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

Alan Zhou
Johns Hopkins University
azhou23@jhu.edu

Colin Wilson
Johns Hopkins University
colin.wilson@jhu.edu

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]

(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 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**
Rahul.M Erg book.FSg read-Pfv.FSg
thii
be.Pst.FSg
Rahul had read (a/the) book.

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

(4) Vivek ne [**kitaab** **parh-nii**]
Vivek.M Erg book.F read-Inf.F
chaah-ii
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 as well as 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 (*probes*) to controllers (*goals*). As in

[1] Example sentences provided throughout the paper follow the glossing and transliteration of the original sources.

Proceedings of the Society for Computation in Linguistics (SCiL) 2024, pages 227-239.
Irvine, California, June 27-29, 2024

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 operation 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 alternative conditions under which search halts (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 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 neural networks (RNNs) and transformer models can capture morphosyntactic agreement (Linzen et al., 2016; Li et al., 2023; Bacon and Regier, 2019; Goldberg, 2019), with 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 non-goal '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 auxiliary verbs agree with several local 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 overall high.

Similar results have been found for verb agreement in French (Li et al., 2023). Evaluating an RNN and a transformer on two different agreement patterns in French, 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 inflection. Additionally, 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 (postsyntactic) 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 in the matrix clause 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 or 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 agreement 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. For a more comprehensive view of Hindi-Urdu agreement and morphosyntax, we refer readers to original sources (e.g., Bhatt, 2005; Butt, 1995; Mohanan, 1994).

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 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 not infinitival agreement, is acceptable according to that source.

(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. We do not formalize this extra mechanism here, and therefore focus on Butt's dialect, which is consistent with the root and infinitival verbs being separate probes. Parasitic agreement should be addressed by future elaborations of the model.

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 (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 **k-I**
Yasin.M.Sg Erg table.F.Sg clean 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 from Mohanan, 1994):

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

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

c. Ilaa ne Mohan ko yaad kiyaa
Ila Erg Mohan Acc memory.F 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 (Mohanan, 1994). 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 (Nominative, Accusative, or Ergative) 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 from Bhatt, 2005).

(10) Akbar ne [meri **madad kar-nii**] **chaahii** **thii**
Akbar Erg my.F help.F do.Inf.F want.Pfv.F 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 5.2.2). 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, as well as 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 the minimal cross-linguistically motivated content. Specifically, separate one-hot vectors are used to embed grammatical category (e.g., noun, verb, auxiliary), each phi-feature separately (e.g., 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 $\mathbf{F}$ following the arbitrary node order. 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 $\mathbf{D}$.

To facilitate our search algorithm, two minor modifications are introduced for each tree. First, we create a 'self' connection from each node to itself that bears its own special dependency relation. This gives the model the option to 'stay' at a node during the search process, rather than being forced to pick from one of its neighbors. Additionally, we introduce a 'default' node to each tree that has in-going connections from every other node, but an out-going connection only to itself. This node is entirely featureless in terms of phi-features, part-of-speech, and Case during the search process, but is associated with default phi-features during the feature valuation step of the model (see below).

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 a stochastic transition matrix conditioned on the topology of the tree and learnable weight vectors via the softmax function. Multiple-step search simply iterates the same transition matrix for a fixed topology and weights.

Within a language, probes seek goals that bear particular features and dependency relations. We formalize this with two 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 weights the 'downward' direction of dependencies — from heads to their dependents. To independently weight the 'upward' direction — from dependents to their heads — we use another vector $\mathbf{u}$. The model has two additional scalar weights, w_{self} and $w_{default}$, which correspond to self and default node dependencies as described above.

Each node assigns a logit 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 logit score to its parent and these are collected in the $N \times N$ matrix $\mathbf{S}_{up}$. Finally, each node also assigns a score to itself according to the self dependency, represented in $\mathbf{S}_{self}$. In our notation, $\odot$ is the elementwise (Hadamard) product and common broadcasting conventions are assumed.

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

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

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

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

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

$$\mathbf{A}_i = \text{softmax}(\hat{\mathbf{A}}_i)$$

The ith row of the $N \times N$ 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 and the default node). To convert these into probabilities, we mask out zero entries of $\mathbf{S}$ and take the row-wise softmax to derived the single-step transition matrix $\mathbf{A}$.

