Sequence-to-Sequence Networks Learn the Meaning of Reflexive Anaphora

Robert Frank[*]
Yale University
bob.frank@yale.edu

Jackson Petty[*]
Yale University
jackson.petty@yale.edu

Abstract

Reflexive anaphora present a challenge for semantic interpretation: their meaning varies depending on context in a way that appears to require abstract variables. Past work has raised doubts about the ability of recurrent networks to meet this challenge. In this paper, we explore this question in the context of a fragment of English that incorporates the relevant sort of contextual variability. We consider sequence-to-sequence architectures with recurrent units and show that such networks are capable of learning semantic interpretations for reflexive anaphora which generalize to novel antecedents. We explore the effect of attention mechanisms and different recurrent unit types on the type of training data that is needed for success as measured in two ways: how much lexical support is needed to induce an abstract reflexive meaning (i.e., how many distinct reflexive antecedents must occur during training) and what contexts must a noun phrase occur in to support generalization of reflexive interpretation to this noun phrase?

1 Introduction

Recurrent neural network architectures have demonstrated remarkable success in natural language processing, achieving state of the art performance across an impressive range of tasks ranging from machine translation to semantic parsing to question answering (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2016). These tasks demand the use of a wide variety of computational processes and information sources (from grammatical to lexical to world knowledge), and are evaluated in coarse-grained quantitative ways. As a result, it is not an easy matter to identify the specific strengths and weaknesses in a network's solution of a task.

In this paper, we take a different tack, exploring the degree to which neural networks successfully master one very specific aspect of linguistic knowledge: the interpretation of sentences containing reflexive anaphora. We address this problem in the context of the task of semantic parsing, which we instantiate as mapping a sequence of words into a predicate calculus logical form representation of the sentence's meaning.

(1) a. Mary runs → $\textsc{run}(\textsc{mary})$

b. John sees Bob → $\textsc{see}(\textsc{john}, \textsc{bob})$

Even for simple sentences like those in (1), which represent the smallest representations of object reflexives in English, the network must learn lexical semantic correspondences (e.g., the input symbol *Mary* is mapped to the output MARY and *runs* is mapped to RUN) and a mode of composition (e.g., for an intransitive sentence, the meaning of the subject is surrounded by parentheses and appended to the meaning of the verb). Of course, not all of natural language adheres to such simple formulas. Reflexives, words like *herself* and *himself*, do not have an interpretation that can be assigned independently of the meaning of the surrounding context.

[*] Equal contribution.

Proceedings of the 3rd Workshop on Computational Models of Reference, Anaphora and Coreference (CRAC 2020), pages 154–164,
Barcelona, Spain (online), December 12, 2020.

(2) a. Mary sees herself → SEE(MARY, MARY)

b. Alice sees herself → SEE(ALICE, ALICE)

In these sentences, the interpretation of the reflexive is not a constant that can be combined with the meaning of the surrounding elements. Rather, a reflexive object must be interpreted as identical to the meaning of verb's subject. Of course, a network could learn a context-sensitive interpretation of a reflexive, so that for any sentence with *Mary* as its subject, the reflexive is interpreted as MARY, and with *Alice* as its subject it is interpreted as ALICE. However, such piecemeal learning of reflexive meaning will not support generalization to sentences involving a subject that has not been encountered as the antecedent of a reflexive during training, even if the interpretation of the subject has occurred elsewhere. What is needed instead is an interpretation of the reflexive that is characterized not as a specific (sequence of) output token(s), but rather as an abstract instruction to duplicate the interpretation of the subject. Such an abstraction requires more than the "jigsaw puzzle" approach to meaning that simpler sentences afford.

Marcus (1998) argues that this kind of abstraction, which he takes to require the use of algebraic variables to assert identity, is beyond the capacity of recurrent neural networks. Marcus's demonstration involves a simple recurrent network (SRN, Elman 1990) language model that is trained to predict the next word over a corpus of sentences of the following form:

(3) a. A rose is a rose.

b. A mountain is a mountain.

All sentences in this training set have identical subject and object nouns. Marcus shows, however, that the resulting trained network does not correctly predict the subject noun when tested with a novel preamble '*A book is a* ...'. Though intriguing, this demonstration is not entirely convincing: since the noun occurring in the novel preamble, *book* in our example, did not occur in the training data, there is no way that the network could possibly have known which (one-hot represented) output should correspond to the reflexive for a sentence containing the novel (one-hot represented) subject noun, even if the network did successfully encode an identity relation between subject and object.

