Advanced behavioral evaluation of NLU models

Christopher Potts

Stanford Linguistics

CS224u: Natural language understanding
Overview
Varieties of evaluation

**Behavioral**

- Standard ("IID"; Independent and Identically Distributed)
- Exploratory
- Hypothesis-driven
- Challenge
- Adversarial
- Security-oriented

**Structural**

- Probing
- Feature attribution
- Interventions
Standard evaluations

1. Create a dataset from a single process.

2. Divide the dataset into disjoint train and test sets, and set the test set aside.

3. Develop a system on the train set.

4. Only after all development is complete, evaluate the system based on accuracy on the test set.

5. Report the results as providing an estimate of the system’s capacity to generalize.
Adversarial evaluations

1. Create a dataset by whatever means you like.

2. Develop and assess the system using that dataset, according to whatever protocols you choose.

3. Develop a new test dataset of examples that you suspect or know will be challenging given your system and the original dataset.

4. Only after all system development is complete, evaluate the system based on accuracy on the new test dataset.

5. Report the results as providing an estimate of the system’s capacity to generalize.
A bit of history

Vol. lix.  No. 236.]

M I N D

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing
A bit of history

Understanding Natural Language

Terry Winograd
Massachusetts Institute of Technology
Cambridge, Massachusetts
A bit of history

On our best behaviour

Hector J. Levesque
Dept. of Computer Science
University of Toronto
Toronto, Ontario
Canada M5S 3A6
hector@cs.toronto.edu
Winograd sentences

1. The trophy doesn’t fit into the brown suitcase because it’s too small. What is too small? The suitcase / The trophy

2. The trophy doesn’t fit into the brown suitcase because it’s too large. What is too large? The suitcase / The trophy

3. The council refused the demonstrators a permit because they feared violence. Who feared violence? The council / The demonstrators

4. The council refused the demonstrators a permit because they advocated violence. Who advocated violence? The council / The demonstrators

Winograd 1972; Levesque 2013
Levesque’s (2013) adversarial framing

Could a crocodile run a steeplechase?
“The intent here is clear. The question can be answered by thinking it through: a crocodile has short legs; the hedges in a steeplechase would be too tall for the crocodile to jump over; so no, a crocodile cannot run a steeplechase.”

Foiling cheap tricks
“Can we find questions where cheap tricks like this will not be sufficient to produce the desired behaviour? This unfortunately has no easy answer. The best we can do, perhaps, is to come up with a suite of multiple-choice questions carefully and then study the sorts of computer programs that might be able to answer them.”
Analytical considerations
Key questions

What can behavioral testing tell us?  
(And what can’t it tell us?)
No need to be adversarial

Here are some questions that start off exploratory and end up being adversarial:

1. Has my system learned anything about numerical terms?
2. Does my system understand how negation works?
3. Does my system work with a new style or genre?
4. This system is supposed to know about numerical terms, but here are some test cases that are outside of its training experiences for such terms...
5. When applied to invented genres, does my system produce socially problematic (e.g., stereotyped) outputs?
6. Are their patterns of random inputs that lead my system to produce problematic outputs?
Limits of behavioral testing

Even/Odd Model 1
Limits of behavioral testing

Even/Odd Model 1

four

even
Limits of behavioral testing

Even/Odd Model 1

four

twenty one

odd
Limits of behavioral testing

Even/Odd Model 1

- four
- twenty one
- thirty two

even
Limits of behavioral testing

Even/Odd Model 1

- four
- twenty one
- thirty two
- thirty six

even
Limits of behavioral testing

Even/Odd Model 1

- four
- twenty one
- thirty two
- thirty six
- sixty three

odd
Limits of behavioral testing

Even/Odd Model 1

- `four`: even
- `twenty one`: odd
- `thirty two`: even
- `thirty six`: even
- `sixty three`: odd
- `else`: odd

Result: odd
Limits of behavioral testing

**Even/Odd Model 1**

- **four**: even
- **twenty one**: odd
- **thirty two**: even
- **thirty six**: even
- **sixty three**: odd
- **else**: odd

- **twenty two**: odd
Limits of behavioral testing

Even/Odd Model 2

- four
- twenty one
- thirty two
- thirty six
- sixty three
- twenty two

even
Limits of behavioral testing
Limits of behavioral testing

Even/Odd Model 2

- four
- twenty one
- thirty two
- thirty six
- sixty three
- twenty two
- five
- eighty nine
- odd
Limits of behavioral testing

Even/Odd Model 2

four
twenty one
thirty two
thirty six
sixty three
twenty two
five
eighty nine
fifty six
Limits of behavioral testing

Even/Odd Model 2

\[ d = \begin{cases} 
  \text{one: odd} \\
  \text{two: even} \\
  \text{three: odd} \\
  \text{four: even} \\
  \text{five: odd} \\
  \text{six: even} \\
  \text{seven: odd} \\
  \text{eight: even} \\
  \text{nine: odd} \\
  \text{else: odd} 
\end{cases} \]

return \( d[\text{input final token}] \)
Limits of behavioral testing

Even/Odd Model 2

\[
d = \begin{cases} 
    \text{one}: \text{odd} \\
    \text{two}: \text{even} \\
    \text{three}: \text{odd} \\
    \text{four}: \text{even} \\
    \text{five}: \text{odd} \\
    \text{six}: \text{even} \\
    \text{seven}: \text{odd} \\
    \text{eight}: \text{even} \\
    \text{nine}: \text{odd} \\
    \text{else}: \text{odd}
\end{cases}
\]

return \(d[\text{input final token}]\)

odd
Limits of behavioral testing

Even/Odd Model 3

data points:
- four
- twenty one
- thirty two
- thirty six
- sixty three
- twenty two
- five
- eighty nine
- fifty six
- sixteen

even
The limitations of accuracy-based metrics are generally left unaddressed by the methods we will explore here, but these limitations should be brought in!
Model failing or dataset failing?

Liu et al. (2019)

“What should we conclude when a system fails on a challenge dataset? In some cases, a challenge might exploit blind spots in the design of the original dataset (dataset weakness). In others, the challenge might expose an inherent inability of a particular model family to handle certain natural language phenomena (model weakness). These are, of course, not mutually exclusive.”
Model failing or dataset failing?

Geiger et al. (2019)

However, for any evaluation method, we should ask whether it is fair. Has the model been shown data sufficient to support the kind of generalization we are asking of it? Unless we can say “yes” with complete certainty, we can’t be sure whether a failed evaluation traces to a model limitation or a data limitation that no model could overcome.
Model failing or dataset failing?

3 5 7 ...
Model failing or dataset failing?

3  5  7  ...  

What number comes next?
Model failing or dataset failing?

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>
Model failing or dataset failing?

\[
\begin{array}{c|c|c}
 p & q & p \rightarrow q \\
 T & T & T \\
 T & F & F \\
 F & T & T \\
 F & F & T \\
\end{array}
\]
Inoculation by fine-tuning

Figure 1: An illustration of the standard challenge evaluation procedure (e.g., Jia and Liang, 2017) and our proposed analysis method. “Original” refers to the a standard dataset (e.g., SQuAD) and “Challenge” refers to the challenge dataset (e.g., Adversarial SQuAD). Outcomes are discussed in Section 2.
Inoculation by fine-tuning

**Outcome 1**
(Dataset weakness)
**(a) Word Overlap**

**Outcome 2**
(Model weakness)
**(c) Spelling Errors**

**Outcome 3**
(Dataset artifacts or other problem)
**(e) Numerical Reasoning**

**Outcome 1**

Outcome 1: Refers to the first outcome of the inoculation by fine-tuning experiment, which is related to dataset weaknesses.

**Outcome 2**

Outcome 2: Refers to the second outcome of the inoculation by fine-tuning experiment, which is related to model weaknesses.

**Outcome 3**

Outcome 3: Refers to the third outcome of the inoculation by fine-tuning experiment, which is related to dataset artifacts or other problems.

---

Liu et al. 2019
Negation as a learning target

Intuitive learning target

If $A$ entails $B$ then $\neg B$ entails $\neg A$

Observation
Top-performing NLI models fail to achieve the learning target (Yanaka et al. 2019, 2020; Hossain et al. 2020; Geiger et al. 2020b).

Tempting conclusion
Top-performing models are incapable of learning negation.

Dataset observation
Negation is severely under-represented in NLI benchmarks.
MoNLI dataset construction

Positive MoNLI (PMoNLI; 1,476 examples)

SNLI hypothesis (A) Food was served.
WordNet pizza ⊑ food
New example (B) Pizza was served.

