeanette K. Gundel. 1997. Centering theory and the givenness hierarchy: Towards a synthesis. In *Centering Theory In Discourse*. Oxford University Press.

Jeanette K. Gundel, Nancy Hedberg, and Ron Zacharski. 1993. Cognitive Status and the Form of Referring Expressions in Discourse. *Language*, 69(2):274–307. Publisher: Linguistic Society of America.

Michael Hahn, Judith Degen, and Richard Futrell. 2021. Modeling word and morpheme order in natural language as an efficient trade-off of memory and surprisal. *Psychological Review*, 128(4):726–756.

John Hale. 2001. A probabilistic Earley parser as a psycholinguistic model. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Carina Kauf and Anna Ivanova. 2023. A better way to do masked language model scoring. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Roger Levy. 2008. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177.

Roger Levy and T. Florian Jaeger. 2007. Speakers optimize information density through syntactic reduction. In Bernhard Schölkopf, John Platt, and Thomas Hofmann, editors, *Advances in Neural Information Processing Systems 19*, pages 849–856. The MIT Press.

Belinda Z. Li, Maxwell Nye, and Jacob Andreas. 2021. Implicit representations of meaning in neural language models. 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 1813–1827, Online. Association for Computational Linguistics.

Lei Liu and Min Zhu. 2023. Bertalign: Improved word embedding-based sentence alignment for Chinese–English parallel corpora of literary texts. *Digital Scholarship in the Humanities*, 38(2):621–634.

Katja Markert, Yufang Hou, and Michael Strube. 2012. Collective classification for fine-grained information status. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 795–804, Jeju Island, Korea. Association for Computational Linguistics.

Clara Meister, Tiago Pimentel, Patrick Haller, Lena Jäger, Ryan Cotterell, and Roger Levy. 2021. Revisiting the Uniform Information Density hypothesis. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 963–980, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ellen F. Prince. 1992. The ZPG Letter: Subjects, Definiteness, and Information-status. In William C. Mann and Sandra A. Thompson, editors, *Pragmatics & Beyond New Series*, volume 16, page 295. John Benjamins Publishing Company, Amsterdam.

C. E. Shannon. 1948. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3):379–423.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Anastasia Kozlova, Vladislav Mikhailov, and Tatiana Shavrina. 2024. mGPT: Few-shot learners go multilingual. *Transactions of the Association for Computational Linguistics*, 12:58–79.

Eleftheria Tsipidi, Franz Nowak, Ryan Cotterell, Ethan Wilcox, Mario Giulianelli, and Alex Warstadt. 2024. Surprise! Uniform Information Density isn't the whole story: Predicting surprisal contours in long-form discourse. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18820–18836, Miami, Florida, USA. Association for Computational Linguistics.

Annemarie Verkerk and Luigi Talamo. 2024. mini-CIEP+ : A shareable parallel corpus of prose. In *Proceedings of the 17th Workshop on Building and Using Comparable Corpora (BUCC) @ LREC-COLING 2024*, pages 135–143, Torino, Italia. ELRA and ICCL.

A Projection coverage on unlabeled data

Figure 4 shows POS-wise coverage of projected IS on the unlabeled portion of the corpus. We observe markedly lower coverage for languages such as Arabic, Armenian, Georgian, Irish, Kurmanji, Latin, Urdu, and Welsh, compared to several higher-resource languages. This pattern is consistent with noisier or sparser alignments (e.g., reduced lexical overlap and weaker cross-lingual representations), which results in fewer confidently aligned tokens and therefore fewer projected IS instances.

B Surprisal estimation

We calculated surprisal on all words that received an IS annotation with mGPT (Shliazhko et al., 2024) using the minicons[2] Python library (Kauf and Ivanova, 2023). For all languages we ran mGPT concurrently on 4 H100 GPUs, with a batch size of 128 per language and a fixed context size of $T = 256$. Code and data needed to run the experiments will be made available on GitHub upon publication. For each language, we calculate channel capacity $\hat{S}_{\mathcal{L}}$ as average surprisal over all words in that language's section of the corpus.

C Corpus statistics

Table 1 reports corpus statistics for all languages considered in this study. For Chinese, Greek, Hindi, Indonesian, Portuguese, Turkish, and Ukrainian, surprisal is computed using manually annotated Information Status labels. For the remaining nine languages, we rely on automatically projected annotations in order to broaden the typological and script diversity of the dataset.

[2] https://github.com/kanishkamisra/minicons

Language	IS (NOUN)	IS (PRON)	IS (PROPN)	#Tokens	#Sents
Human annotation					
Chinese	1,296	484	257	10,286	370
Greek	736	827	59	18,100	892
Hindi	2,558	3,281	558	31,591	1,853
Indonesian	1,331	568	227	8,923	540
Portuguese	4,955	1,647	763	46,607	2,803
Turkish	846	196	160	6,983	499
Ukrainian	1,140	791	211	11,257	838
Projection					
Albanian	11,465	3,449	1,956	133,690	6,401
Arabic	7,911	3,068	4	106,319	7,547
Basque	4,482	310	671	55,120	3,949
Finnish	11,101	5,483	1,755	101,193	6,169
Georgian	4,454	830	548	44,337	2,225
Irish	1,447	849	161	20,535	958
Japanese	14,650	2,758	1,793	199,575	6,637
Lithuanian	11,987	4,604	573	105,226	6,800
Persian	12,509	3,694	1,312	129,444	5,317
Swedish	12,322	10,004	1,975	123,708	5,970

Table 1: Corpus statistics by language. Rows are grouped by annotation type (human annotation vs. automatic projection). IS columns report the number of tokens annotated with Information Status for each POS category (noun, pronoun, proper noun). #Tokens and #Sents denote the total number of tokens and sentences in the corpus, respectively.

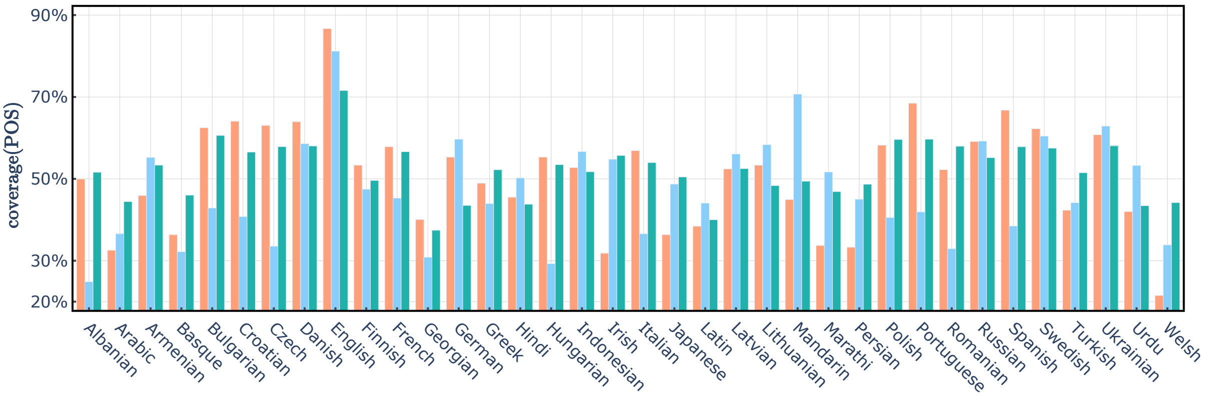


Figure 4: Percentage of **nouns**, **pronouns**, and **proper nouns** that received a POS tag through annotation projection, per language.

What does surprisal have to do with information status?

Andrew Dyer
Language Science and Technology
Saarland University, Germany
andrew.dyer@uni-saarland.de

Abstract

It is common in cognitive computational linguistics to use language model surprisal as a measure of the information content of units in language production. From here, it is tempting to then apply this to information structure and status, considering surprising mentions to be *new* and unsurprising ones to be given, providing us with a ready-made continuous metric of information givenness/newness. To see if this conflation is appropriate, we perform regression experiments to see if surprisal is actually well predicted by information status as manually annotated, and if so, if this effect is separable from more trivial linguistic information such as parts of speech and word frequency. We find that information status alone is at best a very weak predictor of surprisal, and that surprisal can be much better predicted by the effect of parts of speech, which are highly correlated with both information status and surprisal; and word frequency. We conclude that surprisal should not be used as a continuous representation of information status by itself.

1 Introduction

Language model surprisal – a measure of the unexpectedness of a following word in a sequence – is commonly used as a measure of the difficulty of processing language, and is surprisingly adaptable to a wide range of tasks (Goldstein et al., 2022), most notably reading times (de Varda and Marelli, 2022; Wilcox et al., 2023), among others. While the concept of surprisal is agnostic to the architecture of the language model used, increasingly neural, and particularly transformer-based language models are used.

Transformer-based language models are able to pay attention to previous context when making their next-word predictions (Vaswani et al., 2017). For this reason, they are ubiquitous in coreference and anaphora resolution, a task which requires connecting an often ambiguous, closed-class mention of a referent to its previously mentioned antecedent (Ogrodniczuk et al., 2025). This makes them promising in classification of information status – the givenness or newness of mentions and the referents they refer to (Chafe, 1976). Probes on transformer language models have also shown that their hidden representations can be used to predict information status, without finetuning for the task (Loáiciga et al., 2022).

With this in mind, it is tempting to then consider surprisal as a measure of that which which is "novel and unexpected" (Xu and Futrell, 2024), and this would fit with the conception of given information as that which is predictable or topical in context (Givón, 1983). It would certainly be intuitive that, if a mention refers to an entity that has been encountered in discourse previously, it should be less surprising for a language model that can encode this context.

