Discourse Context Primes Hindi Word Order

Anonymous ACL submission

Abstract

Hindi has a flexible word order, yet certain word orders are consistently preferred over others. A number of factors are known to influence Hindi word order preferences in isolation, including information structure and syntactic complexity. However, the relative impact of these factors on Hindi constituent ordering is not well understood. Inspired by prior work on syntactic priming, we investigate how the words and syntactic structures in a sentence influence the word order of the following sentences. Specifically, we extract sentences from the Hindi-Urdu Treebank corpus (HUTB), we permute the preverbal constituents of those sentences, and we build a classifier to predict which sentences actually occurred in the corpus against our generated distractors. The classifier uses a number of discourse-based features and cognitive features to make its predictions, including dependency length, surprisal, and information status. We find that lexical and syntactic priming and referent givenness drive order preferences. Moreover, along the lines of previous work in psycholinguistics, we find that certain verbs are more susceptible to priming than others. We conclude by situating our results within the broader syntactic priming literature.

1 Introduction

Hindi (Indo-Aryan family) has a rich case-marking system and flexible word order. In this work, we investigate the factors that cause certain word orders to be preferred over other orderings that express similar meanings.

Example 1 contains a set of sentences that each expresses a similar meaning but with a different pre-verbal word order.

(1) a. amar ujala-ko **yah** *sukravar*-ko daak-se prapt hua
Amar Ujala-ACC **it** *friday*-on post-INST receive be.PST.SG
Amar Ujala received **it** by post on *Friday*.

b. **yah** amar ujala-ko *sukravar*-ko daak-se prapt hua

c. *sukravar*-ko **yah** amar ujala-ko daak-se prapt hua

Earlier studies of Hindi word order have demonstrated a wide variety of factors that influence order preferences, such as information status (Butt and King, 1996; Kidwai, 2000), prosody (Patil et al., 2008), and semantics (Perera and Srivastava, 2016; Mohanan and Mohanan, 1994). Prior work has also shown that Hindi optimizes processing efficiency by minimizing information load (Ranjan et al., 2019) and dependency length (Ranjan et al., 2021; Vasishth, 2004). The current work investigates how discourse and cognitive factors jointly influence preverbal constituent order in Hindi.

During reading, encountering a syntactic structure eases the comprehension of subsequent sentences with similar syntactic structures as attested in a wide variety of languages (Arai et al., 2007; Husain and Yadav, 2020; Tooley and Traxler, 2010). So in this work we test whether *adapting* a neural language model to inter-sentential discourse information helps better model preverbal Hindi constituent order in the presence of other cognitively grounded controls. Additionally, since information structure influences word order (Arnold et al., 2000), we also test whether *givenness* of the constituents (Clark and Haviland, 1977; Chafe, 1976) influences which order is preferred.

To test ordering preferences, we generated meaning-equivalent grammatical variants of sentences from the Hindi-Urdu Treebank corpus (HUTB; Bhatt et al., 2009) by permuting their preverbal constituent ordering. Subsequently, we used a logistic regression model to separate the original reference sentences from the plausible variants using the cognitive features of interest.

Corroborating the previous findings of adaptation/priming in comprehension (Fine et al., 2013; Fine and Jaeger, 2016) and production (Gries, 2005; Bock, 1986), our results indicate that priming influences word-order preferences in Hindi. Generally, this effect is driven by lexical priming, but we also find that certain object-fronted constructions prime subsequent object-fronting, providing evidence for self-priming of larger syntactic configurations. Verb-specific analyses revealed that priming in Hindi is stronger for certain verb classes, a phenomenon also observed in English spoken and written text (Gries, 2005). Finally, we discuss the implications of our findings for syntactic priming in both comprehension and production.

Our main contribution is that for a low-resource and typologically distinct language, *viz.*, Hindi, we show the impact of discourse context on word order choices using computational methodology. Thus we provide cross-linguistic evidence imperative to validate theories of language processing (Jaeger and Norcliffe, 2009).

2 Background

2.1 Surprisal Theory

Surprisal Theory (Hale, 2001; Levy, 2008) posits that comprehenders construct probabilistic interpretations of sentences based on previously encountered structures. Mathematically, the *surprisal* of the k^{th} word, w_k, is defined as the negative log probability of w_k given the preceding context:

$$S_k = -\log P(w_k|w_{1...k-1}) = \log \frac{P(w_1...w_{k-1})}{P(w_1...w_k)} \qquad (1)$$

These probabilities can be computed either over word sequences or syntactic configurations and reflect the information load (or predictability) of w_k. Both versions of surprisal predict eye-movements in reading times (Levy, 2008; Demberg and Keller, 2008; Staub, 2015) as well as spontaneous speech word durations (Demberg et al., 2012; Dammalapati et al., 2021).

2.2 Dependency Locality Theory

Dependency locality theory (Gibson, 2000) has been shown to be effective at predicting the comprehension difficulty of a sequence, with shorter dependencies generally being easier to process than longer ones (Temperley, 2007; Futrell et al., 2015; Liu et al., 2017, cf. Demberg and Keller, 2008). In this work, we defined dependency length as the number of intervening words between head and dependent units in a dependency tree (Temperley, 2008; Rajkumar et al., 2016).

2.3 Information Status

Information structure strongly influences syntactic choice (Halliday, 1970). Languages generally prefer to mention *given* referents, from earlier in the discourse, before introducing *new* ones (Clark and Haviland, 1977; Chafe, 1976; Kaiser and Trueswell, 2004). The explanation for this is that given information is more accessible compared to new information, so providing the given information first provides a more robust context to ease processing of the new referents (Arnold et al., 2000; Bock and Irwin, 1980).

3 Data and Models

Our dataset comprises 1996 reference sentences containing well-defined subject and object constituents from the HUTB corpus of dependency trees (Bhatt et al., 2009). Figure 1 in Appendix A displays the dependency tree for Example sentence 1a and explains our variant generation procedure in more detail. For each reference sentence, we created artificial variants by permuting the preverbal constituents whose heads were linked to the root node in the dependency tree. Inspired by grammar rules proposed in the NLG literature (Rajkumar and White, 2014), ungrammatical variants were automatically filtered out by detecting dependency relation sequences not attested in the original HUTB corpus. After filtering, we had 72833 variant sentences for our classification task.

3.1 Models

We set up a binary classification task to separate the original HUTB reference sentences from the variants using the cognitive metrics described in Section 2. To alleviate the data imbalance between the two classes (1996 references vs 72833 variants), we transformed our data set using the approach described in Joachims (2002). This technique converts a binary classification problem into a pairwise ranking task by training the classifier on the difference of the feature vectors of each reference and its corresponding variants (see Equation 2 and 3). Equation 2 displays the objective of a standard binary classifier, where the classifier must learn a feature weight (w) such that the dot product of w with the reference feature vector ($\phi(reference)$) is greater than the dot product of w with the variant feature vector ($\phi(variant)$). This objective can be rewritten as equation 3 such that the dot product of w with the difference of the feature vectors is greater than zero.

$$w \cdot \phi(reference) > w \cdot \phi(variant) \tag{2}$$

$$w \cdot (\phi(reference) - \phi(variant)) > 0 \tag{3}$$

Every variant sentence in our dataset was paired with their corresponding reference sentence with order balanced across these pairings (e.g., Example 1 would yield (1a,1b) and (1c,1a)). Thereafter, their feature vectors were subtracted (e.g., 1a-1b and 1c-1a), and binary labels were assigned to each transformed data point. Reference-Variant pairs were coded as "1" and Variant-Reference pairs were coded as "0". The alternate pair ordering thus re-balanced our previously severely imbalanced classification task.