Positive MoNLI (A) neutral (B)
Positive MoNLI (B) entailment (A)

Negative MoNLI (PMoNLI; 1,202 examples)

SNLI hypothesis (A) The children are not holding plants.
WordNet flowers ⊑ plants
New example (B) The children are not holding flowers.

Negative MoNLI (A) entailment (B)
Negative MoNLI (B) neutral (A)
A systematic generalization task

<table>
<thead>
<tr>
<th>NMoNLi Train</th>
<th>NMoNLi Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>person</td>
<td>dog</td>
</tr>
<tr>
<td>instrument</td>
<td>building</td>
</tr>
<tr>
<td>food</td>
<td>ball</td>
</tr>
<tr>
<td>machine</td>
<td>car</td>
</tr>
<tr>
<td>woman</td>
<td>mammal</td>
</tr>
<tr>
<td>music</td>
<td>animal</td>
</tr>
<tr>
<td>tree</td>
<td></td>
</tr>
<tr>
<td>boat</td>
<td></td>
</tr>
<tr>
<td>fruit</td>
<td></td>
</tr>
<tr>
<td>produce</td>
<td></td>
</tr>
<tr>
<td>fish</td>
<td></td>
</tr>
<tr>
<td>plant</td>
<td></td>
</tr>
<tr>
<td>jewelry</td>
<td></td>
</tr>
<tr>
<td>anything</td>
<td></td>
</tr>
<tr>
<td>hat</td>
<td></td>
</tr>
<tr>
<td>man</td>
<td></td>
</tr>
<tr>
<td>horse</td>
<td></td>
</tr>
<tr>
<td>gun</td>
<td></td>
</tr>
<tr>
<td>adult</td>
<td></td>
</tr>
<tr>
<td>shirt</td>
<td></td>
</tr>
<tr>
<td>shoe</td>
<td></td>
</tr>
<tr>
<td>store</td>
<td></td>
</tr>
<tr>
<td>cake</td>
<td></td>
</tr>
<tr>
<td>individual</td>
<td></td>
</tr>
<tr>
<td>clothe</td>
<td></td>
</tr>
<tr>
<td>weapon</td>
<td></td>
</tr>
<tr>
<td>creature</td>
<td></td>
</tr>
</tbody>
</table>

Our models know these lexical relations (high Positive MoNLI accuracy) and will be compelled to combine this knowledge with what they learn about negation during Negative MoNLI fine-tuning.
## MoNLI as challenge dataset

<table>
<thead>
<tr>
<th>Model</th>
<th>Input pretrain</th>
<th>NLI train data</th>
<th>No MoNLI fine-tuning</th>
<th>With NMoNLI fine-tuning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>SNLI</td>
<td>PMoNLI</td>
</tr>
<tr>
<td>BiLSTM</td>
<td>GloVe</td>
<td>SNLI train</td>
<td>81.6</td>
<td>73.2</td>
</tr>
<tr>
<td>ESIM</td>
<td>GloVe</td>
<td>SNLI train</td>
<td>87.9</td>
<td>86.6</td>
</tr>
<tr>
<td>BERT</td>
<td>BERT</td>
<td>SNLI train</td>
<td>90.8</td>
<td>94.4</td>
</tr>
</tbody>
</table>

Diagnosis: Dataset failing!

Geiger et al. 2020b
Reminder: Biological creatures are amazing

Premack 1983; Wasserman et al. 2017; Geiger et al. 2020a
Reminder: Biological creatures are amazing

Premack 1983; Wasserman et al. 2017; Geiger et al. 2020a
Compositionality
Informal statement

Compositionality

The meaning of a phrase is a function of the meanings of its immediate syntactic constituents and the way they are combined.

\[
\begin{align*}
S & \\
NP & \\
Det & N \\
every & student \\
VP & \\
V & NP \\
admired & the \\
D & N \\
the & idea
\end{align*}
\]
The usual motivation

1. Modeling all meaningful units
   \[ \text{[every]} = \lambda f \lambda g \forall x ((f \ x) \rightarrow (g \ x)) \]

2. “Infinite” capacity

3. Creativity

4. Systematicity
Compositionality or systematicity?

Fodor and Pylyshyn (1988:37):
“What we mean when we say that linguistic capacities are systematic is that the ability to produce/understand some sentences is intrinsically connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.
3. the turtle ~ the puppy
4. The turtle loves the puppy.
5. The puppy loves the turtle.
6. The turtle loves Sandy.
7. ...
A worrisome lack of systematicity

<table>
<thead>
<tr>
<th>Example</th>
<th>Gold</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>The bakery sells a mean apple pie.</td>
<td>pos</td>
<td>pos</td>
</tr>
<tr>
<td>They sell a mean apple pie.</td>
<td>pos</td>
<td>pos</td>
</tr>
<tr>
<td>She sells a mean apple pie.</td>
<td>pos</td>
<td>neg</td>
</tr>
<tr>
<td>He sells a mean apple pie.</td>
<td>pos</td>
<td>neg</td>
</tr>
</tbody>
</table>
Compositionality by design

**SHRDLU**

(TINOT 
  (THPROC (X2)) 
  (THGOAL(#IS $?X2 #PYRAMID)) 
  (THGOAL(#SUPPORT $?X1 $?X2)))))

"which supports no pyramids"

(TINOT 
  (THPROC (X2)) 
  (THGOAL(#IS $?X2 #PYRAMID)) 
  (THNOT 
    (THGOAL(#SUPPORT $?X1 $?X2)))))

"which supports every pyramid"

Fig. 52—Quantifiers.

**DCS**

Some river traverses every city.

city traversed by no rivers

(c) Quantifier scope ambiguity (Q, Q)

(d) Quantification (Q, E)

**SST**
No compositionality/systematicity guarantees!

Can we pose behavioral tests that will assess whether models like this have found systematicity solutions?
COGS and ReCOGS
COGS: A Compositional Generalization Challenge Based on Semantic Interpretation

Najoung Kim  
Johns Hopkins University  
n.kim@jhu.edu

Tal Linzen  
New York University  
linzen@nyu.edu

ReCOGS: How Incidental Details of a Logical Form Overshadow an Evaluation of Semantic Interpretation

Zhengxuan Wu  
Christopher D. Manning  
Christopher Potts  
Stanford University  
{wuzhengx, manning, cgpotts}@stanford.edu
Task

COGS
1. ▶ Input: A rose was helped by a dog.
   ▶ Output: \( \text{rose} (x_1) \ AND \ \text{help} \ . \ \text{theme} (x_3, x_1) \ AND \ \text{help} \ . \ \text{agent} (x_3, x_6) \ AND \ \text{dog} (x_6) \)

2. ▶ Input: The sailor dusted a boy.
   ▶ Output: * \( \text{sailor} (x_1) \ ; \ \text{dust} \ . \ \text{agent} (x_2, x_1) \ AND \ \text{dust} \ . \ \text{theme} (x_2, x_4) \ AND \ \text{boy} (x_4) \)

ReCOGS
1. ▶ Input: A rose was helped by a dog.
   ▶ Output: \( \text{rose} (53) \ ; \ \text{dog} (38) \ ; \ \text{help} (7) \ AND \ \text{theme} (7, 53) \ AND \ \text{agent} (7, 38) \)

2. ▶ Input: The sailor dusted a boy.
   ▶ Output: * \( \text{sailor} (48) \ ; \ \text{boy} (53) \ ; \ \text{dust} (10) \ AND \ \text{agent} (10, 48) \ AND \ \text{theme} (10, 53) \)
Motivations

1. Humans easily interpret novel combinations of familiar elements in ways that are systematic.
2. Compositionality is an explanation for this capability.
3. Can our best models generalize this way?
4. Have they too found compositional solutions?

