Modeling morphosyntactic agreement as neural search: a case study of Hindi-Urdu

Anonymous ACL submission

Abstract

Agreement is central to the morphosyntax of many natural languages. Within contemporary linguistic theory, agreement relations have often been analyzed as the result of a structure-sensitive search operation. Neural language models, which lack an explicit bias for this type of operation, have shown mixed success at capturing morphosyntactic agreement phenomena. This paper develops an alternative neural model that formalizes the search operation in a fully differentiable way using gradient neural attention, and evaluates the model's ability to learn the complex agreement system of Hindi-Urdu from a large-scale dependency treebank and smaller synthetic datasets. We find that this model outperforms standard architectures at generalizing agreement patterns to held-out examples and structures.

1 Introduction

Agreement is central to the morphosyntax of many natural languages (e.g., Moravcsik, 1978; Corbett, 2006; Baker, 2008). For example, in Hindi-Urdu sentences such as (1), the main verb and auxiliary agree in number and gender with the subject (as indicated by **bold**; examples here from Bhatt, 2005).

(1) **Rahul** kitaab **paRh-taa thaa**
Rahul.M book.F read-Hab.MSg be.Pst.MSg
Rahul used to read (a/the) book.

Across languages, agreement systems are sensitive to a wide yet restricted range of properties: grammatical categories and features such as Case, grammatical functions such as subject and object, structural positions such as specifier and complement, syntactic relations of dominance and c-command, as well as syntactic locality (shortest-path node distance). Agreement is also distinguished by being 'fallible' (Preminger, to appear): when no suitable controller for agreement exists, the target can take on default features (e.g., masculine and singular).

Verb agreement in Hindi-Urdu illustrates much of this complexity. For example, in (2), the verb and auxiliary agree with the Nominative object instead of the Ergative subject (cf. the Nominative subject in (1)). In (3), verb agreement 'fails' because the subject and object both have overt Case (Ergative and Accusative). Most strikingly, Hindi-Urdu allows 'long-distance' agreement (LDA) as in (4): when all of the local noun phrase arguments have overt Case marking, a verb can agree with the Nominative object of an embedded clause.

(2) Rahul ne **kitaab paRh-ii thii**
Rahul.M Erg book.FSg read-Pfv.FSg be.Pst.FSg
Rahul had read (a/the) book.

(3) Rahul ne kitaab ko paRh-aa thaa
Rahul.M Erg book.F Acc read-Pfv be.Pst.MSg
Rahul had read the book.

(4) Vivek ne [**kitaab parh-nii**] **chaah-ii**
Vivek.M Erg book.F read-Inf.F want-Pfv.FSg
Vivek wanted to read the book.

In this paper, we develop a neural model of morphosyntactic agreement that is capable of representing intricate agreement systems like those attested cross-linguistically, and evaluate its ability to learn the system of Hindi-Urdu from a large dependency treebank and from much smaller synthetic datasets. We begin by situating our model in the context of morphosyntactic theory and previous computational approaches to agreement. Following many contemporary theoretical proposals, our model formalizes agreement as structure-dependent *search* from targets, or *probes*, to controllers, or *goals*. As in some previous models, agreement is implemented with soft neural *attention* (and other differentiable mechanisms) rather than by symbolic tree traversal and feature copying.

2 Related research

2.1 Morphosyntactic theory

In some contemporary linguistic theories, agreement is a fundamental structure-building operations of syntax (e.g., Chomsky, 1995; Deal, 2015). In others, agreement is treated as postsyntactic: a part of morphology that operates on fully-formed syntactic structures (e.g., Bobaljik, 2008). Within both approaches, there is broad consensus that agreement relations are established by tree-based *search* (e.g., Preminger, to appear; Baker, 2008; Ke, 2023).

The details of the search operation remain controversial. Preminger (to appear) argues for strictly serial and 'downward' search in which each agreement probe explores the nodes of its c-command domain in a preset order and halts when it finds a suitable goal, or fails to find a goal before reaching terminal and blocking 'phase' nodes (resulting in default agreement). Others argue for different directionality, allowing a probe to optionally or obligatorily look 'upwards' to nodes that c-command it (e.g., Bjorkman and Zeijlstra, 2019; Baker, 2008). Still others argue for more elaborate operations that can occur as part of the search (Béjar and Rezac, 2009; Deal, 2015), or propose alternatives conditions for halting search (Deal, 2015).

The neural model that we propose is postsyntactic, insofar as it takes complete syntactic structures as inputs, but is otherwise compatible with many theoretical frameworks and varieties of search. We assume minimally that the input structures consist of nodes, that nodes are specified for grammatical category (e.g., noun vs. verb), that some nodes have specifications for phi-features (e.g., person, number, gender) and other morphosyntactically relevant properties such as Case (e.g., Nominative vs. Ergative), that some nodes are designated as agreement probes (or as having 'uninterpretable' phi-features to be satisfied by agreement), and that nodes enter into (labeled) syntactic relations of dominance or dependency with one another. The model is architecturally agnostic about search directionality and our application to Hindi-Urdu uses both 'downward' and 'upward' probing.