Note that the zero-one encoding of adjacencies in $\mathbf{H}$ and $\mathbf{I}_N$ ensure that the transition probabilities of $\mathbf{A}$ are only non-zero from nodes to their immediate neighbors (including the default node). Additionally, the default node has a transition probability of 1 to itself (hence 0 to all other nodes).

Let $\mathbf{p}$ be an N-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 with a one-hot vector at its own position, as stated in the definition of $\mathbf{P}^{(0)}$. Search then proceeds — attention in each row is iteratively reallocated — simply by multiplying the previous $\mathbf{P}^{t-1}$ with $\mathbf{A}$.

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

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

Observe that $\mathbf{A}$ is constant for a given tree and weights, and can therefore be precomputed prior

to search by all probes in the tree. Observe further that rows of $\mathbf{P}^t$ for non-probe nodes are identically zero; these could be ignored in sparse matrix implementations.

The search process is repeated for a fixed number of steps t_{max}, allowing a probe to iteratively explore the tree from its starting position. At the end of search, we take the final attention scores of a probe to be a distribution over the goal nodes that a probe 'returns.' The entire search can thus be viewed as a Markov process, with the nodes of a tree being the states over which the transition matrix operates (e.g., the default node is an absorbing state).

Intuitively, our formalization results in a *gradient breadth-first search*. Note that our structurally-informed transition matrix ensures that for any individual step, attention can only reallocated from a node to itself or its immediate neighbors. Thus, at step t, each probe's attention can only be allocated among nodes that are at most t steps away from its probe node. We additionally observe that after learning this process converges to an approximation of *greedy search*, in which attention for a given probe is nearly one-hot at each step.

4.3 Feature Valuation

The features that are 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 $\mathbf{E}_\phi$, whose ith row contains the concatenation of n_i's one-hot phi-feature embeddings, or the concatenated phi-feature embeddings for a language's default phi-features (masculine singular for Hindi-Urdu) if n_i is the default node. This results in a $N \times D_\phi$ matrix, where D_ϕ is the dimensionality of our concatenated embeddings.

The predicted features for a probe are then the result of multiplying $\mathbf{P}^{(t_{max})}$ by $\mathbf{E}_\phi$:

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

4.4 Objective

During training, the model's predicted features are compared with the correct phi-features on each probe node by cross-entropy loss. Assuming perfect annotation of phi-features on probes and goals, this can be done directly. However, in our naturalistic treebank, many lexical items that are not overtly inflected for phi-features are mislabeled as having null phi-features (e.g. proper nouns and certain auxiliaries that do not inflect for gender). To account for this, we take the argmax of the one-hot feature predictions as the discrete 'prediction' for 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 predicted phi-features that each probe returns.

5 Evaluation

We trained our model on both naturalistic data from the Hindi UD treebank and synthetic data from a hand-designed dependency grammar. As noted above, we assume the dialect from Butt (1993), which does not require a probe to additionally create dependencies during its search. Therefore, we initialized a probe at each verb and auxiliary. A modest value of $t_{max} = 3$ steps was found to be sufficient for these data sets. To test our model's structural generalization ability, we also increased this to $t_{max} = 5$ on a relative clause distractor task (see section 5.2.2 below).

5.1 Transformer Baselines

We compared our model, referred to below as Search, against two transformer baselines: a Cloze transformer that predicts the phi-features of masked-out probes given the entire sentence, and a language model (LM) transformer that predicts the phi-features of masked-out probes given the preceding tokens in a sentence. These two transformer models are identical in architecture, featuring a one-head, one-layer transformer encoder, followed by a linear decoder that maps each token's embedding to a phi-feature prediction.[2]

Linearized (surface order) trees were used as inputs to these models, with each token embedded by stacking one-hot vectors for its part-of-speech, Case, phi-features, and dependency relation from its parent. We further tested 'structural' versions of the models in which the token's parent index is also given as part of the stacked one-hot embedding, but found that this additional information had a negligible impact on model performance in most settings.

[2]These transformers are much smaller than state-of-the-art models. However, our preliminary tests with larger models showed drastic decreases in performance, likely due to the smaller size of our training data.