Frank et al. (2013) explore a related task in the context of SRN interpretation of reflexives. In their experiments, SRNs were trained to map input words to corresponding semantic symbols that are output on the same time step in which a word is presented. For most words in the vocabulary, this is a simple task: the desired output is a constant function of the input (*Mary* corresponds to MARY, *sees* to SEE, etc.). For reflexives however, the target output depends on the subject that occurs earlier in the sentence. Frank et al. tested the network's ability to interpret a reflexive in sentences containing a subject that had not occurred as a reflexive's antecedent during training. However, unlike Marcus' task, this subject and its corresponding semantic symbol did occur in other (non-reflexive) contexts in the training data, and therefore was in the realm of possible inputs and outputs for the network. Nonetheless, none of the SRNs that they trained succeeded at this task for even a single test example.

Since those experiments were conducted, substantial advances have been made on recurrent neural network architectures, some of which have been crucial in the success of practical NLP systems.

- **Recurrent units**: More sophisticated recurrent units like LSTMs (Graves and Schmidhuber, 2005) and GRUs (Cho et al., 2014) have been shown to better encode preceding context than SRNs.

- **Sequence-to-Sequence architectures**: The performance of network models that transduce one string to another, used in machine translation and semantic parsing, has been greatly improved by the use of independent encoder and decoder networks (Sutskever et al., 2014).

- **Attention mechanism**: The ability of a network to produce contextually appropriate outputs even in the context of novel vocabulary items has been facilitated by content-sensitive attention mechanisms (Bahdanau et al., 2016; Luong et al., 2015).

These innovations open up the possibility that modern network architectures may well be able to solve the variable identity problem necessary for mapping reflexive sentences to their logical form. In the experiments we describe below, we explore whether this is the case.

2 Experimental Setup

Our experiments take the form of a semantic parsing task, where sequences of words are mapped into logical form representations of meaning. Following Dong and Lapata (2016), we do this by means of a sequence-to-sequence architecture (Sutskever et al., 2014) in which the input sentence is fully processed by an encoder network before it is decoded into a sequence of symbols in the target domain (cf. Botvinick and Plaut 2006, Frank and Mathis 2007 for antecedents). This approach removes the need to synchronize the production of output symbols with the input words, as in Frank et al. (2013), allowing greater flexibility in the nature of semantic representations.

The sequence-to-sequence architecture is agnostic as to the types of recurrent units for the encoding and decoding phases of the computation, and whether the decoder makes use of an attention mechanism. Here, we explore the effects of using different types of recurrent units and including attention or not. Specifically, we examine the performance and training characteristics of sequence-to-sequence models based on SRNs, GRUs, and LSTMs with and without multiplicative attention (Luong et al., 2015).

In all experiments, we perform 5 runs with different random seeds for each combination of recurrent unit type (one layer of SRN, LSTM or GRU units for both the encoder and decoder) and attention (with or without multiplicative attention). All models used hidden and embedding of size of 256. Training was done using Stochastic Gradient Descent with learning rate of 0.01. Models were trained for a maximum of 100 epochs with early stopping when validation loss fails to decrease by 0.005 over three successive epochs.

We conduct all of our experiments with synthetic datasets from a small fragment of English sentences generated using a simple context-free grammar. This fragment includes simple sentences with transitive and intransitive verbs. Subjects are always proper names and objects are either proper names or a reflexive whose gender matches that of the subject. Our vocabulary includes 8 intransitive verbs, 7 transitive verbs, 15 female names, and 11 male names. The grammar thus generates 5,122 distinct sentences. All sentences are generated with equal probability, subject to the restrictions imposed by each experiment. We use a unification extension to CFG to associate each sentence with a predicate calculus interpretation. The symbols corresponding to the predicates and the entities in our logical language are identical with the verbs and names used by our grammar, yielding representations like those shown in (1) and (2). The output sequences corresponding to the target semantic interpretations include parentheses and commas as separate symbols. Quite clearly, this dataset does not reproduce the richness of English sentence structure or the distribution of reflexive anaphora, and we leave the exploration of syntactically richer domains for future work. However, even this simple fragment instantiate the kind of contextual variable interpretation found in all cases of reflexive interpretation and therefore it allows us to probe the ability of networks to induce a representation of such meanings.

