What Does My QA Model Know? Devising Controlled Probes Using Expert Knowledge

Kyle Richardson and **Ashish Sabharwal**
Allen Institute for AI, Seattle, WA, USA
{kyler,ashishs@allenai.org}

Abstract

Open-domain question answering (QA) involves many knowledge and reasoning challenges, but are successful QA models actually learning such knowledge when trained on benchmark QA tasks? We investigate this via several *new diagnostic tasks* probing whether multiple-choice QA models know definitions and taxonomic reasoning—two skills widespread in existing benchmarks and fundamental to more complex reasoning. We introduce a methodology for automatically building probe datasets from *expert knowledge sources*, allowing for systematic control and a comprehensive evaluation. We include ways to carefully control for artifacts that may arise during this process. Our evaluation confirms that transformer-based multiple-choice QA models are already predisposed to recognize certain types of structural linguistic knowledge. However, it also reveals a more nuanced picture: their performance notably degrades even with a slight increase in the number of ''hops'' in the underlying taxonomic hierarchy, and with more challenging distractor candidates. Further, existing models are far from perfect when assessed at the level of clusters of semantically connected probes, such as all hypernym questions about a single concept.

1 Introduction

Automatically answering questions, especially in the open-domain setting where minimal or no contextual knowledge is explicitly provided, requires considerable background knowledge and reasoning abilities. For example, answering the two questions in the top gray box in Figure 1 requires identifying a specific *ISA relation* (that 'cooking' is a type of 'learned behavior') as well as recalling a concept *definition* (that 'global warming' is defined as a 'worldwide increase in temperature').

Recent success in QA has been driven largely by new benchmarks (Zellers et al., 2018; Talmor et al., 2019b; Bhagavatula et al., 2020; Khot et al., 2020) and advances in model pre-training (Radford et al., 2018; Devlin et al., 2019). This raises a natural question: *Do state-of-the-art multiple-choice QA (MCQA) models that excel at standard benchmarks truly possess basic knowledge and reasoning skills expected in these tasks?*

Answering this question is challenging because of limited understanding of heavily pre-trained complex models and the way existing MCQA datasets are constructed. We focus on the second aspect, which has two limitations: Large-scale crowdsourcing leaves little systematic control over question semantics or requisite background knowledge (Welbl et al., 2017), while questions from real exams tend to mix multiple challenges in a single dataset, often even in a single question (Clark et al., 2018; Boratko et al., 2018).

To address this challenge, we propose systematically constructing model competence probes by exploiting structured information contained in *expert knowledge sources* such as knowledge graphs and lexical taxonomies. Importantly, these probes are diagnostic tasks, designed not to impart new knowledge but to assess what models trained on standard QA benchmarks already know; as such, they serve as proxies for the types of questions that a model might encounter in its original task, but involve a single category of knowledge under various controlled conditions and perturbations.

Figure 1 illustrates our methodology. We start with a set of standard MCQA benchmark tasks $\mathcal{D}$ and a set of models $\mathcal{M}$ trained on $\mathcal{D}$. Our goal is to assess how competent these models are relative to a particular knowledge or reasoning skill S (e.g., definitions) that is generally deemed important for performing well on D. To this end, we

Transactions of the Association for Computational Linguistics, vol. 8, pp. 572–588, 2020. https://doi.org/10.1162/tacl_a_00331
Action Editor: Dipanjan Das. Submission batch: 2/2020; Revision batch: 5/2020; Published 9/2020.

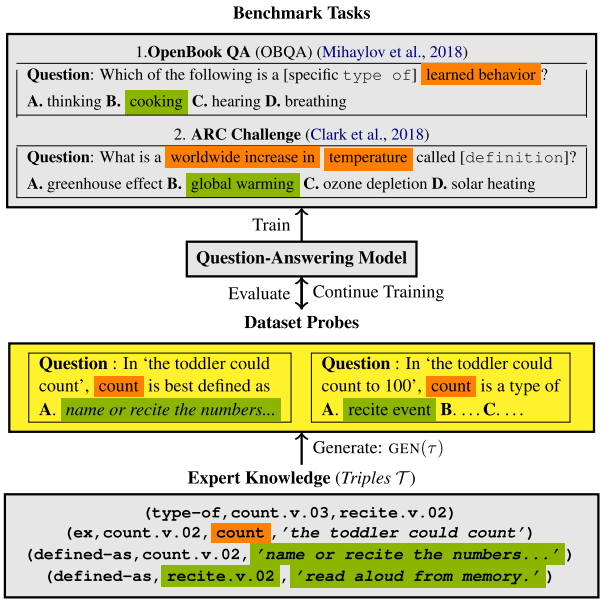


Figure 1: An illustration of our experimental setup and probing methodology. The gray box at the top shows questions from existing open-domain QA benchmarks, requiring background knowledge. The yellow box shows simple examples of multiple-choice questions in our proposed Definition and *ISA* probes.

systematically and automatically generate a set of *dataset probes* P_S from information available in expert knowledge sources. Each probe is an MCQA rendering of the target information (see examples in Figure 1, yellow box). We then use these probes P_S to ask two empirical questions: (1) How well do models in $\mathcal{M}$ already trained on $\mathcal{D}$ perform on probing tasks P_S? (2) With additional nudging, can models be re-trained, using only a modest amount of additional data, to perform well on each probing task P_S with minimal performance loss on their original tasks D (thus giving evidence of prior model competence on S)?

While our methodology is general, our experiments focus on probing state-of-the-art MCQA models in the domain of grade-school level science, which is considered particularly challenging with respect to background knowledge and inference (Clark, 2015; Clark et al., 2019; Khot et al., 2020). In addition, existing science benchmarks are known to involve widespread use of definition and taxonomic knowledge (see detailed analysis by Clark et al. [2018], Boratko et al. [2018]), which is also fundamental to deeper reasoning. Accordingly, we use the most widely used lexical ontology WordNet (Miller, 1995) and publicly available dictionaries as sources of expert knowledge to construct our probes, WordNetQA (Section 3.1) and DictionaryQA (Section 3.2).[1] These probes measure competence in various settings including hypernymy, hyponymy, and synonymy detection, as well as word sense disambiguation.

Our exploration is closely related to the recent work of Talmor et al. (2019a). However, a key difference is that they study language models (LMs), for which there is *no clear a priori expectation* of specific knowledge or reasoning skills. In contrast, we focus on models heavily trained for benchmark QA tasks, where such tasks are known to require certain types of knowledge and reasoning skills. We probe whether such skills are actually learned by QA models, either during LM pre-training or when training for the QA tasks.

Recognizing the need for suitable controls in any synthetic probing methodology (Hewitt and Liang, 2019; Talmor et al., 2019a), we introduce two controls: (a) the probe must be challenging for any model that lacks contextual embeddings, and (b) strong models must have a *low inoculation cost*—that is, when fine-tuned on a few probing examples, the model should mostly retain its performance on its original task.[2] This ensures that the probe performance of a model, even when lightly inoculated on probing data, reflects its knowledge as originally trained for the benchmark task, which is precisely what we aim to uncover.

Constructing a wide range of systematic tests is critical for having definitive empirical evidence of model competence on any given phenomenon. Such tests should cover a broad set of concepts and question *variations* (i.e., systematic adjustments to how the questions are constructed). When assessing *ISA* reasoning, not only is it important to recognize in the question in Figure 1 that *cooking* is a *learned behavior*, but also that *cooking* is a general type of *behavior* or, through a few more inferential steps, a type of *human activity*. Our automatic use of expert knowledge sources allows constructing such high-coverage probes, circumventing pitfalls of solicitation bias and reporting bias.

[1] All data and code are available at https://github.com/allenai/semantic_fragments.

[2] Standard inoculation (Liu et al., 2019a) is known to drop performance on the original task. We use a modified objective (Richardson et al., 2020) to alleviate this issue.

Our results confirm that transformer-based QA models[3] have a remarkable ability to recognize the types of knowledge captured in our probes—even without additional fine-tuning (i.e., in a *zero-shot* setting). Such models can even outperform strong task-specific non-transformer models trained directly on our probing tasks (e.g., +26% compared to a task-specific LSTM). We also show that the same models can be effectively re-fine-tuned on small samples (even 100 examples) of probe data, and that high performance on the probes tends to correlate with a smaller drop in the model's performance on the original QA task.