2.2 Neural models

Previous computational research has explored whether recurrent (RNN) and transformer models can capture morphosyntactic agreement (Linzen et al., 2016; Li et al., 2023; Bacon and Regier, 2019; Goldberg, 2019), which mixed success. Evaluating on English subject-verb agreement, Linzen et al. (2016) find that RNNs require explicit supervision of verb inflection to approximate structure-sensitive dependencies, despite seemingly high accuracy when trained only on a language modeling task. More robust sensitivity to structure is found for transformer architectures (Goldberg, 2019; Wilson et al., 2023), though these models are still not entirely unaffected by distractors and are more susceptible to linearly close distractors than humans.

Previous models further struggle to capture agreement dependencies for languages with more complex agreement phenomena. Ravfogel et al. (2018) find that recurrent neural networks have difficulty learning the agreement system of Basque, in which verbs agree with all of their arguments, instead showing some reliance on surface heuristics instead of syntactic structure. A cross-linguistic evaluation of transformers (Bacon and Regier, 2019), following (Goldberg, 2019), finds that transformers struggle significantly with agreement in a handful of languages, such as Persian, Basque, and Finnish, as well as noting their sensitivity to distractors even when performance is high.

Similar results have been found for verb agreement in French (Li et al., 2023). Evaluating a recurrent neural network and a transformer on two somewhat different agreement patterns in the language, the authors find that both models achieve relatively high accuracy. However, they see a degradation in performance when surface heuristics — such as agreement with the linearly first or most recent noun phrase — fail to predict the correct output. Relatedly, while the attention patterns of the transformer model indicate that it appropriately distinguishes the two agreement patterns, the sensitivity to heuristics makes attention difficult to interpret in a syntactically coherent way.

A separate line of work explores models that explicitly learn agreement rules. Chaudhary et al. (2020) use a decision tree to extract rules predicting agreement across multiple languages in the Universal Dependencies family of treebanks (Nivre et al., 2020). While this works well for certain languages like Greek or Russian, performance varies widely from language to language and especially drops in 'zero-shot' settings with minimal training data. Importantly, this model operates only between nodes that are directly connected within a dependency tree, making it unable to capture long-distance agreement as in example (4) above.

Our contribution shares high-level aspects of these proposals, including the use of continuous embeddings and attention, but differs in its goals and scope. We do not treat morphosyntactic agreement as a language modeling problem, recurrent or otherwise, but rather follow syntactic theory in taking agreement to be essentially a relation among syntactic nodes.

The model that we propose establishes these relations through search — technically, iterative redistribution of attention among nodes — conditioned on the types of morphosyntactic relations and features that are relevant for agreement cross-linguistically. The model does not parse sentences or generate inflected wordforms: it is designed solely to capture agreement but, in virtue of being fully differentiable, could be incorporated into larger neural models for parsing, inflection, or other applications. It has a small number of trainable parameters that can be set for particular agreement patterns, such as that of Hindi-Urdu.

3 Agreement in Hindi-Urdu

Agreement in the language of our case study has been extensively investigated within descriptive and theoretical linguistics (e.g., Pandharipande and Kachru, 1977; Bhatt and Keine, 2017; Mohanan, 1994; Bhatt, 2005; Kachru, 1970; Butt, 1993). A generalization that covers all of the examples in (1) - (4) is that Hindi-Urdu verbs and auxiliaries agree in gender and number with *the highest non-overtly Case-marked noun phrase*, where all Cases other than Nominative/Absolutive are overt.

The notion of 'highest' can be defined in many technical ways (e.g., in terms of proximity to a Tense of Inflection node), but basically tracks the well-known accessibility hierarchy subject > direct object > indirect object > other (e.g., Moravcsik, 1978; cf. Bobaljik, 2008). When there is no such noun phrase, masculine singular is used by default.

Hindi-Urdu is particularly remarkable for allowing long-distance agreement (LDA), and for the intricacies of agreements in light-verb constructions. Below we provide some further details about each of these phenomena, both of which occur in the datasets used to evaluate our model.

3.1 Long Distance Agreement

As illustrated in (4), verbs and auxiliaries can agree with non-overtly case marked arguments of infinitival embedded clauses when no 'higher' noun phrase is suitable. This agreement is optional: (5) below, which differs from (4) in that both the matrix and embedded verbs show default agreement, is also acceptable. Mahajan (1990) notes some interpretation differences between these cases, in which LDA seems to make the object more 'specific' (examples below based on Bhatt, 2005).

(5) Vivek ne [kitaab parh-naa]
Vivek.M Erg book.M read-Inf.M
chaah-aa
want-Pfv.MSg
Vivek wanted to read the book.

Bhatt (2005) also notes a parasitism in LDA, such that the matrix and embedded verb in which the matrix and infinitival verb must either both agree with the same noun phrase or both take default features. Neither (6a), which has infinitival agreement without LDA, nor (6b), which has LDA but no infinitival agreement, is acceptable.

(6) a. *Shahrukh ne [**tehnii kaat-nii**]
Shahrukh Erg branch.F
chaah-aa
cut-Inf.F want-Pfv.MSg
Shahrukh had wanted to cut the branch.

b. *Shahrukh ne [**tehnii** kaat-naa]
Shahrukh Erg branch.F cut-Inf.M
chaah-ii thii
want-Pfv.F be.Psts.FSg
Shahrukh had wanted to cut the branch.

However, this parasiticism may be dialect specific. Butt (1993) provides the following example in which the infinitival verb agrees with its embedded object but the matrix verb agrees with its Nominative subject.

(7) Ram [rotii khaa-nii] **caah-taa**
Ram.M bread.F eat.Inf.FSg want-Impf.M.Sg
thaa
was
Ram wanted to eat the bread.