The COGS and ReCOGS tasks are behavioral tests that seek to resolve 3, and the hope is that this can inform 4.
Understanding COGS logical forms

1. Verbs specify primitive events that have their own core conceptual structure and can involve one more more obligatory or optional roles.
   a. Emma broke a vase:
      \[
      \text{vase (} x \_ 3 \text{)}; \text{break . agent (} x \_ 2 , \text{Emma)} \AND \\
      \text{break . theme (} x \_ 2 , x \_ 3 \text{)}
      \]
   b. The vase broke:
      \[
      \text{vase (} x \_ 3 \text{)}; \text{break . theme (} x \_ 2 , x \_ 1 \text{)}
      \]

2. Variable numbering is determined by linear position in the input sentence.

3. All variables are bound; free variables are existentially bound with widest scope:
   a. \[
   \text{dog (} x \_ 1 \text{)} \AND \text{run . agent (} x \_ 2 , x \_ 1 \text{)}
   \]
   b. \[
   \exists x \_ 1 \exists x \_ 2 \text{dog (} x \_ 1 \text{)} \AND \text{run . agent (} x \_ 2 , x \_ 1 \text{)}
   \]

4. Definite descriptions are marked with *:
   a. The sailor ran.
   b. \[
   * \text{sailor (} x \_ 1 \text{)}; \text{run . agent (} x \_ 2 , x \_ 1 \text{)}
   \]
COGS splits

1. Train: 24,000 examples plus 155 primitives
2. Dev: 10,000 examples
3. Test: 10,000 examples
4. Gen: 21,000 examples
### Generalization categories

<table>
<thead>
<tr>
<th>Case</th>
<th>Training</th>
<th>Generalization</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>S.3.1. Novel Combination of Familiar Primitives and Grammatical Roles</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subject $\rightarrow$ Object (common noun)</td>
<td>A <em>hedgehog</em> ate the cake.</td>
<td>The baby liked the <em>hedgehog</em>.</td>
</tr>
<tr>
<td>Subject $\rightarrow$ Object (proper noun)</td>
<td>Lina gave the cake to Olivia.</td>
<td>A hero shortened <em>Lina</em>.</td>
</tr>
<tr>
<td>Object $\rightarrow$ Subject (common noun)</td>
<td>Henry liked a <em>cockroach</em>.</td>
<td>The <em>cockroach</em> ate the bat.</td>
</tr>
<tr>
<td>Object $\rightarrow$ Subject (proper noun)</td>
<td>The creature grew <em>Charlie</em>.</td>
<td><em>Charlie</em> worshiped the cake.</td>
</tr>
<tr>
<td>Primitive noun $\rightarrow$ Subject (common noun)</td>
<td><em>shark</em></td>
<td>A <em>shark</em> examined the child.</td>
</tr>
<tr>
<td>Primitive noun $\rightarrow$ Subject (proper noun)</td>
<td><em>Paula</em></td>
<td><em>Paula</em> sketched William.</td>
</tr>
<tr>
<td>Primitive noun $\rightarrow$ Object (common noun)</td>
<td><em>shark</em></td>
<td>A chief heard the <em>shark</em>.</td>
</tr>
<tr>
<td>Primitive noun $\rightarrow$ Object (proper noun)</td>
<td><em>Paula</em></td>
<td>The child helped <em>Paula</em>.</td>
</tr>
<tr>
<td>Primitive verb $\rightarrow$ Infinitival argument</td>
<td><em>crawl</em></td>
<td>A baby planned to <em>crawl</em>.</td>
</tr>
<tr>
<td><strong>S.3.2. Novel Combination Modified Phrases and Grammatical Roles</strong></td>
<td>Noah ate the <em>cake on the plate</em>.</td>
<td>The <em>cake on the table</em> burned.</td>
</tr>
<tr>
<td><strong>S.3.3. Deeper Recursion</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Depth generalization: Sentential complements</td>
<td>Emma said <em>that</em> Noah knew <em>that</em> the cat danced.</td>
<td>Emma said <em>that</em> Noah knew <em>that</em> Lucas saw <em>that</em> the cat danced.</td>
</tr>
<tr>
<td>Depth generalization: PP modifiers</td>
<td>Ava saw the ball in the <em>bottle on the table</em>.</td>
<td>Ava saw the ball in the <em>bottle on the table</em> on the floor.</td>
</tr>
<tr>
<td><strong>S.3.4. Verb Argument Structure Alternation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active $\rightarrow$ Passive</td>
<td>The crocodile <em>blessed</em> William.</td>
<td>A muffin was <em>blessed</em>.</td>
</tr>
<tr>
<td>Passive $\rightarrow$ Active</td>
<td>The book <em>was squeezed</em>.</td>
<td>The girl <em>squeezed</em> the strawberry.</td>
</tr>
<tr>
<td>Object-omitted transitive $\rightarrow$ Transitive</td>
<td>Emily <em>baked</em>.</td>
<td>The giraffe <em>baked a cake</em>.</td>
</tr>
<tr>
<td>Unaccusative $\rightarrow$ Transitive</td>
<td>The glass <em>shattered</em>.</td>
<td>Liam <em>shattered</em> the jigsaw.</td>
</tr>
<tr>
<td>Double object dative $\rightarrow$ PP dative</td>
<td>The girl <em>teleported</em> Liam the cookie.</td>
<td>Benjamin <em>teleported</em> the cake to Isabella.</td>
</tr>
<tr>
<td>PP dative $\rightarrow$ Double Object Dative</td>
<td>Jane shipped the cake to John.</td>
<td>Jane shipped John the cake.</td>
</tr>
<tr>
<td><strong>S.3.5. Verb Class</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agent NP $\rightarrow$ Unaccusative subject</td>
<td>The <em>cobra</em> helped a dog.</td>
<td>The <em>cobra</em> froze.</td>
</tr>
<tr>
<td>Theme NP $\rightarrow$ Object-omitted transitive subject</td>
<td>The hippo <em>decomposed</em>.</td>
<td>The hippo <em>painted</em>.</td>
</tr>
<tr>
<td>Theme NP $\rightarrow$ Unergative subject</td>
<td>The hippo <em>decomposed</em>.</td>
<td>The hippo <em>giggled</em>.</td>
</tr>
</tbody>
</table>

Kim and Linzen 2020
## Synthetic leaderboard

<table>
<thead>
<tr>
<th>Model</th>
<th>Obj PP → Subj PP</th>
<th>STRUCT</th>
<th>LEX</th>
<th>Overall %</th>
</tr>
</thead>
<tbody>
<tr>
<td>BART (Lewis et al. 2019)</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>91</td>
</tr>
<tr>
<td>BART+syn (Lewis et al. 2019)</td>
<td>0</td>
<td>5</td>
<td>8</td>
<td>80</td>
</tr>
<tr>
<td>T5 (Raffel et al. 2019)</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>97</td>
</tr>
<tr>
<td>Kim and Linzen 2020</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>73</td>
</tr>
<tr>
<td>Ontanon et al. 2022</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>53</td>
</tr>
<tr>
<td>Akyurek and Andreas 2021</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>96</td>
</tr>
<tr>
<td>Conklin et al. 2021</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>88</td>
</tr>
<tr>
<td>Csordás et al. 2021</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>95</td>
</tr>
<tr>
<td>Zheng and Lapata 2022</td>
<td>0</td>
<td>25</td>
<td>35</td>
<td>99</td>
</tr>
</tbody>
</table>

†Results are copied from Yao and Koller (2022). ‡Model uses pretrained weights and is hyperparameter tuned using data sampled from the generalization splits.

Wu et al. 2023
Why removing redundant tokens matters

COGS:  kitten ( x _ 1 )      COGS:  kitten ( 1 )

Wu et al. 2023
What is behind the 0s for CP/PP recursion?

**Input sentences**

To decouple length from depth, we concatenate existing examples and re-index the variable names to cover the variable names seen at test time.

**Output LFs**

Wu et al. 2023
What is behind the 0s for PP modifiers?

Hypothesis

The train data *teach* the model that PPs occur only with a specific set of variables and positions. When models learn this lesson, they struggle with examples that contradict it.

<table>
<thead>
<tr>
<th>Variant</th>
<th>Sentence</th>
<th>Logical Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preposing + Interjection</td>
<td>The box in the tent Emma was * um um* lended.</td>
<td>* box (x_1); * tent (x_4); box. nmod. in (x_1, x_4) AND lend. theme (x_7, x_1) AND lend. recipient (x_7, Emma)</td>
</tr>
<tr>
<td>Participial VP (Subj)</td>
<td>A leaf <em>painting the spaceship</em> froze.</td>
<td>* spaceship (x_4); leaf (x_1) AND leaf. acl. paint (x_1, x_4) AND freeze. theme (x_5, x_1)</td>
</tr>
</tbody>
</table>

Result

Large performance increases for LSTMs and Transformers.

