What Question Answering can Learn from Trivia Nerds

Jordan Boyd-Graber[⬤],[†]
iSchool, CS, UMIACS, LSC
⬤ University of Maryland
jbg@umiacs.umd.edu

Benjamin Börschinger[†]

† Google Research Zürich
{jbg, bboerschinger}@google.com

Abstract

Question answering (QA) is not just building systems; this NLP subfield also creates and curates challenging question datasets that reveal the best systems. We argue that QA datasets—and QA leaderboards—closely resemble trivia tournaments: the questions agents—humans or machines—answer reveals a "winner". However, the research community has ignored the lessons from decades of the trivia community creating vibrant, fair, and effective QA competitions. After detailing problems with existing QA datasets, we outline several lessons that transfer to QA research: removing ambiguity, identifying better QA agents, and adjudicating disputes.

1 Introduction

This paper takes an unconventional analysis to answer "where we've been and where we're going" in question answering (QA). Instead of approaching the question only as computer scientists, we apply the best practices of trivia tournaments to QA datasets.

The QA community is obsessed with evaluation. Schools, companies, and newspapers hail new SOTAs and topping leaderboards, giving rise to troubling claims (Lipton and Steinhardt, 2019) that an "AI model tops humans" (Najberg, 2018) because it 'won' some leaderboard, putting "millions of jobs at risk" (Cuthbertson, 2018). But what is a leaderboard? A leaderboard is a statistic about QA accuracy that induces a ranking over participants.

Newsflash: this is the same as a trivia tournament. The trivia community has been doing this for decades (Jennings, 2006); Section 2 details this overlap between the qualities of a first-class QA dataset (and its requisite leaderboard). The experts running these tournaments are imperfect, but they've learned from their past mistakes (see Appendix A for a brief historical perspective) and cre-

ated a community that reliably identifies those best at question answering. Beyond the format of the *competition*, trivia norms ensure individual questions are clear, unambiguous, and reward knowledge (Section 3).

We are not saying that academic QA should surrender to trivia questions or the community—far from it! The trivia community does not understand the real world information seeking needs of users or what questions challenge computers. However, they have well-tested protocols to declare that someone is better at answering questions than another. This collection of tradecraft and principles can nonetheless help the QA community.

Beyond these general concepts that QA can learn from, Section 4 reviews how the "gold standard" of trivia formats, Quizbowl can improve traditional QA. We then briefly discuss how research that uses fun, fair, and good trivia questions can benefit from the expertise, pedantry, and passion of the trivia community (Section 5).

2 Surprise, this is a Trivia Tournament!

"My research isn't a silly trivia tournament," you say. That may be, but let us first tell you a little about what running a tournament is like, and perhaps you might see similarities.

First, the questions. Either you write them yourself or you pay someone to write questions by a particular date (sometimes people on the Internet).

Then, you advertise. You talk about your questions: who is writing them, what subjects are covered, and why people should try to answer them.

Next, you have the tournament. You keep your questions secure until test time, collect answers from all participants, and declare a winner. Afterward, people use the questions to train for future tournaments.

These have natural analogs to crowd sourcing

questions, writing the paper, advertising, and running a leaderboard. Trivia nerds cannot help you form hypotheses or write your paper, but they can tell you how to run a fun, well-calibrated, and discriminative tournament.

Such tournaments are designed to effectively find a winner, which matches the scientific goal of knowing which model best answers questions. Our goal is **not to encourage the QA community to adopt to quirks and gimmicks of trivia games**. Instead, it's to encourage experiments and datasets that **consistently and efficiently find the systems that best answer questions**.

2.1 Are we having fun?

Many authors use crowdworkers to establish human accuracy (Rajpurkar et al., 2016; Choi et al., 2018). However, they are not the only humans who should answer a dataset's questions. So should the datasets' creators.