For each reference sentence, our objective was to model the possible syntactic choices entertained by the speaker. In each instance, the author chose to generate the reference order over the variant, implicitly demonstrating an order preference. If the cognitive factors we chose influenced that decision, a logistic regression model should be able to use those factors to predict which syntactic choice was ultimately chosen by the author. Using the transformed features dataset labelled with 1 (denoting a preference for the reference order) and 0 (denoting a preference for the variant order), we trained a logistic regression model to predict each reference sentence (see Equation 4). We report our classification results using 10-fold cross-validation. The regression results are reported on the entire transformed test data for the respective experiments. All the experiments were done with the Generalized Linear Model (GLM) package in R.

$$choice \sim \begin{cases} \delta \text{ dependency length } + \\ \delta \text{ trigram surp } + \delta \text{ pcfg surp } + \\ \delta \text{ IS score } + \delta \text{ lexical repetition surp } + \\ \delta \text{ lstm surp } + \delta \text{ adaptive lstm surp} \end{cases} \tag{4}$$

Here *choice* is encoded by the binary dependent variable as discussed above (1: reference preference and 0: variant preference). To obtain sentence-level surprisal measures, we summed word-level surprisal of all the words in each sentence. The values for independent variables were calculated as follows.

1. **Dependency length**: We computed a sentence-level dependency length measure by summing the head-dependent distances (measured as the number of intervening words) in the dependency trees provided in the HUTB corpus. Since our variants were generated by manipulating the provided dependency trees, we were able to directly compute the dependency length for each variant sentence as well.

2. **Trigram surprisal**: For each word in a sentence, we estimated its local predictability using a 3-gram language model (LM) trained on the EMILLE Hindi Corpus (Baker et al., 2002), which consists of 1 million mixed genre sentences, using the SRILM toolkit (Stolcke, 2002) with Good-Turing discounting.

3. **PCFG surprisal**: The syntactic predictability of each word in a sentence was estimated using the Berkeley latent-variable PCFG parser[1] (Petrov et al., 2006). Hindi is a low resource language as it has only one dependency treebank and no constituency treebank. So 12000 phrase structure trees were created to train the parser by converting Bhatt et al.'s HUTB dependency trees into constituency trees using the approach described in Yadav et al. (2017). Sentence level log-likelihood of each test sentence was estimated by training a PCFG language model on four folds of the phrase structure trees and then testing on a fifth held-out fold.

4. **Information status (IS) score**: We automatically annotated whether each sentence exhibited *given-new* ordering. The subject and object constituents in a sentence were assigned a *Given* tag if any content word within them was

[1] 5-fold CV parser training and testing F1-score metrics were 90.82% and 84.95%, respectively.

mentioned in the preceding sentence (e.g., if "Amar Ujala" had been mentioned in the sentence preceding 1a, it would be annotated as *Given* in 1a) or if the head of the phrase was a pronoun (e.g., "yah" in 1b). All other phrases were tagged as *New*. For each sentence, IS score was computed as follows: a) Given-New order = +1 b) New-Given order = -1 c) Given-Given and New-New = 0. For illustration, see Example 3 in Appendix A, which shows how givenness would be coded after a context sentence (Example 2).

5. **Lexical repetition surprisal**: For each word in a sentence, we accounted for lexical priming by interpolating a 3-gram language model with a unigram cache LM based on the history of words ($H = 100$) containing the preceding sentence. We used the original implementation provided in the SRILM toolkit with a default interpolation weight parameter ($\mu = 0.05$; see Equations 5 and 6) based on the approach described by Kuhn and De Mori (1990). The idea is to keep a count of recently occurring words in the sentence history and then boost their probability within the trigram language model. Words that have occurred recently in the text are likely to re-occur in subsequent sentences (Kuhn and De Mori, 1990; Clarkson and Robinson, 1997).

$$P(w_k|w_1, w_2, ...w_{k-1}) = \mu\, P_{cache}(w_k|w_1, w_2, ...w_{k-1}) + (1-\mu)\, P_{trigram}(w_k|w_{k-2}, w_{k-1}) \tag{5}$$

$$P_{cache}(w_k|w_{k-H}, w_{k-H+1}, ...w_{k-1}) = \frac{w_k \; counts \; in \; cache}{H} \tag{6}$$

6. **LSTM surprisal**: We estimated the predictability for each word according to the entire sentence prefix using a long short-term memory language model (LSTM; Hochreiter and Schmidhuber, 1997) trained on the 1 million sentences of the EMILLE Hindi corpus (Baker et al., 2002). We used the original implementation provided in the neural complexity toolkit[2] (van Schijndel and Linzen, 2018) with default hyper-parameter settings to estimate surprisal using an unbounded neural context.

7. **Adaptive LSTM surprisal**: We estimated the discourse predictability of each word in the sentence using the neural complexity toolkit. van Schijndel and Linzen (2018) proposed a simple way to continuously adapt a neural LM, and found that adaptive surprisal predicts human reading times significantly better than non-adaptive surprisal. Their method takes a pre-trained LSTM LM, and, after generating surprisals for a test sentence, the parameters of the LM get updated based on the cross-entropy loss for that sentence. After that, the revised LM weights are used to predict the next test sentence. In our work, for each test sentence, we used our base (non-adaptive) LSTM LM and adapted it to the preceding context sentence before generating (adaptive) surprisal values for the desired sentence.

[2] https://github.com/vansky/neural-complexity

Learning Rate	0	0.002	0.02	0.2	**2**	20	200
Perplexity	103.29	98.79	87.78	66.64	**56.86**	117.91	$\sim 10^9$

Table 1: Learning rate influence on lexical and syntactic adaptation for the validation set containing 13274 sentences (the initial non-adaptive model performance is when we use a learning rate of 0)

4 Experiments and Results

We tested the hypothesis that information status and surprisal enhanced with inter-sentential discourse information (adaptive LSTM surprisal) predict constituent ordering in Hindi over other baseline cognitive controls, including dependency length, lexical repetition and non-adaptive surprisal. For our adaptation experiments, we used an adaptive learning rate of 2 as it minimized the perplexity of the validation data set (see Table 1).[3] The Pearson's correlation coefficients between different predictors are displayed in Figure 2 in Appendix A. The adaptive LSTM surprisal has a high correlation with all other surprisal features and a low correlation with dependency length and information status score. We report the results of the regression and prediction experiments on the full

[3] Interestingly, van Schijndel and Linzen (2018) found that an adaptive learning rate of 2 minimized validation perplexity in English as well, though we leave further investigation of this to future work.

data set as well as on subsets of the data consisting of two types of non-canonical constructions. We also conducted a fine-grained verb-specific analysis of priming patterns.

4.1 Regression Analysis

Our regression results over the entire data set (Table 2) show that all of our measures are significant predictors for the task of classifying reference and variant sentences. The negative regression coefficients for our surprisal metrics indicate that surprisal is consistently lower in the reference sentences compared with the competing variants. And adding adaptive LSTM surprisal into a model containing all other predictors significantly improved the fit of our regression model (χ^2 = 66.81; p $<$ 0.001). The positive regression coefficient for information status (IS) score indicates that reference sentences adhere to *given-new* ordering. These results support our two core hypotheses that discourse-adaptive surprisal and information status affect word order preferences in Hindi. However, the positive regression coefficient of dependency length suggests that reference sentences exhibit *longer* dependency lengths compared to their variant counterparts, violating locality considerations.