In this study, we aim to test whether this intuition holds by measuring the extent to which surprisal is predicted by information status of mentions in a multilingual corpus. In doing so, we also examine whether information status itself is predicted by two morphosynactic cues: parts of speech and dependency relations, so that we can see whether the effect of information status rises to the surface in surprisal beyond the context-free information provided by these cues. We report results for nine languages: Chinese (Mandarin), English, German, Greek, Hindi, Indonesian, Portuguese, Turkish and Ukrainian.

2 Related work

Loáiciga et al. (2022) conducted a probing experiment on two English-language transformer-based language models (Transformer-XL and GPT2) to determine whether their parameters can be used to predict given- or newness of mentions. They found that hidden representations of tokens in contextual language models were indeed sufficient to clas-

March 29, 2026 ©2026 Association for Computational Linguistics

sify a mention (or mention head) as given or new, though they were less good at extracting mentions from text. They used a part-of-speech baseline in their study, whereby pronouns and definite noun phrases were considered given, and found that the systems regularly beat this baseline.

3 Experimental setup

3.1 Design

Our experiments are simple regression experiments to investigate the following:

1. First, to see whether information status itself is predicted by two syntactic features: universal parts of speech (UPOS) and dependency relations (deprel). We do this using logistic regression, with information status as the response variable and UPOS and deprel as explanatory variables. F1 score is reported in this experiment.
2. Second, to measure the extent to which information status predicts surprisal, independently of the two syntactic predictors (UPOS and deprel) and word frequency. R2 score is reported in this experiment.

In both experiments, we run analyses with different input features, including combinations of features, to see which contribute to the regression model and which can be removed without degrading the fit of the model.

3.2 Data

For our experiments, we use CiepInf (Dyer et al., 2024), a parallel corpus of modern prose built on top of the mini-CIEP+ corpus (Verkerk and Talamo, 2024), annotated into Universal Dependencies (Nivre et al., 2020) using UDPipe v.2 (Straka, 2018), and annotated for information status. The corpus is available in conllu format, and mention annotation follows the CorefUD format (Nedoluzhko et al., 2022). The corpus currently has annotated data in nine languages: English, Chinese (Mandarin), German, Greek, Hindi, Indonesian, Portuguese, Turkish, and Ukrainian. We show the data sizes in Table 1.

We extract mentions from CiepInf using Udapi (Popel et al., 2017). As mentions can have varying lengths, we extract surprisal of the syntactic head of the mention, rather than full spans.

	Sentences (approx)	Mentions New	Mentions Old
Chinese	3500	8294	7940
English	6380	18998	14560
German	560	810	1161
Greek	900	1361	2086
Hindi	3100	6963	3857
Indonesian	1880	3459	3176
Portuguese	2830	4252	4121
Turkish	580	714	813
Ukrainian	900	1310	1042

Table 1: Table of approximate number of sentences and mentions in CiepInf.

3.3 Resources

For all regressions, we use the sk-learn package in Python (Buitinck et al., 2013). For the first experiment (predicting information status from parts of speech and dependency relations) we use logistic regression. For the second experiment (predicting surprisal from given predictors) we use poisson regression, as surprisal is non-negative and has a long tailed distribution. In all cases, we apply the default L2 regularisation to constrain weights and reduce overfitting. We also use eight-fold cross-validation, and report averaged results across runs.

For surprisal, we use the Huggingface mGPT (Shliazhko et al., 2024) model[1], a multilingual model which covers all the languages that are of interest to us, and is autoregressive (i.e. unidirectional), which is suitable for measuring surprisal as the unexpectedness of following words. We tokenize using the mGPT tokenizer with a context size of 256 tokens, and a stride of 128. Because pretrained mGPT models are trained using byte-pair encoding, which produces token boundaries incongruous with Universal Dependencies token boundaries, we enforce byte-pair splitting only within UD token boundaries, and after inference merge these subtokens into their parent UD token, along with their surprisals, which are summed.

For word frequencies, we use the wordfreq package in Python (Speer, 2022) to get word frequencies, and specifically the *zipf_frequency* function to scale frequencies between languages. Since the Chinese lookup is not functional in this package at time of writing, we exclude Chinese from this part of the analysis.

[1] https://huggingface.co/ai-forever/mGPT

	UPOS	deprel	UPOS + deprel	UPOS ∗ deprel
Chinese	.84	.68	.84	.84
English	.82	.68	.82	.82
German	.98	.8	.98	.98
Greek	.88	.76	.88	.88
Hindi	.79	.43	.8	.8
Indonesian	.76	.71	.77	.77
Portuguese	.75	.69	.76	.76
Turkish	.78	.65	.78	.78
Ukrainian	.83	.65	.84	.84

Table 2: logistic regression scores (F1) of information status by UPOS and dependency relations. + means two features being used in a model independently, while ∗ means the interaction between the two of them.

4 Results

4.1 Experiment 1: Correlates of information status

Table 2 shows the F1 scores of the first experiment. We run regression models with different combinations of features: UPOS alone, deprel alone, UPOS and deprel as independent features, and the interaction between UPOS and deprel.

In all languages, we find that information status is well predicted by parts of speech and dependency relations; particularly the former. Dependency relations are a weaker predictor, and when combined with parts of speech in an interaction, the F1 is no higher, suggesting that this feature adds little information not already captured by parts of speech.

Figure 1 shows the weights (coefficients) of UPOS as predictors of surprisal between languages (from the analysis using only UPOS). Pronouns tend to be given, while nouns tend to be new. Proper nouns have a generally weak effect: they can be given or new. The trend is relatively consistent between our nine languages, though Greek shows an unusually higher tendency of proper nouns to signal new information.

So far, this is in line with the observation that pronouns, as reduced, closed class mentions, are indicative of given referents, and open class, full referring expressions are indicative of new ones (Gundel et al., 1993). It also tells us that when it comes to information status, there is a colinear effect of parts of speech that is hard to separate.

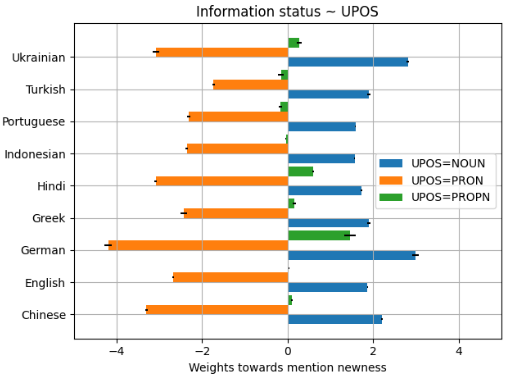


Figure 1: Weights towards information status newness by UPOS per language.

4.2 Experiment 2: Predictors of surprisal

Next, we explore the extent to which surprisal responds to information status as compared to other predictors, including parts of speech. Table 3 shows the results of the poisson regressions. Again, we run analyses with different features. The first two analyses are with UPOS and information status alone, respectively. The third is the interaction between information status and UPOS. The last two use word frequency: in a multiple model with the infstat-UPOS interaction, and alone, respectively.

We see that information status alone seems to have very little predictive power, shown by the low R2 scores. Once again, the effect it has is surpassed by UPOS, and there is little gain from the interaction of the two, telling us that surprisal responds much more to UPOS than information status. The strongest predictor of all appears to be word frequency, and it gains little from the interaction between parts of speech and information status being added to the model.

It is to be expected that a continuous variable (frequency) should provide the best fit to a continuous response variable (surprisal). But it is also remarkable that parts of speech alone are so predictive of surprisal, and considerably more so than information status.

5 Discussion

Our first finding – and one that is fairly intuitive – is that information status is itself well correlated with parts of speech. For example, given referents are very often pronominalised, while fully referring noun phrases are more frequently used for new referents (Gundel et al., 1993). This provides us with an intuitive baseline: if parts of speech are in-

	UPOS	infstat	infstat $*$ UPOS	infstat*UPOS + frequency	frequency
Chinese	.19	.03	.17	–	–
English	.17	.1	.17	.38	.38
German	.14	.16	.16	.36	.36
Greek	.09	.08	.09	.42	.42
Hindi	.21	.14	.21	.42	.42
Indonesian	.12	.09	.13	.22	.22
Portuguese	.02	.02	.03	.23	.23
Turkish	.03	.04	.03	.22	.22
Ukrainian	.17	.12	.19	.4	.4

Table 3: Poisson regression scores (R2) of surprisal by information status, UPOS, and frequency (and interactions). Once again, + means two features being used in a model independently, while $*$ means the interaction between the two of them. Chinese is excluded from all frequency experiments due to limitations in the Python package.

formative of mention givenness/newness, then any proposed or demonstrated sensitivity of surprisal to information status must be shown to be clearly independent from this.

As for surprisal, this also appears to be sensitive to parts of speech, but also, to simple word frequency. Both of these add predictive power to the regression model, but information status itself adds little. The response to parts of speech is intuitive to us as surprisal is, in the end, an output of the probabilities of next-word prediction, and where an open-class part of speech is expected there is a wider range of possible completions. The response to word frequency is also unsurprising from a language-modeling perspective, as language models have previously been shown to increasingly fit to this, to the detriment of fit to measures such as reading time (Oh et al., 2024).

6 Future work

There are baseline systems we could include in our study. For example, this study finds that information status is only weakly predictive of surprisal, despite transformer language models theoretically being able to encode long distance contexts. But do they at least outperform baselines such as LSTM language models, which have some ability to store previous context in a memory representation but no transformer architecture to "look back" at previous context; or a statistical language model such as a Kneser-Ney n-gram language model, which operates within a fixed window size? If the answer to this is *no*, then it appears even more inappropriate to consider surprisal from these language models to be reflective of information status.

We performed our analyses using simple linear regression models in Python, where feature interactions had to be manually programmed. But we see some variables which would be more appropriately modeled as mixed effects (for example, frequency could be grouped by UPOS), and a mixed effects regression may be more appropriate for this use case.

