The Winograd Schemas from Hell

Fabio Gagliardi Cozman[1], **Hugo Neri Munhoz**[1]

[1]Universidade de São Paulo - Brazil

Abstract. *The Winograd Challenge has been advocated as a test of computer understanding with respect to commonsense reasoning. The challenge is based on Winograd Schemas: sentences that contain correferential ambiguities. Most Winograd Schemas are relatively easy for human subjects, and today the best computer systems for the Winograd Challenge can work close to human performance. In this paper, we examine the assumptions behind the Winograd Challenge, and investigate how far we can push the difficulty level of Winograd Schemas, proposing various strategies to build really challenging schemas.*

1. Introduction

As computers master a variety of human activities, from text summarization to photo captioning, many challenging datasets have been concocted to test specific abilities. For instance, the SQuAD dataset challenges one to find a short text fragment, in the middle of a large number of such fragments, that answers a question [Rajpurkar et al. 2016]. Other datasets emphasize the ability to classify images, sound, video, and so on. Computers now often display superhuman performance in several of such narrow settings.

The ultimate test of computer ability still is, at least in popularity, the Turing Test [Turing 1950]: a computer must fool a human judge so that the latter takes the former to be human. There are many difficulties with the Turing Test [Epstein et al. 2009]; above all, it seems its most problematic aspect is that it induces too much effort to be spent in deception: one is worried about creating a device that can trick a human judge, not a device that can be called intelligent in some intelligently original way.

Possibly the most satisfying alternative to the Turing Test that has been proposed during the last decade is the Winograd Challenge [Levesque et al. 2012]. The idea is to ask a computer system to solve a set of Winograd Schemas, and to verify whether the accuracy of the computer rivals the accuracy of human subjects. A Winograd Schema consists of a pair of pronominal disambiguation problems, so that the referent of the pronoun depends on a special word. Of course, many disambiguation problems are very easy; in particular a variety of disambiguation problems can be solved without any serious semantic analysis, just by "clever tricks involving group order or other features of words or groups of words" [Levesque et al. 2012]. An assumption behind the Winograd Challenge is that disambiguation problems that can be solved by "clever tricks" are not powerful enough to determine whether a machine has any degree of intelligence.

The Winograd Challenge instead focuses on pairs such as

> The trophy does not fit in the brown suitcase because *it* is too BIG/SMALL.

What is "it"? If the special word BIG is used, then it is the trophy; if SMALL is used, then it is the suitcase. Presumably, to correctly solve such a Winograd Schema both ways, one must combine some commonsense reasoning with accepted common knowledge.

A collection of relatively short Winograd Schemas has been assembled through the years in a dataset referred to as WSC273 [Davis 2018]. There are 150 pairs in this dataset, a few of which do not strictly qualify as Winograd Schemas because their ambiguity does not lie on a pronoun (they are said to be Winograd Schemas *in the broad sense*). The WSC273 dataset is the gold standard as far as datasets are concerned; however, it is a very small dataset. For years, the performance of computer systems in the Winograd Challenge was mediocre at best; this started to change by the end of 2018. Recently, a much larger set of Winograd Schemas, referred to as the WinoGrande set, has been created and used as the basis of the current Winograd Challenge non-human champion, a specialized version of the UnifiedQA solver [Khashabi et al. 2020]. This solver attains more than 90% accuracy in the Winograd Challenge, a truly impressive figure that is similar to human accuracy [Sakaguchi et al. 2019]. An important feature of the current champion solvers is that they are based on language models learned from large textual datasets; that is, they estimate the probability for each possible solution of a Winograd Schema, and output the most likely solutions.