5.2 Datasets

5.2.1 Hindi UD Treebank

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

5.2.2 Synthetic Data

For more controlled data that includes the agreement phenomena of interest, we also wrote a probabilistic grammar that generates basic syntactic trees within the UD framework. This grammar allowed us to evaluate 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 created 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-Urdu'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 generated a *Minimal Training Set* of examples by enumerating over all 98 structures possible from our grammar and then randomly permuting the number of auxiliaries and the phi-features on noun goals. This resulted in a set of 98 dependency trees. Finally, we created a *Relative Clause Test Set* by randomly appending relative clauses to 25% of the eligible goals in the original 500-sentence test set.

These sets were 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*.

5.3 Results

The average test accuracies over 10 runs of each model are shown in Table 1. Each model was trained for a minimum of 1000 steps and a maximum of 100,000 steps, saving the checkpoint with the lowest validation loss for testing.

We find that the models performed similarly on the naturalistic treebank (HDTB). Our Search model slightly outperforms the transformer models without structural information, but not the Cloze model with access to parent information. Each model also performed similarly on the synthetic task, with both the Search model and the Cloze models reaching perfect or near-perfect test accuracy. However, compared to our Search model, the transformer models see a larger drop-off in synthetic accuracy in the low-data setting of the minimal task. This suggests that our Search model is particularly well-suited to low-resource data.

Most strikingly, while our Search model maintains near-perfect test accuracy on the relative clause generalization task, all of the baseline transformer models show a significant drop in performance compared to other tasks. This demonstrates an ability of our model to generalize agreement patterns to held-out examples and structures that the transformer models do not share. We hypothesize that the poor performance of the latter is due to an overreliance on heuristics— they have difficulty avoiding agreement with the subject and object distractors introduced by the relative clauses because they lack the structural biases of Search.

We note that the transformer models performed similarly with or without access to structural information (parent indexes), with the possible exception of the Cloze model on the naturalistic treebank. This suggests that these models do not consistently assign high weights to structural relations relative to other cues such as dependency relation or part of speech.

5.4 Learned Search Algorithm

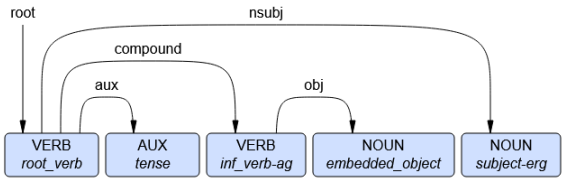
To further examine the search algorithm that our model induces, we dissect a subset of a particular model's learned weights (see Table 2). We can see that the model has learned a coherent search algorithm for Hindi-Urdu agreement. Weights on all phi-features are similar, suggesting that the model