Finally, we may like to extend Loáiciga et al.'s probing experiment to the multilingual case and experiment more to see how well information status is encoded in the embeddings of large language models, separate from simple morphosyntactic cues. Information status may, as they found, be encoded in layers of the model, but not rise to the surface in surprisal, which is a single number derived only from the output layer.

7 Conclusion

Our experiments indicate that language model surprisal, while correlating well with many cognitive linguistic measures, is also well predicted by fairly mundane linguistic information such as parts of speech and word frequency, and only minimally by information status alone, making it problematic to use it as a stand-in for information status, which itself is also easily predicted by the same mundane syntactic information. This does not mean that information status is irrelevant to surprisal, but we do not see the effect shining through when simply looking at the surprisal of mentions. Though this may change with innovations in language models, we maintain that disentangling the effect of information status from that of more trivial and context-free linguistic cues is a must in evaluation.

Limitations

The generalisability of our study is limited by the use of a single language model and hyperparameter set. We originally set out to repeat this experiment over multiple context and stride sizes, but were unable to do this due to time and compute. Further experiments using different hyperparameters and/or models would be beneficial.

The imbalance between language data sizes in CiepInf is a problem, though the consistency of the results suggests that these findings are robust to data size.

Ethics Statement

We are unaware of any concrete harms towards individuals or communities arising as a result of our study.

We use a publicly available large language model, which we ran locally. We do not train or finetune this model. Our use of this model is within its intended use, and there is no possibility of personally identifiable information becoming available from our use, nor of harm to individuals or communities.

The corpus we use is available to share for the purposes of scientific study upon request, though it is not open-source due to copyrighted material contained within it.

Acknowledgments

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 232722074 – SFB 1102.

This work received support from the CA21167 COST action UniDive, funded by COST (European Cooperation in Science and Technology).

References

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Wallace L. Chafe. 1976. Givenness, contrastiveness, definiteness, subjects, topics, and point of view. In Charles N. Li, editor, *Subject and Topic*, pages 25–55. Academic Press, New York.

Andrea de Varda and Marco Marelli. 2022. The effects of surprisal across languages: Results from native and non-native reading. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 138–144, Online only. Association for Computational Linguistics.

Andrew Dyer, Ruveyda Betul Bahceci, Maryam Rajestari, Andreas Rouvalis, Aarushi Singhal, Syahidah Asma Umniyati Yuliya Stodolinska, and Helena Rodrigues Menezes de Oliveira Vaz. 2024. A multilingual parallel corpus for coreference resolution and information status in the literary domain. In *Proceedings of the 22nd International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2024)*, Hamburg, Germany. Association for Computational Linguistics.

Talmy Givón. 1983. Topic continuity in discourse: An introductions. *Topic continuity in discourse: A quantitative cross-language study.*

Ariel Goldstein, Zaid Zada, Eliav Buchnik, Mariano Schain, Amy Price, Bobbi Aubrey, Samuel A. Nastase, Amir Feder, Dotan Emanuel, Alon Cohen, Aren Jansen, Harshvardhan Gazula, Gina Choe, Aditi Rao, Catherine Kim, Colton Casto, Lora Fanda, Werner Doyle, Daniel Friedman, Patricia Dugan, Lucia Melloni, Roi Reichart, Sasha Devore, Adeen Flinker, Liat Hasenfratz, Omer Levy, Avinatan Hassidim, Michael Brenner, Yossi Matias, Kenneth A. Norman, Orrin Devinsky, and Uri Hasson. 2022. Shared computational principles for language processing in humans and deep language models. *Nature Neuroscience*, 25(3):369–380. Publisher: Nature Publishing Group.

Jeanette K. Gundel, Nancy Hedberg, and Ron Zacharski. 1993. Cognitive Status and the Form of Referring Expressions in Discourse. *Language*, 69(2):274–307. Publisher: Linguistic Society of America.

Sharid Loáiciga, Anne Beyer, and David Schlangen. 2022. New or old? exploring how pre-trained language models represent discourse entities. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 875–886, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Anna Nedoluzhko, Michal Novák, Martin Popel, Zdeněk Žabokrtský, Amir Zeldes, and Daniel Zeman. 2022. CorefUD 1.0: Coreference meets Universal Dependencies. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4859–4872, Marseille, France. European Language Resources Association.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Maciej Ogrodniczuk, Michal Novak, Massimo Poesio, Sameer Pradhan, and Vincent Ng, editors. 2025. *Proceedings of the Eighth Workshop on Computational Models of Reference, Anaphora and Coreference*. Association for Computational Linguistics, Suzhou, China.

Byung-Doh Oh, Shisen Yue, and William Schuler. 2024. Frequency explains the inverse correlation of large language models' size, training data amount, and surprisal's fit to reading times. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2644–2663, St. Julian's, Malta. Association for Computational Linguistics.

Martin Popel, Zdeněk Žabokrtský, and Martin Vojtek. 2017. Udapi: Universal API for Universal Dependencies. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 96–101, Gothenburg, Sweden. Association for Computational Linguistics.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Anastasia Kozlova, Vladislav Mikhailov, and Tatiana Shavrina. 2024. mGPT: Few-shot learners go multilingual. *Transactions of the Association for Computational Linguistics*, 12:58–79.

Robyn Speer. 2022. wordfreq: v3.0.

Milan Straka. 2018. UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Annemarie Verkerk and Luigi Talamo. 2024. mini-CIEP+ : A shareable parallel corpus of prose. In *Proceedings of the 17th Workshop on Building and Using Comparable Corpora (BUCC) @ LREC-COLING 2024*, pages 135–143, Torino, Italia. ELRA and ICCL.

Ethan G. Wilcox, Tiago Pimentel, Clara Meister, Ryan Cotterell, and Roger P. Levy. 2023. Testing the predictions of surprisal theory in 11 languages. *Transactions of the Association for Computational Linguistics*, 11:1451–1470.

Weijie Xu and Richard Futrell. 2024. Syntactic dependency length shaped by strategic memory allocation. In *Proceedings of the 6th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 1–9, St. Julian's, Malta. Association for Computational Linguistics.

A Appendix

Appendix material may be provided in the camera-ready submission.

Beyond Multilinguality: Typological Limitations in Multilingual Models for Meitei Language

Badal Nyalang
MWire Labs
Shillong, India
nyalang@mwirelabs.com

Abstract

We present MeiteiRoBERTa, the first publicly available monolingual RoBERTa-based language model for Meitei (Manipuri), a low-resource language spoken by over 1.8 million people in Northeast India. Trained from scratch on 76 million words of Meitei text in Bengali script, our model achieves a perplexity of 65.89, representing a 5.2× improvement over multilingual baselines mBERT (341.56) and MuRIL (355.65). We argue that Meitei's agglutinative morphology and complex word formation present typological challenges that multilingual models with broad language coverage fail to capture effectively. Through comprehensive evaluation on perplexity, tokenization efficiency, and semantic representation quality, we demonstrate that domain-specific pretraining significantly outperforms general-purpose multilingual models for low-resource languages. Our model exhibits superior semantic understanding with 0.769 similarity separation compared to 0.035 for mBERT and near-zero for MuRIL, despite MuRIL's better tokenization efficiency (fertility: 3.29 vs. 4.65). We publicly release the model, training code, and datasets to accelerate NLP research for Meitei and other underrepresented Northeast Indian languages.

1 Introduction

Meitei (also known as Manipuri) is an endangered Tibeto-Burman language spoken by approximately 1.8 million people, primarily in Manipur, India, and parts of Bangladesh and Myanmar. Despite being recognized as one of India's 22 scheduled languages and having a rich literary tradition spanning centuries, Meitei remains critically underrepresented in natural language processing research. The language exhibits complex agglutinative morphology, subject-object-verb word order, and can be written in both Meitei Mayek (indigenous script) and Bengali script, with the latter being more prevalent in digital contexts.

From a typological perspective, Meitei presents specific challenges for multilingual language models. As an agglutinative language, Meitei forms words through extensive suffixation and compounding, resulting in long, morphologically complex word forms that strain tokenization strategies optimized for isolating or mildly inflecting languages. Its SOV word order and predominantly head-final phrase structure differ from the SVO patterns dominant in many high-resource languages represented in multilingual models. These typological characteristics-shared with other Tibeto-Burman languages of the region-suggest that multilingual models trained predominantly on typologically distant languages may allocate insufficient capacity to capture Meitei's linguistic structure, motivating our investigation of monolingual alternatives.

Recent advances in transformer-based language models have revolutionized NLP across high-resource languages (Devlin et al., 2019; Liu et al., 2019). However, these benefits remain largely inaccessible to low-resource languages like Meitei due to their underrepresentation in multilingual models (Joshi et al., 2020; Ponti et al., 2020). While multilingual model such as mBERT (Devlin et al., 2019) and Indic-specific models like MuRIL (Khanuja et al., 2021) and IndicBERT (Kakwani et al., 2020) have attempted to address linguistic diversity, they often allocate insufficient capacity to individual low-resource languages, resulting in suboptimal performance (Conneau et al., 2020; Lauscher et al., 2020).

The critical challenge for endangered languages is not merely technological but existential: without adequate digital infrastructure and NLP tools, these languages risk accelerated decline in the digital age (Moseley, 2010) . Recent work has demonstrated that language-specific models trained from scratch can significantly outperform multilingual alternatives for low-resource languages, even with limited data (de Vries et al., 2019; Virtanen et al., 2019;

Proceedings of the 8th Workshop on Research in Computational Linguistic Typology and Multilingual NLP, pages 32–38
March 29, 2026 ©2026 Association for Computational Linguistics

Martin et al., 2020). This finding is particularly relevant for Northeast Indian languages, which collectively represent over 220 distinct languages but remain marginalized in mainstream NLP research.