The victory of pattern-extracting solvers has obliterated the claim that WSC273 can be used to test whether a machine is intelligent or not, or at least whether a machine displays some commonsense [Kocijan et al. 2020]. There are a few possibilities. One is to look for entirely different tests [Marcus et al. 2016]. Another is to change the Winograd Challenge, say by requiring the machine not only to disambiguate a pronoun, but also to explain the solution — we have examined this *Explaining Winograd Challenge* elsewhere [Cozman and Munhoz 2020]. And yet another path is to look for more difficult Schemas: perhaps WSC273 is still too easy in that statistical regularities in language can crack most of its Schemas, without the need for any deeper understanding. A comment related to the Turing Test by Floridi, Taddeo and Turilli [Floridi et al. 2009] applies here as well: "If you need to test, and we mean really *test*, an artefact, the higher the stakes are, the tougher the procedure should be."

Now, what exactly constitutes a tough Winograd Schema? This paper discusses this question. We examine several strategies that can be used to create hard Winograd Schemas. Some of these strategies succeed in interesting ways, in some cases producing truly diabolical Winograd Schemas.

We make two contributions in this paper.

First, in Section 2 we offer a novel analysis of Winograd Schemas by focusing on the fact that they are required to be easy for humans. We discuss the meaning of this requirement and its consequences, and we then investigate the limiting consequences of letting Winograd Schemas to be arbitrarily hard. By examining exactly the space of possible Winograd Schemas, we stumble upon their limitations and their differences both with human reasoning and with the Turing Test.

Second, in Section 3 we show how to harness a variety of strategies so as to produce tough Winograd Schemas, presenting a few paradigmatic examples that can guide us in producing effective variants of the Winograd Challenge. These examples can be solved by human subjects with significant but realistic effort, while no machine can now crack. We expect that useful datasets can be created employing the strategies outlined here.

2. Winograd Schemas: the easy, the impossible, and the tough

As described in the original paper about the Winograd Challenge [Levesque et al. 2012]: "A Winograd schema is a pair of sentences differing in only one or two words and containing an ambiguity that is resolved in opposite ways in the two sentences and that requires the use of world knowledge and reasoning for its resolution." In the present paper the ambiguity always centers on the referent of a pronoun, even though ambiguity in the referent of possessive adjectives is allowed in the original proposal. Some of the existing Winograd Schemas describe a pair of sets of sentences that differ in a few words; we accept such schemas with the proviso that they cannot be too long — at most say two hundred words.

The original definition of Winograd Schemas added a few guidelines to the broad definition in the previous paragraph. First, a Winograd Schema should not be solved by obvious statistical tests over corpora. Second, it should not be solved by simple techniques such as selectional restrictions. Third, it should be easily disambiguated by the human reader ("so easy that the reader does not even notice that there is an ambiguity").

The first and second guidelines are not really important. They just discard those cases that are too easy both for humans and computers and that consequently have no discriminatory power, but they do not add anything else of interest.

The third guideline carries enormous weight. Why do we need Winograd Schemas to be *easy* for humans? This guideline reveals several assumptions about commonsense that are hidden behind the Winograd Challenge.

2.1. Easy for humans (and thus for computers as well...)

Levesque and the other defenders of the original Winograd Schema do not define precisely what they mean with the "easiness guideline." One can also find the mysterious statement that the schemas could be "progressively difficult" to both humans and computers [Levesque et al. 2012], but no discussion of what makes a schema difficult.

Possibly, they mean that a human reader should easily understand the meaning of a Winograd Schema as it was intended by the schema's creator. If we consider the process of understanding as the (mental) process of decomposing a received input, setting the appropriate relationships amongst its elements, and adding to whatever is absent or implicit in the input some previous acquired knowledge, then "reader easily understands" means: *reader can quickly provide superficial knowledge that is expected from anyone competent enough in a given language and/or someone who was socialized enough in a given cultural/social environment, but that is not explicitly shared in the text*. Within such perspective, commonsense knowledge and commonsense reasoning offer a compression mechanism in which parts of an interaction are suppressed because one party expects the other party to automatically fill in the missing pieces. To make matters a bit more concrete, consider the following Winograd Schema:[1]

> The city councilmen refused the demonstrators a permit because *they* FEARED/ADVOCATED violence.