Our comprehensive assessment also reveals important nuances to the positive trend. For example, we find that the best models still perform 2–10% (absolute) below conservative estimates of human performance (Section 3.1.3) on these tasks. Further, the accuracy of even the best QA model degrades substantially on our hyponym probes (by 8–15%) when going from 1-hop hyponym links to 2-hops. The accuracy on the WordNetQA probe drops by 14–44% under our *cluster-level analysis* (Section 3.1.1), which assesses whether a model knows several facts about each individual concept, rather than only answering correctly isolated questions. This shows that state-of-the-art QA models have much room to improve even in some fundamental building blocks (definitions and taxonomic hierarchies) of more complex forms of reasoning.

2 Related Work

We follow recent work on constructing challenge datasets for probing neural models, which has primarily focused on the task of natural language inference (NLI) (Glockner et al., 2018; McCoy et al., 2019; Rozen et al., 2019; Warstadt et al., 2019). Most of this work looks at constructing data through adversarial generation methods, which have also been found useful for creating stronger models (Kang et al., 2018). There has also been work on using synthetic data of the type we consider in this paper (Poliak et al., 2018a; Geiger et al., 2019; Yanaka et al., 2020; Clark et al., 2020). We closely follow the methodology of Richardson et al. (2020), who use hand-constructed linguistic fragments to probe NLI models and study model re-training using a variant of the *inoculation by fine-tuning* strategy of Liu et al. [2019a]. In contrast, we focus on probing open-domain MCQA models (see Si et al. (2019) for a study on *reading comprehension*) as well as constructing data from much larger sources of structured knowledge.

Our main study focuses on probing the BERT model and fine-tuning approach of Devlin et al. (2019), and other variants thereof, which are all based on the transformer architecture of Vaswani et al. (2017). There have been recent studies into the types of relational knowledge contained in large-scale knowledge models (Schick and Schütze, 2020; Petroni et al., 2019; Jiang et al., 2019), which also probe models using structured knowledge sources. These studies, however, primarily focus on unearthing the knowledge contained in the underlying language models *as is* without further training, using simple (single token) cloze-style probing tasks and templates. Most of these results only provide a *lower-bound* estimate of model performance, since the probing templates being used potentially deviate from what the model has observed during pre-training. In contrast, we focus on understanding the knowledge contained in language models *after* they have been trained for a QA end-task using benchmark datasets in which such knowledge is expected to be widespread. Further, our evaluation is done before and *after* these models are fine-tuned on our small samples of target data. This has the advantage of allowing each model to become informed about the format of each probe. We also explore a more complex set of probing templates.

The use of lexical resources such as WordNet to construct datasets has a long history, and has recently appeared in work on adversarial attacks (Jia and Liang, 2017) and general task construction (Pilehvar and Camacho-Collados, 2019). In the area of MCQA, there is related work on constructing questions from tuples (Jauhar et al., 2016; Talmor et al., 2019b), both of which involve standard crowd annotation to elicit question-answer pairs (see also Seyler et al., 2017; Reddy et al., 2017). In contrast to this work, we focus on generating data in an entirely automatic and *silver-standard* fashion (i.e., in a way that potentially introduces a little noise), which obviates the need for expensive annotation and gives us the flexibility to construct much larger datasets that control a rich set of semantic aspects of the

[3]Different from Talmor et al. (2019a), we find BERT and RoBERTa based QA models to be qualitatively similar, performing within 5% of each other on nearly all probes.

target questions. Following standard practices in MCQA dataset creation (e.g., Khot et al., 2020), however, we perform crowd-sourcing to obtain *conservative* (in the sense of Nangia and Bowman [2019]) estimates of human performance on our main evaluation sets, to compare against model performance.

Although our probing methodology is amenable to any domain, we focus on probing open-domain QA models in the domain of grade-school level science using a standard suite of benchmark QA datasets (see Table 6). Our choice of this domain is based on the following considerations: It is well-studied qualitatively (Davis, 2016), making it relatively easy to know the types of probes and diagnostic tests to construct using existing expert knowledge. For example, the manual analysis of Mihaylov et al. (2018) found that *explicit* definitional and ISA knowledge occurred in around 20% and 18%, respectively, of the questions sampled in one benchmark task. Clark et al. (2013) and Boratko et al. (2018) provide similar results involving other benchmarks used in our study.

We also examined MCQA models trained on closely related datasets tailored to commonsense and situational reasoning (Zellers et al., 2018; Talmor et al., 2019b; Bhagavatula et al., 2020; Sap et al., 2019). However, there has been a limited study of the kinds of knowledge needed in this domain, as well as expert knowledge sources for creating corresponding probes. MCQA models trained in this domain exhibit lower performance on our definition and ISA probes.

3 Dataset Probes and Construction

Our probing methodology starts by constructing challenge datasets (Figure 1, yellow box) from a target set of knowledge resources. Each probing dataset consists of multiple-choice questions that include a *question* **q** and a set of *answer choices* or candidates $\{a_1, ...a_N\}$. This section describes in detail the 5 datasets we build (grouped into **WordNetQA** and **DictionaryQA**), drawn from two publicly available resources: WordNet (Miller, 1995) and the GNU Collaborative International Dictionary of English (GCIDE).[4]

For convenience, we will describe each source of expert knowledge as a directed, edge-labeled graph G. The nodes of this graph are $\mathcal{V} = \mathcal{C} \cup \mathcal{W} \cup \mathcal{S} \cup \mathcal{D}$, where $\mathcal{C}$ is a set of atomic concepts, $\mathcal{W}$ a set of words, $\mathcal{S}$ a set of sentences, and $\mathcal{D}$ a set of definitions (see Table 1 for details for WordNet and GCIDE). Each edge of G is directed from an atomic concept in $\mathcal{C}$ to another node in V, and is labeled with a relation, such as hypernym or $\texttt{isa}^{\uparrow}$, from a set of relations $\mathcal{R}$ (see Table 1).

Set	WordNet (WN)	GCIDE
$\mathcal{R}$	{isa↑, isa↓, def, ex, lemma}	{def, ex, lemma}
$\mathcal{C}$	{**WN synsets**}	{**entry ids**}
$\mathcal{D}$	{**synset glosses**}	{**unique defs**}
$\mathcal{S}$	{**synset sentences**}	{**entry examples**}
$\mathcal{W}$	{**synset lemmas**}	{**all words**}

Atomic Triple Types	Definition
Concept Senses and Definitions	$\mathcal{T}_d \subseteq \{\texttt{def}\} \times \mathcal{C} \times \mathcal{D}$
Concepts with Example Sentences	$\mathcal{T}_e \subseteq \{\texttt{ext}\} \times \mathcal{C} \times \mathcal{S}$
Concepts with Words	$\mathcal{T}_l \subseteq \{\texttt{lemma}\} \times \mathcal{C} \times \mathcal{W}$
ISA Relations (*WN only*)	$\mathcal{T}_i \subseteq \{\texttt{isa}^{\uparrow}, \texttt{isa}^{\downarrow}\} \times \mathcal{C} \times \mathcal{C}$

Table 1: A description of the different resources used to construct the probes, represented as abstract triples.

When defining our probe question templates, it will be useful to view G as a set of *(relation, source, target)* **triples** $\mathcal{T} \subseteq \mathcal{R} \times \mathcal{C} \times \mathcal{V}$. Because of their origin in an expert knowledge source, such triples preserve semantic consistency. For instance, when the *relation* in a triple is def, the corresponding edge maps a concept in $\mathcal{C}$ to a definition in $\mathcal{D}$.

We rely on two heuristic functions, defined below for each individual probe: $\text{GEN}_{\mathcal{Q}}(\tau)$, which generates gold question-answer pairs $(\mathbf{q}, \mathbf{a})$ from a set of triples $\tau \subseteq \mathcal{T}$ and question templates $\mathcal{Q}$, and $\text{DISTR}(\tau')$, which generates distractor answers choices $\{a'_1, ...a'_{N-1}\}$ based on another set of triples τ' (where usually $\tau \subset \tau'$). For brevity, we will use $\text{GEN}(\tau)$ to denote $\text{GEN}_{\mathcal{Q}}(\tau)$.

In generating our dataset probes, our general strategy is to build automatic *silver-standard* training and developments sets, in the latter case at a large scale to facilitate detailed and controlled analysis of model performance. As discussed below, we also provide estimates of human performance on our test sets, and in some cases introduce smaller gold-standard test sets to allow for a direct comparison with model performance.

3.1 WordNetQA

WordNet is a publicly available English lexical database consisting of around 117k concepts, which are organized into groups of *synsets* that each contain a *gloss* (i.e., a definition), a set of representative English words (called *lemmas*), and, in around 33k synsets, example sentences.