In the trivia world, this is called a **play test**: get in the shoes of someone *answering* the questions. If you find them boring, repetitive, or uninteresting, so will crowdworkers. If you can find shortcuts to answer questions (Rondeau and Hazen, 2018; Kaushik and Lipton, 2018), so will a computer.

Concretely, Weissenborn et al. (2017) catalog artifacts in SQuAD (Rajpurkar et al., 2018), the most popular QA leaderboard. If you see a list like "Along with Canada and the United Kingdom, what country. . . ", you can ignore the rest of the question and just type Ctrl+F (Yuan et al., 2019; Russell, 2020) to find the third country—Australia in this case—that appears with "Canada and the UK". Other times, a SQuAD playtest would reveal frustrating questions that are i) answerable given the information but not with a direct span,[1] ii) answerable only given facts beyond the given paragraph,[2] iii) unintentionally embedded in a discourse, resulting in arbitrary correct answers,[3] iv) or non-questions.

---

[1] A source paragraph says "In [Commonwealth countries]. . . the term is generally restricted to. . . Private education in North America covers the whole gamut. . . "; thus, "What is the term private school restricted to in the US?" has the information needed but not as a span.

[2] A source paragraph says "Sculptors [in the collection include] Nicholas Stone, Caius Gabriel Cibber, [...], Thomas Brock, Alfred Gilbert, [...] and Eric Gill", i.e., a list of names; thus, the question "Which British sculptor whose work includes the Queen Victoria memorial in front of Buckingham Palace is included in the V&A collection?" should be unanswerable in SQuAD.

[3] A question "Who *else* did Luther use violent rhetoric towards?" has the gold answer "writings condemning the Jews and in diatribes against Turks".

SearchQA (Dunn et al., 2017), derived from *Jeopardy!*, asks "An article that he wrote about his riverboat days was eventually expanded into *Life on the Mississippi*." The apprentice and newspaper writer who wrote the article is named Samuel Langhorne Clemens; however, the reference answer is his later pen name, Mark Twain. Most QA evaluation metrics would count Samuel Clemens as incorrect. In a real game of *Jeopardy!*, this would not be an issue (Section 3.1).

Of course, fun is relative, and any dataset is bound to contain errors. However, playtesting is an easy way to find systematic problems: unfair, unfun playtests make for ineffective leaderboards. Eating your own dog food can help diagnose artifacts, scoring issues, or other shortcomings early in the process.

The deeper issues when creating a QA task are: i) have you designed a task that is internally consistent, ii) supported by a scoring metric that matches your goals, iii) using gold annotations that reward those who do the task well? Imagine someone who loves answering the questions your task poses: would they have fun on your task? This is the foundation of Gamification (von Ahn, 2006), which can create quality data from users motivated by fun rather than pay. Even if you pay crowdworkers, unfun questions may undermine your dataset goals.

2.2 Am I measuring what I care about?

Answering questions requires multiple skills: identifying answer mentions (Hermann et al., 2015), naming the answer (Yih et al., 2015), abstaining when necessary (Rajpurkar et al., 2018), and justifying an answer (Thorne et al., 2018). In QA, the emphasis on SOTA and leaderboards has focused attention on single automatically computable metrics—systems tend to be compared by their 'SQuAD score' or their 'NQ score', as if this were all there is to say about their relative capabilities. Like QA leaderboards, trivia tournaments need to decide on a single winner, but they explicitly recognize that there are more interesting comparisons.

A tournament may recognize different background/resources—high school, small school, undergraduates (Hentzel, 2018). Similarly, more practical leaderboards would reflect training time or resource requirements (see Dodge et al., 2019) including 'constrained' or 'unconstrained' training (Bojar et al., 2014). Tournaments also give specific awards (e.g., highest score without

incorrect answers). Again, there are obvious leaderboard analogs that would go beyond a single number. In SQuAD 2.0 (Rajpurkar et al., 2018), abstaining contributes the same to the overall F_1 as a fully correct answer, obscuring whether a system is more precise or an effective abstainer. If the task recognizes both abilities as important, reporting a single score risks implicitly prioritizing one balance of the two.