[1] We adopt a few conventions in this paper: the pair of words that differentiate the two instances of the schema appear in small caps, while the pronoun to be disambiguated is italicized.

The creator of this schema presumably holds the belief that demonstrators (generally) defend the instrumental use of violence in their manifestations and that authorities (usually) prevent or contain violence.

The main difficulty here is the assumption of a generalized belief instead of a localized belief. The former leads to the underlying assumption that there is a storage of commonsense that "most people" would know a significant portion of it. But this is misleading. Indeed, take the linguistic phenomenon of *intralingual false friends*. The expression *false friends* indicates lexical items in two languages that are similar in form but different in meaning [Hill 1982, Shlesinger and Malkiel 2005]. We are used to interlingual false friends, and justify the different meanings by resorting to the languages history or structure. However, the fact that there exist lexical false friends within the same language reveals that we should take a step back before assuming any sort of universal (commonsense) knowledge available to everyone. In human normal communication, the (mis)use of intralingual false friends may lead to funny situations, serious blunders, or just neutral sentences that could have been more appropriate with different referents. The requirement that Winograd Schemas should be easy in the sense of "easily understandable" seems clumsy as it ignores the communicating parties. To be easy for all humans, one must stay within a minimum common denominator — and thus be easy for all agents whatsoever.

Moreover, the easiness guideline forces one to limit Winograd Schemas to a small number of sentences referring to a small number of entities. The Winograd Schemas in WSC273 usually mention two events linked by a word that expresses causality. There is hardly an alternative way to knit together two sentences or two phrases meaningfully, if not by describing how some event A leads or affects some event B. It is exactly because of such a regular construction that the resulting Winograd Schemas in WSC273 are easy for humans; most of the schemas in WSC273 end up testing merely whether a human can understand how the elements in a sentence are affected by causal markers in the language.

To be short, the consequence of demanding Winograd Schemas to be easy for humans is that the schemas become easy for computers as well. If a sentence (or a small set of sentences) is to have an ambiguity that any human reader can promptly solve, then this sentence cannot be too long, nor too convoluted. So the sentence must rely on obvious causal relationships between entities, well known facts, or respected social norms.

Indeed, we suspect that if one tries to follow the original guidelines concerning the Winograd Challenge as strictly as possible, then one will be left with Winograd Schemas that resemble the ones in WSC273. We now know that such guidelines limit too much the scope of Winograd Schemas: as demonstrated by recent results on computer solvers, Winograd Schemas that are (relatively) easy for human subjects are (relatively) easy for computers as well [Sakaguchi et al. 2019]. In hindsight this is perhaps unsurprising because language must reflect established facts and rules and social conventions that must appear in large textual corpora. Thus a large enough language model is bound to capture most of the content exploited by Winograd Schemas that are easy for humans.

2.2. Too hard for humans (and, sometimes, impossible for any agent...)

Our proposal is, then, to remove the requirement that a Winograd Schema must be so easy that a human reader does not even notice that there is an ambiguity. Once we remove

this requirement, a vast number of possibilities can be contemplated, and we must in fact place some limits on what can be taken as a Winograd Schema.

Many Winograd Schemas can be solved by looking up facts about given entities. For instance, consider the following schema in WSC273:

> This book introduced Shakespeare to OVID/GOETHE; it was a major influence on *his* writing.

So, who is "his" referring to? Clearly we can only solve the schema if we know facts about each one of the three writers in the schema. And this schema can be made much more difficult if we choose less known writers:

> This book introduced Shakespeare to John GOWER/DRYDEN; it was a major influence on *his* writing.

A factual Winograd Schema that depends on obscure information may be much easier for a computer than for a human: in fact, a quick Wikipedia search can reveal the lifespan of all characters in the latter schema, something a human subject cannot do during a conversation.

By stressing the dependence on esoteric background knowledge, one can create the ultimate Winograd Schema from hell:

> Carol decided to give the apple to Ann if the thousandth bit of Chaitin's Ω_U with respect to the universal Chaitin self-delimiting Turing machine U was ZERO/ONE, and otherwise give the apple to Bella; after checking who would get the apple, Carol gave it to *her*.