[4]See https://wordnet.princeton.edu/ and http://gcide.gnu.org.ua/.

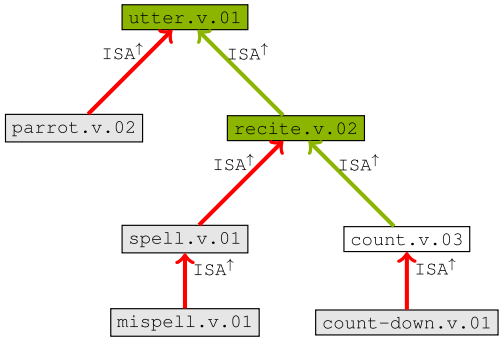
In addition, many synsets have ISA links to other synsets that express complex taxonomic relations. Figure 2 shows an example and Table 1 summarizes how we formulate WordNet as a set of triples $\mathcal{T}$ of various types. These triples together represent a directed, edge-labeled graph G.

Our main motivation for using WordNet, as opposed to a resource such as ConceptNet (Havasi et al., 2007), is the availability of glosses ($\mathcal{D}$) and example sentences ($\mathcal{S}$), which allows us to construct natural language questions that contextualize the types of concepts we want to probe. For example, when probing whether a model has knowledge of a concept such as *bank* (a financial institution), we provide an example sentence *he cashed a check at the bank*, to help disambiguate the particular sense of *bank* we are probing. Sentential contexts also provide additional hints to models in cases of rare or infrequent concepts.[5] Because WordNet is the most authoritative and widely used knowledge resource in NLP, it also has the advantage of having mappings into other knowledge resources (Niles and Pease, 2001; Navigli and Ponzetto, 2010; Tandon et al., 2017), which allows for easily extending our probes to other domains and phenomena.

Example Generation GEN(τ)**.** We build 4 individual datasets based on semantic relations native to WordNet: `hypernymy` (i.e., generalization or ISA reasoning up a taxonomy, ISA$^{\uparrow}$), `hyponymy` (ISA$^{\downarrow}$), `synonymy`, and `definitions`. To generate a set of questions in each case, we use a number of rule templates $\mathcal{Q}$ that operate over tuples. A subset of such templates is shown in Table 2 and were designed to mimic *naturalistic* (i.e., human-authored) questions we observed in our science benchmarks.

For example, suppose we wish to create a question **q** about the definition of a target concept $c \in \mathcal{C}$. We first select a question template from $\mathcal{Q}$ that first introduces the concept c and its lemma $l \in \mathcal{W}$ in context using the example sentence context $s \in \mathcal{S}$, and then asks to identify the corresponding WordNet gloss $d \in \mathcal{D}$, which serves as the gold answer **a**. The same is done for ISA reasoning;

[5]Given the open-domain nature of WordNet, not all probed concepts may have *explicitly* been observed during QA training. Nevertheless, unlike prior probing studies (Petroni et al., 2019), we did not see a substantial performance disparity between observed and unobserved concepts across our models, perhaps owing to the provided contexts.



Graph Triples	Question/Answers
Question+Answer about Hypernymy/ISA$^{\uparrow}$	
(isa$^{\uparrow}$, count.v.03, recite.v.02)	**q**. In the sentence The toddler could count, the word count is a type of: **a**. recite event...
Sister Family Distractors	
(isa$^{\downarrow}$, recite.v.02, spell.v.01)	a'_1. spell event, defined as ... (*1-hop sister distractor*); a'_2 mispell event, defined as... (*2-hop sister*).

Figure 2: A portion of the WordNet ISA graph (top) and an example distractor function DISTR(τ) (bottom) used to generate distractor choices $\{a'_1, a'_2\}$ for a question **q** based on information in the graph.

each question about a hypernym/hyponym relation between two concepts $c \rightarrow^{\uparrow/\downarrow} c' \in \mathcal{T}_i$ (e.g., `dog` $\rightarrow^{\uparrow/\downarrow}$ `animal/terrier`) first introduces a context for c and then asks for an answer that identifies c' (which is also provided with a gloss so as to contain all available context).

In the latter case, the rules $(\texttt{isa}^r, c, c') \in \mathcal{T}_i$ in Table 2 cover only *direct* ISA links from c in direction $r \in \{\uparrow, \downarrow\}$. In practice, for each c and r, we construct tests that cover the set HOPS(c, r) of *all* direct as well as derived ISA relations of c:

$$\text{HOPS}(c, r) := \left\{ (\texttt{isa}^r, c, c') \in \mathcal{T}_i \right\} \cup \text{HOPS}(c', r)$$

This allows us to evaluate the extent to which models are able to handle complex forms of reasoning that require several inferential steps or *hops*.[6]

Distractor Generation: DISTR(τ')**.** Figure 2 shows an example of how distractors are generated, relying on similar principles as above. For each concept c, we choose 4 distractor answers that are close in the WordNet semantic space. For example, when constructing hypernymy tests for c from the set HOPS$(c, \uparrow)$, we draw distractors

[6]In practice, most WordNet synsets have no more than 5 hops. We use this as a default limit when building datasets.

Probe Type	**Triple Input** τ	**Generation Templates** from $\mathcal{Q}$	**Example Questions and Answers** (q, a)
`Definitions`: Defining words in context.	$(\texttt{def}, c_i, d)$ $(\texttt{ex}, c_i, s)$ $(\texttt{word}, c_i, w)$	**q**. *In the sentence* $[s]$*, the word* $[w]$ *is best defined as:* ***a***. $[d]$	**q**. In the sentence The baby nestled her head, the word nestled is best defined as: **a**. position comfortably
`Hypernymy`: ISA$^{\uparrow}$ reasoning in context (symbolically c_i=>$c_{i'}$).	$(\texttt{def}, c_{i'}, d)$ $(\texttt{isa}^{\uparrow}, c_i, c_{i'})$ $(\texttt{ex}, c_i, s)$ $(\texttt{word}, c_i, w)$ $(\texttt{word}, c_{i'}, w')$	**q**. *In* $[s]$*, the word or concept* $[w]$ *is best described as a type of* ***a***. $[w']$ *defined as* $[d]$	**q**. In The thief eluded the police, the word or concept eluded is best described as a type of **a**. escape event defined as to run away from..
`Hyponymy`: ISA$^{\downarrow}$ reasoning given context. (symbolically c_i<=$c_{i'}$)	$(\texttt{def}, c_{i'}, d)$ $(\texttt{isa}^{\downarrow}, c_i, c_{i'})$ $(\texttt{ex}, c_i, s)$ $(\texttt{word}, c_i, w)$ $(\texttt{word}, c_i, w')$	**q**. *Given the context* $[s]$*, which of the following word or concept is a specific type of* $[w]$ ***a***. $[w']$ *defined as* $[d]$	**q**. Given the context they awaited her arrival, which of the following word or concept is a specific type of arrival? **a**. crash landing, defined as an emergency landing under circumstances where....'
`Synonymy`: Related words.	$(\texttt{def}, c_i, d)$ $(\texttt{word}, c_i, w_1)$ $(\texttt{word}, c_i, w_2)$	**q**. *Which words best correspond to* $[d]$*?* ***a***. $[\{w_1, w_2, ...\}]$	**q**. Which set of words best corresponds to the definition a grammatical category in inflected languages governing agreement ....? **a**. gender,...

Table 2: Details of the GEN(τ) function used to construct gold question-answer pairs $(\mathbf{q}, \mathbf{a})$ from a triple graph G.