			Gender Accuracies			Number Accuracies			
		Model	Masculine	Feminine	Total	Singular	Plural	Total	Overall
Dataset	HDTB	Search	0.904 ± 0.026	$\mathbf{0.904 \pm 0.012}$	0.904 ± 0.019	$\mathbf{0.990 \pm 0.003}$	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	$\mathbf{0.846 \pm 0.017}$	0.958 ± 0.001	0.909 ± 0.002
		Cloze*	$\mathbf{0.970 \pm 0.006}$	0.867 ± 0.019	$\mathbf{0.942 \pm 0.004}$	0.987 ± 0.003	0.826 ± 0.013	$\mathbf{0.963 \pm 0.002}$	$\mathbf{0.942 \pm 0.003}$
		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
		LM*	0.947 ± 0.012	0.778 ± 0.028	0.902 ± 0.004	0.977 ± 0.004	0.785 ± 0.020	0.947 ± 0.002	0.888 ± 0.003
	Synth	Search	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$
		Cloze	$\mathbf{1.0 \pm 0}$	0.999 ± 0.0008	0.999 ± 0.0003	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	0.999 ± 0.0003
		Cloze*	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$	$\mathbf{1.0 \pm 0}$
		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	0.983 ± 0.007
		LM*	0.992 ± 0.005	0.992 ± 0.000	0.992 ± 0.003	0.989 ± 0.007	0.989 ± 0.008	0.989 ± 0.006	0.982 ± 0.008
	Minimal	Search	$\mathbf{0.995 \pm 0.01}$	$\mathbf{0.995 \pm 0.014}$	$\mathbf{0.995 \pm 0.011}$	$\mathbf{0.99 \pm 0.027}$	$\mathbf{0.996 \pm 0.014}$	$\mathbf{0.993 \pm 0.02}$	$\mathbf{0.989 \pm 0.029}$
		Cloze	0.990 ± 0.004	0.969 ± 0.015	0.982 ± 0.007	0.979 ± 0.002	0.995 ± 0.009	0.986 ± 0.004	0.972 ± 0.007
		Cloze*	0.989 ± 0.004	0.951 ± 0.017	0.973 ± 0.005	0.980 ± 0.002	0.977 ± 0.018	0.978 ± 0.007	0.960 ± 0.007
		LM	0.987 ± 0.003	0.906 ± 0.073	0.954 ± 0.031	0.943 ± 0.094	0.899 ± 0.165	0.924 ± 0.125	0.896 ± 0.131
		LM*	0.982 ± 0.026	0.846 ± 0.109	0.927 ± 0.059	0.876 ± 0.198	0.853 ± 0.212	0.866 ± 0.156	0.812 ± 0.180
	ReCl	Search	$\mathbf{0.996 \pm 0.008}$	$\mathbf{1.0 \pm 0}$	$\mathbf{0.998 \pm 0.005}$	$\mathbf{0.995 \pm 0.010}$	$\mathbf{1.0 \pm 0}$	$\mathbf{0.997 \pm 0.006}$	$\mathbf{0.997 \pm 0.005}$
		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
		Cloze*	0.829 ± 0.026	0.890 ± 0.025	0.855 ± 0.004	0.852 ± 0.008	0.876 ± 0.014	0.863 ± 0.004	0.787 ± 0.011
		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
		LM*	0.828 ± 0.035	0.866 ± 0.022	0.844 ± 0.013	0.833 ± 0.015	0.878 ± 0.023	0.853 ± 0.008	0.774 ± 0.02

Table 1: Test accuracies for each model broken down by phi-feature type and value, where * indicates that a transformer model had access to structural information about node parents.

does not prioritize any particular phi-feature combination (e.g., masculine singular) over others. Taking the weights on Case and dependency relation together, we see that the model strongly prefers Nominative subjects, and prefers Nominative objects over Ergative subjects. Moreover, the default weight by itself is preferred over an 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 verbs. On the other hand, low priority is given to light verb noun compound dependents, as Nominative nouns are already given high priority

In practice, the learned weights of Search 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 one such 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 can then transition to the embedded object. The tense probe requires an additional iteration, 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. The model has learned a coherent and efficient search path from each probe to the correct goal.



Verb Probe Scores

Node	Step 0	Step 1	Step 2	Step 3
root_verb	**1**	0.004	0.000	0.001
tense	0	0.000	0.0004	0.005
inf_verb-ag	0	**0.879**	0.065	0.005
embedded_object	0	0.000	**0.782**	**0.837**
subject-erg	0	0.058	0.020	0.007
default	0	0.041	0.106	0.136

Tense Probe Scores

Node	Step 0	Step 1	Step 2	Step 3
root_verb	0	**0.917**	0.006	0.001
tense	**1**	0.002	0.000	0.000
inf_verb-ag	0	0.000	**0.807**	0.061
embedded_object	0	0.000	0.001	**0.72**
subject-erg	0	0.000	0.054	0.019
default	0	0.057	0.110	0.174

Figure 1: Attention patterns at each step for the verb and tense probe for a sentence with long distance agreement.

Case	Weight	Phi-Features	Weight	Part of Speech	Weight	Dependencies	Weight
Nominative	7.96	Masculine	2.79	Noun	3.33	Subject dependent	4.36
Accusative	-4.55	Feminine	3.04	Verb	0.003	Object dependent	-4.79
Ergative	-6.58	Singular	2.77	Auxiliary	-1.74	Infinitival Clause dependent	6.40
		Plural	2.85			Light Verb Noun Compound dependent	0.61
						Auxiliary head	10.11
						Default node	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, the default dummy node over Ergative subjects and Accusative objects. We also see a high preference for embedded infinitival clauses (6.40) to overcome the otherwise low preference for verbs (0.003), and a high preference for the heads of auxiliaries (10.11) to allow auxiliary probes to travel to the matrix verb.

6 Conclusion and Future Directions

Artificial neural networks are often seen as black-box models with little or no inductive bias. We present a counterpoint to this view, creating an efficient, minimal, and interpretable neural network model that possesses a strong inductive bias for agreement as structurally-informed search.