No one can solve this Winograd Schema today; Ω_U is an uncomputable and random number that carries, in its first few thousand digits, "answers to more mathematical questions that could be written down in the entire universe" [Bennett and Gardner 1979]. A solution to the Winograd Schema is possible today only if the schema is restricted to a few dozen bits of Ω_U [Calude et al. 2002].

Thus we must limit what counts as common knowledge in a Winograd Schema.

We require that all relevant information about the entities in a Winograd Schema must be assumed known by the intended solvers of the schema. While the schema on Shakespeare may be fair in a conversation between scholars on English literature, it does not qualify as a hard Winograd Schema. And the schema on Chaitin's Ω_U does not qualify regardless of the participant agents.

Alas, if we are to limit common knowledge to facts that are known by all contemporary human beings, then very little will be left in our knowledge base. If we are to really test artificial intelligences, then we must at least assume those facts that are mastered by say an advanced and able undergraduate student — in fact we should take the union, not the intersection, of basic facts mastered by students engaged in the majority of existing professions.

Returning to our quest for hard Winograd Schemas, one strategy that is still possible is to resort to short mathematical questions that belong to the vocabulary of the average undergraduate student. Here is an example, where we adapt our previous schema on Chaitin's number so as to deal with well-known quantities:

> Carol decided to give the apple to Ann if the thousandth decimal of π was ODD/EVEN, and otherwise give the apple to Bella; after checking who would get the apple, Carol gave it to *her*.

Alternatively, Carol might give an apple to Ann if and only if the Riemann hypothesis is true — not everyone knows about the Riemann hypothesis, but it is a piece of mathematical folklore that is somewhat popular (possibly more than Ovid's poems).

However, it does not seem profitable to investigate Winograd Schemas whose solution is impossible for humans, and even less to look at Winograd Schemas whose solution is impossible both for humans and for computers. We must limit the computation that is needed to solve a given schema.

Thus we adopt the following: an *extended* Winograd Schema must follow the definition in the first paragraph of Section 2, but it must require reasoning that is based only on common knowledge for the assumed participants, and it must require at most reasoning that can be written down in a page of notes.

2.3. Just tough enough

In the remainder of this paper we do not repeat the word *extended*; all Winograd Schemas from now on are assumed to be extended ones. As shown later, even with these restrictions one can produce rather tough Winograd Schemas that require substantial reasoning and that would not be immediately solved in the course of a human conversation — schemas that are beyond the best current Winograd Schema solvers.

We have reached what seems to be a working recipe towards building tough Winograd Schemas, but it is worth noting just how far we are from the Turing Test. The latter employs a human judge, so it does not need to avoid questions about the thousandth decimal of π, as it is concerned about the form of the answers, not their accuracy. As an unfortunate consequence, the Turing Test gets bogged down with attempts to fool the judge by resorting to formulaic chatting. The Winograd Challenge moves away from human intervention in a search for objectivity, but tough Winograd Schemas require reasoning that is very removed from commonsense reasoning.

3. Building tough Winograd Schemas

The previous schema on the thousandth digit of π employed a strategy that is worth revisiting, as we can turn many different questions into schemas of the form:

> Carol decided to give the apple to Ann if the question of interest has answer ANSWER 1/ANSWER2, and otherwise give the apple to Bella; after checking who would get the apple, Carol gave it to her.

Here is an example:

> Carol decided to give the apple to Ann if the maximum sum of the distances from the power station to each of its substations, knowing that the power station was located on the boundary of a square region with 10 miles on each side, and the three substations were located inside the square region, was GREATER/SMALLER than 30 miles, and otherwise give the apple to Bella; after checking who would get the apple, Carol gave it to her.