In this work, we present MeiteiRoBERTa, a monolingual RoBERTa-based encoder model trained from scratch on 76 million words of Meitei text. Our contributions are threefold: (1) We release the first publicly available transformer-based language model specifically designed for Meitei in Bengali script, (2) We demonstrate through rigorous evaluation that our model achieves 5.2× better perplexity than multilingual baselines while exhibiting superior semantic understanding, and (3) We provide comprehensive comparative analysis against state-of-the-art multilingual models, offering insights for future low-resource language modeling efforts.

2 Related Work

2.1 Multilingual Language Models

The development of multilingual BERT (mBERT) (Devlin et al., 2019) marked a significant milestone in cross-lingual NLP, demonstrating that a single model could capture linguistic patterns across 104 languages. Subsequent work on XLM-R (Conneau et al., 2020) extended this to 100 languages with improved performance through larger-scale training. For Indian languages specifically, MuRIL (Khanuja et al., 2021) and IndicBERT (Kakwani et al., 2020) were introduced as specialized multilingual models covering 17 and 12 Indian languages respectively. However, recent studies have shown that these multilingual models suffer from the "curse of multilinguality", where increased language coverage leads to decreased per-language performance, particularly for low-resource languages (Pfeiffer et al., 2020; Üstün et al., 2020).

2.2 Low-Resource Language Models

Growing evidence suggests that monolingual models trained from scratch can outperform multilingual alternatives for low-resource languages (de Vries et al., 2019; Virtanen et al., 2019). Recent work on BanglaBERT (Bhattacharjee et al., 2022), AraBERT (Antoun et al., 2020), and CamemBERT (Martin et al., 2020) has demonstrated significant performance gains through language-specific pretraining. For Northeast Indian languages, preliminary efforts include work on Assamese (Nath et al., 2023) and limited experiments with Manipuri NER systems (Singh and Bandyopadhyay, 2009), but no comprehensive pre-trained language model for Meitei has been publicly released prior to this work.

2.3 RoBERTa Architecture

RoBERTa (Liu et al., 2019) introduced key improvements over BERT including dynamic masking, removal of next sentence prediction (NSP), larger batch sizes, and longer training sequences. These modifications have consistently demonstrated superior performance across various benchmarks (Clark et al., 2020; Lan et al., 2020). Recent adaptations of RoBERTa for low-resource languages (Nguyen and Nguyen, 2020; Agerri et al., 2020) have shown that the architecture's efficiency makes it particularly suitable for scenarios with limited computational resources and smaller corpora, motivating our choice of RoBERTa over alternative architectures.

3 Methodology

3.1 Model Architecture

MeiteiRoBERTa follows the RoBERTa-base architecture with 12 transformer layers, 12 attention heads, and a hidden dimension of 768, totaling 125 million parameters. We trained a custom Byte-Pair Encoding (BPE) tokenizer with a vocabulary size of 52,000 tokens, optimized specifically for Meitei morphology in Bengali script. The tokenizer was trained on the full corpus to minimize out-of-vocabulary rates and handle the language's agglutinative morphological structure efficiently.

3.2 Training Data and Preprocessing

Our training corpus comprises 76 million words of Meitei text, representing a curated aggregate of the IndicCorp v2 subset (Gala et al., 2023) and independent crawls of local news archives, government documents, digitized literary collections and web content. The corpus underwent rigorous preprocessing including deduplication, language identification filtering, and quality assessment. Data sources include publicly available digital archives, news portals, and literary collections. The data was chunked into 353,123 blocks of 512 tokens for efficient batch processing during training.

The dataset is publicly available at `https://huggingface.co/datasets/MWirelabs/meitei-monolingual-corpus`.

3.3 Training Configuration

The model was trained from random initialization using masked language modeling (MLM) with a 15% masking probability. We employed an effective batch size of 256, a learning rate of 6×10^{-4} with linear warmup and decay, and trained for 3 epochs. Training was conducted on NVIDIA A40 GPUs. The final training loss converged to 4.1855, indicating successful learning of Meitei language patterns.

4 Evaluation

4.1 Baselines

We compare MeiteiRoBERTa against three multilingual baselines: (1) mBERT (Devlin et al., 2019): 110M parameters covering 104 languages and (2) MuRIL (Khanuja et al., 2021): 235M parameters optimized for 17 Indian languages including transliterated text. These models represent the current state-of-the-art for multilingual NLP and are commonly used for low-resource Indian languages.

4.2 Evaluation Metrics

Perplexity. We evaluate language modeling capability using perplexity on a held-out validation set of 19,038 samples, calculated as PPL = exp(average loss), where the loss is computed through masked language modeling prediction accuracy.

Tokenization Efficiency. We measure subword fertility (average number of subword tokens per word) on a test set of 10 diverse Meitei sentences. Lower fertility indicates more efficient tokenization and better vocabulary alignment with the language's morphology.

Semantic Representation Quality. We assess the quality of learned representations using semantic similarity tests on 4 manually curated sentence pairs (2 semantically similar, 2 dissimilar) drawn from our test set. Sentence pairs were constructed to test the model's ability to distinguish between paraphrases with shared semantic content versus unrelated sentences from different domains. We compute cosine similarity between [CLS] token embeddings, which serve as sentence-level representations in BERT-style models, following standard practice for semantic similarity evaluation (Devlin et al., 2019). For each pair labeled as semantically similar or dissimilar, we measure the model's ability to separate semantically related from unrelated content through the separation score (average high similarity - average low similarity). The choice of [CLS] serves as a typological probe; if a model cannot distinguish basic sentence-level meaning in its pooling layer, it indicates a fundamental failure to map the language's syntax and morphology into a coherent semantic manifold

Model	Parameters	Perplexity
mBERT	110M	341.56
MuRIL	235M	355.65
MeiteiRoBERTa	125M	65.89

Table 1: Perplexity comparison on Meitei validation set (19,038 samples). Lower is better.

Model	Vocab Size	Fertility
mBERT	119K	4.79
MuRIL	197K	3.29
MeiteiRoBERTa	52K	4.65

Table 2: Tokenization efficiency on 10 diverse Meitei sentences. Lower fertility is better.

5 Results and Analysis

5.1 Perplexity Comparison

Table 1 presents the perplexity results on the Meitei validation set. MeiteiRoBERTa achieves a perplexity of 65.89, substantially outperforming mBERT (341.56) and MuRIL (355.65) by factors of 5.2× and 5.4× respectively. This dramatic improvement demonstrates that monolingual pre-training with language-specific tokenization provides superior language modeling capabilities compared to general-purpose multilingual models, even when the latter have significantly more parameters.

5.2 Tokenization Efficiency

Table 2 shows tokenization efficiency measured by subword fertility. MuRIL achieves the best fertility score of 3.29, followed by mBERT (4.79) and MeiteiRoBERTa (4.65). While MuRIL's extensive vocabulary (197K tokens) enables more efficient tokenization, our custom BPE tokenizer with 52K tokens achieves competitive performance while maintaining a more compact model size. The trade-off between vocabulary size and model compactness is favorable for resource-constrained deployment scenarios.

5.3 Semantic Representation Quality

Table 3 presents semantic similarity evaluation results. MeiteiRoBERTa demonstrates exceptional

Model	High Sim	Low Sim	Sep.
mBERT	0.983	0.948	0.035
MuRIL	0.993	0.993	0.000
MeiteiRoBERTa	0.968	0.199	0.769

Table 3: Semantic representation quality on curated Meitei sentence pairs. Higher separation indicates better semantic understanding.

semantic understanding with a separation score of 0.769, vastly outperforming mBERT (0.035) and MuRIL (0.000). The near-perfect similarity scores (>0.95) from multilingual models for both related and unrelated sentence pairs indicate their failure to capture fine-grained semantic distinctions in Meitei. In contrast, MeiteiRoBERTa shows high similarity (0.968) for semantically related pairs while correctly assigning low similarity (0.199) to unrelated pairs, demonstrating genuine semantic comprehension.

The 0.000 separation score for MuRIL may indicate limited semantic differentiation under the current probing setup, where the model's pre-training on Indo-Aryan languages fails to provide the structural handles necessary to differentiate Meitei's Tibeto-Burman semantic features.

6 Discussion

Our results provide strong empirical evidence for the superiority of monolingual language models over multilingual alternatives for low-resource languages. The 5.2× perplexity improvement and dramatically better semantic separation (0.769 vs. 0.035) demonstrate that dedicated models can capture language-specific nuances that multilingual models miss, even when the latter have 2-3× more parameters.

The stark contrast in semantic representation quality is particularly noteworthy. MuRIL's near-zero separation score (0.000) suggests that despite its optimization for Indian languages, it shows limited ability to differentiate semantic contrasts in Meitei under our evaluation setting, assigning uniformly high similarity scores regardless of semantic content. This may indicate representation collapse or insufficient training signal for Meitei, aligning with recent findings on catastrophic interference in massively multilingual models (Artetxe et al., 2020), where low-resource languages receive insufficient training signal to develop meaningful representations.

While MuRIL demonstrates superior tokenization efficiency (fertility: 3.29 vs. our 4.65), reflecting the benefits of its larger vocabulary (197K tokens) optimized across multiple Indic languages, this advantage does not translate to better language understanding in our evaluation. This finding suggests a more nuanced relationship between tokenization efficiency and semantic representation quality than previously assumed. Efficient tokenization may be necessary but not sufficient for low-resource language support-dedicated model capacity and language-specific pre-training appear essential for developing robust linguistic representations. The trade-off between vocabulary size and model compactness remains an important consideration, particularly for deployment in resource-constrained environments where MeiteiRoBERTa's smaller vocabulary offers practical advantages.

The success of MeiteiRoBERTa with only 76 million words of training data (compared to billions used by multilingual models) has important implications for other low-resource languages. It demonstrates that even modest-sized corpora can yield high-quality language models when combined with appropriate architecture and training strategies. This is particularly encouraging for the hundreds of endangered languages worldwide that face similar resource constraints.