As discussed in the previous section, we are interested in whether sequence-to-sequence models can successfully *generalize* their knowledge of the interpretation of sentences containing reflexives to ones having novel antecedents. To do this, we employ a *poverty of the stimulus* paradigm that tests for systematic generalization beyond a finite (and ambiguous) set of training data (Chomsky, 1980). In our experiments, we remove certain classes of examples from the training data set and test the effect on the network's success in interpreting reflexive-containing sentences. Each of our experiments thus defines a set of sentences that are withheld during training. The non-withheld sentences are randomly split 80%–10%–10% between training, validation, and testing sets. Accuracy for each set is computed on a sentence-level basis, i.e., an accurate output requires that all symbols generated by the model be identical to the target. Our experiments focus on two sorts of manipulations of the training data: (1) varying the number of lexical items that do and do not occur as the antecedents of reflexives in the training set, and (2) varying the syntactic positions in which the non-antecedent names occur. As we will see, both of these manipulations substantially impact the success of reflexive generalization in ways that vary across network types.

3 Experiment 1: Can Alice know herself?

In the first experiment, we directly test whether or not networks can generalize knowledge of how to interpret *herself* to a new antecedent. We withhold all examples whose input sequence includes the reflexive *herself* bound by the single antecedent *Alice*, of the form shown in (4).

(4) Alice *verbs* herself → *verb*(ALICE, ALICE)

Sentences of any other form are included in the training-validation-test splits, including those where *Alice* appears without binding a reflexive.

3.1 Results

All network architectures were successful in this task, generalizing the interpretation of *herself* to the novel antecedent *Alice*. Even the simplest networks, namely SRN models without attention, achieve 100% accuracy on the generalization set (sentences of the form shown in (4)). This is in sharp contrast the negative results obtained by Frank et al. (2013), suggesting an advantage for training with a language with more names as well as for instantiating the semantic parsing task in a sequence-to-sequence architecture as opposed to a language model.

4 Experiment 2: Doesn't Alice know Alice?

While the networks in Experiment 1 are not trained on sentences of the form shown in (4), they are trained on sentences that have the same target semantic form, namely sentences in which *Alice* occur as both subject and object of a transitive verb.

(5) Alice *verbs* Alice → *verb*(ALICE, ALICE)

In Experiment 2 we consider whether the presence of such semantically reflexive forms in the training data is helpful to networks in generalizing to syntactically reflexive sentences. We do this by further excluding sentences of the form in (5) from the training data.

4.1 Results

All architectures except SRNs without attention generalize perfectly to the held out items. Inattentive SRNs also generalize quite well, though only at a mean accuracy of 86%. While success at Experiment 1 demonstrates the networks' abilities to generalize to novel input contexts, success at Experiment 2 highlights how models can likewise generalize to produce entirely new outputs.

5 Experiment 3: Who's Alice and who's Claire?

So far, we have considered generalization of reflexive interpretation to a single new name. One possible explanation of the networks' success is that they are simply defaulting to the (held-out) ALICE interpretation when confronted with a new antecedent, as an elsewhere interpretation (but see Gandhi and Lake 2019 for reasons for skepticism). Alternatively, even if the network has acquired a generalized interpretation for reflexives, it may be possible that this happens only when the training data includes overwhelming lexical support (in Experiments 1 and 2, 25 out of the 26 names in our domain appeared in the training data as the antecedent of a reflexive). To explore the contexts under which networks can truly generalize to a range of new antecedents, we construct training datasets in which we progressively withhold more and more names in sentences of the forms shown in (6), i.e., those that were removed in Experiment 2.[1]

[1] Since *himself* and *herself* are different lexical items, it is unclear if the network will learn their interpretations together, and whether sentences containing *himself* will provide support for the interpretation of sentences containing *herself*. We therefore withhold only sentences of this form with names of a single gender. We have also experimented with witholding masculine reflexive antecedents from the training data, but the main effect remains the number of female antecedents that is withheld.

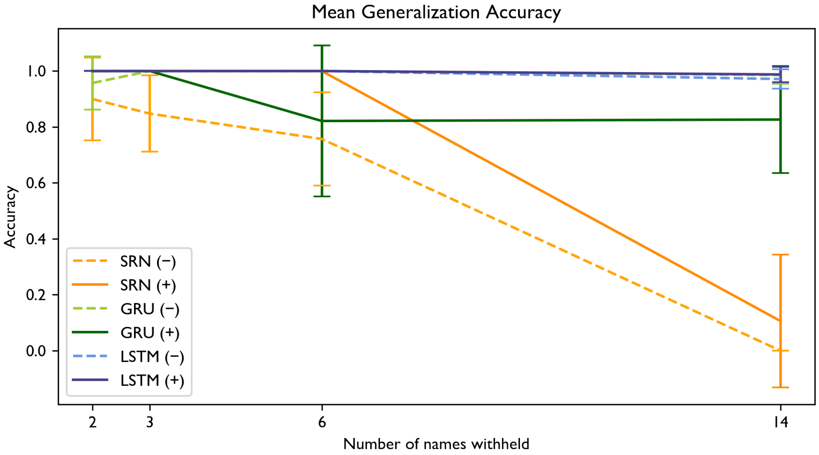

Figure 1: Mean generalization accuracy by number of names withheld in Experiment 3. The (+) or (−) next to the type of recurrent unit indicates the presence or absence of attention. Error bars display the standard deviation of accuracies.