This example is directly based on a practice test for the GRE exam.[2] Most GRE problems could in fact be turned into Winograd Schemas if we were to allow more alternatives than two; for instance:[3]

> Carol decided to give the apple to the Ann if the amount of sugar required for a recipe to make 30 cookies, knowing that the recipe requires 3/2 cups of sugar to make two dozen cookies, is GREATER than two cups, otherwise, give the apple to Bella if the amount of sugar is SMALLER than two cups, and finally, give the apple to Donna if the amount of sugar is EXACTLY two cups; after checking who would get the apple, Carol gave it to her.

At this point is fair to ask: if one can fold any question whatever into a Winograd Schema, then what is the point of the Winograd Challenge? Why not focus attention on the larger class of Question and Answer (QA) datasets, perhaps taking extended Winograd Schemas to provide an interesting class of questions? Indeed, while the original easy-for-humans Winograd Schemas do not really qualify as interesting tests for QA systems [Davis 2016], extended Winograd Schemas can work perfectly in the context of QA, provided the "question" asks for the referent of the ambiguous pronoun.

Perhaps the most profitable way to look at Winograd Schemas is, indeed, as useful members of the larger class of question and answer tests. Yet Winograd Schemas have something distinctive about them. For one thing, they can be, if well designed, much less dry than a GRE problem — as we show in later examples. Secondly, they do depend on ambiguities that are not usually explored in story telling, where one gets a short story and must answer a few questions about it; typically the difficulty lies in capturing the relationships amongst entities so that simple questions can be answered, not in actually interpreting dubious statements.

Thus it makes sense to build a store of tough Winograd Schemas whose purpose is to test aspects of QA systems that are not explored by other tests. In the remainder of this section we discuss strategies that can be used to successfully build tough Winograd Schemas; hopefully they will stimulate the construction of a large dataset containing such schemas.

3.1. Complex social stories

One natural way to build a tough Winograd Schema is to mix several characters and subtle expectations.

The proponents of the Winograd Challenge explicitly debated how to turn a sentence by Jane Austen into a (very strict) Winograd Schema [Levesque et al. 2012]. The original sentence by Austen was "Her mother had died too long ago for her to have more than an indistinct remembrance of her caresses; and her place had been taken by an excellent woman as governess, who had fallen little short of a mother in affection". An easy-for-humans Winograd Schema was then contemplated, one that did not satisfy the proponents of the Winograd Challenge:

> Emma's mother had died long ago, and *her* PLACE HAD BEEN TAKEN/EDUCATION HAD BEEN MANAGED by an excellent woman as governess.

It seems better to produce an extended Winograd Schema by making minimal changes to Austen's sentence, thus producing a much richer and difficult schema:

[2] https://www.ets.org/s/gre/accessible/GRE_Practice_Test_1_Quant_18_point.pdf.

[3] This schema is also based on the same practice test for the GRE exam.

> Her mother had died too long ago for her to have more than an indistinct remembrance of her caresses; and *her* PLACE HAD BEEN TAKEN/EDUCATION HAD BEEN MANAGED by an excellent woman as governess, who had fallen little short of a mother in affection.

If we are looking for beautifully crafted sentences that convey much meaning, we need not stop at the previous example. Here is a Winograd Schema based on a sentence by Marcel Proust (in fact this schema is about half the size of the original sentence in *Swann's Way*):

> The name Gilberte passed close by me, evoking all the more forcibly her whom it labelled in that it did not merely refer to her, as one speaks of a man in his absence, but was directly addressed to her; it passed thus close by me, in action, so to speak, with a force that increased with the curve of its trajectory and as it drew near to its target — carrying in its wake, I could feel, the knowledge, the impression of her to whom it was addressed that belonged not to me but to the friend who called to her, everything that, while she uttered the words, she more or less vividly reviewed, possessed in her memory, of their daily intimacy, of the visits that they paid to each other, of an affection that was so painful to me from being, conversely, so familiar, so tractable to this happy girl —; letting float in the atmosphere the delicious scent which that call had foreshadowed, from the evening to come, at her home — and within those confusing thoughts I wondered, how could *it* carry so much KNOWLEDGE/PAIN?