6.1 Implications for Underserved Communities

Our work directly addresses the digital divide affecting Northeast Indian linguistic communities. By providing the first comprehensive language model for Meitei, we enable potential applications in education, digital governance, cultural preservation, and content moderation. The model can support machine translation, sentiment analysis, and information retrieval systems that were previously unavailable for Meitei speakers. This technological infrastructure is crucial for preventing language endangerment in the digital age and ensuring equitable access to AI technologies.

From a typological perspective, our results highlight systematic limitations in how multilingual models handle morphologically complex, agglutinative languages. The dramatic performance gap suggests that current multilingual pretraining approaches-which distribute model capacity across typologically diverse languages-may inherently disadvantage languages with rich morphology and head-final syntax. This has broader implications for the approximately 220 languages of Northeast

India and hundreds of other agglutinative languages worldwide that share similar typological profiles with Meitei.

7 Conclusion and Future Work

We have presented MeiteiRoBERTa, the first publicly available transformer-based language model for Meitei, achieving state-of-the-art performance with 5.2× better perplexity than multilingual baselines and superior semantic understanding (0.769 separation vs. 0.035 for mBERT). Our comprehensive evaluation demonstrates that dedicated monolingual models remain the most effective approach for low-resource language processing, even in the era of massively multilingual models.

Future work will focus on several directions: (1) extending support to Meitei Mayek script through script-agnostic representations or dual-script training, (2) fine-tuning for downstream tasks including named entity recognition, sentiment analysis, and machine translation, (3) exploring few-shot learning capabilities for related Tibeto-Burman languages, and (4) investigating cross-lingual transfer learning between Bengali-script Meitei and other languages using the same script. Most importantly, we aim to establish partnerships with Meitei language communities to ensure our future work aligns with community priorities and contributes meaningfully to language preservation efforts.

The model (https://huggingface.co/MWirelabs/meitei-roberta) and evaluation datasets (https://huggingface.co/datasets/MWirelabs/meitei-monolingual-corpus) are publicly available to support further research on Meitei and other low-resource Northeast Indian languages.

These findings suggest potential typological limitations in multilingual models when applied to morphologically complex, agglutinative languages such as Meitei, with possible implications for other underrepresented Tibeto-Burman and related agglutinative languages.

Limitations

While our work represents a significant advance for Meitei NLP, several limitations warrant discussion:

Script Coverage: Our model only supports Bengali script, excluding the indigenous Meitei Mayek script still used in some contexts. Future work should explore multi-script models or script-agnostic representations.

Evaluation Scope: We primarily evaluate intrinsic metrics (perplexity, tokenization, semantic similarity). Downstream task evaluation on named entity recognition, sentiment analysis, and machine translation would provide additional insights into practical utility.

Corpus Limitations: Our 76M word corpus, while substantial for a low-resource language, may not capture the full linguistic diversity of Meitei, including dialectal variations and specialized domains. The corpus is also biased toward formal written text from news and government sources.

Community Involvement: Data collection and model development occurred without extensive consultation with Meitei language communities. Future efforts should employ participatory methods to ensure alignment with community needs and values.

Potential Biases: The model may inherit biases present in the training corpus, including underrepresentation of certain demographics, topics, or perspectives. We have not yet conducted comprehensive bias audits.

Computational Resources: While more efficient than training multilingual models, our approach still requires substantial computational resources (GPU training), which may limit reproducibility for researchers with limited access.

Ethical Considerations

Data Sourcing: All training data was collected from publicly available sources. We employed language identification and quality filtering to ensure corpus integrity, but acknowledge that web-scraped data may contain errors, biases, or copyrighted material despite our filtering efforts.

Cultural Sensitivity: Language technology for endangered languages carries responsibility for cultural preservation. While our model enables digital applications, we recognize that technology alone cannot address underlying sociolinguistic challenges. Community-led language revitalization efforts remain paramount.

Dual Use: Like all language models, MeiteiRoBERTa could potentially be misused for generating misinformation, impersonation, or surveillance. We advocate for responsible deployment and recommend implementing appropriate safeguards in downstream applications.

Access and Equity: By releasing our model, code, and data publicly, we aim to democratize access to NLP tools for Meitei. However, meaningful access requires not just open models but also computational resources, technical expertise, and internet connectivity; resources unevenly distributed in Northeast India.

Acknowledgements

We thank the SIGTYP 2026 reviewers for their constructive feedback, which significantly improved the typological framing and clarity of this work. The author would like to acknowledge the **AI4Bharat** team for the **IndicCorp v2** initiative (Gala et al., 2023), which provided a vital foundational dataset for this study. This research was conducted under the research and development initiative at **MWire Labs**, Shillong. We also thank the digital archivists of Northeast India whose efforts in preserving online Meitei texts provided the primary signal for our monolingual pre-training.

References

Rodrigo Agerri, Iñaki San Vicente, Jon Ander Campos, Ander Barrena, Xabier Saralegi, Aitor Soroa, and Eneko Agirre. 2020. Give your text representation models some love: the case for basque. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*, pages 4781–4788.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT)*, pages 9–15.

Mikel Artetxe, Sebastian Ruder, Dani Yogatama, Gorka Labaka, and Eneko Agirre. 2020. A call for more rigor in unsupervised cross-lingual learning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7375–7388.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations (ICLR)*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 8440–8451.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. BERTje: A Dutch BERT model. *arXiv preprint arXiv:1912.09582*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186.

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. 2023. IndicTrans2: Towards high-quality and accessible machine translation models for all 22 scheduled Indian languages. *Transactions on Machine Learning Research*. Survey of IndicCorp v2 coverage.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6282–6293.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. MuRIL: Multilingual representations for Indian languages. *arXiv preprint arXiv:2103.10730*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *International Conference on Learning Representations (ICLR)*.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual transformers. In *Proceedings of the 2020*

Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4483–4499.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: A tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7203–7219.

Christopher Moseley. 2010. *Atlas of the World's Languages in Danger*, 3rd edition. UNESCO Publishing.

Abhijnan Nath, Sheikh Mannan, and Nikhil Krishnaswamy. 2023. AxomiyaBERTa: A phonologically-aware transformer model for Assamese. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11629–11646, Toronto, Canada. Association for Computational Linguistics.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An adapter-based framework for multi-task cross-lingual transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376.

Thoudam Doren Singh and Sivaji Bandyopadhyay. 2009. Named entity recognition for Manipuri using support vector machine. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC)*, pages 811–818.

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. 2020. UDapter: Language adaptation for truly universal dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2302–2315.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: BERT for Finnish. *arXiv preprint arXiv:1912.07076*.

A RAG Approach for Typological Database Completion

Jonathan Hus, Antonios Anastasopoulos
Department of Computer Science, George Mason University
jhus@gmu.edu, antonis@gmu.edu

Abstract

Linguistic reference material is a trove of information that can be utilized for the analysis of languages. The material, in the form of grammar books and sketches, has been used for machine translation, but it can also be used for language analysis. Retrieval Augmented Generation (RAG) has been demonstrated to improve large language model (LLM) capabilities by incorporating external reference material into the generation process. In this paper, we investigate the use of grammar books and RAG techniques to identify language features. We use Grambank for feature definition and ground truth values, and we evaluate on five typologically diverse low-resource languages. We demonstrate that this approach can effectively make use of reference material.[1]

1 Introduction

Grambank is an online database of language features and contains information on more than 2000 languages (Skirgård et al., 2023a). The database is structured around 195 language features. For each language in the database, the features are coded with a value, covering a wide spectrum of typological information such as subject-verb-object order, definite and indefinite article usage, and morphological markings (Skirgård et al., 2023b). This data is intended to be used for linguistic research; for example, Skirgård et al. (2023b) analyzed the data to make an argument for the significance of genealogical inheritance on language diversity.

The database, similar to others in the past like WALS (Dryer and Haspelmath, 2013), is hand-coded by a team of contributors. Questionnaires are provided to coders that contain the feature name, description, possible values, among other information. The intention is that coders can code the features using grammars or grammar books, even if they are not experts in that language.

While Grambank has feature information on more than 2000 languages, that leaves almost 5000 languages that are missing from the database. Questionnaires for a language are typically provided to a single contributor and completing the remainder of the database will require significant human capital. With the database missing data, researchers will not be able to utilize this resource to answer questions about certain languages or language families, nor will they be able to have the broadest possible coverage of languages when making general analyses (e.g., genealogical inheritance vs geographical diffusion in shaping language diversity).

In order to fill this gap, we propose a Retrieval-Augmented Generation (RAG) approach using a Large Language Model (LLM) that can query linguistic reference material. Since the questionnaire is intended to be completed by non-experts using grammar books and sketches, a sufficiently-capable LLM with access to the correct resources may be able to provide feature codes for languages not present in Grambank.

2 Related Work

RAG has become a popular approach applied to many applications of LLMs (Gao et al., 2024). The technique enables the incorporation of external knowledge, stored in a database, into the LLM input context.

Kornilov and Shavrina (2024) built a benchmark to evaluate RAG approaches, with Grambank and WALS feature prediction as targets. In contrast, our work more closely mimicks the process of Gramank coders' efforts, covering all Grambank features.

Hammarström et al. (2020) used term spotting to extract linguistic information from digitized raw-text descriptions, and Virk et al. (2021) describe

[1] Code and data to reproduce our experiments are provided here: https://github.com/jonathanhus/gram_features.

an automated deep learning approach to extract linguistic features from textual language descriptions.

The use of full-length grammar books has previously been explored when constructing prompts for LLMs. In Machine Translation from One Book, Tanzer et al. (2023) used a grammar book to perform machine translation between English and Kalamang, and more recently Hus and Anastasopoulos (2024) expanded this translation approach to 15 additional low-resource languages.