Wu et al. 2023
Modifications for ReCOGS
Modifications for ReCOGS

**Input Sentence:** Mia ate a cake.
Modifications for ReCOGS

**Input Sentence**: Mia ate a cake.

**COGS LF**: eat. agent (x₁, Mia) AND eat. theme (x₁, x₃) AND cake (x₃)
Modifications for ReCOGS

**Input Sentence**: Mia ate a cake.

**COGS LF**: 
\[
\text{eat} \cdot \text{agent} (x_1, \text{Mia}) \text{ AND eat} \cdot \text{theme} (x_1, x_3) \text{ AND cake} (x_3)
\]

\[
\downarrow
\]

**Redundant Token Removal**
Modifications for ReCOGS

**Input Sentence:** Mia ate a cake.

**COGS LF:** eat . agent ( x_1 , Mia ) AND eat . theme ( x_1 , x_3 ) AND cake ( x_3 )

↓

**Redundant Token Removal**

↓

**Meaning-Preserving Data Augmentation**

Wu et al. 2023
Modifications for ReCOGS

**Input Sentence:** Mia ate a cake.

**COGS LF:**

\[
\text{eat} . \text{agent}(\ x_1, \ Mia) \text{AND} \text{eat} . \text{theme}(\ x_1, \ x_3) \text{AND} \text{cake}(\ x_3)
\]

\[
\downarrow
\]

**Redundant Token Removal**

\[
\downarrow
\]

**Meaning-Preserving Data Augmentation**

\[
\downarrow
\]

**Arbitrary Variable Renaming**

Wu et al. 2023
Modifications for ReCOGS

Input Sentence: Mia ate a cake.

**COGS LF:** eat. agent (x_1, Mia) AND eat. theme (x_1, x_3) AND cake (x_3)

↓

Redundant Token Removal

↓

Meaning-Preserving Data Augmentation

↓

Arbitrary Variable Renaming

**ReCOGS LF:** Mia (3); cake (21); eat (6) AND agent (6, 3) AND theme (6, 21)

Wu et al. 2023
Modifications for ReCOGS

Input Sentence: Mia ate a cake.

COGS LF: eat. agent(\(x_1\), Mia) AND eat. theme(\(x_1, x_3\)) AND cake(\(x_3\))

↓

Redundant Token Removal

↓

Meaning-Preserving Data Augmentation

↓

Arbitrary Variable Renaming

ReCOGS LF: Mia(3); cake(21); eat(6) AND agent(6, 3) AND theme(6, 21)

Performance

LEX
STRUCT

Wu et al. 2023
ReCOGS results

Wu et al. 2023
Conceptual questions

1. How can we test for **meaning** if we are predicting **logical forms**?

2. What is a *fair* generalization test in the current context?
   - a. Models are shown a world that manifests specific restrictions.
   - b. In some cases we want them not to learn those restrictions.
   - c. In other cases we do want them to learn those restrictions.

3. What are the limits of compositionality *for humans* and how should that inform our generalization tests?

4. If we have goals that are not supported by our datasets but that seem like good goals for models to reach, how should we express that in our tasks and our models?
Adversarial testing
## SQUaD leaderboards

### Leaderboard

SQuAD2.0 tests the ability of a system to not only answer reading comprehension questions, but also abstain when presented with a question that cannot be answered based on the provided paragraph.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model</th>
<th>Model Details</th>
<th>EM</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IE-Net (ensemble)</td>
<td>RICOH_SRCB_DML</td>
<td>90.939</td>
<td>93.214</td>
</tr>
<tr>
<td>2</td>
<td>FPNet (ensemble)</td>
<td>Ant Service Intelligence Team</td>
<td>90.871</td>
<td>93.183</td>
</tr>
<tr>
<td>3</td>
<td>IE-NetV2 (ensemble)</td>
<td>RICOH_SRCB_DML</td>
<td>90.860</td>
<td>93.100</td>
</tr>
<tr>
<td>4</td>
<td>SA-Net on Albert (ensemble)</td>
<td>QIANXIN</td>
<td>90.724</td>
<td>93.011</td>
</tr>
<tr>
<td>5</td>
<td>SA-Net-V2 (ensemble)</td>
<td>QIANXIN</td>
<td>90.679</td>
<td>92.948</td>
</tr>
<tr>
<td>31</td>
<td>RoBERTa+Verify (single model)</td>
<td>CW</td>
<td>86.448</td>
<td>89.586</td>
</tr>
<tr>
<td>31</td>
<td>BERT + ConvLSTM + MTL + Verifier (ensemble)</td>
<td>Layer 6 AI</td>
<td>86.730</td>
<td>89.286</td>
</tr>
</tbody>
</table>

Rajpurkar et al. 2016
SQUaD adversarial testing

Passage
Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager.

Question
What is the name of the quarterback who was 38 in Super Bowl XXXIII?
Passage
Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager.

Question
What is the name of the quarterback who was 38 in Super Bowl XXXIII?

Answer
John Elway

Jia and Liang 2017
Passage

Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Quarterback Leland Stanford had jersey number 37 in Champ Bowl XXXIV.

Question

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

Answer

John Elway
**SQUaD adversarial testing**

**Passage**

Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Quarterback Leland Stanford had jersey number 37 in Champ Bowl XXXIV.

**Question**

What is the name of the quarterback who was 38 in Super Bowl XXXIII?

**Answer**

John Elway  

Model: Leland Stanford  

Jia and Liang 2017
SQUaD adversarial testing

Passage
Quarterback Leland Stanford had jersey number 37 in Champ Bowl XXXIV. Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager.

Question
What is the name of the quarterback who was 38 in Super Bowl XXXIII?

Answer
John Elway

Jia and Liang 2017
SQUaD adversarial testing

Passage
Quarterback Leland Stanford had jersey number 37 in Champ Bowl XXXIV. Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager.

Question
What is the name of the quarterback who was 38 in Super Bowl XXXIII?

Answer
John Elway

Model: Leland Stanford

Jia and Liang 2017
## SQUaD adversarial testing

<table>
<thead>
<tr>
<th>System</th>
<th>Original</th>
<th>Adversarial</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReasoNet-E</td>
<td>81.1</td>
<td>39.4</td>
</tr>
<tr>
<td>SEDT-E</td>
<td>80.1</td>
<td>35.0</td>
</tr>
<tr>
<td>BiDAF-E</td>
<td>80.0</td>
<td>34.2</td>
</tr>
<tr>
<td>Mnemonic-E</td>
<td>79.1</td>
<td>46.2</td>
</tr>
<tr>
<td>Ruminating</td>
<td>78.8</td>
<td>37.4</td>
</tr>
<tr>
<td>jNet</td>
<td>78.6</td>
<td>37.9</td>
</tr>
<tr>
<td>Mnemonic-S</td>
<td>78.5</td>
<td>46.6</td>
</tr>
<tr>
<td>ReasoNet-S</td>
<td>78.2</td>
<td>39.4</td>
</tr>
<tr>
<td>MPCM-S</td>
<td>77.0</td>
<td>40.3</td>
</tr>
<tr>
<td>SEDT-S</td>
<td>76.9</td>
<td>33.9</td>
</tr>
<tr>
<td>RaSOR</td>
<td>76.2</td>
<td>39.5</td>
</tr>
<tr>
<td>BiDAF-S</td>
<td>75.5</td>
<td>34.3</td>
</tr>
<tr>
<td>Match-E</td>
<td>75.4</td>
<td>29.4</td>
</tr>
<tr>
<td>Match-S</td>
<td>71.4</td>
<td>27.3</td>
</tr>
<tr>
<td>DCR</td>
<td>69.4</td>
<td>37.8</td>
</tr>
<tr>
<td>Logistic</td>
<td>50.4</td>
<td>23.2</td>
</tr>
</tbody>
</table>
## SQUaD adversarial testing

<table>
<thead>
<tr>
<th>System</th>
<th>Original Rank</th>
<th>Adversarial Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReasoNet-E</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>SEDT-E</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>BiDAF-E</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>Mnemonic-E</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Ruminating</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>jNet</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Mnemonic-S</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>ReasoNet-S</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>MPCM-S</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>SEDT-S</td>
<td>10</td>
<td>13</td>
</tr>
<tr>
<td>RaSOR</td>
<td>11</td>
<td>4</td>
</tr>
<tr>
<td>BiDAF-S</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>Match-E</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>Match-S</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>DCR</td>
<td>15</td>
<td>8</td>
</tr>
<tr>
<td>Logistic</td>
<td>16</td>
<td>16</td>
</tr>
</tbody>
</table>
Comparison with regular testing

Plot of Original vs. Adversarial scores for SQuaD
Example: NLI

SNLI leaderboard: Systems over time

F1 score

Human

Bowman et al. 2015
Example: NLI

MultiNLI leaderboard: Systems over time

Human: 92.6

Bowman et al. 2015
An SNLI adversarial evaluation

<table>
<thead>
<tr>
<th>Premise</th>
<th>Relation</th>
<th>Hypothesis</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Train</strong> A little girl kneeling in the dirt crying.</td>
<td>entails</td>
<td>A little girl is very sad.</td>
</tr>
<tr>
<td><strong>Adversarial</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Train</strong> An elderly couple are sitting outside a restaurant, enjoying wine.</td>
<td>entails</td>
<td>A little girl is very unhappy.</td>
</tr>
<tr>
<td><strong>Adversarial</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Glockner et al. 2018
An SNLI adversarial evaluation