This schema tells a complete and delicate story. We can imagine the setting; here is a narrator who is clearly much in love, painfully feeling envy that anyone would know anything about Gilberte, and hoping to soon meet her. There is not factual knowledge to be looked up in Wikipedia; we just know that such things happen through subtle social expectations built across centuries of human experience. Moreover, Proust's original sentence is not pointlessly labyrinthine; the many sentences within the sentence communicate the fact that the narrator is confused. Human readers get such a nuance through some sort of commonsense entirely missed by the Winograd Schemas in WSC273.

An important point is that the latter schema moves away from the strict pattern of WSC273, where two entities appear, then either something happens to one of them, or something else happens to the other (this is clearly the case in the first rendering of Austen's sentence, where either Emma is educated or her mother is replaced). In the latter schema, there are several characters; there is Gilberte, and there is her name, there is a friend and her knowledge, and their intimacy, their affection. It is *not* the case that random guessing gets one a 50% accuracy!

In fact, here are answers provided by UnifiedQA for the question "What did carry the KNOWLEDGE/PAIN?" followed by the corresponding instance of the latter Winograd Schema.[4] With respect to KNOWLEDGE, we have

```
Prediction [small, 60 million parameters]: the impression of her to whom it was addressed.
Prediction [large, 770 million parameters]: Gilberte
```

and with respect to PAIN, we have

```
Prediction [small, 60 million parameters]: scent
Prediction [large, 770 million parameters]: that name
```

[4]Answers produced in July 4th 2020 through the site https://unifiedqa.apps.allenai.org/. Note that the available interface to UnifiedQA does not use the specialized version of UnifiedQA that attains the best current performance in the Winograd Challenge.

We should note that UnifiedQA seems to randomize answers, so it is not the case that it produces twice the same answer with the same input. For instance, with PAIN, the system variously produced (with the large) model: Gilberte, the name, a name, that name, the name Gilberte, me the bearer of pain, and even the correct answer, affection.

There is an endless number of alluring one-sentence stories in the literature, and therefore an endless number of charming extended Winograd Schemas to be produced. To avoid repeating ourselves, we move to a different strategy.

3.2. Logical puzzles

In our quest for tough Winograd Schemas that do not have the dry feel of multiple choice quizes turned into sentences, a reasonable strategy is to look for logical puzzles that can be concisely formulated. There are many different kinds of riddles; to be concrete, here we explore a few possibilities.

Here is a fun Winograd Schema that is similar to countless logical riddles:

> Miss Marple was looking for the jewel, so she asked the girls about it: Ann said she took the jewel, Bella said Donna was the thief, Carol said she did not even see the jewel, and Donna said Ann DID/DIDN'T in fact take the jewel. When Miss Marple learned that only one of the girls was telling the truth, she immediately knew who had the jewel, and she smiled to *her*.

Again we have a situation where random guessing will not produce 50% accuracy, as there are four possible thieves — but only two can be guilty, depending on whether we choose DID or DIDN'T. Indeed, if Donna said Ann did take the jewel, then Bella took it; if Donna said otherwise, then Carol took it.

Here is how UnifiedQA answers the question "Who took the jewel?", both with DID and DIDN'T:

```
Prediction [small, 60 million parameters]: Ann
Prediction [large, 770 million parameters]: Ann
```

We now move to another strategy that can generate many tough Winograd Schemas. A variety of logical puzzles actually asks whether there is a sequence of actions that achieves some goal; apparently such puzzles are not related to pronoun disambiguation as they do not have a simple answer. Yet we can build a Winograd Schema whose solution depends on determining the required strategy. For instance, here is a tough Winograd Schema that requires one to think about actions:

> Ann, Bella and Carol were carrying boxes with fruits, respectively with labels "Apples", "Oranges", and "Apples and Oranges", but all boxes had been wrongly labeled. Donna wanted to buy a box surely containing only apples, and she could only try one fruit — she thought about it, took a fruit from a box, and saw it was an APPLE/ORANGE, then looked at the lady carrying the box she wanted and paid *her*.