Target Concept	**Example Question**	**Inferences** *(target answers in symbolic form)*
trouser.n.01, **gloss**: *a garment extending from the waist to the knee or ankle covering each leg..*	**q**. In he had a sharp crease in his trousers, the word/phrase trousers is best defined as a type of	`trouser.n.01 =>` **`consumer_goods.n.01`** `trouser.n.01 =>` **`garment.n.01`** `trouser.n.01 =>` **`commodity.n.01`** `trouser.n.01 =>` **`clothing.n.01`**
oppose.v.06, **gloss**: *be resistant to*	**q**. In the sentence or expression The board opposed his motion, the following is a more specific type of opposed [or opposition]	`oppose.v.06 <=` **`protest.v.02`** `oppose.v.06 <=` **`veto.v.01`** `oppose.v.06 <=` **`demonstrate.v.04`**
poet_laureate.n.01, **gloss**: *a poet who is . . . holding an honorary position...*	**q**. Given the fragment he is the poet laureate of Arkansas, poet laureate . . . is best described as a type of	`poet_laureate.n.01=>`**`poet.n.01`** `poet_laureate.n.01=>`**`communicator.n.01`** `poet_laureate.n.01=>`**`writer.n.01`**

Table 3: Semantic clusters for three target concepts, involving ISA reasoning.

from HOPS$(c, \downarrow)$, as well as from the *ℓ-deep sister family* of c, defined as follows. The 1-deep sister family is simply c's siblings or sisters (i.e., the other children $\tilde{c} \neq c$ of the parent node c' of c). For $\ell > 1$, the ℓ-deep sister family also includes all descendants of each $\tilde{c}$ up to $\ell - 1$ levels deep, denoted HOPS$_{\ell-1}(\tilde{c}, \downarrow)$. Formally:

$$\text{SISTER}_{\ell}(c) := \Big\{ x \in \text{HOPS}_{\ell-1}(\tilde{c}, \downarrow) \mid (\texttt{isa}^{\uparrow}, c, c') \in \mathcal{T}_i, (\texttt{isa}^{\uparrow}, \tilde{c}, c') \in \mathcal{T}_i, \tilde{c} \neq c \Big\}$$

For definitions and synonyms, we build distractors from all of these sets (with a similar depth limit for SISTER distractors), enabling a systematic investigation via a wide range of distractors.

3.1.1 Perturbations and Semantic Clusters

For each concept c (an atomic WordNet synset) and probe type (definitions, hypernymy, etc.), we have a wide variety of questions related to c that manipulate (1) the complexity of reasoning that is involved (e.g., the number of inferential *hops*) and (2) the types of distractors (or *distractor perturbations*) that are used. We call such sets *semantic clusters*.

Table 3 shows three examples, capturing ISA reasoning about the following target concepts: *trousers*, *opposing*, and *poet laureate*. Such clusters enable new types of evaluation of the comprehensiveness and consistency of a model's knowledge of target concepts.

Probe	# Questions (Unique / w Perturb.)	Cluster Size (Avg.)	# Synsets (or concepts)
Hypernymy	19,705 / 35,094	5	7,849
Hyponymy	6,697 / 35,243	11	3,452
Synonymy	28,254 / 91,069	6	15,632
Definitions	31,380 / 148,662	10	15,159
WordSense	∼7,000 / –	1	∼7,000

Table 4: Details of our dataset probes, including both the number of *unique* (**q**, **a**) pairs (for **WordNetQA**) and the number of all questions including distractor choice perturbations (*w Perturb.*).

3.1.2 Summary of Probe Datasets

Details of the individual datasets, including average cluster sizes, are summarized in Table 4.

From these sets, we follow Richardson et al. (2020) in allocating a maximum of 3k examples for *inoculating* the models in the manner described in the next section (i.e., for continuing to train QA models and introduce them to the format of our probes), and reserve the rest for development and testing. In particular, we build large development sets, which are important for performing detailed analysis and cluster-based evaluation.

3.1.3 Human Performance

We report human scores on the individual test sets in WordNetQA (see bottom of Table 7). This is done in two ways.

First, for our test sets generated for definitions and synonyms that cover a large set of disconnected concepts in the WordNet graph and where it is infeasible to annotate individual instances of concepts, we estimate human performance by having crowd-workers on Amazon Mechanical Turk answer a random sample of 500 test questions. Scores are computed by taking the majority vote for each question among 5 annotators. This follows exactly the evaluation protocol used by Nangia and Bowman (2019) and is a *conservative* estimate in that crowd annotators received virtually no training and no qualification exam before participating in the task.

Second, for our hypernymy and hyponymy test sets, which cover a smaller number of densely connected concepts, we annotated smaller *gold-standard* test sets that include a sample of around 2,000 random questions that cover a large proportion of the concepts being probed and that have high human performance. To do this, we follow the annotation strategy described above, and greedily apply filtering to remove questions incorrectly answered by human annotators, which follows prior work on building evaluation sets for MCQA (Mihaylov et al., 2018; Talmor et al., 2019b; Khot et al., 2020).

GCIDE Dictionary Entries
word: gift, **pos**: n., **definition**: Anything given; anything voluntarily transferred by one person to another without compensation; a present; **entry example**: None.
word: gift, **pos**: n., **definition**: A bribe; anything given to corrupt. **entry example**: None.
word: gift, **pos**: n., **definition**: Some exception inborn quality or characteristic; a striking or special talent or aptitude;.. **entry example**: *the gift of wit; a gift for speaking.*

Table 5: Example dictionary entries for the word *gift*.

3.2 DictionaryQA

The DictionaryQA dataset is created from the English dictionary GCIDE built largely from the Webster's Revised Unabridged Dictionary (Webster, 1913), which has previously been used in other NLP studies because of its large size and public availability (Hill et al., 2016). Each dictionary entry consists of a word, its part-of-speech, its definition, and an optional example sentence, as shown for an example in Table 5. Overall, 33k entries (out of a total of 155k) contain example sentences/usages. As with the WordNet probes, we focus on this subset so as to contextualize each word being probed. Because GCIDE does not have ISA relations or explicit synsets, we take each unique entry to be a distinct sense. Our probe centers around word-sense disambiguation.

To buildQA examples, we use the same generation templates for *definitions* exemplified in Table 2 for WordNetQA. To construct distractors, we simply take alternative definitions for the target words that represent a different word sense (e.g., the alternative definitions of *gift* in Table 5), and randomly chosen definitions if needed to create a 5-way multiple choice question. As above, we reserve a maximum of 3k examples for training, and use the same amount for development.

Science Datasets	#Questions	N
OpenBookQA Mihaylov et al. 2018	4,957	4
SciQ Welbl et al. 2017	11,675	4
TextBookQA Kembhavi et al. 2017	7,611	4/5
ARC Dataset++ Clark et al. 2018	4,035	4/5
MCQL Liang et al. 2018	6,318	4
Science Collection (total)	34,596	5

Table 6: The MCQA training datasets used. **#Question** denotes the number of training samples in our version of each dataset, N the number of choices.

Our initial attempts at building this dataset via standard random splitting resulted in certain systematic biases, revealed by high performance of the **choice-only** model we used as a control. Among other factors, we found the use of definitions from entries without example sentences as distractors (see again Table 5) to have a surprising correlation with such biases. Filtering such distractors helped improve the quality of this probe.

For assessing human performance, we annotated a smaller gold-standard test set consisting of around 1,100 questions using the crowd-sourcing elicitation setup described in Section 3.1.

4 Probing Methodology and Modeling

Given the probes above, we now can start to answer the empirical questions posed at the beginning. Our main focus is on looking at transformer-based MCQA models trained on science benchmarks in Table 6. We start with our target MCQA models, as well as several control baselines.

4.1 Task Definition and Modeling

Given a dataset $D = \{(\mathbf{q}^{(d)}, \{a_1^{(d)}, \ldots, a_N^{(d)}\})\}_d^{|D|}$ consisting of pairs of questions stems $\mathbf{q}$ and answer choices a_i, the goal is to find the correct answer a_{i^*} that correctly answers each $\mathbf{q}$. Throughout this paper, we look at 5-way multiple-choice problems (i.e., where each $N = 5$).

Question+Answer Encoder. Our investigation centers around the use of the transformer-based BERT encoder and fine-tuning approach of Devlin et al. (2019) (see also Radford et al., 2018). For each question and individual answer pair $q_{a_i}^{(j)}$, we assume the following rendering of this input:

$$q_{a_i}^{(j)} := \texttt{[CLS]}\ \ \mathbf{q}^{(j)}\ \ \texttt{[SEP]}\ \ a_i^{(j)}\ \ \texttt{[SEP]}$$

This is run through the pre-trained BERT encoder to generate a representation for $q_{a_i}^{(j)}$ using the hidden state representation for `CLS` (i.e., the *classifier token*): $\mathbf{c}_i^{(j)} = \text{BERT}(q_{a_i}^{(j)}) \in \mathbb{R}^H$. The probability of a given answer $p_i^{(j)}$ is then standardly computed using an additional classification layer over $\mathbf{c}_j$, which is optimized (along with the full transformer network) by taking the final loss of the probability of each correct answer p_{i^*} over all answer choices, i.e., $\mathcal{L} = \sum_{d \in |D|} -\log p_{i^*}^{(d)}$.

We specifically use **BERT-large** uncased with whole-word masking, as well as the **RoBERTa-large** model from Liu et al. (2019b), which is a more robustly trained version of the original BERT model. Our system uses the implementations provided in AllenNLP (Gardner et al., 2018) and Huggingface (Wolf et al., 2019).