We also examined the contribution of each predictor on two non-canonical constructions, *DO-fronted* and *IO-fronted* constructions, which have been studied extensively in the sentence comprehension literature. Prior work has shown that salient objects tend to occur early in the sentence, thus leading to fronting (Wierzba and Fanselow, 2020; Kaiser and Trueswell, 2004). In the specific context of Hindi, Vasishth (2004) examined the role of locality effects in processing non-canonical word orders (direct and indirect object fronting) in salient as well as non-salient contexts. He showed that the increased distance from direct object (DO) fronting leads to high self-paced reading time at the inner-most verb as compared to its canonical counterpart in both salient and non-salient conditions. However, in indirect object (IO) fronted constructions, he found that salient contexts alleviated the processing difficulty which was caused by increased distance. Based on these findings, we predict that **adaptive surprisal should be more effective in IO-fronted than DO-fronted constructions**.

Predictor	$\hat{\beta}$	$\hat{\sigma}$	t
intercept	1.50	0.001	1496.47
trigram surprisal	-0.08	0.005	-14.53
dependency length	0.02	0.001	15.55
pcfg surprisal	-0.07	0.002	-39.46
IS score	0.01	0.001	11.32
lex-rept surprisal	-0.03	0.005	-5.31
lstm surprisal	-0.14	0.016	-9.26
adaptive lstm surprisal	-0.13	0.016	-8.18

Table 2: Regression model on full data set ($N = 72833$; all significant predictors denoted by |t|>2)

We isolated two types of non-canonical constructions from our data set. In the first type, the reference sentence has a direct object (DO) fronted structure while the variant has the canonical order where the subject precedes the DO. In the second type, the reference sentence has an indirect object (IO) fronted structure while the variant has the canonical order where the subject precedes the IO. Table 3a and Table 3b present regression results for DO- and IO-fronted constructions respectively. These subsets constitute a very small fraction of our data set due to the infrequency of these constructions in Hindi. The regression coefficient for adaptive LSTM surprisal was significantly negative for both subsets, indicating that the non-canonical structures are more common in the context of similarly non-canonical structures. This pattern is more robust for IO-fronted reference sentences (χ^2 = 90.90; p $<$ 0.001) than for DO-fronted reference sentences (χ^2 = 4.03; p = 0.04), validating our proposed prediction about these constructions. Furthermore, in contrast to the IO-fronted subset, the regression coefficient for dependency length in DO-fronted items is significantly negative suggesting that locality considerations are limited to constructions involving a high dependency length difference[4] between reference and variants, a similar finding to that reported in Ranjan et al. (2021) on a similar task.

4.2 Prediction Accuracy

While the previous section explored how predictors contribute to Hindi ordering preferences across all of the data in aggregate, in this section we frame our model as a classification task on held-out data to determine how many sentences are affected by

[4]The average dependency length difference for DO-subset is 13.92 and IO-subset is 7.77 words

Predictor	$\hat{\beta}$	$\hat{\sigma}$	t
intercept	**1.49**	0.008	171.18
trigram surp	**-0.28**	0.049	-5.84
dep length	**-0.05**	0.008	-6.22
pcfg surp	0.001	0.014	0.12
IS score	**0.04**	0.006	7.04
lex repetition surp	0.07	0.044	1.67
lstm surp	0.03	0.114	0.23
adaptive lstm surp	**-0.23**	0.113	-2.00

(a) Direct objects (DO; 1663 points) fronted cases

Predictor	$\hat{\beta}$	$\hat{\sigma}$	t
intercept	**1.51**	0.008	188.49
trigram surp	**-0.18**	0.039	-4.54
dep length	0.02	0.012	1.77
pcfg surp	**-0.13**	0.015	-8.34
IS score	-0.01	0.005	-1.87
lex repetition surp	0.03	0.036	0.92
lstm surp	**1.21**	0.154	7.87
adaptive lstm surp	**-1.50**	0.155	-9.67

(b) Indirect objects (IO; 1353 points) fronted cases

Table 3: Discourse adaptation regression model on DO/IO fronted cases (all significant predictors denoted by |t|>2)

Predictors	**Full Accuracy %**	**DO**	**IO**
a = IS score	51.84	53.88	50.92
b = dep length	62.31***	68.49***	58.91***
c = pcfg surp	86.86***	65.90	78.86***
d = lex repetition surp	90.07***	77.33***	85.07***
e = 3-gram surp	91.18***	78.95*	87.29**
f = lstm surp	**94.01*** **	79.55	87.28
g = adaptive lstm surp	94.06	**79.97**	**88.32*** **

Predictors	**Full Accuracy %**	**DO**	**IO**
Collective: with repetition effects			
base1 = a+b+c+d+e+f	**95.05**	80.99	89.06
base1 + g	95.06	81.06	**89.65***
Collective: without repetition effects			
base2 = a+b+c+e+f	95.06	81.24	89.65
base2 + g	**95.09***	81.42	89.80

Table 4: Prediction performances (Full data set (72833 points), Direct objects (DO; 1663 points) and indirect object (IO; 1353 points) fronted cases; each row refers to a distinct model; *** McNemar's two-tailed significance compared to model on previous row)

each predictor. This enables us to examine the relative performance of different predictors in identifying Hindi reference sentences amidst artificially generated grammatical variants and to conduct more detailed error analysis of our results. We used 10-fold cross-validation to evaluate model classification accuracy, *i.e.* the percentage of data points where a model correctly predicted the referent sentence over a paired variant, for different subsets of predictors (see Table 4).

Non-adaptive LSTM surprisal (94.01% accuracy) and adaptive LSTM surprisal (94.06%) yielded the best classification accuracies when no other predictors were included. Over a baseline model comprised of every other feature except lexical repetition surprisal (see base2 in Table 4), adaptive LSTM surprisal induced a small but significant increase of 0.03% in accuracy (p = 0.04 using McNemar's two-tailed test). When we included lexical repetition surprisal in the baseline model (see base1 in Table 4), adaptive LSTM surprisal ceases to be a significant predictor. This suggests that, in the general case, adaptive LSTM surprisal reflects the influence of lexical priming on word order. Apart from the content words, adaptive LSTM surprisal accounts for the re-occurrence of function words (e.g., case markers) which have been shown to modulate syntactic priming and drive parsing processes (Husain and Yadav, 2020).

To study prediction accuracy on non-canonical constructions, we restricted our analyses to IO- and DO-fronted items in the test partition (still training the classifier on the full training partition for each fold). In contrast to the DO-fronted subset, adaptive surprisal was a significant predictor of IO-fronted syntactic choice, even in the presence of lexical repetition surprisal, as evident from the significant increase of 0.6% in accuracy (p = 0.02 using McNemar's two-tailed test; see the rightmost IO column in Table 4). This result indicates that syntactic priming is effective in predicting IO-fronting in sentences that follow other IO-fronted sentences. Both our regression and classification results demonstrate that adaptation is more effective in IO-fronted than DO-fronted constructions, mirroring the findings in Hindi sentence comprehension, where Vasishth (2004) showed that discourse context could compensate for the processing difficulty induced by indirect object fronting.