The solution requires some thinking about the strategy adopted by Donna. There are three options: she can take a fruit from Ann, from Bella, or from Carol; however, the only option that decides the matter decisively is to take the fruit from Carol (the other options may lead to ambiguity depending on the fruit that is taken). Thus Donna should take a fruit from Carol, and then: if Donna saw an APPLE, she paid Carol; if Donna saw an ORANGE, she paid Bella.

Here is how UnifiedQA answers the question "Who is carrying the box that Donna wants?" when Donna saw an APPLE:

```
Prediction [small, 60 million parameters]: lady
Prediction [large, 770 million parameters]: the lady.
```

and here is what happens if Donna saw an ORANGE:

```
Prediction [small, 60 million parameters]: the lady
Prediction [large, 770 million parameters]: the lady
```

In a sense, this is a smart answer: obviously, the lady is carrying the box Donna wanted. But then one might destroy the original ambiguity by saying: "Well, the pronoun *her* refers to the lady carrying the box Donna wanted." This strategy destroys any ambiguity whatever: for instance, take the first Winograd Schema of this paper; there we have that *it* is the thing that doesn't fit in the brown suitcase. In any case, UnifiedQA fails miserably if we just replace the word "lady" by "person", an apparently innocuous change. We get, with APPLE:

```
Prediction [small, 60 million parameters]: Ann
Prediction [large, 770 million parameters]: Bella
```

and with ORANGE:

```
Prediction [small, 60 million parameters]: her mother
Prediction [large, 770 million parameters]: the person carrying the box she wants.
```

There are other puzzles that resort to some epistemic reasoning; for those, we must think about the information that various characters hold at various times. Here is a Winograd Schema based on a well-known epistemic puzzle:[5]

> Cheryl first described to both Albert and Bernard the list of possible couples: either Ann with Carl or Dom or Gabriel, or Bella with Erik or Felix, or Carol with Alex or Bob or Dom, or Donna with Alex or Bob or Carl or Erik, or Ella simply with Alex; then she told Albert the selected bride and told Bernard the selected groom. Albert said that indeed Bernard did not know the name of the bride; then Bernard said that he didn't know the name of the bride before and he DID/DIDN'T know it now; and Albert then stated that he knew the name of the groom — at that point Miss Marple discovered the name of the bride and smiled at *her*.

The reader is invited to solve this Winograd Schema. As for UnifiedQA, here is how it answers the question "Who is the bride?" with DID:

```
Prediction [small, 60 million parameters]: Albert the chosen bride.
Prediction [large, 770 million parameters]: Camilla.
```

and with DIDN'T:

```
Prediction [small, 60 million parameters]: on her wedding day.
Prediction [large, 770 million parameters]: miss marple
```

By again exploring epistemic reasoning, here is a fun and diabolic Winograd Schema:[6]

[5]The original puzzle appeared in an edition of the Singapore and Asian Schools Math Olympiad; it is often referred to as "Cheryl's Birthday Problem" and its solution can be found in many venues.

[6]This Schema is based on a "Riddle of the Week" by Jay Bennett in *Popular Mechanics*, 2017 (the commented solution can be found there).

> Donna took 4 red stamps and 4 green stamps, threw two of them away, and affixed two of them in the back of each of three logicians Ann, Bella, and Carol, so that each logician could only see the stamps in the other two logician's backs; Miss Marple, who could not see any of the stamps, asked Ann, Bella and Carol in this sequence whether they knew the colors of their own stamps, and they replied No, No, No in sequence — and when Miss Marple asked Ann again whether she knew the colors of her own stamps, and she replied YES/NO, she concluded that one logician surely had distinct stamps in her back, and she smiled at *her*.

