The Acquisition of Semantic Representations of Anaphora in Complex Expressions

Jackson Petty
jackson.petty@yale.edu

Summer 2020
Yale University
New Haven, Connecticut

Advised by Robert Frank

Contents

1 Introduction

Well-formed •–∘ constructions of a natural language have both an observable representation, written or spoken words, and a semantic representation, the meaning behind these utterances. Taken together, the observable and semantic represenations of a phrase constitue a form-meaning pair. Knowledge of that language is then the ability to translate between form and meaning, interpreting the observable form of an construction and producing new observable representations in kind from semantic representations. While to us the connection between a form and its meaning is intuitive, to machines this relationship is not innate and must then be taught.

One challenge in teaching machines to understand language comes from words or phrases whose meaning is necessarily context-dependent. Simple words, like proper nouns or verbs, may have meanings which are derivable from the words themselves. For instance, consider a simple language consiting of names and intransitive verbs like the one shown below in (1).

(1) a. ⟦Alice⟧ = the girl named Alice
    b. ⟦Bob⟧ = the boy named Bob
    c. ⟦thinks⟧ = the act of thinking

The meanings of sentences in this simple language are nicely composed of the meanings of the constituent words. For example, in (2) we see that the sentence "Alice thinks" is understood of one knows the meanings of "Alice," "thinks," and the rules about how sentences are formed from constituent parts.

(2) ⟦Alice thinks⟧ ≈ ⟦Alice⟧ + ⟦thinks⟧ = the girl named Alice thinks

Not all words, however, have this nice property that their meanings are independent of the words surrounding them. Consider the case of anaphora, where reflexive pronouns refer to other nouns in a sentence, like in (3).

(3) a. Alice sees herself
    b. Claire sees herself

The sentences here can't be easily interpreted from the meanings of the individual words in the same way that (2) can. For one, it's clear that the word "herself" doesn't have a meaning independent of "Alice" or "Claire." Furthermore, it's apparent that the meaning of "herself" changes depending on the context. These are in fact the defining properties of reflexive pronouns: they must be bound by some independently-defined noun in order to have meaning. Cases like this pose challenges for machines being trained to translate forms into meaning.

Another problem facing linguists and computer scientists is trying to figure out what exactly is happening when machines "learn" a task. Defined extentially, one might think that a machine has learned a task when it can successfully complete the problems set before it. That is, if we give a machine a list of sentences and ask it to translate those sentences into semantic representations, and it does so correctly, we might think that the machine has learned how to interpret language.

2 Background

2.1 Form-Meaning Pairs

2.2 Anaphora and Binding Theory

2.3 Neural Networks and Anaphora

The question of whether or not neural networks can learn the semantic representations of anaphora is not new. [1] posed this problem and investigated whether or not a real-time RNN network architecture could predict the meanings of anaphors. Their results found that real-time networks were unable to successfully.

3 The Seq2Seq Network Architecture

3.1 Specifying the Form and Meaning domains

In order to generate training data consisting of form-meaning pairs, I use a Featural Context-Free Grammar to proceduraly generate sentences and then parse them into semantic representations using the `nltk` Python library. The Form domain is then all possible sentences generated by the grammar, while the meaning domain is the corresponding collection of semantic representations.

An example, minimal grammar is given below along with the sentences is generates and their associated semantic representations.

```
# file: grammar.fcfg
% start S

# Grammatical Rules
S[SEM = <?pred(?subj)>] -> NP[SEM = ?subj] VP[SEM = ?pred]
VP[SEM = <?v(?obj)>] -> V[SEM = ?v] NP[SEM = ?obj]

# Lexical Rules
NP[SEM = <\P.P(alice)>] -> Alice
VP[SEM = <\x.know(x)>] -> knows
```

4 Experiment: Generalizing anaphora to new antecedents

One of the simplest types of sentences involving reflexive pronouns are sentences containg only names, transitive verbs, and reflexive pronouns. An example of one such sentence is shown below in (4).

(4) Alice sees herself.