Further linguistic analyses in IO-fronted con-

Type	Freq (%)	Baseline	Baseline + Adaptive LSTM
Verb Class			
DO	48.68	96.82	96.82
GIVE	19.35	93.86	**93.98**
SOCIAL	8.00	92.90	92.95
COMMUNICATE	6.25	93.94	93.98
LODGE	4.04	94.29	94.22
MOTION	3.87	90.87	90.76
PUT	2.97	95.28	95.28
DESTROY	2.42	95.58	95.63
PERCEPTION	0.73	87.48	87.10
OTHERS	3.69	90.63	90.22
Alternations			
S-DO	71.89	95.35	95.33
S-DO-IO	12.74	93.39	**93.50**
S-IO	15.37	94.98	95.04

Table 5: Prediction performance of verb-specific and subject-objects alternations (72833 points); Baseline denotes *base1* shown in Table 4; bold denotes McNemar's two-tailed significance compared to baseline model in the same row)

structions revealed that LSTM adaptation also captured the priming of *given-given* items, potentially modeling the preferred ordering of multiple *given* items, a case not captured by IS score or lexical repetition surprisal. Refer to Appendix G for full details of this analysis.

4.3 Verb-specific Priming

Individual verb biases also influence structural choices during language production (Ferreira and Schotter, 2013; Thothathiri et al., 2017; Yi et al., 2019). Therefore, we grouped Hindi verbs based on Levin's syntactico-semantic classes using the heuristics proposed by Begum and Sharma (2017). Then we analyzed the efficacy of adaptive surprisal at classifying reference and variant instances of Levin's verb classes (still training the classifier on the full training partition for each fold). Our results (Table 5, top block) indicate that the GIVE verb class was susceptible to priming, with adaptive surprisal producing a significant improvement of 0.5% in classification accuracy (p = 0.01 using McNemar's two-tailed test) over the baseline model. Other verb frames did not show a syntactic priming effect.

Our results are in line with previous work in the priming literature that show GIVE to be especially susceptible to priming, thus providing cross-linguistic support to verb-based priming effects (Pickering and Branigan, 1998; Gries, 2005; Bock, 1986). The GIVE verb class in our data set includes different verbs that are semantically similar to *give* in English, such as *de, baant, saup, bhej, maang, dila, lauTaa, vasul, thama, vaapas*. We found that all these verbs strongly exhibited double object constructions (Begum and Sharma, 2017) and their arguments are heavily case marked (see Table 6 in Appendix B).

Our results also reveal (Table 5, bottom block) that syntactic priming is more influential in double object constructions (S-DO-IO) than in single object constructions as attested by a significant improvement of 0.1% in classification accuracy (p = 0.04 using McNemar's two-tailed test). Double object constructions are also highly case marked (see Table 7 in Appendix C) and 57.82% of these items contain verbs that belong to GIVE class (see Table 8 in Appendix D). We present a more nuanced discussion on the effects of case-markers and verb's combinatorial properties on priming in Section 5. The regression coefficients on Levin's GIVE verb classes and double object alternations follow similar trends as reported in the previous section (see Appendices E and F).[5]

Our analyses suggest that different verbs display varying strength of priming effects, corroborating previous findings in the literature (Gries, 2005). Ditransitive constructions (denoted by S-DO-IO ordering) prime more strongly than other orderings, where verbs from the GIVE class have a strong preference for canonical argument ordering.[6]

4.4 What causes priming?

In the priming literature, there is debate as to whether priming is driven by residual neural activation (short-lived effects) or by humans learning and updating their language expectations (long-lived effects). Bock and Griffin (2000) showed that syntactic priming in humans persisted even when prime and target sentences were separated by 10 intervening sentences, supporting the implicit learning (long-lived) hypothesis of syntactic priming. In or-

[5] We provide an analysis of an example item in Appendix H to show how discourse priming (via adaptive surprisal) can interact with the other factors we studied to jointly predict the correct ordering preference in double-object constructions.

[6] For example, out of 284 instances, 89.79% of the give lemma 'de' occurs with canonical argument ordering in our test data set.

der to test this effect on constituent ordering choice, we repeated our adaptation experiment by adapting to additional context sentences from the preceding discourse. Adaptive LSTM surprisal and lexical repetition surprisal were estimated by adapting the base LSTM LM and trigram LM, respectively, to five preceding context sentences, rather than the single context sentence we used for our other analyses. We found that for non-canonical IO/DO-fronted constructions, additional context sentences do not improve the adaptive LSTM LM's word order predictions, suggesting that priming may be driven by short-term residual activation (see Table 13 in the Appendix I).

5 Discussion

Our main findings suggest that lexical priming, structural priming, and information status all influence the word order preferences of Hindi. Lexical priming is most influential in canonical sentence contexts, but syntactic priming does influence preferences in non-canonical contexts. We also show that certain verb classes are more susceptible to priming than others. Specifically, verbs selecting double objects are most prone to priming, a case demonstrated in English as well (Gries, 2005), thus providing cross-linguistic support for the finding.

Below, we discuss the implications of our findings in terms of the 4 factors affecting syntactic priming discussed in detail by Reitter et al. (2011): *inverse frequency interaction, decay, lexical boost, and cumulativity*. The IO-fronted construction is very rare (0.76%) compared to DO-fronted non-canonical sentences (1%) in the HUTB corpus of 13274 sentences. We find strong priming effects in the case of IO-fronted constructions but weak priming in the case of DO-fronted constructions, providing evidence for an *inverse frequency interaction* (Scheepers, 2003; Jaeger and Snider, 2007).

Our finding that priming is not aided by long-term contexts indicates a *decay effect* in priming, which supports the residual activation (short-lived) hypothesis of priming in comprehension (Pickering and Branigan, 1998). Nevertheless, there has been evidence for implicit learning effects in comprehension as well (Luka and Barsalou, 2005; Wells et al., 2009). In an experimental study examining the impact of preverbal case markers on syntactic priming in Hindi comprehension, Husain and Yadav (2020) provide counter-evidence against the residual activation account. They argue that researchers must incorporate more syntactic properties of target sentences into priming studies. Additional research is required to tease apart the exact processes reflected by priming, a point raised by Tooley and Traxler (2010) in their comprehensive literature review.

Previous work suggests that lexical overlap between prime and target sentences enhances syntactic priming (Pickering and Branigan, 1998; Gries, 2005). The repeated lexical items become cues during sentence planning and bias the speaker to produce similar structures that those repeated lexical items tend to occur in. Overall, we find that lexical repetition drives Hindi syntactic choice; however, syntactic priming is observed over and above lexical repetition in non-canonical and double object constructions. Our verb-specific priming analyses indicate that prime sentences need not share the same main verb as the target sentence; instead successive sentences may have a similar argument structure (subcategorization frame). Our results provide evidence for a generalized *lexical boost effect* which operates over verb classes and not simply string-identical verbs, validating similar findings on English (Snider, 2009). However, Husain and Yadav (2020) showed that the combinatory properties of the verb need not be the sole driver of priming in Hindi. In their self-paced reading experiment involving identical critical verbs in both prime and target sentences, they observed a speedup in reading times only in the condition where nominals were marked by a locative case marker (in contrast to accusative and ergative conditions). So the impact of case markers on priming strength needs to be explored more thoroughly in future inquiries.

Finally, with regards to the *cumulativity* of priming, Jaeger and Snider (2007) showed in their corpus study of production of passives and *that* insertion/omission that the effect of priming increases with the number of primes preceding it. Our work does not investigate this specifically, and more controlled experiments would be required.