(6) a. P *verbs* herself → $verb(P,P)$

b. P *verbs* P → $verb(P,P)$

Our domain contains 15 distinct feminine antecedents; we perform several iterations of this experiment, withholding progressively more feminine names from appearing in the contexts in (6), until only a single feminine name is included in the training data as the antecedent of a reflexive.

5.1 Results

As shown in Figure 1, reducing the set of names that serve as antecedents to reflexives in the training data resulted in lower accuracy on the generalization set. SRNs, especially without attention, show significantly degraded performance when high numbers of names are withheld from reflexive contexts during training. With attention, SRN performance degrades only when reflexives are trained with a single feminine antecedent (i.e., 14 names are held out). In contrast, LSTMs both with and without attention maintain near-perfect accuracy on the generalization set even when the training data allows only a single antecedent for the feminine reflexive *herself*. The performance of GRUs varies with the presence of an attention mechanism: without attention, GRUs achieve near perfect generalization accuracy even for the most demanding case (training with a single feminine antecedent), while the performance of GRUs with attention has mean accuracy near 80%.

We also explored how recurrent unit type and attention affect *how* models learn to generalize. One way to gauge this is by examining how quickly networks go from learning reflexive interpretation for a single name to learning it for every name. Table 1 shows the mean number of epochs it takes from when a network attains 95% accuracy on a single antecedent contexts[2] to when it has attained more than 95% accuracy on *all* held out antecedent contexts.[3]

This ‘time to learn’ highlights the disparate impact of attention depending on the type of recurrent unit; SRNs with attention and LSTMs with attention acquire the generalization much faster than their attentionless counterparts, while attention increases the length of time it takes for GRUs to learn for all but the condition in which 14 antecedents were withheld. Figure 2 illustrates another important aspect of reflexive generalization: it proceeds in a piecemeal fashion, where networks first learn to interpret reflexives for the trained names and then generalize to the held out antecedents one by one. In Figure 2 we show an SRN without attention, but the same pattern is representative of the other networks tested.

[2] An ‘antecedent context’ is the set of all reflexive sentences with a particular antecedent.

[3] Note that this doesn’t mean that models retained more than 95% accuracy on all contexts — some models learned a context, only to forget it later in training; this measurement does not reflect any such unlearning by models.

Architecture	*# contexts withheld*			
	2	3	6	14
SRN (−)	7.5	5.0	—	—
SRN (+)	0.6	0.6	0.6	—
GRU (−)	1.8	2.2	3.4	9.4
GRU (+)	2.2	3.6	5.3	1.5
LSTM (−)	1.2	2.2	4.4	12.2
LSTM (+)	0.6	0.8	1.4	3.4

Table 1: Average number of epochs between having learned one context and having learned all contexts, calculated as the mean difference among runs which succeeded in eventually learning all contexts once. A '—' in a row indicates that no models were able to achieve this degree of generalization.

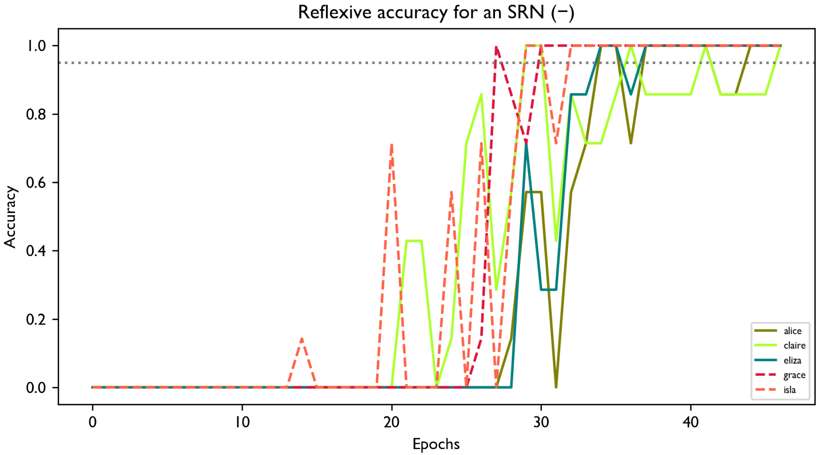

Figure 2: Reflexive accuracy with different antecedents during training of an SRN without attention. *Alice*, *Claire* and *Eliza* were withheld during training while *Grace* and *Isla* present in the training data.