Parasiticism motivates Bhatt to propose an additional operation that allows a probe to create dependencies between heads as part of the search process. Because of this, our present focus is on Butt's dialect, which is consistent with the root and infinitival verbs being separate probes, and set parasitic agreement aside for future research.

3.2 Light Verb Agreement

Light-verb constructions make up a majority of verbal predications in the language (e.g., Ahmed et al., 2012; Vaidya et al., 2019, 2016). In these constructions, a semantically less meaningful *light* verb (e.g. *kar* 'do', *ho* 'be') combines with a more meaningful noun, verb, or adjective to form a single predicate (example from Ahmed et al., 2012):

(8) a. **NAdiyah** hans **paR-I**
Nadiya.F.Sg laugh fall.Perf.F.Sg

Nadya burst out laughing.

b. YAsIn nE **mEz** s3Af
Yasin.M.Sg Erg table.F.Sg clean
k-I
do.Perf.F.Sg

Yasin made the table clean.

Agreement morphology in these constructions is always on the light verb. In both the V-V (8a) and Adj-V (8b) constructions, agreement follows from the same generalizations discussed earlier. However, a somewhat different pattern is found in N-V light verb constructions (examples below from Mohanan, 1994):

(9) a. Ilaa ne mohan kii **prasamsaa**
Ila Erg Mohan Gen praise.F
kii.
do.Perf.F

Ila praised Mohan.

b. Ilaa ne **kissaa** yaad
Ila.F Erg incident.M memory.F
kiyaa.
do.Perf.M

Ila remembered the incident.

c. Ilaa ne Mohan ko yaad
Ila Erg Mohan Acc memory.F
kiyaa
do.Perf.M

Ila remembered Mohan.

Unlike for Adj-V and V-V, members of one class of nouns in N-V constructions are eligible for agreement, as shown in (9a). When conjoined with a light verb, these nouns select either an object with oblique Case (e.g., Genitive in (9a)), or no object at all. Members of another class of nouns do not agree in N-V constructions, as in (9b, 9c). These form a predicate that selects for a direct case object, and agreement patterns follow as expected.

LDA and light-verb constructions can occur together. For example, in (10) the embedded infinite clause contains an N-V predicate. Both the matrix and embedded verbs agree with the noun component of the light verb (example below from Bhatt, 2005).

(10) Akbar ne [meri **madad kar-nii**] **chaahii**
Akbar Erg my.F help.F do.Inf.F want.Pfv.F
thii
be.pst.FSg

Akbar had wanted to help me.

4 Model

The neural model that we propose takes as input a syntactic tree, with certain nodes designated as agreement probes, and outputs predicted phi-feature values for each probe. Here we apply the model to Hindi-Urdu dependency trees (Bhat et al., 2017; Palmer et al., 2009) and synthetic trees based on those (see section 5.1.2). The edges between nodes are therefore directed and labeled by UD relations (Nivre et al., 2020, e.g., nsubj, obj, aux). Future research could experiment with constituency trees of the type that are more familiar in generative syntax, perhaps with minimal labeling of edges (e.g., specifier vs. complement).

Below we describe our neural embedding of dependency trees, the search process that distributes attention from probes to goals (or defaults), the transfer of predicted features to probes, and the loss function and other model details. We also describe two baseline transformer models, and compare the performance of our model to those on learning Hindi-Urdu verb agreement.

4.1 Tree embedding

The N nodes of a given syntactic tree are assumed to be arbitrarily ordered $(n_0, n_1, ...)$ and represented as feature vectors with minimal content. Specifically, separate one-hot vectors are used to embed grammatical category (e.g., noun, verb, auxiliary), each phi-feature (i.e., person, gender, number), and Case (e.g., Nominative, Accusative, Ergative). Zero vectors are used for unspecified features (e.g., root verbs are not specified for Case). These vectors are stacked into a single embedding $\mathbf{f}_i$ for each node n_i, and the embeddings are arranged as rows in a matrix $\mathbf{F}$ for the entire tree. Each node also has a separate one-hot embedding $\mathbf{d}_i$ of the dependency relation that it bears with its (unique) parent, and these are likewise arranged as rows in a matrix $\mathbf{D}$.

Because dependency relations are embedded as properties of child nodes, including edge labels would be redundant. Therefore, the edges of a tree are represented with a binary adjacency matrix $\mathbf{H}$, where $H_{ij} = 1$ indicates that node n_i is the head of node n_j. The transposed adjacency matrix $\mathbf{H}^T$ relates dependents in rows to heads in columns.

4.2 Searching from probes to goals

Each designated probe in a tree searches for a goal with which to agree by initially attending to itself and then iteratively redistributing attention to other nodes in the tree. The single-step redistribution of attention is determined by the topology of the tree, learnable weight vectors, and the softmax function. Multiple-step search simply iterates the process for a fixed topology and weights.

Within a language, probes seek goals that bear particular features and dependency relations. We formalize this with weight vectors $\mathbf{w}$ (of the same dimensionality as each $\mathbf{f}_i$) and $\mathbf{v}$ (of the same dimensionality as $\mathbf{d}_i$). The latter vector weights the 'downward' direction of dependencies — from heads to their dependents. To independently weight the 'upward' direction — from dependents to their heads — we use a vector $\mathbf{u}$. The model has two additional scalar weights, w_{self} and $w_{default}$, whose roles are explained momentarily.