We define a simple predicate logic where verbs are mapped to predicates, and subjects and objects are mapped to the arguments to those predicates, as in (5) below.

(5) Bob sees Alice → see(bob, alice)

In cases where the object of the verb is a reflexive pronoun, like in (4), the subject and object arguments to the predicate are simply the subject of the sentence, as in (6).

(6) Alice sees herself → see(alice, alice)

Intransitive sentences are mapped to predicates with a single argument, as in (7).

(7) Bob sleeps → sleep(bob)

In order to test whether ot not a Seq2Seq model can be trained to generalize knowledge of reflexive sentences, we will selectively withhold certain sentences generated by the language grammar from the training set and see how the network performs on these novel cases. In this set of experiments, we will examine whether the Seq2Seq models can learn to parse 'Alice-reflexive' sentences like in 4, which are of the form of (8) below.

(8) Alice *verbs* herself

Our Seq2Seq model is made of recurrent encoders and decoders and can optionally implement attention. We conduct these tests using SRN, GRU, and LSTM architectures with No Attention, Additive Attention, and Multiplicative attention to gauge the effect that network architecture has on our models' abilities to learn to interpret reflexive pronouns and generalize this knowledge to new cases. For each combination of recurrent unit and attention, we train three models separately to see if there is any variance in the networks' abilities.

(9)

Grammar		
S	→	Name VP
VP	→	V_i ∣ V_t Name ∣ V_t Refl
Name	→	Alice ∣ Bob ∣ … ∣ Zelda
Refl	→	himself ∣ herself
V_i	→	walks ∣ sleeps ∣ eats ∣ runs ∣ sings ∣ dances ∣ flies ∣ slumbers
V_t	→	sees ∣ meets ∣ likes ∣ dislikes ∣ throws ∣ notices ∣ knows

Table 4.1 Context-free grammar used for Alice-* experiments.

The sentences produced by this grammar are matched to semantic representations in the following way.

$x\ \mathrm{V}_i$	→	$verb(x)$	Alice sleeps	→	sleep(alice)
$x\ \mathrm{V}_t\ y$	→	$verb(x, y)$	Bob knows Claire	→	know(bob, claire)
$x\ \mathrm{V}_t$ Refl	→	$verb(x, x)$	Zelda sees herself	→	see(zelda, zelda)

Table 4.2 Semantic representations of generated sentences for Alice-* experiments.

For example, examples in (10) show how various types of sentences produced by the grammar are parsed into form-meaning pairs.

(10) a. P *verbs* → *verb*(P) [intransitive]
b. P *verbs* Q → *verb*(P, Q) [transitive]
c. P *verbs* (himself | herself) → *verb*(P, P) [reflexive]

Note that in (10b) the names P and Q need not be distinct. This grammar produces sentences like (11a), where the subject and the object of the sentence are the same name. As shown by (11b), these sentences have identical semantic representations to reflexive sentences.

(11) a. Alice sees Alice → see(Alice, Alice)
b. Alice sees herself → see(Alice, Alice)

4.1 Alice-α: Can Alice know herself?

The Alice-α experiment explores whether a Seq2Seq model can generalize knowledge of the semantic representations of reflexive sentences to a novel antecedent. Here, sentences of the form of (12), where "Alice" is the antecedent of the reflexive pronoun "herself" are withheld from the training, testing, and validation data.

(12) Alice *verbs* herself → *verb*(Alice, Alice) [withheld in Alice-α]

All other sentences generated by the grammar, including transitive sentences where "Alice" appears as both the subject and object like in (13a), intransitive sentences where "Alice" is the subject like in (13b), sentences where "Alice" appears only as the object like in (13c), and sentences where "Alice" appears only as the subject like in (13d) where $P \neq$ "Alice", are included in the training, validation, and testing data.

(13) a. Alice *verbs* Alice → *verb*(Alice, Alice) [present in Alice-α]
b. Alice *verbs* → *verb*(Alice) —
c. P *verbs* Alice → *verb*(P, Alice) —
d. Alice *verbs* P → *verb*(Alice, P) —

All sentences generated by the grammar not involving "Alice," such as those types enumerated in (14) where $P, Q \neq$ "Alice", are also present in the training, validation, and testing data.