<table>
<thead>
<tr>
<th>Model</th>
<th>Train set</th>
<th>SNLI test set</th>
<th>New test set</th>
<th>Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decomposable Attention</td>
<td>SNLI</td>
<td>84.7%</td>
<td>51.9%</td>
<td>-32.8</td>
</tr>
<tr>
<td></td>
<td>MultiNLI + SNLI</td>
<td>84.9%</td>
<td>65.8%</td>
<td>-19.1</td>
</tr>
<tr>
<td></td>
<td>SciTail + SNLI</td>
<td>85.0%</td>
<td>49.0%</td>
<td>-36.0</td>
</tr>
<tr>
<td>ESIM (Chen et al., 2017)</td>
<td>SNLI</td>
<td>87.9%</td>
<td>65.6%</td>
<td>-22.3</td>
</tr>
<tr>
<td></td>
<td>MultiNLI + SNLI</td>
<td>86.3%</td>
<td>74.9%</td>
<td>-11.4</td>
</tr>
<tr>
<td></td>
<td>SciTail + SNLI</td>
<td>88.3%</td>
<td>67.7%</td>
<td>-20.6</td>
</tr>
<tr>
<td>Residual-Stacked-Encoder</td>
<td>SNLI</td>
<td>86.0%</td>
<td>62.2%</td>
<td>-23.8</td>
</tr>
<tr>
<td>(Nie and Bansal, 2017)</td>
<td>MultiNLI + SNLI</td>
<td>84.6%</td>
<td>68.2%</td>
<td>-16.8</td>
</tr>
<tr>
<td></td>
<td>SciTail + SNLI</td>
<td>85.0%</td>
<td>60.1%</td>
<td>-24.9</td>
</tr>
<tr>
<td>WordNet Baseline</td>
<td>-</td>
<td>-</td>
<td>85.8%</td>
<td>-</td>
</tr>
<tr>
<td>KIM (Chen et al., 2018)</td>
<td>SNLI</td>
<td>88.6%</td>
<td>83.5%</td>
<td>-5.1</td>
</tr>
</tbody>
</table>

Table 3: Accuracy of various models trained on SNLI or a union of SNLI with another dataset (MultiNLI, SciTail), and tested on the original SNLI test set and the new test set.
An SNLI adversarial evaluation

RoBERTA-MNLI, off-the-shelf

```
[1]:  import nli, os, torch
     from sklearn.metrics import classification_report

[2]:  # Available from https://github.com/BIU-NLP/Breaking_NLI:
     breaking_nli_src_filename = os.path.join("../new-data/data/dataset.jsonl")
     reader = nli.NLIReader(breaking_nli_src_filename)

[3]:  exs = [((ex.sentence1, ex.sentence2), ex.gold_label) for ex in reader.read()]

[4]:  X_test_str, y_test = zip(*exs)

[5]:  model = torch.hub.load('pytorch/fairseq', 'roberta.large.mnli')
     _ = model.eval()

     Using cache found in /Users/cgpotts/.cache/torch/hub/pytorch_fairseq_master

[6]:  X_test = [model.encode(*ex) for ex in X_test_str]

[7]:  pred_indices = [model.predict('mnli', ex).argmax() for ex in X_test]

[8]:  to_str = {0: 'contradiction', 1: 'neutral', 2: 'entailment'}

[9]:  preds = [to_str[c.item()] for c in pred_indices]
```
An SNLI adversarial evaluation

RoBERTA-MNLI, off-the-shelf

[10]: `print(classification_report(y_test, preds))`

<table>
<thead>
<tr>
<th></th>
<th>precision</th>
<th>recall</th>
<th>f1-score</th>
<th>support</th>
</tr>
</thead>
<tbody>
<tr>
<td>contradiction</td>
<td>0.99</td>
<td>0.97</td>
<td>0.98</td>
<td>7164</td>
</tr>
<tr>
<td>entailment</td>
<td>0.86</td>
<td>1.00</td>
<td>0.92</td>
<td>982</td>
</tr>
<tr>
<td>neutral</td>
<td>0.15</td>
<td>0.15</td>
<td>0.15</td>
<td>47</td>
</tr>
<tr>
<td>accuracy</td>
<td></td>
<td></td>
<td>0.97</td>
<td>8193</td>
</tr>
<tr>
<td>macro avg</td>
<td>0.67</td>
<td>0.71</td>
<td>0.68</td>
<td>8193</td>
</tr>
<tr>
<td>weighted avg</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
<td>8193</td>
</tr>
</tbody>
</table>
## A MultiNLI adversarial evaluation

<table>
<thead>
<tr>
<th>Category</th>
<th>Premise</th>
<th>Relation</th>
<th>Hypothesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antonyms</td>
<td>I love the Cinderella story.</td>
<td>contradicts</td>
<td>I hate the Cinderella story.</td>
</tr>
<tr>
<td>Numerical</td>
<td>Tim has 350 pounds of cement in 100, 50, and 25 pound bags.</td>
<td>contradicts</td>
<td>Tim has less than 750 pounds of cement in 100, 50, and 25 pound bags.</td>
</tr>
<tr>
<td>Word overlap</td>
<td>Possibly no other country has had such a turbulent history.</td>
<td>entails</td>
<td>The country’s history has been turbulent and true is true</td>
</tr>
<tr>
<td>Negation</td>
<td>Possibly no other country has had such a turbulent history.</td>
<td>entails</td>
<td>The country’s history has been turbulent and false is not true</td>
</tr>
</tbody>
</table>

Also ‘Length mismatch’ and ‘Spelling errors’; Naik et al. 2018
# A MultiNLI adversarial evaluation

<table>
<thead>
<tr>
<th>Category</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antonym</td>
<td>1,561</td>
</tr>
<tr>
<td>Length Mismatch</td>
<td>9,815</td>
</tr>
<tr>
<td>Negation</td>
<td>9,815</td>
</tr>
<tr>
<td>Numerical Reasoning</td>
<td>7,596</td>
</tr>
<tr>
<td>Spelling Error</td>
<td>35,421</td>
</tr>
<tr>
<td>Word Overlap</td>
<td>9,815</td>
</tr>
</tbody>
</table>

Naik et al. 2018
A MultiNLI adversarial evaluation

<table>
<thead>
<tr>
<th>System</th>
<th>Original MultiNLI Dev</th>
<th>Competence Test</th>
<th>Distraction Test</th>
<th>Noise Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Antonymy</td>
<td>Word Overlap</td>
<td>Length Mismatch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Numerical</td>
<td>Negation</td>
<td>Spelling Error</td>
</tr>
<tr>
<td></td>
<td>Mat</td>
<td>Mis</td>
<td>Mat</td>
<td>Mis</td>
</tr>
<tr>
<td>NB</td>
<td>74.2</td>
<td>74.8</td>
<td>15.1</td>
<td>19.3</td>
</tr>
<tr>
<td>CH</td>
<td>73.7</td>
<td>72.8</td>
<td>11.6</td>
<td>9.3</td>
</tr>
<tr>
<td>RC</td>
<td>71.3</td>
<td>71.6</td>
<td>36.4</td>
<td>32.8</td>
</tr>
<tr>
<td>IS</td>
<td>70.3</td>
<td>70.6</td>
<td>14.4</td>
<td>10.2</td>
</tr>
<tr>
<td>BiLSTM</td>
<td>70.2</td>
<td>70.8</td>
<td>13.2</td>
<td>9.8</td>
</tr>
<tr>
<td>CBOW</td>
<td>63.5</td>
<td>64.2</td>
<td>6.3</td>
<td>3.6</td>
</tr>
</tbody>
</table>

Table 3: Classification accuracy (%) of state-of-the-art models on our constructed stress tests. Accuracies shown on both genre-matched and mismatched categories for each stress set. For reference, random baseline accuracy is 33%. 

3.3 Noise Test Construction
This class consists of an adversarial example set which tests model robustness to spelling errors. Spelling errors occur often in MultiNLI data, due to involvement of Turkers and noisy source text (Ghaeini et al., 2018), which is problematic as some NLI systems rely heavily on word embeddings. Inspired by Belinkov and Bisk (2017), we construct a stress test for “spelling errors” by performing two types of perturbations on a word sampled randomly from the hypothesis: random swap of adjacent characters within the word (for example, “I saw Tipper with him at the movie.”), and random substitution of a single alphabetical character with the character next to it on the English keyboard. For example, “Agencies have been further restricted and given less choice in selecting contracting methods.” 