Each node assigns a softmax score to its dependents on the basis of their features and their relations. These scores are represented in the $N \times N$ matrix $\mathbf{S}_{down}$ as defined below. Similarly, each node assigns a softmax score to its parent and these are collected in the $N \times N$ matrix $\mathbf{S}_{up}$. We have also found it useful to consider each node as having a reflexive 'self' dependency and associated softmax score, as in the definition of S_{self}. In our notation, $\odot$ is the elementwise (Hadamard) product and common broadcasting conventions are assumed.

$$\mathbf{S}_{down} = \mathbf{H} \odot [\, \underbrace{(\mathbf{F}\,\mathbf{w})^T}_{1\times N} + \underbrace{(\mathbf{D}\,\mathbf{v})^T}_{1\times N} \,]$$

$$\mathbf{S}_{up} = \mathbf{H}^T \odot [\, \underbrace{(\mathbf{F}\,\mathbf{w})^T}_{1\times N} + \underbrace{(\mathbf{D}\,\mathbf{u})}_{N\times 1} \,]$$

$$\mathbf{S}_{self} = \mathbf{I}_N \odot [\, \underbrace{(\mathbf{F}\,\mathbf{w})^T}_{1\times N} + w_{self} \,]$$

$$\hat{\mathbf{S}} = \mathbf{S}_{down} + \mathbf{S}_{up} + \mathbf{S}_{self}$$

$$\mathbf{S} = \left[\begin{array}{c:c} & w_{default} \\ \hat{\mathbf{S}} & \vdots \\ & w_{default} \\ \hdashline \multicolumn{2}{c}{\vec{0}} \end{array}\right]$$

$$\hat{\mathbf{A}} = \left[\hat{A}_{ij} = \begin{cases} S_{ij} & \text{if } S_{ij} \neq 0 \\ -\infty & \text{if } S_{ij} = 0 \end{cases} \right]$$

$$\mathbf{A} = \left[\mathbf{A}_i = \text{softmax}(\hat{\mathbf{A}}_i) \right]$$

The ith row of the $(N+1) \times (N+1)$ matrix $\mathbf{S}$ contains the logit scores that node n_i assigns to every other node n_j with which it is related by dependency (including self-dependency). To allow for the possibility of default agreement, we append a column vector with a constant score of $w_{default}$ and a row vector of all zeros. To convert these into probabilities, we mask out zero entries of $\mathbf{S}$ and take the row-wise softmax to get the transition matrix $\mathbf{A}$.

Let $\mathbf{p}$ be an $(N+1)$-dimensional binary vector that indicates which nodes of the tree are probes (with a final zero element for the default). The search process begins with each probe node attending fully to itself, as stated in the definition of $\mathbf{P}^{(0)}$. Search then proceeds — attention in each row is iteratively reallocated — through multiplying the previous $\mathbf{P}^{t-1}$ by $\mathbf{A}$.

$$\mathbf{P}^{(0)} = \mathbf{I}_{N+1} \odot \mathbf{p}$$

$$\mathbf{P}^{t} = \mathbf{P}^{t-1}\,\mathbf{A}$$

Observe that $\mathbf{A}$ is constant for a given tree and weights, and therefore can be precomputed, and further that rows of $\mathbf{P}^t$ for non-probe nodes are identically zero and could be ignored by sparse matrix implementations.

This iterative process is repeated for a fixed number of steps S, allowing a probe to iteratively explore the tree from its starting position. At the end of search, we take the final attention distribution for each probe to be the probe's final distributed 'position'. The features that then get copied to the probe are the weighted sum of phi-features from each node the probe attends to.

To compute this, we construct a phi-feature matrix $\hat{\mathbf{E}}_\phi$. The first N rows $\boldsymbol{e}_i$ of this matrix are each a concatenation of n_i's one-hot phi-feature embeddings, and the last row of this matrix a concatenation of a language's default phi-feature embeddings (for Hindi, Masculine Singular). This results in a $(N + 1) \times D_\phi$ matrix, where D_ϕ is the dimensionality of our concatenated embeddings.

The final predicted features can then be taken by multiplying $\mathbf{P}^{(S)}\mathbf{E}_\phi$.

$$\mathbf{Y}_{pred} = \mathbf{P}^{(S)}\mathbf{E}_\phi$$

4.2.1 Objective

During training, these predicted features are compared against the true phi-features on the node corresponding to the probe with the cross-entropy loss. Assuming perfect annotation of phi-features on probes and goals, this can be done directly. However, in our naturalistic treebank, a number of lexical items are annotated as having null features because they are uninflected despite being present in the morphology, such as auxiliaries or proper nouns that are not inflected for gender. To account for this, we take the argmax of the one-hot feature predictions as the discrete "prediction" of a probe, and mask out the parts of the cross-entropy loss where either this prediction or the true feature value is null. We similarly use the argmax at test time to determine the prediction of a probe.

5 Evaluation

We train our model on both naturalistic data from the Hindi UD treebank and synthetic data from a hand-designed dependency grammar. We assume the dialect from Butt (1993), which does not require a probe to additionally create dependencies during its search, and initialize a probe at each verb and auxiliary that each traverse the tree over 3 steps.