6 Experiment 4: What if Alice doesn't know anyone?

The experiments we have described thus far removed from the training data input sentences and logical forms that were exactly identical to those associated with reflexive sentences. The next pair of experiments increases the difficulty of the generalization task still further, by withholding from the Experiment 2 training data all sentences containing the withheld reflexive antecedent, *Alice*, in a wider range of grammatical contexts, and testing the effect that this has on the network's ability to interpret *Alice*-reflexive sentences.

Experiment 4a starts by withholding sentences where *Alice* appears as the subject of a transitive verb (including those with reflexive objects, which we already removed in earlier experiments). This manipulation tests the degree to which the presence of *Alice* as a subject more generally is crucial to the network's generalization of reflexive sentences to a novel name. We also run a variation of this experiment (Experiment 4b) in which sentences containing *Alice* as the subject of intransitives are also removed, i.e., sentences of the following form:

(7) Alice *verbs* → *verb*(ALICE)

If subjecthood is represented in a uniform manner across transitive and intransitive sentences, the absence of such sentences from the training data might further impair the network's ability to generalize to reflexive sentences.

Experiment 4a	SRN (−)	SRN (+)	GRU (−)	GRU (+)	LSTM (−)	LSTM (+)
Alice-reflexive	0.00	0.80	0.03	0.26	0.00	**1.00**
Alice-subject (trans)	0.02	**0.83**	0.04	0.29	0.03	0.28
Experiment 4b	SRN (−)	SRN (+)	GRU (−)	GRU (+)	LSTM (−)	LSTM (+)
Alice-reflexive	0.00	0.63	0.00	0.80	0.00	**0.83**
Alice-subject (trans)	0.00	0.25	0.01	**0.78**	0.03	0.23
Alice-subject (intrans)	0.00	0.80	0.58	0.95	0.98	**1.00**

Table 2: Mean accuracy on generalization sets for Experiments 4a and 4b.

6.1 Results

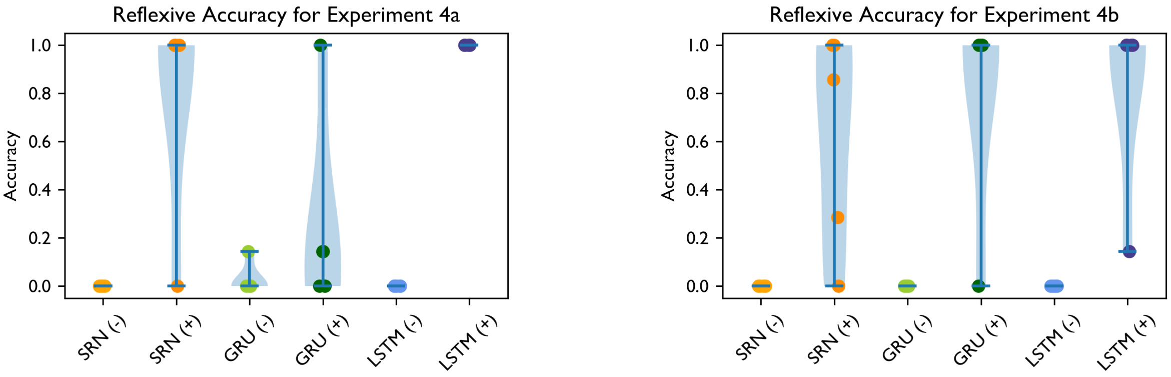

Figure 3: Mean accuracy on *Alice*-reflexive sentences in Experiments 4a (left) and 4b (right).

Experiment 4a The left plot in Figure 3 shows the reflexive generalization accuracy for the runs of the different architectures in the first variant of this experiment. Models without attention uniformly perform poorly across all recurrent unit types. With attention, performance is more variable: LSTMs perform at ceiling and SRNs do well for most random seeds, while GRUs perform poorly for most initializations with a single seed performing at ceiling. The top portion of Table 2 contrasts the means of these results with the generalization performance on transitives with *Alice* subjects. Here again LSTMs without attention performed poorly while those with attention did much worse on *Alice*-transitives than on *Alice*-reflexive sentences.

This result at once highlights the role that attention plays in learning this type of systematic generalization; attention appears to be necessary for recurrent architectures to generalize in this context. The pattern of results also demonstrates a substantial effect of model architecture: attentive SRNs substantially outperform the more complex LSTM and GRU architectures on generalization to *Alice*-transitives, though this was not the case for reflexive sentences, where LSTMs showed a substantial advantage.