2.3 Do my questions separate the best?

Assume that you have picked a metric (or a set of metrics) that captures what you care about. A leaderboard based on this metric can rack up citations as people chase the top spot. But your leaderboard is only useful if it is **discriminative**: the best system reliably wins.

There are many ways questions might not be discriminative. If every system gets a question right (e.g., abstain on non-questions like "asdf" or correctly answer "What is the capital of Poland?"), the dataset does not separate participants. Similarly, if every system flubs "what is the oldest north-facing kosher restaurant", it is not discriminative. Sugawara et al. (2018) call these questions "easy" and "hard"; we instead argue for a three-way distinction.

In between easy questions (system answers correctly with probability 1.0) and hard (probability 0.0), questions with probabilities nearer to 0.5 are more interesting. Taking a cue from Vygotsky's proximal development theory of human learning (Chaiklin, 2003), these discriminative questions—rather than the easy or the hard ones—should most improve QA systems. These Goldilocks[4] questions (not random noise) decide who tops the leaderboard. Unfortunately, existing datasets have many easy questions. Sugawara et al. (2020) find that ablations like shuffling word order (Feng et al., 2018), shuffling sentences, or only offering the most similar sentence do not impair systems. Newer datasets such as DROP (Dua et al., 2019) and HellaSwag (Zellers et al., 2019) are harder for *today*'s systems; because Goldilocks is a moving target, we propose annual evaluations in Section 5.

2.4 Why so few Goldilocks questions?

This is a common problem in trivia tournaments, particularly pub quizzes (Diamond, 2009), where
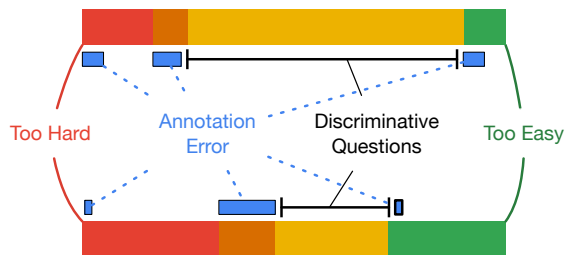

Figure 1: Two datasets with 0.16 annotation error: the top, however, better discriminates QA ability. In the good dataset (top), most questions are challenging but not impossible. In the bad dataset (bottom), there are more trivial or impossible questions *and* annotation error is concentrated on the challenging, discriminative questions. Thus, a smaller fraction of questions decide who sits atop the leaderboard, requiring a larger test set.

challenging questions can scare off patrons. Many quiz masters prefer popularity with players and thus write easier questions.

Sometimes there are fewer Goldilocks questions not by choice, but by chance: a dataset becomes less discriminative through annotation error. All datasets have some annotation error; if this annotation error is concentrated on the Goldilocks questions, the dataset will be less useful. As we write this in 2020, humans and computers sometimes struggle on the same questions.

Figure 1 shows two datasets of the same size with the same annotation error. However, they have different difficulty *distributions* and *correlation* of annotation error and difficulty. The dataset that has more discriminative questions and consistent annotator error has fewer questions that do not discriminate the winner of the leaderboard. We call this the effective dataset proportion ρ (higher is better). Figure 2 shows the test set size required to reliably discriminate systems for different ρ, based on a simulation (Appendix B).

At this point, you may despair about how big a dataset you need.[5] The same terror besets trivia tournament organizers. Instead of writing more questions, they use pyramidality (Section 4) to make every question count.

3 The Craft of Question Writing

Trivia enthusiasts agree that questions need to be well written (despite other disagreements). Asking "good questions" requires sophisticated pragmatic

---

[4] In a British folktale first recorded by Robert Southey, the character Goldilocks finds three beds: one too hard, one not hard enough, and one "just right".