These results are then compared with two transformer baselines: a "cloze"-task transformer that must predict the phi-features of masked-out probes given the entire sentence, and a "language model (LM)"-task transformer that must predict the phi-features of masked-out probes given the preceding tokens in a sentence. Trees are linearized to surface order, and each token is given to the transformer as the same stacked one-hot encoding of its part-of-speech, case, phi-features, and dependency relation from its parent. Both transformers are one-head, one-layer, out-of-the-box models, trained only on the cross-entropy of the feature prediction of each probe encoding and the true phi features of each probe.

5.1 Datasets

5.1.1 Hindi UD Treebank

To evaluate our model on naturalistic data, we source trees from the Hindi Universal Dependencies Treebank (HDTB) (Bhat et al., 2017; Palmer et al., 2009), a treebank of manually annotated trees sourced from news articles, heritage and tourism sites, and a small amount of conversational data. This treebank contains 13,304 sentences in its training set, 1,659 sentences in its validation set, and 1,286 sentences in its test set.

5.1.2 Synthetic Data

For more controlled trees that capture the agreement phenomena of interest, we also handwrite a probabilistic grammar that generates basic syntactic trees along the UD framework. This grammar was written to capture the verbal agreement phenomena of interest, allowing us to test models without the annotation inconsistencies present in parts of HDTB, as well as to precisely control the types and frequencies of structures in the learning data. Specifically, we create production rules that generate transitive, intransitive, and ditransitive sentence frames in the perfective, progressive, and habitual aspects. Acceptable case marking patterns are defined according to Hindi's split-ergativity (Keine, 2007; Mohanan, 1994; Butt, 1995). Verbs can either be simple predicates or light verb constructions, and can also introduce an embedded infinitival clause. To account for optionality, we introduce a flag on the infinitival clauses in which LDA is desired. Embedded infinitivals can also introduce an agreeing light verb construction as in (10). The full grammar can be found in the Appendix.

A *Full Set* of trees is generated by normalizing probability across each structure type. This contains 1700 sentences total, of which 1000 are used for training, 200 reserved for validation, and 500 are reserved for evaluation. We additionally generate a *Minimal Training Set* of examples by enumerating over all 98 structures possible from our grammar and then randomly permute the number of auxiliaries and the phi features on the noun goals of each structure. This results in a set of 98 dependency trees. Finally, we create a *Relative Clause Test Set* by randomly appending relative clauses to 25% of the eligible goals in the 500-sentence test set.

These sets are used in three tasks: a **Synthetic (Synth)** task that is trained, validated, and tested on the *Full Set*, a **Minimal** task that is trained on the *Minimal Training Set* but validated and tested on the *Full Set*, and a **Relative Clause (ReCl)** task that is trained and validated on the *Full Set* but tested on the *Relative Clause Test Set*.

5.2 Results

The average test accuracies over 10 runs of each model on each dataset is shown in Table 1. Each model is trained for a minimum of 50 epochs and a maximum of 500 epochs, saving the checkpoint with the least validation loss for testing.

We find that the models perform similarly on the naturalistic treebank (HDTB). Our Search model slightly outperforms the two transformer models, but performance is generally comparable. The three models also perform similarly on the synthetic task, with both the Search model and the Cloze model reaching near-perfect test accuracy. However, the the two transformer models show dips in performance compared to the full synthetic task. This difference is even more apparent on the relative clause task: while our Search model remains near ceiling, both transformer models show significant dips in performance.

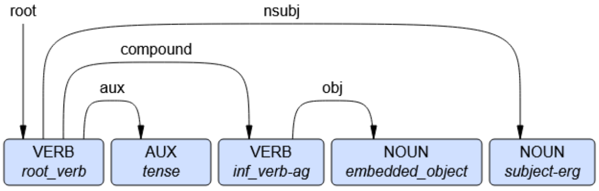
This suggests that our Search model is not only capable of matching performance to standard architectures on large-scale data, but also generalizing agreement patterns to held-out examples and structures. Especially in the case of the relative clause task, we hypothesize that the poor performance of the transformer models is due to an overreliance on heuristics– they have difficulty accouonting for the subject and object distractors introduced by the relative clauses. However, our structurally informed model is able to generalize to this new structure, maintaining the same high accuracy as on the synthetic task.

5.3 Learned Search Algorithm

To further examine the search algorithm that our model is learning, we dissect a particular model's learned weights (see Table 2). We can see that the model learns a coherent search algorithm for Hindi-Urdu agreement. Weights on phi-features are relatively similar, suggesting that the model does not prioritize a particular phi feature combination over another. Taking the weights on case and dependency relation together, we see that the model strongly prefers nominative subjects, but still prefers nominative objects over ergative subjects. Moreover, the default weight by itself is preferred over a ergative subject and an accusative object. To additionally handle LDA and light verb agreement, we see a very high weight on embedded infinitival clauses, likely to overcome the otherwise low priority given to words. On the other hand, low priority is given to noun child compound dependents of light verbs, as nominative nouns are already given high priority

We find that in practice, these weights encourage the softmax that the model takes at each time step to be close to one-hot. Thus, by examining the softmax scores at each time step, we can recover the "path" that a probe takes to reach its goal.

We sketch such a path in figure 1. In this example of long-distance agreement, both probes must take multiple steps to reach their goal. The verb probe must first take the compound transition to its embedded infinitival clause, from where it then sees the embedded object. The tense probe must travel a further step, first taking the auxiliary arc to the root verb, then the compound arc to the infinitival verb, and then finally the object arc to the embedded object. We can see that the model has learned a coherent and efficient search path from probe to target.