Experiment 4b The right plot in Figure 3 shows the impact of withholding *Alice*-intransitive sentences from training. As before, models without attention fail on interpreting *Alice*-reflexive sentences. LSTMs and SRNs with attention perform nearly as well as in Experiment 4a, with some seeds performing at ceiling and a somewhat larger number than before failing to doing so. In contrast, the performance of attentive GRUs is improved in this context. The bottom of Table 2 shows the mean generalization accuracy for transitive and intransitive sentences with *Alice* subjects. In some cases the transitive subject performance is as in Experiment 4a or worse, but in one case, namely attentive GRUs, it improves in this more difficult context, paralleling what we saw for reflexive generalization.

The reversal of GRU (+) and SRN (+) accuracies better lines up with what we might expect given the complexity of the network architectures, with the more complex GRUs now outperforming the simpler SRNs. These results also reinforce the connection observed in those from Experiment 4b on the effects of

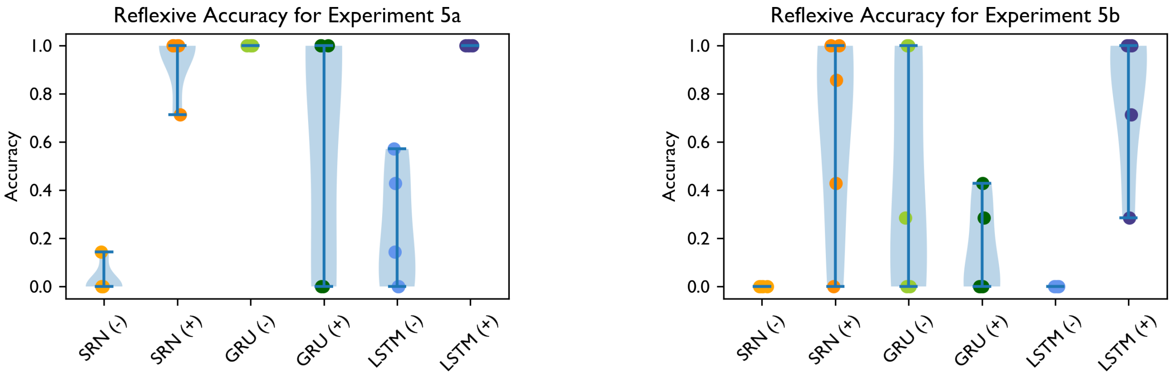

Figure 4: Mean accuracy on *Alice*-reflexive sentences in Experiments 5a (left) and 5b (right).

attention in generalization.

While withholding more information during training as we move from Experiment 4a to 4b might be expected to impair generalization for attentive GRUs, as it did for all other architectures, we in fact see an increase in performance on *Alice*-reflexive sentences. One possible explanation of this surprising result is that the attentive GRU networks in experiment 4a have learned from the training data a context-sensitive regularity concerning the distribution of the withheld name *Alice*, namely that it occurs only as the subject of intransitive verbs. In Experiment 4b, however, the absence of evidence concerning the types of predicates with which *Alice* may occur allows the network to fall back to a context-free generalization about *Alice*, namely that it has the same distribution as the other names in the domain. Note that this explanation is possible only if the network treats intransitive and transitive subjects in a similar way.

7 Experiment 5: What if nobody knows Alice?

In the final experiment, we restrict the grammatical context in which *Alice* appears by removing from the training data of Experiment 2 all instances of transitive sentences with *Alice* in object position (but it is retained in subject position, apart from reflexive sentences). In a second variant (Experiment 5b), we further restrict the training data to exclude all intransitive sentences with *Alice* subjects. Although English, as a language with nominative-accusative alignment, treats subjects of intransitives in a grammatically parallel fashion to subjects of transitives, other languages (with ergative-absolutive alignment) treat intransitive subjects like transitive objects. Though the word order of our synthetic language suggests nominative-accusative alignment, intransitive subjects have in common with transitive objects being the final argument in the logical form, which might lead to them being treated in similar fashion.

7.1 Results

Experiment 5a The left plot in Figure 4 shows reflexive generalization accuracy when the missing antecedent *Alice* is withheld from transitive objects. In contrast to the results in Experiment 4, the effect of attention is more varied here. While SRNs and LSTMs without attention perform poorly, GRUs without attention perform well (for some seeds). As the top panel in Table 3 shows, no models without attention performed well on sentences with *Alice* in object position. For the models with attention, SRNs and LSTMs performmed uniformly well while the performance of GRUs was more mixed. On *Alice*-object sentences attentive SRNs again showed excellent performance, whereas the GRUs and LSTMs fared less well. At the same time, while GRUs with attention outperformed GRUs without attention on *Alice*-object sentences (25% to 4%), they greatly underperformed them on the reflexive sentences (60% to 98%).