4 Experiments
4.1 Experimental Setup
We focus on the following sentence-encoder models, which achieve strong performance on MultiNLI: 

Nie and Bansal (2017) (NB): This model uses a sentence encoder consisting of stacked BiLSTM-RNNs with shortcut connections and fine-tuning of embeddings. It achieves the top non-ensemble result in the RepEval-2017 shared task (Nangia et al., 2017).

Chen et al. (2017) (CH): This model also uses a sentence encoder consisting of stacked BiLSTM-RNNs with shortcut connections. Additionally, it makes use of character-composition word embeddings learned via CNNs, intra-sentence gated attention and ensembling to achieve the best overall result in the RepEval-2017 shared task.

Balazs et al. (2017) (RiverCorners - RC): This model uses a single-layer BiLSTM with mean pooling and intra-sentence attention.

Conneau et al. (2017) (InferSent - IS): This model uses a single-layer BiLSTM-RNN with max-pooling. It is shown to learn robust universal sentence representations which transfer well across several inference tasks.

We also set up two simple baseline models:

BiLSTM: The simple BiLSTM baseline model described by Nangia et al. (2017).

CBOW: A bag-of-words sentence representation from word embeddings.

4.2 Model Performance on Stress Tests
Table 3 shows the classification accuracy of all six models on our stress tests and the original MultiNLI development set. We see that performance of all models drops across all stress tests. On competence stress tests, no model is a clear winner, with RC and CH performing best on antonymy and numerical reasoning respectively. On distraction tests, CH is the best-performing model, suggesting that their gated-attention mechanism handles shallow word-level distractions to some extent. Interestingly, our...
A MultiNLI adversarial evaluation

### Outcome 1
**Dataset weakness**

(a) Word Overlap

(b) Negation

### Outcome 2
**Model weakness**

(c) Spelling Errors

(d) Length Mismatch

### Outcome 3
**Dataset artifacts or other problem**

(e) Numerical Reasoning

---

Liu et al. 2019;

Antonym not tested because its label is always ‘contradiction’
Adversarial NLI
Adversarial NLI: Dataset creation

A direct response to adversarial test failings *NLI datasets:

1. The annotator is presented with a premise sentence and a condition (entailment, contradiction, neutral).

2. The annotator writes a hypothesis.

3. A state-of-the-art model makes a prediction about the premise–hypothesis pair.

4. If the model’s prediction matches the condition, the annotator returns to step 2 to try again.

5. If the model was fooled, the premise–hypothesis pair is independently validated by other annotators.
## Adversarial NLI: Example

<table>
<thead>
<tr>
<th>Premise</th>
<th>Hypothesis</th>
<th>Reason</th>
<th>Label</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>A melee weapon is any weapon used in direct hand-to-hand combat; by contrast with ranged weapons which act at a distance. The term “melee” originates in the 1640s from the French word “mélée”, which refers to hand-to-hand combat, a close quarters battle, a brawl, a confused fight, etc. Melee weapons can be broadly divided into three categories</td>
<td>Melee weapons are good for ranged and hand-to-hand combat.</td>
<td>Melee weapons are good for hand to hand combat, but NOT ranged.</td>
<td>E</td>
<td>N</td>
</tr>
</tbody>
</table>
### Adversarial NLI results

<table>
<thead>
<tr>
<th>Model</th>
<th>Data</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>ANLI</th>
<th>ANLI-E</th>
<th>SNLI</th>
<th>MNLI-m/-mm</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT</td>
<td>S,M*1</td>
<td>00.0</td>
<td>28.9</td>
<td>28.8</td>
<td>19.8</td>
<td>19.9</td>
<td>91.3</td>
<td>86.7 / 86.4</td>
</tr>
<tr>
<td></td>
<td>+A1</td>
<td>44.2</td>
<td>32.6</td>
<td>29.3</td>
<td>35.0</td>
<td>34.2</td>
<td>91.3</td>
<td>86.3 / 86.5</td>
</tr>
<tr>
<td></td>
<td>+A1+A2</td>
<td>57.3</td>
<td>45.2</td>
<td>33.4</td>
<td>44.6</td>
<td>43.2</td>
<td>90.9</td>
<td>86.3 / 86.3</td>
</tr>
<tr>
<td></td>
<td>+A1+A2+A3</td>
<td>57.2</td>
<td>49.0</td>
<td>46.1</td>
<td>50.5</td>
<td>46.3</td>
<td>90.9</td>
<td>85.6 / 85.4</td>
</tr>
<tr>
<td></td>
<td>S,M,F,ANLI</td>
<td>57.4</td>
<td>48.3</td>
<td>43.5</td>
<td>49.3</td>
<td>44.2</td>
<td>90.4</td>
<td>86.0 / 85.8</td>
</tr>
<tr>
<td>XLNet</td>
<td>S,M,F,ANLI</td>
<td>67.6</td>
<td>50.7</td>
<td>48.3</td>
<td>55.1</td>
<td>52.0</td>
<td>91.8</td>
<td>89.6 / 89.4</td>
</tr>
<tr>
<td>RoBERTa</td>
<td>S,M</td>
<td>47.6</td>
<td>25.4</td>
<td>22.1</td>
<td>31.1</td>
<td>31.4</td>
<td>92.6</td>
<td>90.8 / 90.6</td>
</tr>
<tr>
<td></td>
<td>+F</td>
<td>54.0</td>
<td>24.2</td>
<td>22.4</td>
<td>32.8</td>
<td>33.7</td>
<td>92.7</td>
<td>90.6 / 90.5</td>
</tr>
<tr>
<td></td>
<td>+F+A1*2</td>
<td>68.7</td>
<td>19.3</td>
<td>22.0</td>
<td>35.8</td>
<td>36.8</td>
<td>92.8</td>
<td>90.9 / 90.7</td>
</tr>
<tr>
<td></td>
<td>+F+A1+A2*3</td>
<td>71.2</td>
<td>44.3</td>
<td>20.4</td>
<td>43.7</td>
<td>41.4</td>
<td>92.9</td>
<td>91.0 / 90.7</td>
</tr>
<tr>
<td></td>
<td>S,M,F,ANLI</td>
<td>73.8</td>
<td>48.9</td>
<td>44.4</td>
<td>53.7</td>
<td>49.7</td>
<td>92.6</td>
<td>91.0 / 90.6</td>
</tr>
</tbody>
</table>

Table 3: Model Performance. ‘Data’ refers to training dataset (‘S’ refers to SNLI, ‘M’ to MNLI dev (-m=matched, -mm=mismatched), and ‘F’ to FEVER); ‘A1–A3’ refer to the rounds respectively. ‘-E’ refers to test set examples written by annotators exclusive to the test set. Datasets marked ‘*n’ were used to train the base model for round n, and their performance on that round is underlined.
A vision for future development

Zellers et al. (2019)
“a path for NLP progress going forward: towards benchmarks that adversarially co-evolve with evolving state-of-the-art models.”

Nie et al. (2019)
“This process yields a “moving post” dynamic target for NLU systems, rather than a static benchmark that will eventually saturate.”
Dynabench

Rethinking AI Benchmarking

Dynabench is a research platform for dynamic data collection and benchmarking. Static benchmarks have well-known issues: they saturate quickly, are susceptible to overfitting, contain exploitable annotator artifacts and have unclear or imperfect evaluation metrics.

This platform in essence is a scientific experiment: can we make faster progress if we collect data dynamically, with humans and models in the loop, rather than in the old-fashioned static way?
Dynabench

1. NLI (see Nie et al. 2020)
2. QA (see Bartolo et al. 2020)
3. Sentiment (DynaSent; Potts et al. 2021)
4. Hate Speech (Vidgen et al. 2020)
DynaSent
Overview and resources

- Data, code, and models: https://github.com/cgpotts/dynasent
- 121,634 sentences, across two rounds, each with 5 gold labels
- Paper: Potts et al. 2021
- Dynabench: https://dynabench.org
DynaSent overview

Model 0
RoBERTa fine-tuned on sentiment benchmarks

Model 0 used to find challenging naturally occurring sentences

Round 1 Dataset

Human validation

Model 1
RoBERTa fine-tuned on sentiment benchmarks + Round 1 Dataset

Dynabench used to crowdsource sentences that fool Model 1

Round 2 Dataset

Human validation
Round 1

Model 0
RoBERTa fine-tuned on sentiment benchmarks

Model 0 used to find challenging naturally occurring sentences

Round 1 Dataset

Human validation

Human validation
Model 0: RoBERTa-based classifier

### Training data

<table>
<thead>
<tr>
<th></th>
<th>CR</th>
<th>IMDB</th>
<th>SST-3</th>
<th>Yelp</th>
<th>Amazon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>2,405</td>
<td>12,500</td>
<td>42,672</td>
<td>260,000</td>
<td>1,200,000</td>
</tr>
<tr>
<td>Negative</td>
<td>1,366</td>
<td>12,500</td>
<td>34,944</td>
<td>260,000</td>
<td>1,200,000</td>
</tr>
<tr>
<td>Neutral</td>
<td>0</td>
<td>0</td>
<td>81,658</td>
<td>130,000</td>
<td>600,000</td>
</tr>
<tr>
<td>Total</td>
<td>3,771</td>
<td>25,000</td>
<td>159,274</td>
<td>650,000</td>
<td>3,000,000</td>
</tr>
</tbody>
</table>