Overall, our results demonstrate that Hindi word order preferences are driven by lexical and syntactic priming as well as *Given-New* ordering patterns of discourse referents.

References

Manabu Arai, Roger PG Van Gompel, and Christoph Scheepers. 2007. Priming ditransitive structures in comprehension. *Cognitive psychology*, 54(3):218–250.

Jennifer E. Arnold, Thomas Wasow, Anthony Losongco, and Ryan Ginstrom. 2000. Heaviness vs. newness: The effects of structural complexity and discourse status on constituent ordering. *Language*, 76:28–55.

Paul Baker, Andrew Hardie, Tony McEnery, Hamish Cunningham, and Robert Gaizauskas. 2002. Emille: a 67-million word corpus of indic languages: data collection, mark-up and harmonization. In *Proceedings of LREC 2002*, pages 819–827. Lancaster University.

Rafiya Begum and Dipti Misra Sharma. 2017. Development and analysis of verb frame lexicon for hindi. *Linguistics and Literature Studies*, 5(1):1–22.

Rajesh Bhatt, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Misra Sharma, and Fei Xia. 2009. A multi-representational and multi-layered treebank for Hindi/urdu. In *Proceedings of the Third Linguistic Annotation Workshop*, ACL-IJCNLP '09, pages 186–189, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bock and Z. Griffin. 2000. The persistence of structural priming: Transient activation or implicit learning. *Journal of Experimental Psychology*, 2(120):177–192.

J.Kathryn Bock. 1986. Syntactic persistence in language production. *Cognitive Psychology*, 18(3):355 – 387.

Kathryn Bock and David E. Irwin. 1980. Syntactic effects of information availability in sentence production. *Journal of Verbal Learning and Verbal Behavior*, 19(4):467 – 484.

Miriam Butt and Tracy Holloway King. 1996. Structural topic and focus without movement. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the First LFG Conference*. CSLI Publications, Stanford.

Wallace Chafe. 1976. Givenness, contrastiveness, definiteness, subjects, topics, and point of view. *Subject and topic*.

H. H. Clark and S. E. Haviland. 1977. Comprehension and the Given-New Contract. In R. O. Freedle, editor, *Discourse Production and Comprehension*, pages 1–40. Ablex Publishing, Hillsdale, N. J.

P.R. Clarkson and A. J. Robinson. 1997. Language model adaptation using mixtures and an exponentially decaying cache. In *In Proceedings of ICASSP-97*, pages 799–802.

Samvit Dammalapati, Rajakrishnan Rajkumar, and Sumeet Agarwal. 2021. Effects of Duration, Locality, and Surprisal in Speech Disfluency Prediction in English Spontaneous Speech. In *Proceedings of the Society for Computation in Linguistics*, volume 4, page 10.

Vera Demberg and Frank Keller. 2008. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193–210.

Vera Demberg, Asad B. Sayeed, Philip J. Gorinski, and Nikolaos Engonopoulos. 2012. Syntactic surprisal affects spoken word duration in conversational contexts. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 356–367, Stroudsburg, PA, USA. Association for Computational Linguistics.

Victor S Ferreira and Elizabeth R Schotter. 2013. Do verb bias effects on sentence production reflect sensitivity to comprehension or production factors? *Quarterly Journal of Experimental Psychology*, 66(8):1548–1571.

Alex B Fine and T Florian Jaeger. 2016. The role of verb repetition in cumulative structural priming in comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 42(9):1362.

Alex B Fine, T Florian Jaeger, Thomas A Farmer, and Ting Qian. 2013. Rapid expectation adaptation during syntactic comprehension. *PloS one*, 8(10):e77661.

Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Edward Gibson. 2000. Dependency locality theory: A distance-based theory of linguistic complexity. In Alec Marantz, Yasushi Miyashita, and Wayne O'Neil, editors, *Image, Language, brain: Papers from the First Mind Articulation Project Symposium*. MIT Press, Cambridge, MA.

Stefan Th. Gries. 2005. Syntactic priming: A corpus-based approach. *Journal of Psycholinguistic Research*, 34(4):365–399.

John Hale. 2001. A probabilistic Earley parser as a psycholinguistic model. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, NAACL '01, pages 1–8, Pittsburgh, Pennsylvania. Association for Computational Linguistics.

Michael AK Halliday. 1970. Language structure and language function. *New horizons in linguistics*, 1:140–165.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Samar Husain and Himanshu Yadav. 2020. Target complexity modulates syntactic priming during comprehension. *Frontiers in Psychology*, 11:454.

T. Florian Jaeger and Elizabeth Norcliffe. 2009. The cross-linguistic study of sentence production: State of the art and a call for action. *Language and Linguistic Compass*, 3(4):866–887.

T. Florian Jaeger and Neal Snider. 2007. Implicit learning and syntactic persistence: Surprisal and cumulativity. *University of Rochester Working Papers in the Language Sciences*, 3:26–44.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 133–142, New York, NY, USA. ACM.

Elsi Kaiser and John C Trueswell. 2004. The role of discourse context in the processing of a flexible word-order language. *Cognition*, 94(2):113–147.

Ayesha Kidwai. 2000. *XP-Adjunction in Universal Grammar: Scrambling and Binding in Hindi-Urdu: Scrambling and Binding in Hindi-Urdu*. Oxford studies in comparative syntax. Oxford University Press.

Roland Kuhn and Renato De Mori. 1990. A cache-based natural language model for speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 12(6):570–583.

Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. University of Chicago press.

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

Haitao Liu, Chunshan Xu, and Junying Liang. 2017. Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of Life Reviews*, 21:171 – 193.

Barbara J Luka and Lawrence W Barsalou. 2005. Structural facilitation: Mere exposure effects for grammatical acceptability as evidence for syntactic priming in comprehension. *Journal of Memory and Language*, 52(3):436–459.

K.P. Mohanan and Tara Mohanan. 1994. Issues in word order in south asian languages: Enriched phrase structure or multidimensionality? In Miriam Butt, Tracy Holloway King, and Gillian Ramchand, editors, *Theoretical perspectives on word order in South Asian languages*, pages 153–184. Center for the Study of Language and Information, Stanford, CA.

Umesh Patil, Gerrit Kentner, Anja Gollrad, Frank Kügler, Caroline Féry, and Shravan Vasishth. 2008. Focus, word order and intonation in Hindi. *Journal of South Asian Linguistics*, 1(1):55–72.

C. K. Perera and A. K. Srivastava. 2016. Animacy-based accessibility and competition in relative clause production in Hindi and malayalam. *Journal of Psycholinguistic Research*, 45(4):915–930.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 433–440, Stroudsburg, PA, USA. Association for Computational Linguistics.

Martin J. Pickering and Holly P. Branigan. 1998. The Representation of Verbs: Evidence from Syntactic Priming in Language Production. *Journal of Memory and Language*, 39(4):633–651.

Rajakrishnan Rajkumar, Marten van Schijndel, Michael White, and William Schuler. 2016. Investigating locality effects and surprisal in written english syntactic choice phenomena. *Cognition*, 155:204–232.

Rajakrishnan Rajkumar and Michael White. 2014. Better surface realization through psycholinguistics. *Language and Linguistics Compass*, 8(10):428–448. ISSN: 1749-818X.