3 Methodology

Our RAG architecture consists of two main components: ChromaDB for the vector database and GPT-4o for the LLM.

The vector database is organized into collections, where a collection represents a single language. When loading the vector database, we chose the grammar books that are referenced and used by the Grambank coders when they coded language features. Some languages use a single grammar book while others use multiple. We obtained our grammar books from the DReaM corpus (Virk et al., 2020), which contains digitized versions of thousands of linguistic documents. These grammar books are vectorized and loaded into ChromaDB, with each page stored as a single entry.

Each feature in Grambank has a number of attributes that we use to format our LLM prompts. Each feature is written in the form of a question. Additionally, a summary is provided, which contains amplifying information, and last we provide instructions on how to label the feature. For each feature in Grambank, the attributes are used to create a query, which returns the top N similar entries from the vector database. The Grambank attributes are also used to formulate the prompt for the LLM. The database entries, which ideally contain the relevant grammatical information, are included in the prompt. An example prompt is illustrated in Appendix C. As indicated in the prompt, the LLM outputs the predicted feature code.

The full pipeline, therefore, operates as follows. Feature information is obtained from Grambank files and is used to query for relevant grammar book pages from ChromaDB. These retrieved pages, as well as task information from the Grambank files, are then formatted into a prompt for an LLM.

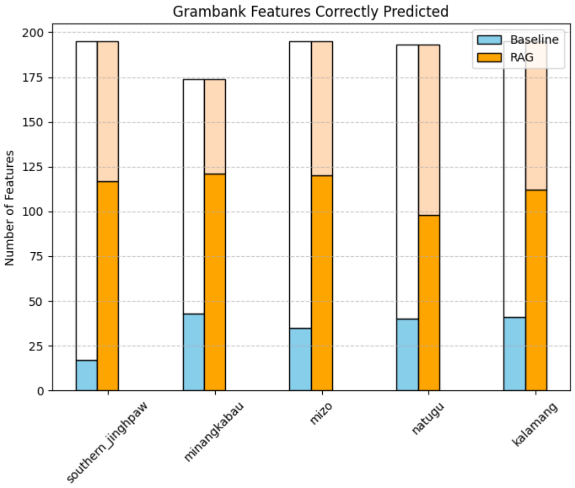


Figure 1: Baseline vs RAG Feature Prediction. RAG approach shows significant improvement over the baseline in identifying grammar features.

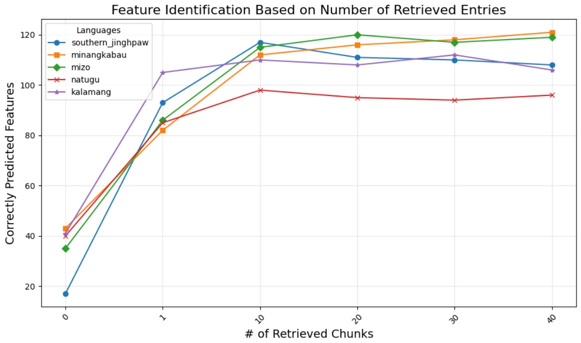


Figure 2: Number of correctly predicted features based on number of retrieved pages that are included in the prompt. Considerable improvements observed when including up to 10 pages in the prompt.

4 Results

4.1 Baseline vs RAG

The first basic question is whether RAG with grammar books improves the performance compared to simply asking the LLM for an answer. We created prompts that both did and did not contain the retrieved document pages from the database. Baseline values (Figure 1) show the performance with no retrieved documents included. RAG values show the number of features correctly predicted when including retrieved pages in the prompt. For all languages, RAG showed a marked improvement over the baseline, with increases of 30% to 52%, indicating the effectiveness of this approach.

4.2 Document Retrieval

Given the performance boost obtained from incorporating RAG, a natural extension would be to include more excerpts from grammar books to see

Language	Sources	Features w/ Pages	≥1 Match	Mean	Max	Skyline	Full Pipeline
Mizo (lus)						120/170	
– without summary	1	170	90	0.125	0.750		
– with summary	1	170	92	0.120	0.667		103/170
Southern Jinghpaw (kac)						112/195	
– without summary	3	195	134	0.143	0.571		
– with summary	3	195	132	0.144	0.500		117/195
Kalamang (kgv)						40/58	
– without summary	1	58	38	0.177	0.500		
– with summary	1	58	39	0.212	0.600		41/58
Minangkabau (min)						91/111	
– without summary	3	111	46	0.078	0.500		
– with summary	3	111	49	0.083	0.400		79/111
Natugu (ntu)						39/70	
– without summary	6	70	25	0.042	0.333		
– with summary	6	70	30	0.57	0.333		53/70

Table 1: Pipeline analysis characterizing the performance of the vector database search, the LLM prediction (skyline), and the full pipeline.

if that further increases performance. For each language, we performed experiments that included an increasing number of pages in the prompt (Figure 2), up to 40. Our results are consistent with other published results (Leng et al., 2024) that show a performance increase as more documents are added. However, more documents give diminishing and eventually negative returns.

4.3 Pipeline Analysis

The system components, namely the database and the LLM, are largely independent. Their performance can be separately assessed to analyze their impact on the system. The experiments and results in this section only look at the features for a given language that have page numbers identified.

Vector Database When coders are specifying feature values for a language in Grambank, they are encouraged to include both the reference book and page number(s) used to determine that value. We can use this information to evaluate the performance of our vector database retrieval. For the features with pages numbers identified, we evaluate the similarity with the pages retrieved via vector search. For each language, Table 1 shows the number of reference sources that the Grambank coders used to determine the feature values in the "Sources" column. Since coders do not always provide page numbers with a feature, we evaluate similarity only on the features that have page numbers. The feature, which is expressed in the form of a question, is always included in the search term. We evaluated both including and omitting the summary in the search, shown in the "with summary" and "without summary" rows, respectively.

For each feature, we retrieve four documents from the database. We calculate the Jaccard similarity between this set of documents and the ground truth provided by Grambank. Within a language, we calculate the mean and maximum of all the similarity scores, shown in the "Mean" and "Max" columns of Table 1, respectively. Not surprisingly, languages with more source documents have lower average similarity scores. While this could be due to a larger search space, it is also possible that the correct answer can be found in a document other than what the coder chose. As a coarser indication of similarity, we evaluate the number of features where at least one of the retrieved pages matches the ground truth, shown in the "≥1 Match" column.

Overall, the RAG approach shows a good deal of agreement between the database search and human coding, with more than 50% of features across all languages having at least one matching page. In general, including the summary in the prompt improves the similarity scores, although the mean similarity score is lower in Mizo when using the summary and the number of features with at least one match is lower in Southern Jinghpaw. Natugu has the worst similarity scores among all the languages, which may be attributed to it also having the most source documents.

Main domain	kac	min	lus	ntu	kgv
Clause	52.0 (26/50)	51.06 (24/47)	62.0 (31/50)	48.0 (24/50)	60.0 (30/50)
Nominal domain	64.79 (46/71)	81.25 (52/64)	61.97 (44/71)	50.0 (35/70)	67.61 (48/71)
Numeral	50.0 (2/4)	75.0 (3/4)	100.0 (4/4)	25.0 (1/4)	25.0 (1/4)
Pronoun	83.33 (10/12)	72.73 (8/11)	33.33 (4/12)	50.0 (6/12)	58.33 (7/12)
Verbal domain	56.9 (33/58)	70.83 (34/48)	63.79 (37/58)	56.14 (32/57)	44.83 (26/58)
Total	60.0 (117/195)	69.54 (121/174)	61.54 (120/195)	50.78 (98/193)	57.44 (112/195)

Table 2: Match accuracy by main domain for each language. Values are formatted as percentage (correct/total).

	LLM output code						
label	**0**	**1**	**2**	**3**	**?**	**IDK**	**Total**
0	312	94	2	0	76	11	495
1	29	144	2	1	24	1	201
2	0	2	9	3	2	1	17
?	84	27	2	0	103	23	239
Total	425	267	15	4	205	36	952

Table 3: Aggregate distribution of code values by category for all languages. We highlight correct predictions.

Full Pipeline vs Skyline To evaluate how well the LLM uses the provided material to determine the feature value, we bypassed the database search and instead directly included the referenced pages in the prompt. As with the vector database characterization, we are only able to utilize this approach for features that have associated reference page numbers in the Grambank dataset. The "Skyline" column in Table 1 shows the number of features correctly predicted. We also evaluate the full pipeline (database and LLM) on the set of features that have reference pages, similar to above. Surprisingly, comparing to the skyline approach, the full pipeline sometimes but not always outperforms skyline. Natugu shows the largest increase from the skyline to the full pipeline. This may be attributed to the increased number of grammar sources, which could increase the amount of highly-relevant excerpts included in the prompt.

4.4 Feature Analysis

Confusion matrices are provided in the appendix for each language, with an aggregate shown in Table 3. When it comes to errors, the LLM is more likely to predict a feature is present when in fact it is absent than to predict a feature is absent when in fact it is present. Such false positives are the most common errors observed in the data. The next highest error categories are a true code "0" (absent feature) with a predicted "?" (unknown) and a true code "?" with a predicted code "0". Both of these situations involve using information from the grammar books to prove that something does not exist. Unless the grammar books explicitly state that some aspect of a grammar is not present in a particular language, the LLMs will struggle to identify excerpts that support this prediction.

The prompt includes an instruction for the LLM to output "IDK", refraining from a prediction if it is unable to determine a code value. In some senses, this is duplicative with the "?" value, but it does provide the LLM with an option instead of a forced guess. That being said, the LLM only returned "IDK" 36 times (3.8%). There were four instances (0.4%) in which the LLM did not conform to the answer options specified in the task instructions and instead chose a "3", which is not a valid choice for any feature.