[5] Using a more sophisticated simulation approach, the TREC 2002 QA test set (Voorhees, 2003) could not discriminate systems with less than a seven absolute score point difference.
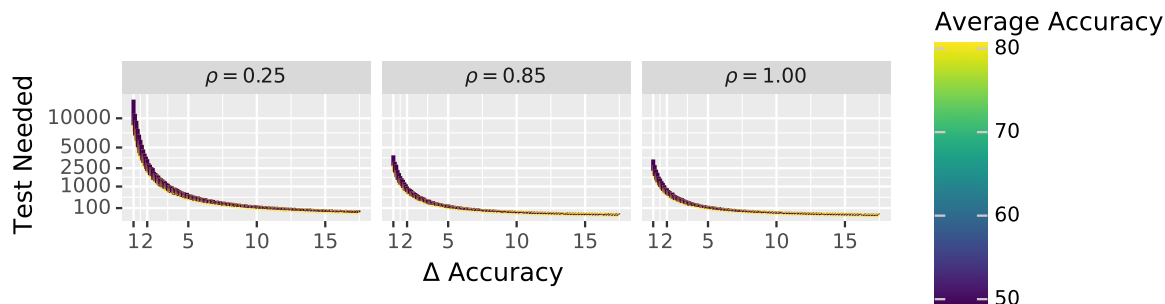
Figure 2: How much test data do you need to discriminate two systems with 95% confidence? This depends on both the difference in accuracy between the systems (x axis) and the average accuracy of the systems (closer to 50% is harder). Test set creators do not have much control over those. They do have control, however, over how many questions are discriminative. If all questions are discriminative (right), you only need 2500 questions, but if three quarters of your questions are too easy, too hard, or have annotation errors (left), you'll need 15000.

reasoning (Hawkins et al., 2015), and pedagogy explicitly acknowledges the complexity of writing effective questions for assessing student performance (Haladyna, 2004, focusing on multiple choice questions).

QA datasets, however, are often collected from the wild or written by untrained crowdworkers. Crowdworkers lack experience in crafting questions and may introduce idiosyncrasies that short-cut machine learning (Geva et al., 2019). Similarly, data collected from the wild such as Natural Questions (Kwiatkowski et al., 2019) or AmazonQA (Gupta et al., 2019) by design have vast variations in quality. In the previous section, we focused on how datasets as a whole should be structured. Now, we focus on how specific *questions* should be structured to make the dataset as valuable as possible.

3.1 Avoiding ambiguity and assumptions

Ambiguity in questions not only frustrates answerers who resolve the ambiguity 'incorrectly'. Ambiguity also frustrates the goal of using questions assess knowledge. Thus, the US Department of Transportation explicitly bans ambiguous questions from exams for flight instructors (Flight Standards Service, 2008); and the trivia community has likewise developed rules and norms that prevent ambiguity. While this is true in many contexts, examples are rife in format called Quizbowl (Boyd-Graber et al., 2012), whose very long questions[6] showcase trivia

---

[6]Like *Jeopardy!*, they are not syntactically questions but still are designed to elicit knowledge-based responses; for consistency, we still call them questions.

writers' tactics. For example, Quizbowl author Zhu Ying (writing for the 2005 PARFAIT tournament) asks participants to identify a fictional character while warning against possible confusion [emphasis added]:

> He's **not Sherlock Holmes**, but his address is 221B. He's **not the Janitor on Scrubs**, but his father is played by R. Lee Ermy. [...] For ten points, name this misanthropic, crippled, Vicodin-dependent central character of a FOX medical drama.
> **ANSWER:** Gregory House, MD