Sidharth Ranjan, Sumeet Agarwal, and Rajakrishnan Rajkumar. 2019. Surprisal and Interference Effects of Case Markers in Hindi Word Order. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 30–42, Minneapolis, Minnesota. Association for Computational Linguistics.

Sidharth Ranjan, Rajakrishnan Rajkumar, and Sumeet Agarwal. 2021. Locality and Expectation Effects in Hindi Preverbal Constituent Ordering. *Cognition*, in press.

David Reitter, Frank Keller, and Johanna D. Moore. 2011. A computational cognitive model of syntactic priming. *Cognitive Science*, 35(4):587–637.

C Scheepers. 2003. Syntactic priming of relative clause attachments: persistence of structural configuration in sentence production. *Cognition*, 89:179–205.

Neal Snider. 2009. Similarity and structural priming. In *Proceedings of the 31th Annual Conference of the Cognitive Science Society*, pages 815–820.

Adrian Staub. 2015. The effect of lexical predictability on eye movements in reading: Critical review and theoretical interpretation. *Language and Linguistics Compass*, 9(8):311–327.

Andreas Stolcke. 2002. SRILM — An extensible language modeling toolkit. In *Proc. ICSLP-02*.

David Temperley. 2007. Minimization of dependency length in written English. *Cognition*, 105(2):300–333.

David Temperley. 2008. Dependency-length minimization in natural and artificial languages. *Journal of Quantitative Linguistics*, 15(3):256–282.

Malathi Thothathiri, Daniel G. Evans, and Sonali Poudel. 2017. Verb bias and verb-specific competition effects on sentence production. *PLOS ONE*, 12(7):1–18.

Kristen M Tooley and Matthew J Traxler. 2010. Syntactic priming effects in comprehension: A critical review. *Language and Linguistics Compass*, 4(10):925–937.

Marten van Schijndel and Tal Linzen. 2018. A neural model of adaptation in reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4704–4710.

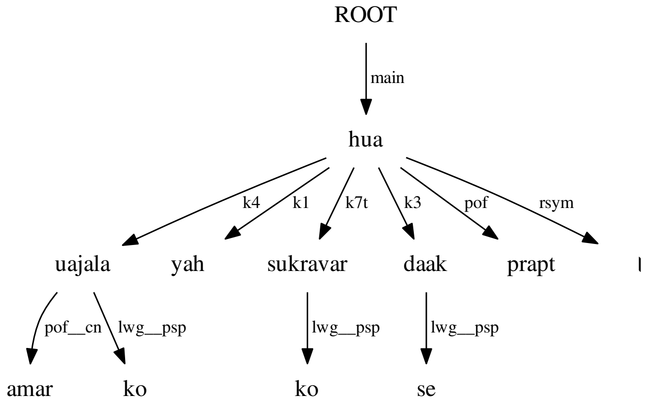
S. Vasishth. 2004. Discourse context and word order preferences in Hindi. *Yearbook of South Asian Languages*, pages 113–127.

Justine B Wells, Morten H Christiansen, David S Race, Daniel J Acheson, and Maryellen C MacDonald. 2009. Experience and sentence processing: Statistical learning and relative clause comprehension. *Cognitive psychology*, 58(2):250–271.

Marta Wierzba and Gisbert Fanselow. 2020. Factors influencing the acceptability of object fronting in german. *The Journal of Comparative Germanic Linguistics*, 23(1):77–124.

Himanshu Yadav, Ashwini Vaidya, and Samar Husain. 2017. Keeping it simple: Generating phrase structure trees from a Hindi dependency treebank. In *TLT*.

Eunkyung Yi, Jean-Pierre Koenig, and Douglas Roland. 2019. Semantic similarity to high-frequency verbs affects syntactic frame selection. *Cognitive Linguistics*, 30(3):601–628.



(a) Dependency tree

Label	Dependency relation
Invariant syntactic relations	
k1	subject/agent
k2	object/patient
k3	instrument
k4	object/recipient
k7t	location in time
Complex predicate relation	
pof	parts of conjunct verb
pof_cn	parts of compound noun
Local word group (lwg)	
lwg_psp	postposition
lwg_vaux	auxilliary verb
Symbols	
rsym	symbol relation

(b) Dependency relations

Figure 1: Example HUTB dependency tree and relation labels

A Variant Generation

(2) **Context sentence**

amar ujala-ki bhumika nispaksh rehti hai
Amar Ujala-GEN role unbiased remain be.PRS.SG

Amar Ujala's role remains unbiased.

(3) a. amar ujala-ko **yah** *sukravar*-ko daak-se prapt hua [Given-Given = 0] **(Reference)**
Amar Ujala-ACC **it** *friday*-on post-INST receive be.PST.SG

Amar Ujala received **it** by post on *Friday*.

b. **yah** amar ujala-ko *sukravar*-ko daak-se prapt hua [Given-Given = 0] **(Variant 1)**

c. *sukravar*-ko **yah** amar ujala-ko daak-se prapt hua [New-Given = -1] **(Variant 2)**

This work uses sentences from the Hindi-Urdu Treebank (HUTB) corpus of dependency trees (Bhatt et al., 2009) containing well-defined subject and object constituents. Figure 1 displays the dependency tree (and a glossary of relation labels) for reference sentence 3a. The grammatical variants were created using an algorithm that took as input the dependency tree corresponding to each HUTB reference sentence. The re-ordering algorithm permuted the preverbal dependents of the root verb and linearized the resulting tree to obtain variant sentences. For example, corresponding to the reference sentence 3a and its root verb "hai" (see figure 1a), the preverbal constituents with parents as "ujala", "yah", "suravar", "daak", and "prapt" were permuted to generate the artificial variants (3b and 3c). The ungrammatical variants were automatically filtered out using dependency relation sequences (denoting grammar rules) attested in the gold standard corpus of HUTB trees. In the dependency tree 1a, "k4-k1", "k7t-k1", "k3-k7t", and "pof-k3" are dependency relation sequences. In cases where the total number of variants exceeded 100[7], we chose 99 non-reference variants randomly along with the reference sentence.

[7]Higher and lower cutoffs do not affect our results.