Baselines and Sanity Checks. When creating synthetic datasets, it is important to ensure that systematic biases, or *annotation artifacts* (Gururangan et al., 2018), are not introduced into the resulting probes and that the target datasets are sufficiently challenging (or *good*, in the sense of Hewitt and Liang [2019]). To test for this, we use several of the MCQA baseline models first introduced in Mihaylov et al. (2018), which take inspiration from the LSTM-based models used in Conneau et al. (2017) for NLI and various *partial-input* baselines based on these models.

Following Mihaylov et al. (2018)'s notation, for any sequence s of tokens in $\{q^{(j)}, a_1^{(j)}, \ldots, a_N^{(j)}\} \in D$, an encoding of s is given as the following:

$$h_s^{(j)} = \textbf{BiLSTM}(\text{EMBED}(s)) \in \mathbb{R}^{|s| \times 2h},$$

where h is the dimension of the hidden state in each directional network, and $\text{EMBED}(\cdot)$ assigns a token-level embeddings to each token in s.[7] A contextual representation for each s is then built by applying an element-wise `max` operation over h_s as follows:

$$r_s^{(j)} = \texttt{max}(h_s^{(j)}) \in \mathbb{R}^{2h}$$

With these contextual representations, different baseline models can be constructed. For example, a **Choice-Only** model, a variant of the well-known *hypothesis-only* baseline used in NLI (Poliak et al.,

[7] As in Mihaylov et al. (2018), we experiment with using both **GloVe** (Pennington et al., 2014) and **ELMo** (Peters et al., 2018) pre-trained embeddings for EMBED.

2018b), scores each choice c_i in the following way: $\alpha_i^{(j)} = \mathbf{W}^T r_{c_i}^{(j)} \in \mathbb{R}$ for $\mathbf{W}^T \in \mathbb{R}^{2h}$ independently of the question and assigns a probability to each answer $p_i^{(j)} \propto e^{\alpha_i^{(j)}}$.

A slight variant of this model, the **Choice-to-choice** model, tries to single out a given answer choice relative to other choices by scoring all choice pairs $\alpha_{i,i'}^{(j)} = \textsc{Att}(r_{c_i}^{(j)}, r_{c_{i'}}^{(j)}) \in \mathbb{R}$ using a learned attention mechanism Att and finding the choice with the minimal similarity to other options (for full details, see their original paper). In using these partial-input baselines, which we train directly on each target probe, we can check whether systematic biases related to answer choices were introduced into the data creation process.

A **Question-to-choice** model, in contrast, uses the contextual representations for each question and individual choice and an attention model Att model to get a score $\alpha_{q,i}^{(j)} = \textsc{Att}(r_q^{(j)}, r_{c_i}^{(j)}) \in \mathbb{R}$ as above. Here we also experiment with using **ESIM** (Chen et al., 2017) to generate the contextual representations for q, c_i (which includes token-wise attention), as well as a **VecSimilarity** model that measures the average (cosine) vector similarity between question and answer tokens: $\alpha_{q,i}^{(j)} = \textsc{Sim}(\textsc{embed}(q^{(j)}), \textsc{embed}(c_i^{(j)}))$. These sets of baselines, which have been shown to be weak on other benchmark MCQA tasks, are primarily used not as competitive models but to check for artifacts between questions and answers that are not captured in the partial-input baselines. This helps ensure that the overall MCQA probing tasks are sufficiently difficult.

4.2 Inoculation and Pre-training

Using the various models introduced above, we train these models on benchmark tasks in the science domain and look at model performance on our probes with and without additional training on samples of probe data, building on the idea of *inoculation* from Liu et al. (2019a). Model inoculation is the idea of continuing to train models on new challenge tasks (in our cases, separately for each probe) using only a small amount of examples. Unlike in ordinary fine-tuning, the goal is not to learn an entirely re-purposed model, but to improve on (or *vaccinate* against) particular phenomena (e.g., our synthetic probes) that potentially deviate from a model's original training distribution.

Following a variant proposed by Richardson et al. (2020), for each pre-trained (science) model and architecture M_a we continue training the model on k new probe examples (with a maximum of k = 3,000) under a set of hyper-parameter configurations $\{1, \ldots, J\}$ and identify, for each k, the model $M_*^{a,k}$ with the best aggregate performance S on the original (*orig*) and *new* task:

$$M_*^{a,k} = \underset{M \in \{M_1^{a,k}, \ldots, M_J^{a,k}\}}{\arg\max} \textsc{avg}\Big(S_{\text{new}}(M), S_{\text{orig}}(M)\Big)$$

As in Richardson et al. (2020), we performed comprehensive hyperparameter searches that target especially learning rates and # training iterations.

Using this methodology, we can see how much exposure to new data it takes for a given model to master a new task, and whether there are phenomena that stress particular models (e.g., lead to catastrophic forgetting of the original task). Given the restrictions on the number of fine-tuning examples, our assumption is that when models are able to maintain good performance on their original task during inoculation, *the quickness with which they are able to learn the inoculated task provides evidence of prior competence*, which is precisely what we aim to probe. To measure past performance, we define a model's **inoculation cost** as the difference in the performance of this model on its original task before and after inoculation, which serves as a *control* on the target QA model.

We pre-train on an aggregated training set of all benchmark science exams in Table 6.[8]

In line with our goal of obtaining insights into the strongest QA models, we first pre-trained our **RoBERTa**-large model on the RACE dataset (Lai et al., 2017), a recipe used by several leading models on science benchmarks. and created an aggregate development set of ∼4k science questions for evaluating overall science performance and inoculation cost. To handle a varying number of answer choices in these sets, we made all sets 5-way by adding empty answers as needed. We also experimented with a slight variant of inoculation, called **add-some inoculation**, which involves balancing the inoculation training sets

[8]To save space, we do not report scores for each individual science dataset, but we did verify that our best models achieve results comparable to the state of the art for each dataset.

Model	WordNetQA Definitions (Dev/Test)	WordNetQA Synonymy (Dev/Test)	WordNetQA Hypernymy (Dev/Test)	WordNetQA Hyponymy (Dev/Test)	DictionaryQA Word sense (Dev/Test)
Group 1: **Baselines** (direct training on 3k probes)					
Random	19.9 / 20.0	19.8 / 19.8	19.9 / 20.0	20.2 / 21.0	20.0 / 19.0
Choice-Only-GloVe	26.6 / 26.1	36.9 / 36.1	42.5 / 46.0	34.3 / 34.4	35.0 / 32.1
Choice-Only-BERT	22.9 / 23.2	**41.1** / 39.4	**63.8** / 54.4	35.7 / 35.1	36.6 / 31.7
Choice-Only-RoBERTa	**26.8** / **28.6**	40.9 / **40.1**	62.3 / **57.3**	**37.8** / **37.5**	**38.0** / **31.7**
Choice-to-Choice-GloVe	26.4 / 28.1	40.1 / 35.0	47.0 / 35.5	35.4 / 36.1	37.3 / 33.3
Question-to-Choice-VecSimilarity	33.4 / 32.1	31.7 / 30.7	28.9 / 33.0	26.2 / 28.8	29.5 / 33.1
Group 2: **Task-Specific (non-transformer) Models**					
Question-to-Choice-GloVe	**53.6** / **51.8**	57.3 / 55.3	50.4 / 47.0	**61.6** / **64.2**	**53.2** / **53.5**
Question-to-Choice-ELMO	42.3 / 41.6	**58.6** / **56.0**	**56.0** / **51.5**	54.8 / 56.3	51.6 / 52.1
Group 3: **Science Models** (no fine-tuning or direct training on probes)					
ESIM-GloVe	27.5 / 28.3	25.1 / 26.1	27.0 / 33.0	23.6 / 24.8	31.9 / 32.5
ESIM-ELMO	23.1 / 24.0	21.1 / 21.5	27.1 / 32.7	18.0 / 18.5	28.3 / 31.5
BERT	54.1 / 55.7	58.8 / 60.9	43.2 / 51.0	24.0 / 27.0	43.0 / 42.9
RoBERTa	**74.1** / **77.1**	**61.1** / **64.2**	**53.2** / **71.0**	**48.5** / **58.6**	**53.0** / **55.1**
Group 4: **Science Models** (best aggregate model M_* fine-tuned on probes; **inoculation cost** is shown in parenthesis)					
ESIM-GloVe	46.2 / 42.4 (−6.27)	50.4 / 47.3 (−6.84)	56.6 / 52.9 (−5.69)	59.1 / 61.1 (−5.10)	50.0 / 55.3 (−7.09)
BERT	84.0 / 84.1 (−1.15)	79.6 / 79.7 (−0.44)	73.8 / 82.7 (−0.49)	79.8 / 88.0 (−0.92)	75.6 / 79.1 (−2.84)
RoBERTa	**89.0** / **89.3** (−1.33)	**81.2** / **81.3** (−1.31)	**77.7** / **87.7** (−0.74)	**81.2** / **89.4** (−1.64)	**80.0** / **85.9** (−2.23)
Human Performance (estimates)	– / 91.2%	– / 87.4%	– / 96%†	– / 95.5%†	– / 95.6%†

Table 7: **Instance-level** accuracy (%) of all baselines (group 1), task-specific non-transformer QA models (group 2), pre-trained MCQA models (zero-shot, group 3), and MCQA models after fine-tuning on our probes (group 4). Human scores marked with † represent scores on gold-standard annotated test sets.

with naturalistic science questions. We reserve the MCQL dataset in Table 6 for this purpose, and experiment with balancing each probe example with one science example (*x1 matching*) and adding twice as many science questions (*x2 matching*, up to 3k) for each new example.