In contrast, QA datasets often contain ambiguous and under-specified questions. While this sometimes reflects real world complexities such as actual under-specified or ill-formed search queries (Faruqui and Das, 2018; Kwiatkowski et al., 2019), ignoring this ambiguity is problematic. As a concrete example, Natural Questions (Kwiatkowski et al., 2019) answers "what year did the us hockey team win the Olympics" with 1960 and 1980, ignoring the US women's team, which won in 1998 and 2018, and further assuming the query is about *ice* rather than *field* hockey (also an Olympic event). Natural Questions associates a page about the United States men's national ice hockey team, arbitrarily removing the ambiguity *post hoc*. However, this does not resolve the ambiguity, which persists in the original question: information retrieval arbitrarily provides one of many interpretations. True to their name, Natural Questions are often under-specified when users ask a question online.

The problem is neither that such questions exist nor that machine reading QA considers questions

given an associated context. The problem is that tasks do not explicitly acknowledge the original ambiguity and gloss over the implicit assumptions in the data. This introduces potential noise and bias (i.e., giving a bonus to systems that make the same assumptions as the dataset) in leaderboard rankings. At best, these will become part of the measurement error of datasets (no dataset is perfect). At worst, they will recapitulate the biases that went into the creation of the datasets. Then, the community will implicitly equate the biases with correctness: you get high scores if you adopt this set of assumptions. These enter into real-world systems, further perpetuating the bias. Playtesting can reveal these issues (Section 2.1), as implicit assumptions can rob a player of correctly answered questions. If you wanted to answer 2014 to "when did Michigan last win the championship"—when the Michigan State Spartans won the Women's Cross Country championship—and you cannot because you chose the wrong school, the wrong sport, and the wrong gender, you would complain as a player; researchers instead discover latent assumptions that creep into the data.[7]

It is worth emphasizing that this is not a purely hypothetical problem. For example, Open Domain Retrieval Question Answering (Lee et al., 2019) deliberately avoids providing a reference context for the question in its framing but, in re-purposing data such as Natural Questions, opaquely relies on it for the gold answers.

3.2 Avoiding superficial evaluations

A related issue is that, in the words of Voorhees and Tice (2000), "there is no such thing as a question with an obvious answer". As a consequence, trivia question authors delineate acceptable and unacceptable answers.

For example, in writing for the trivia tournament Harvard Fall XI, Robert Chu uses a mental model of an answerer to explicitly delineate the range of acceptable correct answers:

> In Newtonian gravity, this quantity satisfies Poisson's equation. [...] For a dipole, this quantity is given by negative the dipole moment dotted with the electric field. [...] For 10 points, name this form of energy contrasted with kinetic.
> **ANSWER:** potential energy (*prompt on energy; accept specific types like electrical potential energy or gravitational potential energy; do not accept or prompt on just "potential"*)

Likewise, the style guides for writing questions stipulate that you must give the answer type clearly and early on. These mentions specify whether you want a book, a collection, a movement, etc. It also signals the level of specificity requested. For example, a question about a date must state "day and month required" (September 11, "month and year required" (April 1968), or "day, month, and year required" (September 1, 1939). This is true for other answers as well: city and team, party and country, or more generally "two answers required". Despite these conventions, no pre-defined set of answers is perfect, and every worthwhile trivia competition has a process for adjudicating answers.

In major high school and college national competitions and game shows, if low-level staff cannot resolve the issue by throwing out a single question or accepting minor variations (America instead of USA), the low-level staff contacts the tournament director. The tournament director, who has a deeper knowledge of rules and questions, often decide the issue. If not, the protest goes through an adjudication process designed to minimize bias:[8] write the summary of the dispute, get all parties to agree to the summary, and then hand the decision off to mutually agreed experts from the tournament's phone tree. The substance of the disagreement is communicated (without identities), and the experts apply the rules and decide.

For example, when a particularly inept *Jeopardy!* contestant[9] did not answer laproscope to "Your surgeon could choose to take a look inside you with this type of fiber-optic instrument". Since the van Doren scandal (Freedman, 1997), every television trivia contestant has an advocate assigned from an auditing company. In this case, the advocate initiated a process that went to a panel of judges who then ruled that endoscope (a more general term) was also correct.