Grambank categorizes each feature into one of five domains. The domains are broad linguistic topics that are designed to enable further analysis. For the best performing RAG system for each language, the performance for each domain is shown in Table 2. Across all languages, our RAG system performs the best on the "Nominal domain" category, with an accuracy of 65%, while it struggles the most with the "Clause" domain, with an accuracy of 55%. This suggests that reference material about nominal domain features (e.g., case marking, articles, possession) is able to be extracted easier from the grammar books than text about clause domain features (e.g., sentence structure, subordination, coordination).

5 Conclusion

In this paper, we showed the benefit of incorporating grammar books into RAG pipelines. We evaluated the performance on several languages and showed that this approach is capable of answering questions about typological structures of languages. Our work spotlights one application where this approach yields benefits and shows that linguistic reference material is a valuable resource for research on low resource languages.

6 Limitations

Grammar books are vectorized and stored in a database for RAG. In order to take advantage of this approach, we use grammar books in PDF format that contain text. However, high-quality grammar books are difficult to obtain for many languages. The DReaM corpus does an admirable job of curating and digitizing many linguistic references, but not all languages have reference material in the necessary format. Additionally, tables lose information that is conveyed by the location of text relative to other text on the page. The LLMs, therefore, are most likely not taking full advantage of that information.

We used an OpenAI model (gpt-4o-mini). While this model is quite performant, there are some drawbacks. OpenAI models are truly closed models, with only an API available. The architecture, weights, and training scheme are not available to researchers.

Acknowledgements

This work was partially supported by the National Science Foundation under CAREER award 2439202. This work was partially supported by resources provided by the Office of Research Computing at George Mason University (URL: https://orc.gmu.edu) and funded in part by grants from the National Science Foundation (Award Number 2018631).

References

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Harald Hammarström, One-Soon Her, and Marc Allassonnière-Tang. 2020. Term spotting: A quick-and-dirty method for extracting typological features of language from grammatical descriptions. In *Selected contributions from the Eighth Swedish Language Technology Conference (SLTC-2020)*, pages 27–34, Göteborg, Sweden.

Jonathan Hus and Antonios Anastasopoulos. 2024. Back to school: Translation using grammar books. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20207–20219, Miami, Florida, USA. Association for Computational Linguistics.

Albert Kornilov and Tatiana Shavrina. 2024. From mteb to mtob: Retrieval-augmented classification for descriptive grammars. *Preprint*, arXiv:2411.15577.

Quinn Leng, Jacob Portes, Sam Havens, Matei Zaharia, and Michael Carbin. 2024. Long context rag performance of large language models. *Preprint*, arXiv:2411.03538.

Hedvig Skirgård, Hannah J. Haynie, Damián E. Blasi, Harald Hammarström, Jeremy Collins, Jay J. Latarche, Jakob Lesage, Tobias Weber, Alena Witzlack-Makarevich, Sam Passmore, Angela Chira, Luke Maurits, Russell Dinnage, Michael Dunn, Ger Reesink, Ruth Singer, Claire Bowern, Patience Epps, Jane Hill, Outi Vesakoski, Martine Robbeets, Noor Karolin Abbas, Daniel Auer, Nancy A. Bakker, Giulia Barbos, Robert D. Borges, Swintha Danielsen, Luise Dorenbusch, Ella Dorn, John Elliott, Giada Falcone, Jana Fischer, Yustinus Ghanggo Ate, Hannah Gibson, Hans-Philipp Göbel, Jemima A. Goodall, Victoria Gruner, Andrew Harvey, Rebekah Hayes, Leonard Heer, Roberto E. Herrera Miranda, Nataliia Hübler, Biu Huntington-Rainey, Jessica K. Ivani, Marilen Johns, Erika Just, Eri Kashima, Carolina Kipf, Janina V. Klingenberg, Nikita König, Aikaterina Koti, Richard G. A. Kowalik, Olga Krasnoukhova, Nora L. M. Lindvall, Mandy Lorenzen, Hannah Lutzenberger, Tânia R. A. Martins, Celia Mata German, Suzanne van der Meer, Jaime Montoya Samamé, Michael Müller, Saliha Muradoglu, Kelsey Neely, Johanna Nickel, Miina Norvik, Cheryl Akinyi Oluoch, Jesse Peacock, India O. C. Pearey, Naomi Peck, Stephanie Petit, Sören Pieper, Mariana Poblete, Daniel Prestipino, Linda Raabe, Amna Raja, Janis Reimringer, Sydney C. Rey, Julia Rizaew, Eloisa Ruppert, Kim K. Salmon, Jill Sammet, Rhiannon Schembri, Lars Schlabbach, Frederick W. P. Schmidt, Amalia Skilton, Wikaliler Daniel Smith, Hilário de Sousa, Kristin Sverredal, Daniel Valle, Javier Vera, Judith Voß, Tim Witte, Henry Wu, Stephanie Yam, Jingting Ye, Maisie Yong, Tessa Yuditha, Roberto Zariquiey, Robert Forkel, Nicholas Evans, Stephen C. Levinson, Martin Haspelmath, Simon J. Greenhill, Quentin D. Atkinson, and Russell D. Gray. 2023a. Grambank reveals the importance of genealogical constraints on linguistic diversity and highlights the impact of language loss. *Science Advances*, 9(16):eadg6175.

Hedvig Skirgård, Hannah J. Haynie, Harald Hammarström, Damián E. Blasi, Jeremy Collins, Jay Latarche, Jakob Lesage, Tobias Weber, Alena Witzlack-Makarevich, Michael Dunn, Ger Reesink, Ruth Singer, Claire Bowern, Patience Epps, Jane Hill, Outi Vesakoski, Noor Karolin Abbas, Sunny Ananth, Daniel Auer, Nancy A. Bakker, Giulia Barbos, Anina Bolls, Robert D. Borges, Mitchell Browen, Lennart Chevallier, Swintha Danielsen, Sinoël Dohlen, Luise Dorenbusch, Ella Dorn, Marie Duhamel, Farah El Haj Ali, John Elliott, Giada Falcone, Anna-Maria Fehn, Jana Fischer, Yustinus Ghanggo Ate, Hannah

Gibson, Hans-Philipp Göbel, Jemima A. Goodall, Victoria Gruner, Andrew Harvey, Rebekah Hayes, Leonard Heer, Roberto E. Herrera Miranda, Nataliia Hübler, Biu H. Huntington-Rainey, Guglielmo Inglese, Jessica K. Ivani, Marilen Johns, Erika Just, Ivan Kapitonov, Eri Kashima, Carolina Kipf, Janina V. Klingenberg, Nikita König, Aikaterina Koti, Richard G. A. Kowalik, Olga Krasnoukhova, Kate Lynn Lindsey, Nora L. M. Lindvall, Mandy Lorenzen, Hannah Lutzenberger, Alexandra Marley, Tânia R. A. Martins, Celia Mata German, Suzanne van der Meer, Jaime Montoya, Michael Müller, Saliha Muradoğlu, HunterGatherer, David Nash, Kelsey Neely, Johanna Nickel, Miina Norvik, Bruno Olsson, Cheryl Akinyi Oluoch, David Osgarby, Jesse Peacock, India O.C. Pearey, Naomi Peck, Jana Peter, Stephanie Petit, Sören Pieper, Mariana Poblete, Daniel Prestipino, Linda Raabe, Amna Raja, Janis Reimringer, Sydney C. Rey, Julia Rizaew, Eloisa Ruppert, Kim K. Salmon, Jill Sammet, Rhiannon Schembri, Lars Schlabbach, Frederick W. P. Schmidt, Dineke Schokkin, Jeff Siegel, Amalia Skilton, Hilário de Sousa, Kristin Sverredal, Daniel Valle, Javier Vera, Judith Voß, Daniel Wikalier Smith, Tim Witte, Henry Wu, Stephanie Yam, Jingting Ye, Maisie Yong, Tessa Yuditha, Roberto Zariquiey, Robert Forkel, Nicholas Evans, Stephen C. Levinson, Martin Haspelmath, Simon J. Greenhill, Quentin D. Atkinson, and Russell D. Gray. 2023b. Grambank v1.0. Dataset.

Garrett Tanzer, Mirac Suzgun, Eline Visser, Dan Jurafsky, and Luke Melas-Kyriazi. 2023. A benchmark for learning to translate a new language from one grammar book. In *Arxiv*.

Shafqat Mumtaz Virk, Daniel Foster, Azam Sheikh Muhammad, and Raheela Saleem. 2021. A deep learning system for automatic extraction of typological linguistic information from descriptive grammars. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1480–1489, Held Online. INCOMA Ltd.

Shafqat Mumtaz Virk, Harald Hammarström, Markus Forsberg, and Søren Wichmann. 2020. The DReaM corpus: A multilingual annotated corpus of grammars for the world's languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 878–884, Marseille, France. European Language Resources Association.

A Additional Results

Tabular results for Figure 2 are provided in Table 4.

Confusion matrices for each of the languages are combined into an aggregate table in section 4. The results for each language is provided here.

Results mapped to Grambanks' "finer grouping" categories are provided in Table 10.

B Resources

For our experiments, we gathered grammar books, which were tokenized and stored in the vector database. We used the OpenAI text-embedding-ada-002 model to generate text embeddings from the grammar book text. Grammar books were obtained from DReaM (Virk et al., 2020) and converted to the format required by the code. We used the grammar books that were referenced by Grambank coders for each language. We note the filename that was downloaded from DReaM. There were a handful of instances in which the document was not loaded into the vector database, as is noted in Table 11.

	baseline	**rag_1**	**rag_10**	**rag_20**	**rag_30**	**rag_40**	**total**
southern_jinghpaw	17	93	117	111	110	108	195
minangkabau	43	82	112	116	118	121	174
mizo	35	86	115	120	117	119	195
natugu	40	85	98	95	94	96	193
kalamang	41	105	110	108	112	106	195

Table 4: Number of correctly predicted features based on number of retrieved pages included in the prompt.