(14) a. P *verbs* → *verb*(P) [present in Alice-α]
b. P *verbs* Q → *verb*(P, Q) —
c. P *verbs* (himself | herself) → *verb*(P, P) —

The networks' task then is to generalize knowledge of the semantic representations of sentences like those in (13) and (14) to sentences like those in (12).

4.2 Alice-β: But wait, doesn't Alice know Alice?

To be successful at the task presented in Alice-α, networks need to be capable of lexical generalization in a new input: the token "Alice" does not appear as the antecedent of "herself" in training data and the network must learn to interpret exactly these types of sentences. However, models trained on the Alice-α do have one potentially useful piece of information at their disposal: although not exposed to sentences like "Alice *verbs* herself" in training, they are exposed to the closely-related sentences of the form "Alice *verbs* Alice." Though these sentences do not involve reflexive pronouns, their semantic representations are identical to reflexive sentences. This means that models in Alice-α may be biased in favor of producing the correct semantic parses since they already have knowledge that there exist sentences of the form of (13a) whose representation is the target output for novel constructions like (12).

To account for this possibility and increase the difficulty of the task set before the network, the Alice-β experiment further withholds sentences like those in (15b) below, where "Alice" is both the subject and the object, along with the Alice-reflexive (15a) sentences withheld in Alice-α.

(15) a. Alice *verbs* herself → *verb*(Alice, Alice) [withheld in Alice-β]
b. Alice *verbs* Alice → *verb*(Alice, Alice) —

The remaining types of sentences involving "Alice," intransitive sentences like those in (16a), transitive sentences where "Alice" is the subject but not the object like those in (16b), and transitive sentences where "Alice" is the object but not the subject like those in (16c), are all present in the training, validation, and testing datasets.

(16) a. Alice *verbs* → *verb*(Alice) [present in Alice-β]
b. Alice *verbs* P → *verb*(Alice, P) —
c. P *verbs* Alice → *verb*(P, Alice) —

Finally, all sentences involving subjects and objects other than "Alice" are also present in the training, validation, and testing dataset, as shown below in (17) where $P, Q \neq$ Alice.

(17) a. P *verbs* → *verb*(P) [present in Alice-β]
b. P *verbs* Q *verb*(P, Q) —
c. P *verbs* (himself | herself) → *verb*(P, P) —

To successfully generalize knowledge of (16) and (17) to interpret the Alice-reflexive sentences of (15b), a model must not only contend with new inputs but must also successfully produce a new semantic representation not previously encountered in training.

4.3 Alice-γ: What if nobody knows Alice?

Given the networks' relatively good performance on the harder task of Alice-β, it seems that Seq2Seq models are capable of lexical generalization to novel antecedents. To explore the models' limits to this kind of generalization Alice-γ further restricts the kinds of sentences present in the training data by withholding all sentences like (18c), where "Alice" appears as the object, in addition to the (18ab) sentences withheld in Alice-β.

(18) a. Alice *verbs* herself → *verb*(Alice, Alice) [withheld in Alice-γ]
b. Alice *verbs* Alice → *verb*(Alice, Alice) —
c. P *verbs* Alice → *verb*(P, Alice) —

Non-reflexive sentences where "Alice" serves only as the subject, like the intransitive sentences of (19a) or transitive sentences of (19b) where $P \neq$ "Alice", are included in the training, validation, and testing datasets.

(19) a. Alice *verbs* → *verb*(Alice) [present in Alice-γ]
b. Alice *verbs* P → *verb*(Alice, P) —

Additionally, sentences of the form of (20) for $P, Q \neq$ "Alice", where "Alice" does not appear at all, are likewise included in the training, validation, and testing sets.