4.3 Evaluating Model Competence

We use **instance-level accuracy**, the standard overall accuracy of correct answer prediction (as in Table 7). In addition, we also propose to measure a model's **cluster-level** (or *strict cluster*) **accuracy**, which requires correctly answering all questions in a *semantic cluster* (cf. Section 3.1.1).

Our cluster-based analysis is motivated by the idea that if a model truly knows the meaning of a given concept then it should be able to answer arbitrary questions about this concept without sensitivity to varied distractors. Although our strict cluster metric is simplistic, it takes inspiration from work on visual QA (Shah et al., 2019), and allows us to evaluate a model's *consistency* and *robustness* across our different probes, and to get insight into whether errors are concentrated on a small set of concepts or widespread across different clusters.

The ability of a model to answer several questions about a single concept can be thought of as a type of *certificate* (i.e., further justification and demonstration) of general understanding of that concept in the sense of Ranta (2017).

5 Results and Findings

We begin with an assessment to ensure that our probes are sufficiently difficult to provide meaningful insights into strong models (Section 5.1), then assess the strength of pre-trained QA models (Section 5.2) and whether they can be effectively inoculated (Section 5.3), and finally present a cluster-based consistency analysis (Section 5.4).

5.1 Are Our Probes Sufficiently Challenging?

Partial-input baseline models, **Choice-Only** and **Choice-to-Choice**, generally performed poorly on our probes (cf. Table 7, group 1), indicating limited biases in distractor generation. Initial versions of DictionaryQA had unforeseen biases partly related to distractors sampled from entries without example sentences (cf. Section 3.2), which resulted in high (56%) Choice-Only-GloVe scores before such distractors were filtered out.

One exception is our hypernymy probe where, despite several attempts at filtering data and de-duplicating splits (with respect to correct answer and distractor types), the Choice-to-Choice-BERT/RoBERTa models achieve over 60% accuracy. The nature of the biases here remains unclear, highlighting the importance of having rigorous baselines as unintended biases in expert knowledge can carry over to resulting datasets. We also note the large gap between the BERT/RoBERTa versus GloVe choice-only models, emphasizing the need for using the best available models even in partial-input baselines.

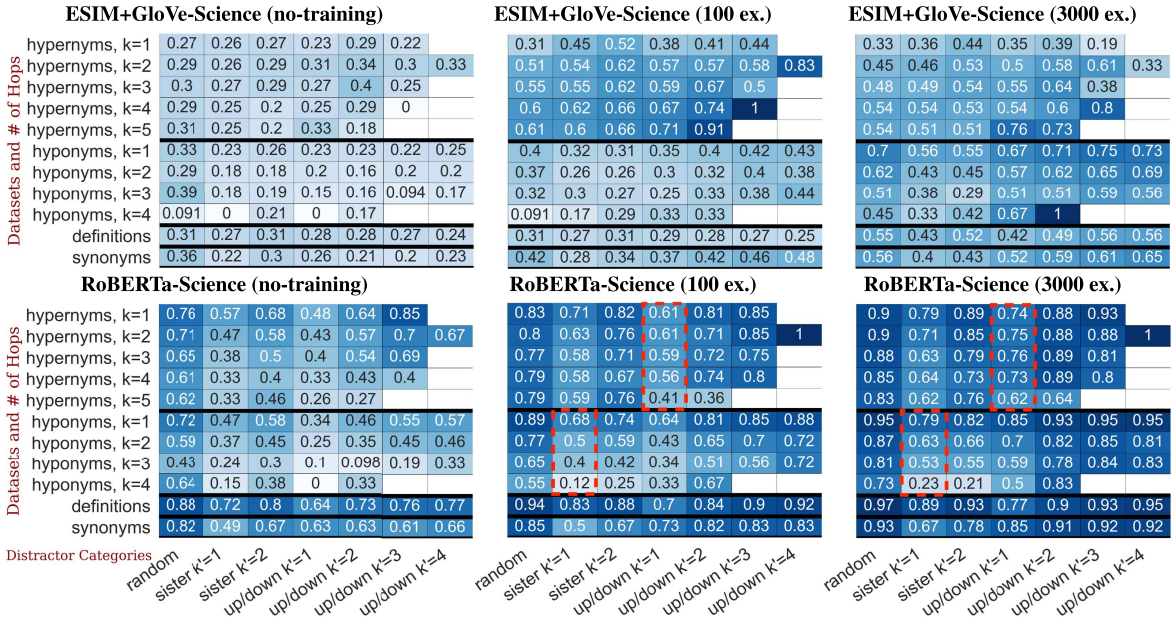


Figure 3: Combined model accuracies on the different WordNetQA datasets (divided by 4 bold lines) broken down (where possible) into number of hops k (rows) and types of distractor sets and hops k' (rows) across the different stages of inoculation (**# ex.**). The 4 dashed lines show some trends related to multi-hop inference.

A more conventional set of *Task-Specific* QA models (i.e., the LSTM-based **Question-to-Choice** models trained directly on the probes) is not particularly strong on any of the datasets (cf. Table 7, group 2), suggesting that our probes are indeed sufficiently challenging and largely immune from overt artifacts. The poor performance of the *VecSimilarity* (which uses pre-trained Word2Vec embeddings without additional training) provides additional evidence of the insufficiency of elementary lexical matching strategies.

5.2 How Strong Are Pre-trained QA Models?

Non-transformer science models, such as **ESIM** with GloVe or ELMo, struggle with all probes (cf. Table 7, group 3), often scoring near random chance. In sharp contrast, the transformer models have mixed results, the most striking being **RoBERTa** QA models on the definitions, synonymy and hypernymy test probes (achieving 77%, 64%, and 71% respectively), which substantially outperform even task-specific LSTM models trained directly on the probes. Throughout all of these results, however, model performance is significantly behind human performance.

At first glance, these zero-shot results suggest RoBERTa's high competence on these phenomena. A closer scrutiny enabled by our controlled probes, however, provides a more subtle picture. Each heat map in Figure 3 breaks down the performance of an ESIM or RoBERTa QA model based on the difficulty of the probe dataset (rows) and the nature of the distractors (columns).

Across all datasets and number of hops in the question (i.e., all rows), zero-shot model performance for RoBERTa (bottom-left heat map) is consistently highest among examples with random distractors (the first column) and lowest when distractors are closest in WordNet space (e.g., sister and ISA, or *up/down*, distractors at distance $k' = 1$). For example, RoBERTa's zero-shot score drops from 88% to 64% when going from random distractors to *up/down* distractors at $k' = 1$.

Further, model performance also clearly degrades for hypernymy and hyponymy as k, the number of hops in the question, increases (see red dashed boxes). For example, the accuracy on questions involving hyponym reasoning with sister distractors of $k' = 1$ (column 2) degrades from 47% to only 15% as k increases from 1 to 4. This general tendency persists despite additional fine-tuning, providing evidence of the limited ability of these models to perform multi-hop inference.

5.3 Can Models Be Effectively Inoculated?

How well probe generation templates align with the science training distribution (which we know little about) can significantly impact zero-shot performance (Petroni et al., 2019). Zero-shot results above thus provide a *lower bound* on model competence on the probed phenomena. We next consider a probe-specific fine-tuning or *inoculation* step, allowing models to learn target templates and couple this with knowledge acquired during pre-training and science training.