Experiment 5b The right plots in Figure 4 shows the effects of further withholding *Alice*-intransitive sentences for *Alice*-reflexive sentences. This manipulation has devastating effects on the performance of all models without attention. For models with attention, there is also a negative impact on reflexive generalization, but not as severe. As shown in the bottom portion of Table 3, this manipulation has little impact on the network's performance on *Alice*-object sentences, with SRNs with attention continuing to

Experiment 5a	SRN (−)	SRN (+)	GRU (−)	GRU (+)	LSTM (−)	LSTM (+)
Alice-reflexive	0.03	0.94	0.98	0.60	0.23	**1.00**
Alice-object	0.00	**0.97**	0.04	0.25	0.04	0.37
Experiment 5b	SRN (−)	SRN (+)	GRU (−)	GRU (+)	LSTM (−)	LSTM (+)
Alice-reflexive	0.00	0.65	0.45	0.14	0.00	**0.80**
Alice-object	0.00	**0.94**	0.03	0.09	0.03	0.17
Alice-subject (intrans)	0.00	0.13	0.00	0.00	0.00	**0.40**

Table 3: Mean accuracy on generalization sets for Experiments 5a and 5b.

perform strongly and the other models performing less well. GRUs continue to interact with attention in unusual ways. While they perform poorly on *Alice*-object and *Alice*-intransitive sentences with and without attention, inattentive GRUs continue to outperform attentive ones on reflexive sentences.

Overall, as in Experiment 4, LSTMs with attention show the highest accuracy on the *Alice*-reflexive sentences by a wide margin, while SRNs with attention attain the best performance on *Alice*-object sentences. Unlike in Experiment 4, withholding the *Alice*-intransitive sentences from training does not yield any benefit for GRUs with attention in performance on the reflexive set, in fact the opposite is true. This may be interpreted once again as evidence that GRUs are treating transitive and intransitive subjects as belonging to the same category. In Experiment 5a, *Alice* occurs in both positions, leading the network to treat it as a subject like any other, and therefore potentially capable of serving as a subject of a reflexive. *Alice*'s absence from object position does not impact the formation of this generalization. In Experiment 5b, on the other hand, where *Alice* occurs only as a transitive subject, it leads the attentive GRU to treat it as name with a distinctive distribution, which impairs generalization to reflexive sentences.

8 Conclusions

Because of their abstract meaning, reflexive anaphora present a distinctive challenge for semantic parsing that had been thought to be beyond the capabilities of recurrent networks. The experiments described here demonstrate that this was incorrect. Sequence-to-sequence networks with a range of recurrent unit types are in fact capable of learning an interpretation of reflexive pronouns that generalizes to novel antecedents. Our results also show that such generalization is nonetheless contingent on the appearance of the held-out antecedent in a variety of syntactic positions as well as the diversity of antecedents providing support for the reflexive generalization. Additionally successful generalization depends on the network architecture in ways that we do not fully understand. It is at present unknown whether the demands that any of these architecture impose on the learning environment for successful learning of reflexives are consistent with what children experience, but this could be explored with both corpus and experimental work. Future work will also be necessary to elucidate the nature of the networks' representations of reflexive interpretation and to understand how they support lexical generalization (or not).

The question we have explored here is related to, but distinct from, the issue of systematicity (Fodor and Pylyshyn, 1988; Hadley, 1994), according to which pieces of representations learned in distinct contexts can freely recombine. This issue has been addressed using sequence-to-sequence architectures in recent work with the synthetic SCAN robot command interpretation dataset (Lake and Baroni, 2018) and on language modeling (Kim and Linzen, 2020), in both cases with limited success. One aspect of the SCAN domain that is particularly relevant to reflexive interpretation is commands involving adverbial modifiers such as *twice*. Commands like *jump twice* must be interpreted by duplicating the meaning of the verb, i.e., as JUMP JUMP, which is similar to what we require for the interpretation of the reflexive object, though in a way that does not require sensitivity to syntactic structure that we have not explored here. Recently, Lake (2019), Li et al. (2019) and Gordon et al. (2020) have proposed novel architectures that increase systematic behavior, and we look forward to exploring the degree to which these impact performance on reflexive interpretation.