Verb Probe Scores

Node	Step 0	Step 1	Step 2	Step 3
root_verb	**1**	0.004	0.0004	0.0006
tense	0	0.00006	0.0004	0.005
inf_verb-ag	0	**0.879**	0.065	0.005
embedded_object	0	0.0003	**0.782**	**0.837**
subject-erg	0	0.058	0.0204	0.0071
default	0	0.0409	0.1055	0.136

Tense Probe Scores

Node	Step 0	Step 1	Step 2	Step 3
root_verb	0	**0.917**	0.0056	0.0005
tense	**1**	0.002	0.0002	0.0004
inf_verb-ag	0	0.0004	**0.807**	0.061
embedded_object	0	0.0004	0.001	**0.72**
subject-erg	0	0.0004	0.054	0.019
default	0	0.0565	0.1104	0.174

Figure 1: Trajectories for the verb and tense probe for a sentence with long distance agreement.

6 Conclusion and Future Directions

We present a efficient, minimal, neural network model that is able to accurately learn a search algorithm for the verb agreement pattern in Hindi-Urdu.

			Gender Accuracies			Number Accuracies			
		Model	Masculine	Feminine	Total	Singular	Plural	Total	Overall
Dataset	HDTB	Search	0.904 ± 0.026	0.904 ± 0.012	0.904 ± 0.019	0.990 ± 0.003	0.796 ± 0.032	0.96 ± 0.005	0.924 ± 0.011
		Cloze	0.965 ± 0.004	0.808 ± 0.011	0.924 ± 0.003	0.978 ± 0.003	0.846 ± 0.017	0.958 ± 0.001	0.909 ± 0.002
		LM	0.940 ± 0.009	0.782 ± 0.033	0.898 ± 0.006	0.975 ± 0.003	0.778 ± 0.02	0.945 ± 0.002	0.881 ± 0.005
	Synth	Search	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0
		Cloze	1.0 ± 0	0.999 ± 0.0008	0.999 ± 0.0003	1.0 ± 0	1.0 ± 0	1.0 ± 0	0.999 ± 0.0003
		LM	0.991 ± 0.005	0.992 ± 0.001	0.991 ± 0.003	0.984 ± 0.009	0.995 ± 0.001	0.989 ± 0.005	0.923 ± 0.007
	Minimal	Search	0.995 ± 0.01	0.995 ± 0.014	0.995 ± 0.011	0.99 ± 0.027	0.996 ± 0.014	0.993 ± 0.02	0.989 ± 0.029
		Cloze	0.989 ± 0.002	0.961 ± 0.013	0.978 ± 0.005	0.979 ± 0.000	0.996 ± 0.008	0.986 ± 0.004	0.968 ± 0.005
		LM	1.0 ± 0	0.0 ± 0	0.594 ± 0	0.858 ± 0.315	0.1533 ± 0.322	0.544 ± 0.035	0.3374 ± 0.033
	ReCl	Search	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0	1.0 ± 0
		Cloze	0.828 ± 0.013	0.904 ± 0.013	0.861 ± 0.004	0.833 ± 0.013	0.915 ± 0.016	0.870 ± 0.001	0.797 ± 0.004
		LM	0.846 ± 0.016	0.894 ± 0.009	0.867 ± 0.006	0.820 ± 0.041	0.928 ± 0.018	0.869 ± 0.015	0.802 ± 0.02

Table 1: Test accuracies for the UD model broken down by phi-feature type and value.

Case	Weight	Phi Features	Weight	Part of Speech	Weight	Dependencies	Weight
Nominative	7.96	Masculine	2.79	Noun	3.33	Subject Child	4.36
Accusative	-4.55	Feminine	3.04	Verb	0.003	Object Child	-4.79
Ergative	-6.58	Singular	2.77	Auxiliary	-1.74	Infinitival Clause Child	6.40
		Plural	2.85			Child Noun Compound of Light Verb	0.61
						Parent from Auxiliary	10.11
						Default	3.49

Table 2: A subset of learned weights for a model trained on synthetic data. Taken together, we see that the model prefers nominative (unmarked) subjects over all objects, nominative (unmarked) objects over ergative subjects, and default over ergative subjects and accusative objects. We also see a high preference for embedded infinitival clauses to overcome the otherwise low preference for verbs, and high preference for the parents of auxiliaries to allow auxiliary probes to travel to the matrix verb.

Despite this, however, our model is limited by a need for the full morphosyntactic specification of a given utterance. In order to predict the phi-features of a verb given the sentence it appears in, we require not only the full syntactic structure of that sentence, but also the exact phi features that appear in every potential noun goal. However, we note that the tree representations that our model uses are fully vectorized. We hope to explore methods of learning these vector representations of trees in future work, towards an end-to-end differentiable model of syntactic agreement that does not require fully specified trees.

The search process our model implements is also insufficient to capture certain agreement phenomenon. Agreement with coordinated phrases, in which phi-features of the full coordinated phrase must be computed from the features of its constituents (Bhatia, 2013), is difficult for our current model, which is specified to predict a simple weighted combination of existing phi-features in the tree. Our model is thus struggles unable to account for more complex agreement phenomena that depend on multiple goals (Shen, 2019). While our model is in theory able to combine featural information from multiple goals, we find that in practice the model converges to near-one-hot attention patterns after training. Future tests on these patterns will show if our model is capable of learning these patterns, and what modifications might facilitate this behavior.