In this case, UnifiedQA answered the question "Who had distinct stamps in her back?", for both options YES and NO, with a trivially correct solution:

```
Prediction [small, 60 million parameters]: one logician
Prediction [large, 770 million parameters]: logician
```

Now if we replace the last appearance of "logician" with person, then we get, with YES:

```
Prediction [small, 60 million parameters]: one person
Prediction [large, 770 million parameters]: Ann.
```

and with NO:

```
Prediction [small, 60 million parameters]: one person
Prediction [large, 770 million parameters]: Logician
```

4. Discussion

We hope the present paper contributes to the ongoing debate within AI as to what is commonsense reasoning and how to measure it in artificial agents. We have argued that, contrary to what the proponents of the original Winograd Challenge proposed, we should not try to evaluate commonsense reasoning, and overall intelligence, by focusing on easy-for-humans Winograd Schemas. The consequence of an exaggerated focus on such schemas is the reliance on customary causal settings and entrenched social norms that can be surely detected through large textual corpora — thus making it possible for ever growing language models to succeed in the Winograd Challenge.

Instead, we should pay attention to the broad class of Q&A problems that emerge for extended Winograd Schemas. The first step in building a satisfactory dataset that can support the progress of AI is to understand what exactly are these extended Winograd Schemas, and to show how they can be built. This is the main contribution of this paper: we have presented a number of strategies that can be used to produce a large number of tough yet solvable Winograd Schemas — schemas that the current state of art in Q&A cannot handle at all.

Future work is, obviously, to build artificial intelligences that can solve tough Winograd Schemas. This seems to require some real intelligence.

Acknowledgements

The first author has been partially supported by the Conselho Nacional de Desenvolvimento Cientifico e Tecnológico (CNPq), grant 312180/2018- 7. The second author has been supported by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), grant 2018/09681-4.

The work was also supported by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), grants 2016/18841-0 and 2019/07665-4, and also by the Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior (CAPES) - finance code 001.

References

Bennett, C. H. and Gardner, M. (1979). The random number omega bids fair to hold the mysteries of the universe. *Scientific American*, 241:20–34.

Calude, C. S., Dinneen, M. J., and Shu, C.-K. (2002). Computing a glimpse of randomness. *Experimental Mathematics*, 11.

Cozman, F. G. and Munhoz, H. N. (2020). Some thoughts on knowledge-enhanced machine learning. *International Journal of Approximate Reasoning*, submitted.

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

Davis, E. (2018). Collection of Winograd schemas.

Epstein, R., Roberts, G., and Beber, G. (2009). *Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer*. Springer.

Floridi, L., Taddeo, M., and Turilli, M. (2009). Turings imitation game: Still an impossible challenge for all machines and some judges — an evaluation of the 2008 Loebner contest. *Minds & Machines*, 19:145–150.

Hill, R. (1982). *A Dictionary of False Friends*. Macmillan Press.

Khashabi, D., Khot, T., Sabhwaral, A., Tafjord, O., Clark, P., and Hajishirzi, H. (2020). UnifiedQA: Crossing format boundaries with a single QA system. Technical report, arXiv:2005.00700.

Kocijan, V., Lukasiewicz, T., Davis, E., Marcus, G., and Morgenstern, L. (2020). A review of Winograd Schema Challenge datasets and approaches. Technical report, arXiv 2004.13831.

Levesque, H. J., Davis, E., and Morgenstern, L. (2012). The Winograd schema challenge. In *International Conference on Principles of Knowledge Representation and Reasoning*, page 552561.

Marcus, G., Rossi, F., and Veloso, M. (2016). Beyond the Turing test. *AI Magazine*, 37(1):3–4.

Rajpurkar, P., Zhang, J., Lopyrey, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Sakaguchi, K., Bras, R. L., Bhagavatula, C., and Choi, Y. (2019). Winogrande: An adversarial Winograd schema challenge at scale. Technical report, arXiv.1907.10641.

Shlesinger, M. and Malkiel, B. (2005). Comparing modalities: Cognates as a case in point. *Across Languages and Cultures*, 6.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, LIX:433–460.