Accuracy after inoculation on 3K probe instances is shown (with inoculation cost in parenthesis) in group 4 of Table 7, for the model with the highest aggregate score on the original task and new probe. Transformer-based models again outperform non-transformer ones, and *better models correlate with lower inoculation costs*. For example, on synonymy, ESIM's inoculation cost is 7%, but only ∼1% for BERT and RoBERTa. This emphasizes the high capacity of transformer QA models to absorb new phenomena at minimal cost, as observed earlier for NLI (Richardson et al., 2020).

Figure 4 shows the corresponding learning curves. Transformer QA models learn most tasks quickly while maintaining constant scores on their original tasks (flat dashed lines, plots 1–4), providing evidence of high competence. For BERT and RoBERTa, **add-some inoculation** (a) improves scores on the probing tasks (*solid* black and blue lines, plot 1) and (b) minimizes loss on the original task (*dashed* blue and black lines, plots 2–4).

ESIM behaves quite the opposite (plots 5–6), generally unable to learn individual probes without degrading on its original task. More science data during inoculation confuses it on both tasks.

As the middle-bottom plot of Figure 3 shows, RoBERTa's performance improves significantly (e.g., from 59% to 77% on 2-hop hyponymy with random distractors) even after inoculation with a mere 100 examples, providing strong evidence of prior competence. After 3k examples, it performs well on virtually all probes. However, results still notably degrade with the number of hops and distractor complexity, as discussed earlier, and we still find its performance to be between 2% and 10% behind human performance.

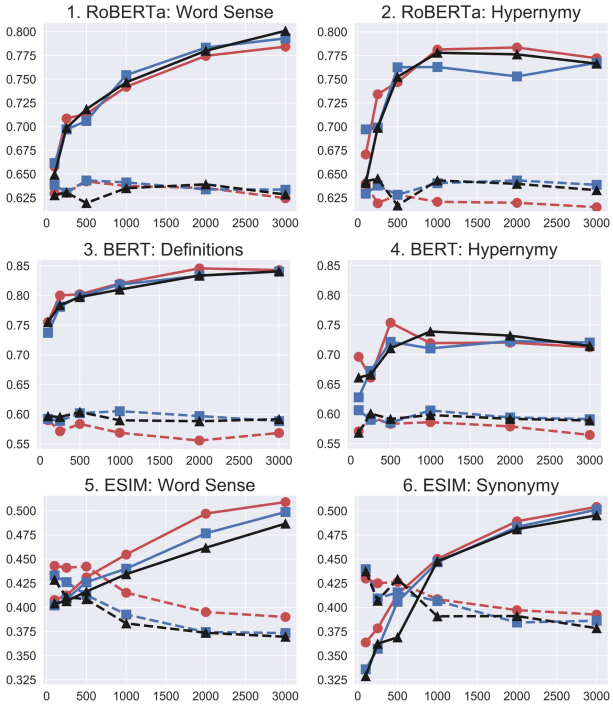


Figure 4: Inoculation plots with accuracy on challenge tasks (red/circle solid lines) and original tasks (red/circle dashed lines) using the best aggregate model $M_*^{a,k}$ at each k challenge examples (x axis). The effect of using **add-some inoculation** is shown in the blue/square (*x1 match*) and black/triangle (*x2 match*) lines.

Model	Definitions	Synonymy	Hypernymy	Hyponymy
	Strict Cluster Accuracy (Δ)			
Choice-Only	14.7 (−12.0)	18.5 (−22.3)	34.6 (−27.6)	4.1 (−33.7)
ESIM	30.2 (−15.9)	23.3 (−26.9)	29.2 (−27.3)	15.2 (−43.8)
BERT	68.5 (−15.5)	58.1 (−21.5)	49.0 (−24.8)	34.0 (−45.4)
RoBERTa	75.0 (−13.9)	61.7 (−19.4)	54.0 (−23.2)	36.7 (−44.4)

Table 8: **Cluster-level** accuracies (%) on the WordNetQA dev. sets for inoculated models and best **Choice-only** model. Δ show the absolute difference in percentage points with instance-level accuracies.

5.4 Are Models Consistent Across Clusters?

Table 8 shows mixed results for **cluster-level** accuracy across the different WordNetQA probes. Our best model is rather robust on the definitions probe. RoBERTa QA's cluster accuracy is 75%, meaning it can answer *all* questions correctly for 75% of the target concepts, and that errors are concentrated on a small minority (25%) of concepts. On synonymy and hypernymy, both BERT and RoBERTa are less strong but appear robust on a majority of concepts. In contrast, our best model on hyponymy has an accuracy of only 36%, indicating that the RoBERTa QA models knows only partially about a vast majority

of concepts, leaving substantial room for further improvement.

We emphasize that these results only provide a crude look into model consistency and robustness. Recalling dataset details in Table 4, probes differ in terms of the average size of clusters. For example, hyponymy, in virtue of having many more questions per cluster, might simply be a much more difficult dataset for our cluster-based evaluation. In addition, such a strict evaluation does not take into account potentially erroneous questions within clusters, which is an important issue that we leave for future work.

6 Discussion

We presented a new methodology for automatically building challenge datasets from knowledge graphs and taxonomies. We introduced several new silver-standard datasets for systematically probing state-of-the-art open-domain QA models. Although our focus was on probing definitions and ISA reasoning, the methodology is amendable to any target knowledge resource or QA domain. We see synthetic datasets and our general methodology as an inexpensive supplement to recent large-scale investment in *naturalistic* QA dataset construction (Zellers et al., 2018; Sakaguchi et al., 2020) to help better understand today's models.

We found transformer-based QA models to have a remarkable ability to reason with complex forms of relational knowledge, both *with* and *without* exposure to our new tasks. In the latter case (zero-shot), a newer RoBERTa QA model trained only on benchmark data outperforms several *task-specific* LSTM-based models trained directly on our probes. When *inoculated* using small samples (e.g., 100 examples) of probing data, RoBERTa masters many aspects of our probes with virtually no performance loss on its original QA task—which we use as a control on the probing quality.

Because these models seem to already contain considerable amounts of relational knowledge, our simple inoculation strategy, which nudges models to bring out this knowledge explicitly while retaining performance on their original task (hence allowing a fairer probe of its knowledge by giving the model the opportunity to learn the probe format), could serve as a simpler alternative to designing new model architectures explicitly encoding such knowledge (Peters et al., 2019).

Regarding our focus on preserving a model's performance on its original task, one might expect that re-training on relevant knowledge should *improve* performance. Following other work in this area (Richardson et al., 2020; Yanaka et al., 2020), we found that maintaining performance after additional fine-tuning on specialized datasets is already a tall order given that models are susceptible to over-specialization; indeed, similar issues have been noticed in recent work on large-scale transfer learning (Raffel et al., 2019). We believe that using inoculation for the sole purpose of improving model performance, which is beyond the scope of this paper, would likely require a more sophisticated inoculation protocol. Devising more complex loss functions extending our inoculation strategy to help balance old and new information could help in this endeavor.

The main appeal of automatically generated probes is the ability to systematically manipulate probe complexity, which in turn enables more controlled experimentation as well as new forms of evaluation. It allowed us to study in detail the effect of different types of distractors and the complexity of required reasoning. This study showed that even the best QA models, despite additional fine-tuning, struggle with harder categories of distractors and with multi-hop inferences. For some probes, our cluster-based analysis revealed that errors are widespread across concept clusters, suggesting that models are not always consistent and robust. These results, taken together with our findings about the vulnerability of synthetic datasets to systematic biases and comparison with human scores, suggest that there is much room for improvement and that the positive results should be taken with a grain of salt. Developing better ways to evaluate semantic clusters and model robustness would be a step in this direction.

We emphasize that using synthetic versus naturalistic QA data comes with important trade-offs. Although we are able to generate large amounts of systematically controlled data at virtually no cost or need for manual annotation, it is much harder to validate the quality of such data at such a scale and such varying levels of complexity. Conversely, with benchmark QA datasets, it is much harder to perform the type of careful manipulations and cluster-based analyses we report here. While we assume that the expert knowledge we use by virtue of being hand-curated by human experts, is generally correct by design,

we know that such resources are fallible and error-prone. We propose measuring human performance via small samples of probing data, and leave more scalable methods of removing potential noise and adding human annotation to future work.