### Performance on external assessment datasets

<table>
<thead>
<tr>
<th></th>
<th>SST-3</th>
<th>Yelp</th>
<th>Amazon</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dev</td>
<td>Test</td>
<td>Dev</td>
</tr>
<tr>
<td>Positive</td>
<td>85.1</td>
<td>89.0</td>
<td>88.3</td>
</tr>
<tr>
<td>Negative</td>
<td>84.1</td>
<td>84.1</td>
<td>88.8</td>
</tr>
<tr>
<td>Neutral</td>
<td>45.4</td>
<td>43.5</td>
<td>58.2</td>
</tr>
<tr>
<td>Macro avg</td>
<td>71.5</td>
<td>72.2</td>
<td>78.4</td>
</tr>
</tbody>
</table>
Harvesting sentences

Favor sentences where the review is 1-star and Model 0 predicts positive, and where the review is 5-star and Model 0 predicts negative.
Validation

### Instructions

You will be shown 10 sentences from reviews of products and services. For each, your task is to choose from one of four labels:

- **Positive**: The sentence conveys information about the author’s *positive evaluative sentiment*.
- **Negative**: The sentence conveys information about the author’s *negative evaluative sentiment*.
- **No sentiment**: The sentence *does not convey anything* about the author’s positive or negative sentiment.
- **Mixed sentiment**: The sentence conveys a *mix of positive and negative sentiment* with *no clear overall sentiment*.

Here are some simple examples of the labels:

- **Sentence**: This is an under-appreciated little gem of a movie.
  This is **Positive** because it expresses a positive overall opinion.

- **Sentence**: I asked for my steak medium-rare, and they delivered this perfectly!
  This is **Positive** because it puts a positive spin on an aspect of the author’s experience.

- **Sentence**: The screen on this device is a little too bright.
  This is **Negative** because it negatively evaluates an aspect of the product.

- **Sentence**: The book is 972 pages long.
  This is **No sentiment** because it describes a factual matter with no evaluative component.

- **Sentence**: The waiting room is drab but the examination rooms are cheery enough.
  This is **Mixed sentiment** because two different sentiment evaluations are balanced against each other.

- **Sentence**: The entrees are delicious, but the service is so bad that it’s not worth going.
  This is **Negative** because the negative statement outweighs the positive one.

---

1
Sentence: The host did a great job of making me feel unwanted.

- **Positive**: The sentence conveys information about the author’s positive evaluative sentiment.
- **Negative**: The sentence conveys information about the author’s negative evaluative sentiment.
- **No sentiment**: The sentence does not convey anything about the author’s positive or negative sentiment.
- **Mixed sentiment**: The sentence conveys a mix of positive and negative sentiment with no clear overall sentiment.
## Resulting dataset

<table>
<thead>
<tr>
<th></th>
<th>Dist Train</th>
<th>Majority Label</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train</td>
<td>Train</td>
</tr>
<tr>
<td>Positive</td>
<td>130,045</td>
<td>21,391</td>
</tr>
<tr>
<td>Negative</td>
<td>86,486</td>
<td>14,021</td>
</tr>
<tr>
<td>Neutral</td>
<td>215,935</td>
<td>45,076</td>
</tr>
<tr>
<td>Mixed</td>
<td>39,829</td>
<td>3,900</td>
</tr>
<tr>
<td>No Majority</td>
<td>–</td>
<td>10,071</td>
</tr>
<tr>
<td>Total</td>
<td>472,295</td>
<td>94,459</td>
</tr>
</tbody>
</table>

47% adversarial examples
Model 0 versus the humans

Model 0

<table>
<thead>
<tr>
<th></th>
<th>SST-3</th>
<th></th>
<th>Yelp</th>
<th></th>
<th>Amazon</th>
<th></th>
<th>Round 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dev</td>
<td>Test</td>
<td>Dev</td>
<td>Test</td>
<td>Dev</td>
<td>Test</td>
<td>Dev</td>
</tr>
<tr>
<td>Positive</td>
<td>85.1</td>
<td>89.0</td>
<td>88.3</td>
<td>90.5</td>
<td>89.1</td>
<td>89.4</td>
<td>33.3</td>
</tr>
<tr>
<td>Negative</td>
<td>84.1</td>
<td>84.1</td>
<td>88.8</td>
<td>89.1</td>
<td>86.6</td>
<td>86.6</td>
<td>33.3</td>
</tr>
<tr>
<td>Neutral</td>
<td>45.4</td>
<td>43.5</td>
<td>58.2</td>
<td>59.4</td>
<td>53.9</td>
<td>53.7</td>
<td>33.3</td>
</tr>
<tr>
<td>Macro avg</td>
<td>71.5</td>
<td>72.2</td>
<td>78.4</td>
<td>79.7</td>
<td>76.5</td>
<td>76.6</td>
<td>33.3</td>
</tr>
</tbody>
</table>

Five annotators synthesized from our crowd

<table>
<thead>
<tr>
<th></th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>88.1</td>
<td>87.8</td>
</tr>
<tr>
<td>Negative</td>
<td>89.2</td>
<td>89.3</td>
</tr>
<tr>
<td>Neutral</td>
<td>86.6</td>
<td>86.9</td>
</tr>
<tr>
<td>Macro avg</td>
<td>88.0</td>
<td>88.0</td>
</tr>
</tbody>
</table>

Note: 614/1,280 workers never disagreed with the majority label.
## Randomly sampled (short) examples

<table>
<thead>
<tr>
<th>Sentence</th>
<th>Model 0</th>
<th>Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good food nasty attitude by hostesses .</td>
<td>neg</td>
<td>mix, mix, mix, neg, neg</td>
</tr>
<tr>
<td>Not much of a cocktail menu that I saw.</td>
<td>neg</td>
<td>neg, neg, neg, neg, neg, neg</td>
</tr>
<tr>
<td>I scheduled the work for 3 weeks later.</td>
<td>neg</td>
<td>neu, neu, neu, neu, neu, pos</td>
</tr>
<tr>
<td>I was very mistaken, it was much more!</td>
<td>neg</td>
<td>neg, pos, pos, pos, pos, pos</td>
</tr>
<tr>
<td>It is a gimmick, but when in Rome, I get it.</td>
<td>neu</td>
<td>mix, mix, mix, neu, neu</td>
</tr>
<tr>
<td>Probably a little pricey for lunch.</td>
<td>neu</td>
<td>mix, neg, neg, neg, neg, neg</td>
</tr>
<tr>
<td>But this is strictly just my opinion.</td>
<td>neu</td>
<td>neu, neu, neu, neu, neu, pos</td>
</tr>
<tr>
<td>The price was okay, not too pricey.</td>
<td>neu</td>
<td>mix, neu, pos, pos, pos, pos</td>
</tr>
<tr>
<td>The only downside was service was a little slow.</td>
<td>pos</td>
<td>mix, mix, mix, neg, neg</td>
</tr>
<tr>
<td>However there is a 2 hr seating time limit.</td>
<td>pos</td>
<td>mix, neg, neg, neg, neg, neu</td>
</tr>
<tr>
<td>With Alex, I never got that feeling.</td>
<td>pos</td>
<td>neu, neu, neu, neu, neu, pos</td>
</tr>
<tr>
<td>Its ran very well by management.</td>
<td>pos</td>
<td>pos, pos, pos, pos, pos, pos</td>
</tr>
</tbody>
</table>
Round 2