Finally, our model as-implemented is not a perfect match for theories of agreement generally discussed by syntacticians. While most theoretic work on agreement is oriented on dependency trees, our model is tested on dependency trees. However, our model can be minimally adapted to operate any tree structure, including constituency trees, giving us the potential to questions regarding directionality or feature specification. Additionally, while many mainstream theories situate the agreement search process as part of the derivational structure-building process itself, our model assumes a fully postsyntactic structure. However, our model's fully differentiable nature allows it to easily slot into other neural models of structure buidling, which we hope to explore in future work.

References

Tafseer Ahmed, Miriam Butt, Annette Hautli, and Sebastian Sulger. 2012. A reference dependency bank for analyzing complex predicates. In *Proceedings of the Eighth International Conference on Language Re-*

sources and Evaluation (LREC'12), page 3145–3152, Istanbul, Turkey. European Language Resources Association (ELRA).

Geoff Bacon and Terry Regier. 2019. Does BERT agree? Evaluating knowledge of structure dependence through agreement relations. (arXiv:1908.09892). ArXiv:1908.09892 [cs].

Mark C. Baker. 2008. *The Syntax of Agreement and Concord*. Cambridge Studies in Linguistics. Cambridge University Press, Cambridge.

Riyaz Ahmad Bhat, Rajesh Bhatt, Annahita Farudi, Prescott Klassen, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Misra Sharma, Ashwini Vaidya, Sri Ramagurumurthy Vishnu, et al. 2017. The Hindi/Urdu treebank project. In *Handbook of Linguistic Annotation*. Springer Press.

Archna Bhatia. 2013. *Agreement in the context of coordination*. Scholars' Press.

Rajesh Bhatt. 2005. Long distance agreement in Hindi-Urdu. *Natural Language Linguistic Theory*, 23(4):757–807.

Rajesh Bhatt and Stefan Keine. 2017. Long-Distance Agreement. In Martin Everaert and Henk C. van Riemsdijk, editors, *The Wiley Blackwell Companion to Syntax, Second Edition*, pages 1–30. John Wiley & Sons, Inc., Hoboken, NJ, USA.

Bronwyn M. Bjorkman and Hedde Zeijlstra. 2019. Checking up on Agree. *Linguistic Inquiry*, 50(3):527–569.

Jonathan David Bobaljik. 2008. *Where's phi? Agreement as a post-syntactic operation*, pages 295–328.

Miriam Butt. 1993. A reanalysis of long distance agreement in Urdu. *Annual Meeting of the Berkeley Linguistics Society*, 19(11):52–63.

Miriam Butt. 1995. *The structure of complex predicates in Urdu*. Center for the Study of Language.

Susana Béjar and Milan Rezac. 2009. Cyclic agree. *Linguistic Inquiry*, 40(1):35–73.

Aditi Chaudhary, Antonios Anastasopoulos, Adithya Pratapa, David R. Mortensen, Zaid Sheikh, Yulia Tsvetkov, and Graham Neubig. 2020. Automatic extraction of rules governing morphological agreement. (arXiv:2010.01160). ArXiv:2010.01160 [cs].

Noam Chomsky. 1995. *The Minimalist Program*. The MIT Press. MIT Press, London, England.

G.G. Corbett. 2006. *Agreement*. Agreement. Cambridge University Press.

Amy Rose Deal. 2015. Interaction and satisfaction in phi-agreement. LingBuzz Published In: In Thuy Bui and Deniz Ozyildiz (eds.), Proceedings of NELS 45, Volume 1, pp. 179-192. Amherst: GLSA.

Yoav Goldberg. 2019. Assessing BERT's syntactic abilities. *arXiv*.

Yamuna Kachru. 1970. An introduction to Hindi syntax. *Journal of Linguistics*, 6(1):151–152.

Alan Hezao Ke. 2023. Can Agree and Labeling be reduced to minimal search? *Linguistic Inquiry*, page 1–22.

Stefan Keine. 2007. Reanalysing Hindi split-ergativity as a morphological phenomenon. *Linguistische Arbeits Berichte*, 85:73–127.

Bingzhi Li, Guillaume Wisniewski, and Benoît Crabbé. 2023. Assessing the capacity of transformer to abstract syntactic representations: A contrastive analysis based on long-distance agreement. *Transactions of the Association for Computational Linguistics*, 11:18–33.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Anoop Kumar Mahajan. 1990. *The A/A-bar distinction and movement theory*. Thesis, Massachusetts Institute of Technology. Accepted: 2008-02-28T15:45:42Z.

Tara Mohanan. 1994. *Argument structure in Hindi*. Center for the Study of Language.

Edith A. Moravcsik. 1978. Agreement. In Charles A. Ferguson Edith A. Moravcsik Joseph H. Greenberg, editor, *Universals of Human Language. Vol. IV: Syntax*, page 331–374. Stanford University Press, Stanford, CA.

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.

Martha Palmer, Rajesh Bhatt, Bhuvana Narasimhan, Owen Rambow, Dipti Misra Sharma, and Fei Xia. 2009. Hindi syntax: Annotating dependency, lexical predicate-argument structure, and phrase structure. In *The 7th International Conference on Natural Language Processing*, pages 14–17.

Rajeshwari Pandharipande and Yamuna Kachru. 1977. Relational grammar, ergativity, and Hindi-Urdu. *Lingua*, 41(3):217–238.