One of the overarching goals of our approach to model probing is to uncover whether black box models are able to reason in a consistent and correct manner. Our assumption, similar to Clark et al. (2020), is that the ability of a model to mimic the input-output behavior of data generated using expert knowledge gives some evidence of correctness in virtue of such data being *correct by construction* (see discussion by Ranta (2017)). We emphasize, however, that there are limits to how much we can learn through this type of behavioral testing, given that models are susceptible to exploiting systematic biases in synthetic data and the general difficulty of disentangling a model's knowledge acquired during pre-training versus fine-tuning (Talmor et al., 2019a). We therefore see efforts to combine behavioral testing with various other *analysis methods* (Belinkov and Glass, 2019) that aim to uncover correlations and causal patterns between internal model representations and discrete structures (Chrupała and Alishahi, 2019; Vig et al., 2020; Geiger et al., 2020) as a promising direction for future work. This, in combination with extending our probing strategy to other forms of expert knowledge, could prove to be an effective way to engage others working on linguistics and other areas of AI in state-of-the-art NLP research.

Acknowledgments

We thank the Action Editor and the three anonymous reviewers for their thoughtful comments and feedback. Thanks also to our colleagues at AI2, in particular Peter Clark, Daniel Khashabi, Tushar Khot, Oyvind Tafjord, and Alon Talmor, for feedback on earlier drafts of this work and assistance with various aspects of modeling. Special thanks to Daniel Khashabi for helping with some of the earlier human evaluation experiments.

References

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. 2020. Abductive commonsense reasoning. *Proceedings of ICLR*.

Michael Boratko, Harshit Padigela, Divyendra Mikkilineni, Pritish Yuvraj, Rajarshi Das, Andrew McCallum, Maria Chang, Achille Fokoue-Nkoutche, Pavan Kapanipathi, Nicholas Mattei, et al. 2018. A systematic classification of knowledge, reasoning, and context within the ARC dataset. In *Proceedings of Machine Reading for Question Answering (MRQA) Workshop at ACL*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of ACL*.

Grzegorz Chrupała and Afra Alishahi. 2019. Correlating neural and symbolic representations of language. In *Proceedings of ACL*.

Peter Clark. 2015. Elementary school science and math tests as a driver for AI: Take the aristo challenge! In *Twenty-Seventh IAAI Conference*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv: 1803.05457*.

Peter Clark, Oren Etzioni, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, Sumithra Bhakthavatsalam, Dirk Groeneveld, Michal Guerquin, and Michael Schmitz. 2019. From 'F' to 'A' on the NY regents science exams: An overview of the aristo project. *arXiv preprint arXiv:1909. 01958*.

Peter Clark, Philip Harrison, and Niranjan Balasubramanian. 2013. A study of the knowledge base requirements for passing an elementary science test. In *Proceedings of AKBC*.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. *Proceedings of IJCAI*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *Proceedings of EMNLP*.

Ernest Davis. 2016. How to write science questions that are easy for people and hard for computers. *AI Magazine*, 37(1):13–22.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep didirectional transformers for language understanding. In *Proceedings of NAACL*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. *arXiv preprint arXiv:1803.07640*.

Atticus Geiger, Ignacio Cases, Lauri Karttunen, and Chris Potts. 2019. Posing fair generalization tasks for natural language inference. In *Proceedings of EMNLP*.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Modular representation underlies systematic generalization in neural natural language inference models. *arXiv preprint arXiv: 2004.14623*.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of ACL*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of NAACL*.

Catherine Havasi, Robert Speer, and Jason Alonso. 2007. ConceptNet 3: A flexible, multilingual semantic network for common sense knowledge. In *Proceedings of Recent Advances in NLP*.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of EMNLP*.

Felix Hill, Kyunghyun Cho, Anna Korhonen, and Yoshua Bengio. 2016. Learning to understand phrases by embedding the dictionary. *TACL*, 4:17–30.

Sujay Kumar Jauhar, Peter Turney, and Eduard Hovy. 2016. Tables as semi-structured knowledge for question answering. In *Proceedings of ACL*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of EMNLP*.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2019. How can we know what language models know? *arXiv preprint arXiv: 1911.12543*.

Dongyeop Kang, Tushar Khot, Ashish Sabharwal, and Eduard H. Hovy. 2018. AdvEntuRe: Adversarial training for textual entailment with knowledge-guided examples. In *Proceedings of ACL*.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Are you smarter than a sixth grader? Textbook question answering for multimodal machine comprehension. In *Proceedings of CVPR*.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. QASC: A dataset for question answering via sentence composition. In *Proceedings of AAAI*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale reading comprehension dataset from examinations. In *Proceedings of EMNLP*.

Chen Liang, Xiao Yang, Neisarg Dave, Drew Wham, Bart Pursel, and C. Lee Giles. 2018. Distractor generation for multiple choice questions using learning to rank. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*.

Nelson F. Liu, Roy Schwartz, and Noah A. Smith. 2019a. Inoculation by Fine-Tuning: A Method for Analyzing Challenge Datasets. *Proceedings of NAACL*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A robustly optimized bert retraining approach. *arXiv preprint arXiv:1907.11692*.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *Proceedings of ACL*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *Proceedings of EMNLP*.

George A. Miller. 1995. Wordnet: a Lexical Database for English. *Communications of the ACM*, 38(11):39–41.

Nikita Nangia and Samuel R. Bowman. 2019. Human vs. muppet: A conservative estimate of human performance on the glue benchmark. *arXiv preprint arXiv:1905.10425*.

Roberto Navigli and Simone Paolo Ponzetto. 2010. Babelnet: Building a very large multilingual semantic network. In *Proceedings of ACL*.

Ian Niles and Adam Pease. 2001. Towards a standard upper ontology. In *Proceedings of the International Conference on Formal Ontology in Information Systems-Volume*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL*.

Matthew E. Peters, Mark Neumann, I. V. Logan, L. Robert, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of EMNLP*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? In *Proceedings of EMNLP*.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: The word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of NAACL*.

Adam Poliak, Aparajita Haldar, Rachel Rudinger, J. Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. 2018a. Collecting diverse natural language inference problems for sentence representation evaluation. In *Proceedings of EMNLP*.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018b. Hypothesis only baselines in natural language inference. In *Proceedings of *SEM*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Aarne Ranta. 2017. Explainable machine translation with interlingual trees as certificates. In *Proceedings of the Conference on Logic and Machine Learning in Natural Language*.

Sathish Reddy, Dinesh Raghu, Mitesh M. Khapra, and Sachindra Joshi. 2017. Generating natural language question-answer pairs from a knowledge graph using a rnn based question generation model. In *Proceedings of EACL*.

Kyle Richardson, Hai Hu, Lawrence S. Moss, and Ashish Sabharwal. 2020. Probing natural language inference models through semantic fragments. In *Proceedings of AAAI*.

Ohad Rozen, Vered Shwartz, Roee Aharoni, and Ido Dagan. 2019. Diversify your datasets: Analyzing generalization via controlled variance in adversarial datasets. In *Proceedings of CoNLL*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WINOGRANDE: An adversarial winograd schema challenge at scale. *Proceedings of AAAI*.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. SocialIQA: Commonsense reasoning about social interactions. In *Proceedings of EMNLP*.

Timo Schick and Hinrich Schütze. 2020. Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking. *Proceedings of AAAI*.

Dominic Seyler, Mohamed Yahya, and Klaus Berberich. 2017. Knowledge questions from knowledge graphs. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*.

Meet Shah, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. 2019. Cycle-consistency for robust visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Chenglei Si, Shuohang Wang, Min-Yen Kan, and Jing Jiang. 2019. What does BERT learn from multiple-choice reading comprehension datasets? *arXiv preprint arXiv:1910.12391*.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2019a. oLMpics – On what language model pre-training captures. *ArXiv:*, arXiv preprint arXiv:1912.13283.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019b. Commonsense QA: A question answering challenge targeting commonsense knowledge. In *Proceedings of NAACL*.

Niket Tandon, Gerard De Melo, and Gerhard Weikum. 2017. Webchild 2.0: Fine-grained commonsense knowledge distillation. In *Proceedings of ACL*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NeurIPS*, pages 5998–6008.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. 2020. Causal mediation analysis for interpreting neural NLP: The case of gender bias. *arXiv preprint arXiv: 2004.12265*.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, Sheng-Fu Wang, Jason Phang, Anhad Mohananey, Phu Mon Htut, Paloma Jeretič, and Samuel R. Bowman. 2019. Investigating BERT's knowledge of language: Five analysis methods with NPIs. In *Proceedings of EMNLP*.

Noah Webster. 1913. *Webster's revised unabridged dictionary of the english language*, G. & C. Merriam Company.

Johannes Welbl, Nelson F Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, and Kentaro Inui. 2020. Do neural models learn systematicity of monotonicity inference in natural language? In *Proceedings of ACL*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of EMNLP*.