	LLM output code						
code	**0**	**1**	**2**	**3**	**?**	**IDK**	**Total**
0	70	25	0	0	19	1	115
1	9	39	0	0	4	1	53
2	0	0	1	2	1	0	4
?	12	2	0	0	7	2	23
Total	91	66	1	2	31	4	195

Table 5: Southern Jinghpaw - Distribution of code values by category.

	LLM output code						
code	**0**	**1**	**2**	**3**	**?**	**IDK**	**Total**
0	85	18	0	0	16	4	123
1	2	30	0	1	4	0	37
2	0	1	2	0	1	0	4
?	4	2	0	0	4	0	10
Total	91	51	2	1	25	4	174

Table 6: Minangkabau - Distribution of code values by category.

	LLM output code					
code	**0**	**1**	**2**	**?**	**IDK**	**Total**
0	78	32	0	16	4	130
1	10	32	2	5	0	49
2	0	0	3	0	0	3
?	1	5	0	7	0	13
Total	89	69	5	28	4	195

Table 7: Mizo - Distribution of code values by category.

	LLM output code					
code	**0**	**1**	**2**	**?**	**IDK**	**Total**
0	57	5	2	10	1	75
1	3	12	0	6	0	21
2	0	1	2	0	0	3
?	39	9	1	41	6	96
Total	99	27	5	57	7	195

Table 8: Kalamang - Distribution of code values by category.

	LLM output code						
code	**0**	**1**	**2**	**3**	**?**	**IDK**	**Total**
0	22	14	0	0	15	1	52
1	5	31	0	0	5	0	41
2	0	0	1	1	0	1	3
?	28	9	1	0	44	15	97
Total	55	54	2	1	64	17	193

Table 9: Natugu - Distribution of code values by category.

Feature Type	**kac**	**min**	**lus**	**ntu**	**kgv**
TAME	42.11 (8/19)	83.33 (10/12)	73.68 (14/19)	36.84 (7/19)	21.05 (4/19)
VP (other)	43.75 (7/16)	78.57 (11/14)	37.5 (6/16)	75.0 (12/16)	31.25 (5/16)
Argument marking (core)	70.0 (14/20)	68.75 (11/16)	65.0 (13/20)	52.63 (10/19)	75.0 (15/20)
Argument marking (non-core)	66.67 (8/12)	41.67 (5/12)	41.67 (5/12)	63.64 (7/11)	33.33 (4/12)
Class	82.35 (14/17)	88.24 (15/17)	70.59 (12/17)	58.82 (10/17)	70.59 (12/17)
Clause order	69.57 (16/23)	57.14 (12/21)	56.52 (13/23)	52.17 (12/23)	69.57 (16/23)
Deixis	66.67 (6/9)	66.67 (6/9)	55.56 (5/9)	11.11 (1/9)	44.44 (4/9)
Non-verbal predication	35.0 (7/20)	70.0 (14/20)	65.0 (13/20)	55.0 (11/20)	45.0 (9/20)
Number	77.27 (17/22)	85.71 (18/21)	68.18 (15/22)	31.82 (7/22)	77.27 (17/22)
Order in NP	50.0 (5/10)	66.67 (4/6)	60.0 (6/10)	60.0 (6/10)	70.0 (7/10)
Quantification	50.0 (4/8)	62.5 (5/8)	87.5 (7/8)	62.5 (5/8)	62.5 (5/8)
Valency	64.29 (9/14)	69.23 (9/13)	57.14 (8/14)	64.29 (9/14)	78.57 (11/14)
Verb complex	40.0 (2/5)	20.0 (1/5)	60.0 (3/5)	20.0 (1/5)	60.0 (3/5)
Total	60.0 (117/195)	69.54 (121/174)	61.54 (120/195)	50.78 (98/193)	57.44 (112/195)

Table 10: Match accuracy by finer grouping for each language. Values are formatted as percentage (correct/total).

Language	Grambank Sources	Chroma (from DReaM)
Mizo	Chhangte, Lalnunthangi. 1986. *A Preliminary Grammar of the Mizo Language*.	chhangte_mizo-grammar1986_o.pdf
Minangkabau	Crouch, Sophie. 2009. *Voice and verb morphology in Minangkabau, a language of West Sumatra, Indonesia*.	crouch_minangkabau2009.pdf
	Reibaud, Rusmidar. 2004. *Parlons Minangkabau. Paris: L'Harmattan.*.	reibaud_minangkabau2004_o.pdf
	Roesli, Oleh. 1967. *Bahasa Minangkabau. Jakarta: Bhratara*.	roesli_minangkabau1967v2_o.pdf
Southern Jinghpaw (Kachin)	Kurabe, Keita. 2012. *Jingpho Dialogue Texts with Grammatical Notes*.	kurabe_jingpho-dialogue2012.pdf
	Kurabe, Keita. 2017. *Jinghpaw*.	kurabe_jinghpaw2017_o.pdf
	Qingxia, Dai and Diehl, Lon. 2003. *Jinghpo*.	Qingxia-diehl_jingpho2003_o.pdf
Kalamang	Visser, Eline. 2016. *A grammar sketch of Kalamang, with a focus on phonetics and phonology*.	visser_kalamang2016_o.pdf
	Galis, Klaas Wilhelm. 1955. *Talen en dialecten van Nederlands Nieuw-Guinea*.	Not loaded. No feature identifies a page number from this source.
Natugu	Næss, Åshild and Boerger, Brenda H. 2008. Reefs-Santa Cruz as Oceanic: Evidence from the Verb Complex. Oceanic Linguistics 47.	naess-boerger_rsc-verb2008.pdf
	Ross, Malcolm D. 2007. Two Kinds of Locative Construction in Oceanic Languages: A Robust Distinction. In Siegel, Jeff and Lynch, John and Eades, Diana (eds.), Language Description, History and Development: Linguistic Indulgence in Memory of Terry Crowley,	Not loaded. Couldn't find free version
	Wurm, Stephen A. 1969. The Linguistics Situation in the Reef and Santa Cruz Islands. In Papers in Linguistics of Melanesia No. 2, 47-105. Canberra: Research School of Pacific and Asian Studies, Australian National University	wurm_reef-santa-cruz1969.pdf
	Wurm, Stephen A. 1971. The Papuan linguistic situation. In Sebeok, Thomas A. (ed.), Linguistics in Oceanea, 541-657. Berlin: Mouton de Gruyter.	sebeok_oceania1971_o.pdf
	Wurm, Stephen A. 1972. Notes on Indication of Possession with Nouns in Reef and Santa Cruz Islands Languages. In Papers in Linguistics of Melanesia 3, 85-113. Canberra: Research School of Pacific and Asian Studies, Australian National University.	wurm_reef-santa-cruz1972v2_o.pdf
	Wurm, Stephen A. 1976. The Reef Islands-Santa Cruz Family. In Wurm, Stephen A. (ed.), New Guinea Area Languages and Language Study Vol 2: Austronesian Languages, 637-674. Canberra: Research School of Pacific and Asian Studies, Australian National University.	wurm_reef-santa-cruz1976_o.pdf
	Wurm, Stephen. (1978) Reef-Santa Cruz: Austronesian, but . . . !. In Stephen Wurm and Lois Carrington (eds.), Proceedings of the 2nd International Conference on Austronesian Linguistics: Fascicle 2 (Pacific Linguistics: Series C 61), 969-1010. Canberra: Research School of Pacific and Asian Studies, Australian National University.	Not loaded. PDF scan had 2 text pages per scanned page.
	van den Berg, Rene and Boerger, Brenda H. 2011. A Proto-Oceanic Passive? Evidence from Bola and Natügu. Oceanic Linguistics 50. 221-246	vandenberg_bola-natugu2011.pdf

Table 11: Grammar books for each language.

C Prompt Format

Each feature to be predicted is formatted into a prompt for GPT-4. In the following sections, we show the format of the prompt by example.

Prompt Template

> You are an expert linguist with extensive knowledge about many languages. Answer the following question about the language {language_name}. You are also provided with additional information about the question and you are given a procedure that indicates allowable answers for the question. You MUST provide an answer following the procedure. If you do not know the answer, answer 'IDK'. Output the answer in JSON format with the following key-value pairs: 'code': code, 'comment': other_data. Please format the response as valid JSON that I can parse.

> {question}
> Here is a summary about the question:
> {summary}
> {context}
> Here is the procedure to follow and the allowable responses:
> {procedure}

Example Using GB020-lush1249

Are there definite or specific articles?

Here is a summary about the question:

An article is a marker that accompanies the noun and expresses notions such as (non-)specificity and (in)definiteness. Sometimes these notions of specificity and definiteness are summed up in the term ’identifiability’. The formal expression is irrelevant; articles can be free, bound, or marked by suprasegmental markers such as tone.

Articles are different from demonstratives in that demonstratives occur in a paradigm of markers that have a clear spatial deictic function. As demonstratives can grammaticalize into definite or specific articles, they form a natural continuum, making it hard to define discrete categories, but to qualify as an article a marker should be used in some cases to express definiteness without also expressing a spatial deictic meaning.

To help answer the question, here is relevant data retrieved from a grammar book

GRAMMAR BOOK EXCERPTS RETRIEVED FROM DATABASE

Here is the procedure to follow and the allowable responses:

1. Code 1 if there is a morpheme that can mark definiteness or specificity without also conveying a spatial deictic meaning.
2. Code 0 if the source does not mention a definite article and you cannot find one in examples or texts in an otherwise comprehensive grammar.
3. Code ? if the grammar does not contain enough analysis to determine whether there is a definite article or not.
4. If you have coded 1 for GB020 and 0 for GB021 and GB022, please write a comment explaining the position of the definite or specific article.

Author Index