Model 1
RoBERTa fine-tuned on sentiment benchmarks + Round 1 Dataset

Dynabench used to crowdsource sentences that fool Model 1

Round 2 Dataset

Human validation
## Model 1: RoBERTa-based classifier

### Training data

<table>
<thead>
<tr>
<th></th>
<th>CR</th>
<th>IMDB</th>
<th>SST-3</th>
<th>Yelp</th>
<th>Amazon</th>
<th>Round 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>2,405</td>
<td>12,500</td>
<td>128,016</td>
<td>29,841</td>
<td>133,411</td>
<td>339,748</td>
</tr>
<tr>
<td>Negative</td>
<td>1,366</td>
<td>12,500</td>
<td>104,832</td>
<td>30,086</td>
<td>133,267</td>
<td>252,630</td>
</tr>
<tr>
<td>Neutral</td>
<td>0</td>
<td>0</td>
<td>244,974</td>
<td>30,073</td>
<td>133,322</td>
<td>431,870</td>
</tr>
<tr>
<td>Total</td>
<td>3,771</td>
<td>25,000</td>
<td>477,822</td>
<td>90,000</td>
<td>400,000</td>
<td>1,024,248</td>
</tr>
</tbody>
</table>

### Performance on external assessment datasets and Round 1

<table>
<thead>
<tr>
<th></th>
<th>SST-3 Dev</th>
<th>SST-3 Test</th>
<th>Yelp Dev</th>
<th>Yelp Test</th>
<th>Amazon Dev</th>
<th>Amazon Test</th>
<th>Round 1 Dev</th>
<th>Round 1 Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>84.6</td>
<td>88.6</td>
<td>80.0</td>
<td>83.1</td>
<td>83.3</td>
<td>83.3</td>
<td>81.0</td>
<td>80.4</td>
</tr>
<tr>
<td>Negative</td>
<td>82.7</td>
<td>84.4</td>
<td>79.5</td>
<td>79.6</td>
<td>78.7</td>
<td>78.8</td>
<td>80.5</td>
<td>80.2</td>
</tr>
<tr>
<td>Neutral</td>
<td>40.0</td>
<td>45.2</td>
<td>56.7</td>
<td>56.6</td>
<td>55.5</td>
<td>55.4</td>
<td>83.1</td>
<td>83.5</td>
</tr>
<tr>
<td>Macro avg</td>
<td>69.1</td>
<td>72.7</td>
<td>72.1</td>
<td>73.1</td>
<td>72.5</td>
<td>72.5</td>
<td>81.5</td>
<td>81.4</td>
</tr>
<tr>
<td>Model 0</td>
<td>71.5</td>
<td>72.2</td>
<td>78.4</td>
<td>79.7</td>
<td>76.5</td>
<td>76.6</td>
<td>33.3</td>
<td>33.3</td>
</tr>
</tbody>
</table>
Dynabench interface

SENTIMENT ANALYSIS

Find examples that fool the model

Your goal: enter a negative statement that fools the model into predicting positive.

Please pretend you are reviewing a place, product, book or movie.

This year's NAACL was very different because of Covid

Model prediction: positive
Well done! You fooled the model.

 Optionally, provide an explanation for your example: Draft. Click out of input box to save.

Covid is clearly not a good thing
The model probably doesn't know what Covid is

The model inspector shows the layer integrated gradients for the input token layer of the model.

This year's NAACL was very different because of Covid

Switch to next context Submit
The prompt condition

**Sentiment Analysis**

Find examples that fool the model

Your goal: enter a **negative** statement that fools the model into predicting positive or neutral.

**Inspirational Prompt (you can use this as a starting point but it might not be negative):**

The waitress periodically stopped by to say sorry or that it was coming up soon, but we didn't actually get food until almost 7:50.

The waitress periodically stopped by to say sorry in a very nice way, but we didn't actually get food until almost 7:50.

Model prediction: **positive**

*You fooled the model!* It predicted **positive**, but a person would say this sentence is **negative**.

Thank you! You are **required** to confirm that you judge this sentence to be **negative** before you can submit this HIT!

Yes, I confirm that I judge this sentence to be **negative**.

No, I judge this sentence to be **positive or neutral**.

---

The waitress periodically stopped by to say sorry in a very nice way, but we didn't actually get food until almost 7:50.
Validation

Same as in Round 1.
### Resulting dataset

<table>
<thead>
<tr>
<th></th>
<th>Dist Train</th>
<th>Majority Label</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train</td>
<td>Train</td>
</tr>
<tr>
<td>Positive</td>
<td>32,551</td>
<td>6,038</td>
</tr>
<tr>
<td>Negative</td>
<td>24,994</td>
<td>4,579</td>
</tr>
<tr>
<td>Neutral</td>
<td>16,365</td>
<td>2,448</td>
</tr>
<tr>
<td>Mixed</td>
<td>18,765</td>
<td>3,334</td>
</tr>
<tr>
<td>No Majority</td>
<td>–</td>
<td>2,136</td>
</tr>
<tr>
<td>Total</td>
<td>92,675</td>
<td>18,535</td>
</tr>
</tbody>
</table>

19% adversarial examples
## Model 1 versus the humans

### Model 1

<table>
<thead>
<tr>
<th></th>
<th>SST-3</th>
<th>Yelp</th>
<th>Amazon</th>
<th>Round 1</th>
<th>Round 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dev</td>
<td>Test</td>
<td>Dev</td>
<td>Test</td>
<td>Dev</td>
</tr>
<tr>
<td>Positive</td>
<td>84.6</td>
<td>88.6</td>
<td>80.0</td>
<td>83.1</td>
<td>83.3</td>
</tr>
<tr>
<td>Negative</td>
<td>82.7</td>
<td>84.4</td>
<td>79.5</td>
<td>79.6</td>
<td>78.7</td>
</tr>
<tr>
<td>Neutral</td>
<td>40.0</td>
<td>45.2</td>
<td>56.7</td>
<td>56.6</td>
<td>55.5</td>
</tr>
<tr>
<td>Macro avg</td>
<td>69.1</td>
<td>72.7</td>
<td>72.1</td>
<td>73.1</td>
<td>72.5</td>
</tr>
</tbody>
</table>

### Five annotators synthesized from our crowd

<table>
<thead>
<tr>
<th></th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>91.0</td>
<td>90.9</td>
</tr>
<tr>
<td>Negative</td>
<td>91.2</td>
<td>91.0</td>
</tr>
<tr>
<td>Neutral</td>
<td>88.9</td>
<td>88.2</td>
</tr>
<tr>
<td>Macro avg</td>
<td>90.4</td>
<td>90.0</td>
</tr>
</tbody>
</table>

Note: 116/244 workers *never* disagreed with the majority label.
## Randomly sampled (short) examples

<table>
<thead>
<tr>
<th>Sentence</th>
<th>Model 1</th>
<th>Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>The place was somewhat good and not well</td>
<td>neg</td>
<td>mix, mix, mix, mix, neg</td>
</tr>
<tr>
<td>I bought a new car and met with an accident.</td>
<td>neg</td>
<td>neg, neg, neg, neg, neg, neg</td>
</tr>
<tr>
<td>The retail store is closed for now at least.</td>
<td>neg</td>
<td>neu, neu, neu, neu, neu, neu</td>
</tr>
<tr>
<td>Prices are basically like garage sale prices.</td>
<td>neg</td>
<td>neg, neu, pos, pos, pos</td>
</tr>
<tr>
<td>That book was good. I need to get rid of it.</td>
<td>neu</td>
<td>mix, mix, mix, neg, pos</td>
</tr>
<tr>
<td>I REALLY wanted to like this place</td>
<td>neu</td>
<td>mix, neg, neg, neg, pos</td>
</tr>
<tr>
<td>I’m going to leave my money for the next vet.</td>
<td>neu</td>
<td>neg, neu, neu, neu, neu, neu</td>
</tr>
<tr>
<td>once the model made a super decision.</td>
<td>neu</td>
<td>pos, pos, pos, pos, pos</td>
</tr>
<tr>
<td>I cook my caribbean food and it was okay</td>
<td>pos</td>
<td>mix, mix, mix, pos, pos</td>
</tr>
<tr>
<td>This concept is really cool in name only.</td>
<td>pos</td>
<td>mix, neg, neg, neg, neu</td>
</tr>
<tr>
<td>Wow, it’d be super cool if you could join us</td>
<td>pos</td>
<td>neu, neu, neu, neu, neu, pos</td>
</tr>
<tr>
<td>Knife cut thru it like butter! It was great.</td>
<td>pos</td>
<td>pos, pos, pos, pos, pos</td>
</tr>
</tbody>
</table>
Conclusions
Key open questions

1. Can adversarial training improve systems? (See Jia and Liang 2017:§4.6; Alzantot et al. 2018:§4.3; Liu et al. 2019; Iyyer et al. 2018.)

2. What constitutes a *fair* non-IID generalization test?

3. Can hard behavioral testing provide us with the insights we need when it comes to certifying systems as trustworthy? If so, which tests? If not, what should be done instead?

4. Are systems finding systematic solutions?

5. Where humans generalize in ways that are unsupported by direct experience, how should AI respond in terms of system design?
References I


References III