The need for a similar process seems to have been well-recognized in the earliest days of QA system bake-offs such as TREC-QA, and Voorhees (2008) notes that

> [d]ifferent QA runs very seldom return exactly the same [answer], and it is quite difficult to determine automatically whether the difference [...] is significant.

In stark contrast to this, QA datasets typically only provide a single string or, if one is lucky, several

---

[7] Where to draw the line is a matter of judgment; computers—which lack common sense—might find questions ambiguous where humans would not.

strings. A correct answer means *exactly* matching these strings or at least having a high token overlap F_1, and failure to agree with the pre-recorded admissible answers will put you at an uncontestable disadvantage on the leaderboard (Section 2.2).

To illustrate how current evaluations fall short of meaningful discrimination, we qualitatively analyze two near-SOTA systems on SQuAD V1.1: the original XLNet (Yang et al., 2019) and a subsequent iteration called XLNet-123.[10]

Despite XLNet-123's margin of almost four absolute F_1 (94 vs 98) on development data, a manual inspection of a sample of 100 of XLNet-123's wins indicate that around two-thirds are 'spurious': 56% are likely to be considered not only equally good but essentially identical; 7% are cases where the answer set omits a correct alternative; and 5% of cases are 'bad' questions.[11]

Our goal is not to dwell on the exact proportions, to minimize the achievements of these strong systems, or to minimize the usefulness of quantitative evaluations. We merely want to raise the limitation of *blind automation* for distinguishing between systems on a leaderboard.

Taking our cue from the trivia community, we present an alternative for MRQA. Blind test sets are created for a specific time; all systems are submitted simultaneously. Then, all questions and answers are revealed. System authors can protest correctness rulings on questions, directly addressing the issues above. After agreement is reached, quantitative metrics are computed for comparison purposes—despite their inherent limitations they at least can be trusted. Adopting this for MRQA would require creating a new, smaller test set every year. However, this would gradually refine the annotations and process.

This suggestion is not novel: Voorhees and Tice (2000) accepts automatic evaluations "for experiments internal to an organization where the benefits of a reusable test collection are most significant (*and the limitations are likely to be understood*)" (our emphasis) but that "satisfactory techniques for [automatically] evaluating new runs" have not been found yet. We are not aware of any change on this front—if anything, we seem to have become more insensitive as a community to just how limited our current evaluations are.

---

[10]We could not find a paper describing XLNet-123, the submission is by http://tia.today.

[11]Examples in Appendix C.

3.3 Focus on the bubble

While every question should be perfect, time and resources are limited. Thus, authors and editors of tournaments "focus on the bubble", where the "bubble" are the questions most likely to discriminate between top teams at the tournament. These questions are thoroughly playtested, vetted, and edited. Only after these questions have been perfected will the other questions undergo the same level of polish.

For computers, the same logic applies. Authors should ensure that these discriminative questions are correct, free of ambiguity, and unimpeachable. However, as far as we can tell, the authors of QA datasets do not give any special attention to these questions.

Unlike a human trivia tournament, however—with finite patience of the participants—this does not mean that you should necessarily remove all of the easy or hard questions from your dataset. This could inadvertently lead to systems unable to answer simple questions like "who is buried in Grant's tomb?" (Dwan, 2000, Chapter 7). Instead, focus more resources on the bubble.

4 Why Quizbowl is the Gold Standard

We now focus our thus far wide-ranging QA discussion to a specific format: Quizbowl, which has many of the desirable properties outlined above. We have no delusion that mainstream QA will universally adopt this format (indeed, a monoculture would be bad). However, given the community's emphasis on fair evaluation, computer QA can borrow *aspects* from the gold standard of human QA.