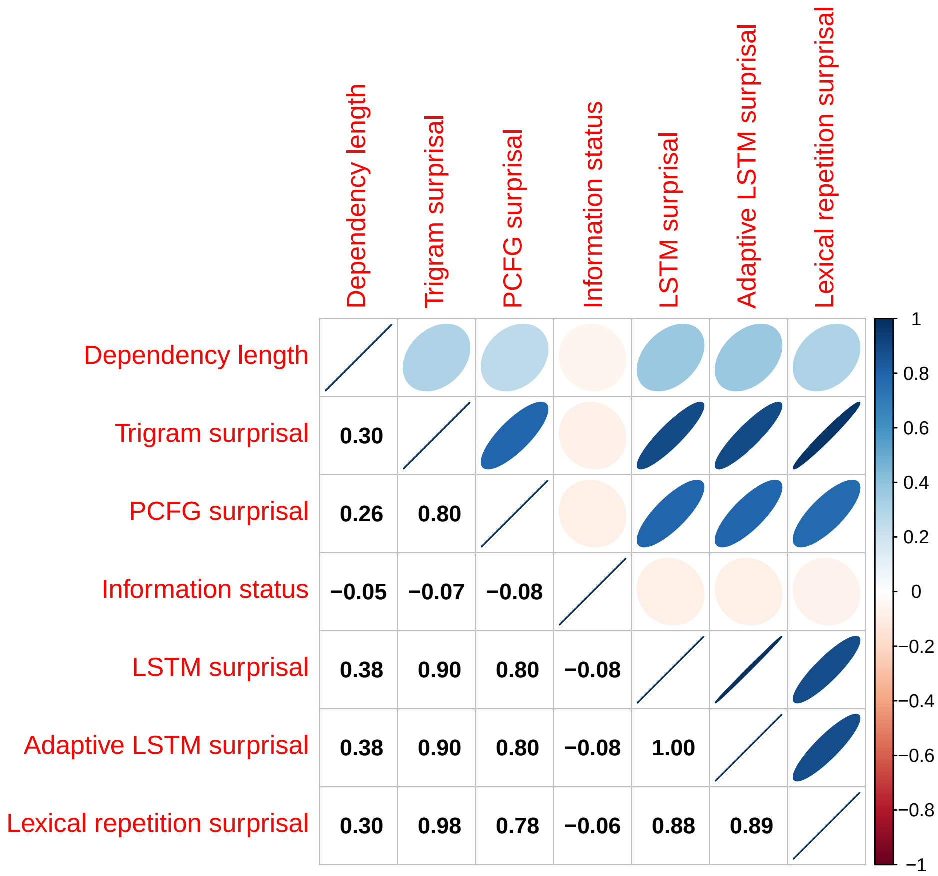


Figure 2: Pearson's coefficient of correlation between different pairs of predictors

B Levin's Verb Class and Case Density 903

Verb Types	**Case density**	**Freq**	**Freq (%)**
GIVE	0.45	372	18.64
DO	0.39	726	36.37
COMMUNICATION	0.67	264	13.23
MOTION	0.39	93	4.66
SOCIAL	0.4	242	12.12
PERCEPTION	0.32	36	1.8
DESTROY	0.63	34	1.7
LODGE	0.32	95	4.76
PUT	0.4	52	2.61
OTHERS	0.43	82	4.11
Full	0.44	1996	100

Table 6: Levin's verb semantic classes and case density (i.e., number of case markers per constituent in a sentence)

C Argument Ordering and Case Density

Alternation	Case density	Freq	Freq (%)
S-DO-IO	0.48	185	9.27
S-DO	0.39	1417	70.99
S-IO	0.59	394	19.74
Full	0.44	1996	100

Table 7: Argument ordering and case density (i.e., number of case markers per constituent in a sentence)

D Levin's classes of verbs within Double Object (S-IO-DO) alternation

Verb Lemma	Frequency	Freq (%)	Verb Types	Freq (%)
chah	127	1.37	SOCIAL	2.59
nawaja	5	0.05		
mil	5	0.05		
bech	104	1.12		
daal	99	1.07	PUT	2.13
jutaa	75	0.81		
pilaa	23	0.25		
dikha	28	0.3	PERCEPTION	0.3
badal	99	1.07	LODGE	1.07
de	3240	34.92	**GIVE**	**57.82**
saup	1090	11.75		
bhej	569	6.13		
maang	419	4.52		
dilaa	46	0.5		
kar	1737	18.72	DO	24.03
karaa	465	5.01		
chipaa	23	0.25		
ban	5	0.05		
kah	883	9.52	COMMUNICATION	12.06
sunaa	198	2.13		
likh	23	0.25		
bataa	15	0.16		
Full (S-IO-DO)	9278	100		12.74% of 72388

Table 8: Levin's syntactico-semantic classes of verbs within S-DO-IO data points from Table 5

E GIVE Verb Class Regression Model

Predictor	$\hat{\beta}$	$\hat{\sigma}$	t
intercept	**1.50**	0.002	638.32
trigram surprisal	**-0.11**	0.013	-8.57
dependency length	**0.01**	0.003	2.78
pcfg surprisal	**-0.08**	0.004	-18.87
IS score	**0.02**	0.002	10.01
lex-rept surprisal	0.01	0.012	0.46
lstm surprisal	**0.08**	0.036	2.25
adaptive lstm surprisal	**-0.36**	0.037	-9.86

Table 9: Regression model on lemma verb GIVE data set (14094 data points; all significant predictors denoted by |t|>2)

F Double Object (S-DO-IO) Alternation Regression Model

Predictor	$\hat{\beta}$	$\hat{\sigma}$	t
intercept	**1.50**	0.003	506.77
trigram surprisal	**-0.14**	0.017	-8.30
dependency length	**0.02**	0.003	6.20
pcfg surprisal	**-0.11**	0.005	-20.8
IS score	**0.02**	0.003	5.43
lex-rept surprisal	**0.06**	0.016	4.07
lstm surprisal	**0.31**	0.081	3.81
adaptive lstm surprisal	**-0.59**	0.081	-7.23

Table 10: Regression model on double object construction S-DO-IO data set (9278 data points; all significant predictors denoted by |t|>2)

G Information Profile for IO-fronted Example

Reference sentence 3a is correctly predicted by the model containing adaptive LSTM surprisal and all other features (i.e., *base1+g* in Table 4) but a model without adaptive LSTM surprisal (i.e., base1) predicts the variant Example 3b. Table 11 (first block) presents the exact scores of different predictors for the referent-variant pairs (3a and 3b). All predictors but LSTM and adaptive LSTM surprisal assign high score for the reference sentence with respect to its paired variant. Adaptive LSTM surprisal assigns a low per-word surprisal for the phrase *amar ujala* when it comes at the first position in the reference sentence (3a) with respect to when it comes at the second position in the variant (3b), potentially modeling *givenness* as this word occurred in the previous sentence (2) as well. See Figure 3 for the information profile of the reference-variant pairs.