Our goal in building this model is not necessarily to adjudicate between neural networks and traditional symbolic models as opposing models of language or cognition. Rather, we aim to show that insights from symbolic modeling can provide useful inductive biases for neural network models. Indeed, our structure-dependent model is capable of correctly learning a search algorithm for the agreement pattern in Hindi-Urdu, and matches or exceeds performance compared to much larger models without such biases. Our model is also capable of achieving near-perfect performance on a structural generalization task, something that more generic models could not match.

While we tested our model on the complex agreement system of Hindi-Urdu, our model is theoretically capable of accounting for a range of agreement phenomena cross-linguistically. For example, an agreement system in which a verb obligatorily agrees with the subject of a clause can be easily accounted for by setting a high weight on the *subject* dependency (nsubj). Our model can also capture the various sensitivities that agreement has with Case in languages other than Hindi-Urdu. Nepali, for example, allows agreement with Ergative subjects as well as Nominative subjects, while Gujarati allows agreement with Accusative objects but not Ergative subjects (Bhatt, 2005). Our model can capture the Nepali case with an equal setting of our Case weights for Nominative and Ergative, and the Gujarati case with a positive weight on Accusative and a negative weighting of Ergative.

However, there do exist some agreement phenomenon that our model cannot yet account for. Our model is specified to return a simple weighted combination of phi-features among existing nodes in a tree, making it impossible to account for agreement with coordinated noun phrases that have phi-features computed by 'resolution rules' applied to their constituents (Bhatia, 2011). Additionally, the weighted combination that our model returns is often exactly the phi-features from a single node, as the model typically converges to near one-hot attention patterns after training. Thus, it seems unlikely that the model can account for agreement phenomena that depend on multiple goals (Shen, 2019) — though distribution of attention over multiple nodes does remain a logical possibility and may be encouraged by some training patterns.

Finally, the model as deployed here does not provide a perfect match to the theories of agreement typically proposed by syntacticians. While most theoretical work on agreement is oriented around constituency trees, our model was trained and tested on dependency trees. However, the model can be minimally adapted to operate on any tree structure, including constituency trees, giving us the potential to address questions regarding directionality and feature weighting in other settings.

Acknowledgements

Thanks to Paul Smolensky, members of the Phon-Morph and the Neurosymbolic Computation labs at Johns Hopkins University, and two anonymous SCiL reviewers for useful comments and questions. We would also like to thank Rajesh Bhatt for his insights on Hindi-Urdu agreement and syntax. This research was partially supported by NSF grant BCS-1941593 to CW.

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 Resources 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 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. 2011. *Agreement in the context of coordination Hindi as a case study*. Ph.D. thesis, University of Illinois at Urbana-Champaign.

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.

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. In Daniel Harbour, David Adger, and Susana Béjar, editors, *Phi-Theory: Phi features across interfaces and modules*, pages 295–328.

Miriam Butt. 1993. A reanalysis of long distance agreement in Urdu. In B.Kaiser and C. Zoll, editors, *Proceedings of the Nineteenth Annual Meeting of the Berkeley Linguistics Society: General Session and Parasession on Semantic Typology and Semantic Universals (1993)*, volume 19, page 52–63.

Miriam Butt. 1995. *The structure of complex predicates in Urdu*. CSLI Publications, Stanford, CA.

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. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5212–5236, Online. Association for Computational Linguistics.

Noam Chomsky. 1995. *The Minimalist Program*. MIT Press, Cambridge, MA.

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

Amy Rose Deal. 2015. Interaction and satisfaction in phi-agreement. In Thuy Bui and Deniz Ozyildiz, *Proceedings of NELS 45*, Volume 1, page 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*, pages 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*. Ph.D. thesis, Massachusetts Institute of Technology.

Tara Mohanan. 1994. *Argument structure in Hindi*. CSLI Publications, Stanford, CA.

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.

Michael Wilson, Zhenghao Zhou, and Robert Frank. 2023. Subject-verb agreement with seq2seq transformers: Bigger is better, but still not best. *Society for Computation in Linguistics*, 6:278–288.

A Synthetic grammar

Our synthetic grammar, designed to capture the agreement phenomena of interest in the paper, 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 are 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
```