(20) a. P *verbs* → *verb*(P) [present in Alice-γ]
b. P *verbs* Q *verb*(P, Q) —
c. P *verbs* (himself | herself) → *verb*(P, P) —

4.4 Alice-δ: What if Alice doesn't know anyone?

In the same vein of further restricting the distribution of "Alice" in the training dataset, we also investigate what happens when the network is never exposed to sentences where "Alice" serves as the subject. This means withholding both Alice-reflexive (21a) and Alice-Alice (21b) sentences, along with transitive sentences like (21c) where $P \neq$ "Alice".

(21) a. Alice *verbs* herself → *verb*(Alice, Alice) [withheld in Alice-δ]
b. Alice *verbs* Alice → *verb*(Alice, Alice) —
c. Alice *verbs* P → *verb*(Alice, P) —

There is the additional question of intransitive sentences where "Alice" is the subject. These (22) are distinct from the reflexive sentences we are testing but are the only other case where "Alice" is the subject. To explore how these intransitive sentences affect models' abilities to interpret the reflexive sentences in question, we run two version of the Alice-δ experiment: one where Alice-intransitive sentences are present in the training, validation, and testing data, and one where they are withheld.

(22) Alice *verbs* → *verb*(Alice) [present in Alice-δ, withheld in Alice-δ^*]

Sentences where "Alice" is only the object, like (23a) for $P \neq$ "Alice", are present in the training, validation, and testing data, as are all sentences (23b–d) not involving "Alice."

(23) a. P *verbs* Alice → *verb*(P, Alice) [present in Alice-δ]
b. P *verbs* → *verb*(P) —
c. P *verbs* Q → *verb*(P, Q) —
d. P *verbs* (himself | herself) → *verb*(P, P) —

4.5 Alice-ε: Who's Alice and who's Claire?

The previous Alice-* experiments have thus far dealt with the task of lexical generalization of a single novel antecedent; we withhold some combinations of sentences containing the token "Alice" and test to see whether or not the network can learn to interpret sentences where "Alice" is the antecedent of a reflexive pronoun. While the results Alice-β experiment were successful in this regard, there is a concern that the networks may have been able to interpret Alice-reflexive sentences in a negative sense: that is, by learning the interpretations of all other *Person*-reflexive sentences through direct example and then "filling in the blank" with "Alice" when confronted with a *Person*-reflexive combination which it has not yet been taught. When dealing with only a single novel antecedent, this outcome would look the same as if the network had truly acquired the generalization of (24).

(24) P *verbs* (himself | herself) → *verb*(P, P)

To determine if these networks are truly capable of the latter inference, we extend the Alice-β experiment by withholding progressively more sentences of the form

(25) *P verbs* (*P* | himself | herself)

and explore the networks' abilities to generalize knowledge of the interpretation of reflexive pronouns to new antecedents.

In withholding more than just a single *Person*-reflexive sentence, we must consider the fact that while speakers of English know that "himself" and "herself" have the same meaning, distinguished only by φ-features, the networks here have no such awareness. There is no *a-priori* reason for a network to associate "himself" with "herself" aside from their similarity in positional distribution in the training data. Because of this, withholding both "Alice *verbs* (Alice | herself)" and "Bob *verbs* (Bob | himself)" may not be qualitatively different for the network than withholding only "Alice *verbs* (Alice | herself)" (with respect to the network's performance at interpreting the latter). Therefore, we will begin by removing progressively more *Person*-reflexive sentences with feminine antecedents.

Alice-ε-2: We begin by withholding *Person*-reflexive sentences where "Alice" and "Claire" are the antecedents, shown below in (26).

(26) a. Alice *verbs* (Alice | herself)
b. Claire *verbs* (Claire | herself)

The training set then included all *Person*-reflexive sentences with masculine antecedents and all remaining *Person*-reflexive sentences with feminine antecedents. [2]

5 References

[1] R. Frank, D. Mathis, and W. Badecker. *The Acquisition of Anaphora by Simple Recurrent Networks*. 18 June 2013. *Language Acquisition*.

[2] R. Frank, D. Mathis, and W. Badecker. *The Acquisition of Anaphora by Simple Recurrent Networks*. 18 June 2013. *Language Acquisition*.