Our current work has focused exclusively on recurrent networks, ranging from SRNs to GRUs and

LSTMs. Recent work by Vaswani et al. (2017) shows that Transformer networks attain superior performance on a variety of sequence-to-sequence tasks while dispensing with recurrent units altogether. Examining both the performance and training characteristics of Transformers will allow us to compare the effects of attention and recurrence on the anaphora interpretation task. This is especially interesting given the impact that attention had on performance in our experiments.

Finally, while our current experiments are revealing about the capacity of recurrent networks to learn generalizations about context-sensitive interpretation, there are nonetheless limited in a number of respects because of simplifications in the English fragment we use to create our synthetic data. Reflexives famously impose a structural requirement on their antecedents (c-command). In the following example, the reflexive's antecedent must be STUDENT and cannot be TEACHER.

(8) The student near the teacher sees herself → $\textsc{see}(\textsc{student}, \textsc{student})$

We do not know whether the architectures that have succeed on our experiments would do similarly well if the relevant generalization required reference to (implicit) structure. Past work has explored the sensitivity of recurrent networks to hierarchical structure, with mixed results (Linzen et al., 2016; McCoy et al., 2020). In ongoing work, we are exploring this question by studying more complex synthetic domains both with the kind of recurrent sequence-to-sequence network used here as well networks that explicitly encode or decode sentences in a hierarchical manner. A second simplification concerns the distribution of reflexives themselves. English reflexives can appear in a broader range of syntactic environments apart from transitive objects (Storoshenko, 2008). It would be of considerable interest to explore the reflexive interpretation in a naturalistic setting that incorporate this broader set of distributions.

Acknowledgments

For helpful comments and discussion of this work, we are grateful to Shayna Sragovicz, Noah Amsel, Tal Linzen and the members of the Computational Linguistics at Yale (CLAY) and the JHU Computation and Psycholinguistics labs. This work has been supported in part by NSF grant BCS-1919321 and a Yale College Summer Experience Award. Code for replicating these experiments can be found on the Computational Linguistics at the CLAY Lab GitHub `transductions` and `logos` repositories.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural machine translation by jointly learning to align and translate.

Matthew M. Botvinick and David C. Plaut. 2006. Short-term memory for serial order: A recurrent neural network model. *Psychological Review*, 113:201–233.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Noam Chomsky. 1980. *Rules and Representations*. Columbia University Press.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 33–43, Berlin, Germany. Association for Computational Linguistics.

Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science*, 14(2):179–211.

Jerry A. Fodor and Zenon W. Pylyshyn. 1988. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28:3–71.

Robert Frank and Donald Mathis. 2007. Transformational networks. In *Proceedings of the 3rd Workshop on Psychocomputational Models of Human Language Acquisition*.

Robert Frank, Donald Mathis, and William Badecker. 2013. The acquisition of anaphora by simple recurrent networks. *Language Acquisition*, 20:181–227.

Kanishk Gandhi and Brenden M. Lake. 2019. Mutual exclusivity as a challenge for deep neural networks.

Jonathan Gordon, David Lopez-Paz, Marco Baroni, and Diane Bouchacourt. 2020. Permutation equivariant models for compositional generalization in language. In *International Conference on Learning Representations*.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks*, 18(5-6):602–610.

Robert F. Hadley. 1994. Systematicity in connectionist language learning. *Mind and Language*, 9:247–272.

Najoung Kim and Tal Linzen. 2020. COGS: A compositional generalization challenge based on semantic interpretation. In *The 2020 Conference on Empirical Methods in Natural Language Processing*.

Brenden M. Lake. 2019. Compositional generalization through meta sequence-to-sequence learning. In *Advances in Neural Information Processing Systems 32*, pages 9791–9801.

Brenden M. Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research*, pages 2873–2882, Stockholm, Sweden.

Yuanpeng Li, Liang Zhao, Jianyu Wang, and Joel Hestness. 2019. Compositional generalization for primitive substitutions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP/IJCNLP)*, pages 4293—4302, Hong Kong, China.

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(1).

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Gary F. Marcus. 1998. Can connectionism save constructionism? *Cognition*, 66:153–182.

R. Thomas McCoy, Robert Frank, and Tal Linzen. 2020. Does syntax need to grow on trees? Sources of hierarchical inductive bias in sequence-to-sequence networks. *Transactions of the Association for Computational Linguistics*, 8:125–140.

Dennis Ryan Storoshenko. 2008. The distribution of reflexive pronouns in English: A corpus analysis. In *Proceedings of the 24th Northwest Linguistics Conference*, pages 67–74.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.