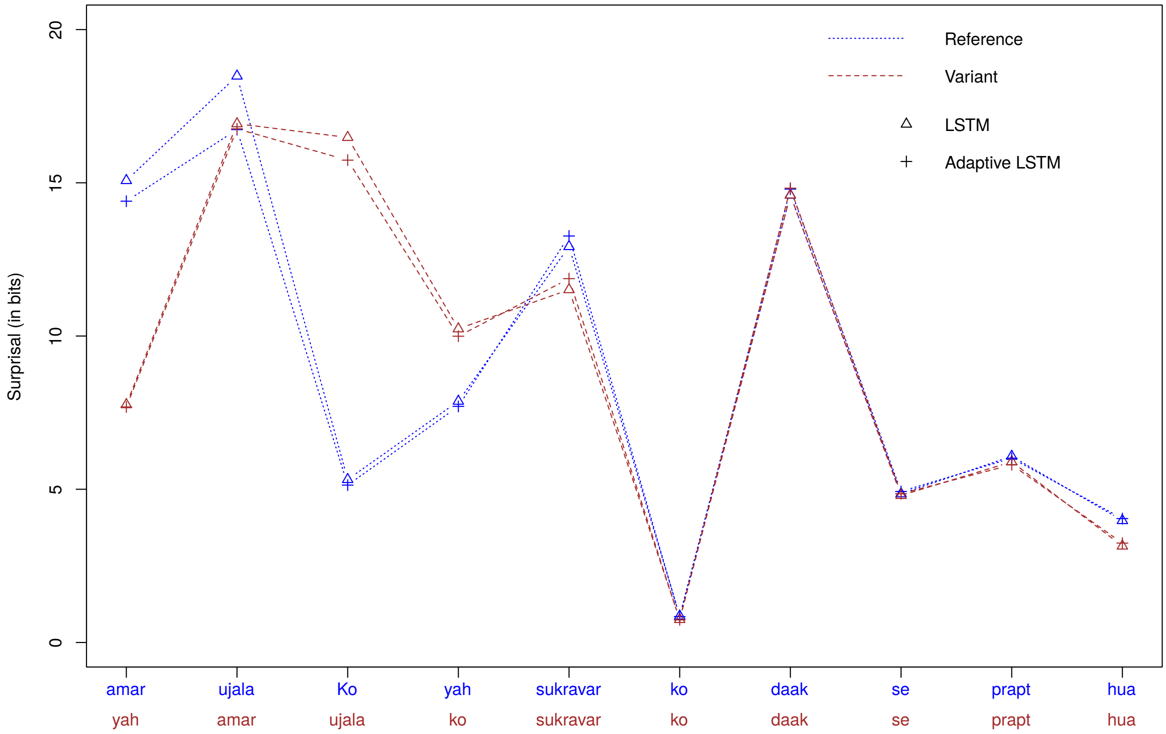


Figure 3: Information profile for the reference-variant pair 3a and 3b

H Information Profile for Double Object Ordering Example

To see how adaptive LSTM surprisal is able to capture ordering preferences, see Example 5:

(4) **Context Sentence**

collingwood 8 aur jones 0 aur blackville 10-par hi harbhajan-ki firki-ka sikaar ban gaye
collingwood 8 and jones 0 and blackville 10-PSP EMPH harbhajan-GEN spin-GEN victim become-PST.PL

Collingwood became the victim of Harbhajan's spin on 8 and Jones on 0 and Blackville on just 10.

(5) a. plunket 14-par pathan-ki gend-par Gambhir-ko kaetch de baethe **(Reference)**
plunket 14-PSP pathan-GEN ball-PSP gambhir-GEN catch give.PST.SG

Plunket ended up giving a catch to Gambhir on 14 off Pathan's bowling.

b. 14-par plunket pathan-ki gend-par gambhir-ko kaetch de baethe **(Variant)**

The LSTM LM when adapted to the previous sentence (4) learns the argument structure of the verb 'become' *(ban)* which when tested on referent-variants pairs (5) assigns a lower surprisal score to reference sentence (5a) than its competing variant (5b) owing to similar double object construction for the verb 'GIVE' *(de)* in reference sentence (see Table 11 for sentence-level predictor values, and see Figure 4 for information profiles).

Type		Trigram surp	Deplen	PCFG surp	IS score	LSTM surp	Adaptive LSTM surp	Repetition surp
Example 3a	Reference	24.69	18	61.13	0	**91.80**	**89.52**	23.80
Example 3b	Variant	23.80	20	60.67	0	93.78	93.17	22.19
Example 5a	Reference	34.27	24	107.04	0	173.06	**156.88**	36.45
Example 5b	Variant	33.92	23	105.11	0	171.49	165.86	36.45

Table 11: Predictor scores for reference-variant pairs

Non-canonical HUTB Sentences	**Frequency Count (%)**	**Baseline Perplexity**	**Adapted Perplexity (Prev1)**	**Perplexity Dip (Prev1)**	**Adapted Perplexity (Prev5)**	**Perplexity Dip (Prev5)**
DO	133 (1%)	183.92	103.40	-80.52	77.33	-106.59
IO	101 (0.76%)	138.78	88.26	-50.52	68.45	-70.33

Table 12: Effect of adaptation on discourse sentences (Prev1: Preceding one sentence in discourse, Prev5: Preceding five sentences in discourse)

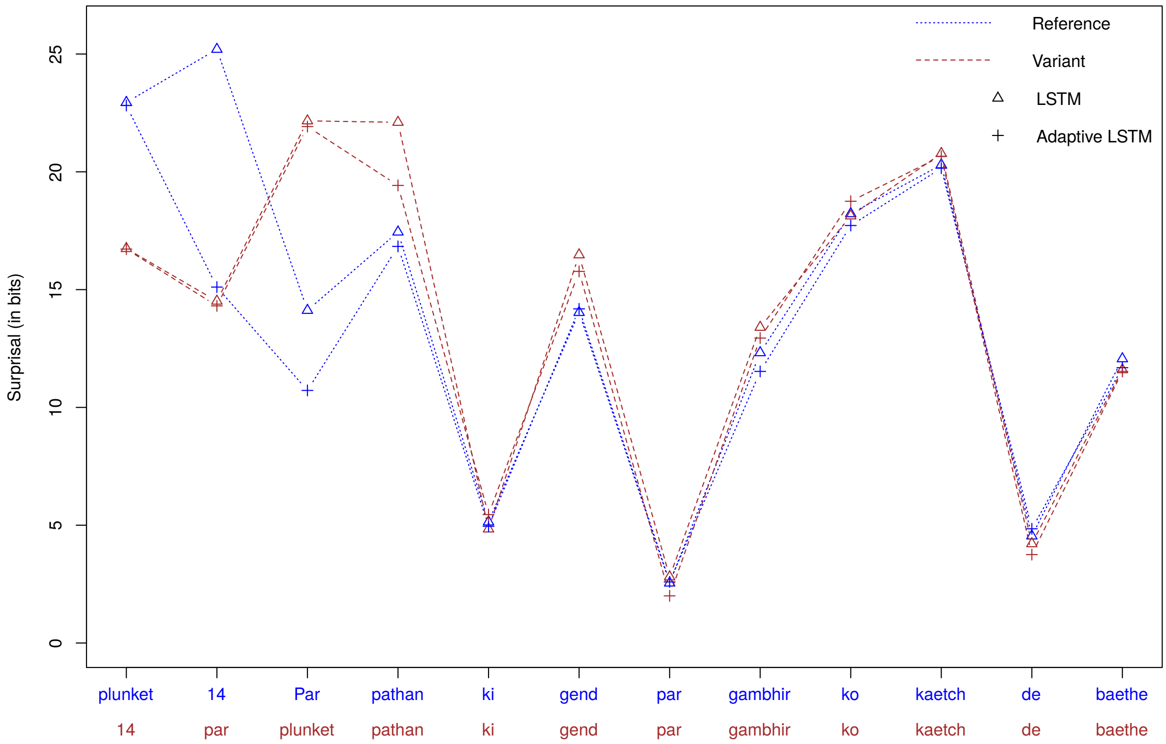


Figure 4: Information profiles for the reference-variant pair 5a and 5b

I Contextual Adaptation on One Vs. Multiple Sentences for DO/IO Constructions

We investigated if adapting the LSTM LM to the preceding five contextual sentences instead of one contextual sentence will help predict word-ordering patterns better for IO/DO constructions. Table 12 showcases perplexity dip on test sentences during 3 vs. 5 contextual sentence adaptation. Table 13 highlights classification accuracy of different models containing combination of features. Our results indicate that adding *Prev5-adaptive* LSTM surprisal in the machine learning model above and beyond every other features including *Prev1-adaptive* surprisal does not significantly boost prediction accuracy for both IO- and DO-fronted subset.

Type	Baseline	+ Prev5 Adaptive LSTM
DO	81.06	81.12
IO	89.65	89.73

Table 13: Prediction performance (Direct objects (DO: 1663 points), Indirect Objects (IO: 1353 points)); Baseline denotes *base1+g* shown in Table 4; bold denotes McNemar's two-tailed significance compared to baseline model in the same row