Omer Preminger. to appear. Phi-feature agreement in syntax. In Kleanthes K. Grohmann and Evelina Leivada, editors, *The Cambridge Handbook of Minimalism*. Cambridge University Press, Cambridge.

Shauli Ravfogel, Yoav Goldberg, and Francis Tyers. 2018. Can LSTM learn to capture agreement? the case of Basque. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, page 98–107, Brussels, Belgium. Association for Computational Linguistics.

Zheng Shen. 2019. The multi-valuation agreement hierarchy. *Glossa: a journal of general linguistics*, 4(11).

Ashwini Vaidya, Sumeet Agarwal, and Martha Palmer. 2016. Linguistic features for Hindi light verb construction identification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1320–1329, Osaka, Japan. The COLING 2016 Organizing Committee.

Ashwini Vaidya, Owen Rambow, and Martha Palmer. 2019. Syntactic composition and selectional preferences in hindi light verb constructions. *Linguistic Issues in Language Technology*, 17(1).

Michael Wilson, Zhenghao Zhou, and Robert Frank. 2023. Subject-verb agreement with seq2seq transformers: Bigger is better, but still not best.

A Synthetic grammar

Our synthetic grammar designed to capture the agreement phenomena of interest is shown below. Each row corresponds to an expansion rule of the grammar. The leftmost number of each row corresponds to the weight of that expansion rule, while the first entry immediately after the number corresponds to the parent node that the expansion rule targets. The remaining entries are nodes that get added to the tree as children of the parent node. Entries with parentheses are optional and generated with 50% probability. For example, the rule `1.35 root_verb subject-erg object-nom (tense)` denotes a rule with weight 1.35 that expands a `root_verb` node with an ergative subject child, an nominative object child, and an optional tense child. In practice, each node is fully specified for features, dependency relation, and part of speech, but this has been truncated here for readability.

```
# ROOT
2 R root_verb
1 R root_verb_prog

# HABITUAL AND PERFECTIVE
# Simple Transitive
1.35 root_verb subject-erg object-nom (tense)
1.35 root_verb subject-nom object-nom (tense)
1.35 root_verb subject-nom object-acc (tense)
1.35 root_verb subject-erg object-acc (tense)
# Simple Intransitive
2.7 root_verb subject-erg (tense)
2.7 root_verb subject-nom (tense)
# Simple Ditransitive
2.7 root_verb subject-erg object-dat object-nom (tense)
2.7 root_verb subject-nom object-dat object-nom (tense)
# Light Verb Constructions
0.385 root_verb subject-nom object-nom host_adj (tense)
0.385 root_verb subject-nom object-acc host_adj (tense)
0.385 root_verb subject-nom object-nom host_verb (tense)
0.385 root_verb subject-nom object-acc host_verb (tense)
0.385 root_verb subject-nom object-nom host_noun (tense)
0.385 root_verb subject-nom object-acc host_noun (tense)
0.385 root_verb subject-nom host_noun_agreeing (tense)
0.385 root_verb subject-erg object-nom host_adj (tense)
0.385 root_verb subject-erg object-nom host_verb (tense)
0.385 root_verb subject-erg object-nom host_noun (tense)
0.385 root_verb subject-erg host_noun_agreeing (tense)
0.385 root_verb subject-erg object-acc host_adj (tense)
0.385 root_verb subject-erg object-acc host_verb (tense)
0.385 root_verb subject-erg object-acc host_noun (tense)
# Infinitivals
1.08 root_verb subject-erg inf_verb-agree (tense)
1.08 root_verb subject-nom inf_verb-nonagree (tense)
1.08 root_verb subject-nom inf_verb-nonagree-acc (tense)
1.08 root_verb subject-erg inf_verb-nonagree (tense)
1.08 root_verb subject-erg inf_verb-nonagree-acc (tense)

# PROGRESSIVE
# Simple Transitive
1.2 root_verb_prog subject-nom object-nom aspect (tense)
1.2 root_verb_prog subject-nom object-acc aspect (tense)
# Simple Intransitive
2.4 root_verb_prog subject-nom aspect (tense)
# Simple Ditransitive
2.4 root_verb_prog subject-nom object-dat object-nom aspect (tense)
# Light Verb Constructions
0.34 root_verb_prog subject-nom object-nom host_adj aspect (tense)
0.34 root_verb_prog subject-nom object-acc host_adj aspect (tense)
0.34 root_verb_prog subject-nom object-nom host_verb aspect (tense)
0.34 root_verb_prog subject-nom object-acc host_verb aspect (tense)
0.34 root_verb_prog subject-nom object-nom host_noun aspect (tense)
0.34 root_verb_prog subject-nom object-acc host_noun aspect (tense)
0.34 root_verb_prog subject-nom host_noun-agreeing aspect (tense)
# Infinitivals = 1
2.4 root_verb_prog subject-nom inf_verb-nonagree aspect (tense)

# EXPANSIONS
# Light Verb Construction Expansions
```

```
1 host_agreeing object-gen
1 host_agreeing object-loc
1 host_agreeing object-ins
1 host_agreeing

# Agreeing Infinitival Expansions
1 inf_verb-agreeing object-nom
1 inf_verb-agreeing host_noun-agreeing

# Non-Agreeing Infinitival Clause Expansions
1 inf_verb-non object-nom
1 inf_verb-non object-acc
```