We have shown example of Quizbowl questions, but we have not explained how the format works; see Rodriguez et al. (2019) for more. You might be scared off by how long the questions are. However, in real Quizbowl trivia tournaments, they are not finished because the questions are *interruptible*.

Interruptible A moderator reads a question. Once someone knows the answer, they use a signaling device to "*buzz in*". If the player who buzzed is right, they get points. Otherwise, they lose points and the question continues for the other team.

Not all trivia games with buzzers have this property, however. For example, take *Jeopardy!*, the subject of Watson's *tour de force* (Ferrucci et al., 2010). While *Jeopardy!* also uses signaling devices, these only work *once the question has been*

read in its entirety; Ken Jennings, one of the top *Jeopardy!* players (and also a Quizbowler) explains it on a *Planet Money* interview (Malone, 2019):

Jennings: The buzzer is not live until Alex finishes reading the question. And if you buzz in before your buzzer goes live, *you actually lock yourself out for a fraction of a second.* So the big mistake on the show is people who are all adrenalized and are buzzing too quickly, too eagerly.
Malone: OK. To some degree, *Jeopardy!* is kind of a video game, and a *crappy video game where it's, like, light goes on, press button*—that's it.
Jennings: (Laughter) Yeah.

Jeopardy!'s buzzers are a gimmick to ensure good television; however, Quizbowl buzzers discriminate knowledge (Section 2.3). Similarly, while TriviaQA (Joshi et al., 2017) is written by knowledgeable writers, the questions are not pyramidal.

Pyramidal Recall that effective datasets discriminate the best from the rest—the higher the proportion of effective questions ρ, the better. Quizbowl's ρ is nearly 1.0 because discrimination happens *within* a question: after every word, an answerer must decide if know enough to answer. Quizbowl questions are arranged so that questions are maximally *pyramidal*: questions begin with hard clues—ones that require deep understanding—to more accessible clues that are well known.

Well-Edited Quizbowl questions are created in phases. First, the *author* selects the answer and assembles (pyramidal) clues. A *subject editor* then removes ambiguity, adjusts acceptable answers, and tweaks clues to optimize discrimination. Finally, a *packetizer* ensures the overall set is diverse, has uniform difficulty, and is without repeats.

Unnatural Trivia questions are fake: the asker already knows the answer. But they're no more fake than a course's final exam, which—like leaderboards—are designed to test knowledge.

Experts know when questions are ambiguous (Section 3.1); while "what play has a character whose father is dead" could be *Hamlet*, *Antigone*, or *Proof*, a good writer's knowledge avoids the ambiguity. When authors omit these cues, the question is derided as a "hose" (Eltinge, 2013), which robs the tournament of fun (Section 2.1).

One of the benefits of contrived formats is a focus on specific phenomena. Dua et al. (2019) exclude questions an existing MRQA system could answer to focus on challenging quantitative reasoning. One of the trivia experts consulted in Wallace et al. (2019) crafted a question that tripped up neural QA by embedding the phrase "this author opens

Crime and Punishment" into a question; the top system confidently answers Fyodor Dostoyevski. However, that phrase was in a longer question "The narrator in *Cogwheels* by this author opens *Crime and Punishment* to find it has become *The Brothers Karamazov*". Again, this shows the inventiveness and linguistic dexterity of the trivia community.

A counterargument is that real-life questions—e.g., on Yahoo! Questions (Szpektor and Dror, 2013), Quora (Iyer et al., 2017) or web search (Kwiatkowski et al., 2019)—ignore the craft of question writing. Real humans react to unclear questions with confusion or divergent answers, explicitly answering with how they interpreted the original question ("I assume you meant...").

Given real world applications will have to deal with the inherent noise and ambiguity of unclear questions, our systems must cope with it. However, addressing the real world cannot happen by glossing over its complexity.

Complicated Quizbowl is more complex than other datasets. Unlike other datasets where you just need to decide *what* to answer, in Quizbowl you also need to choose *when* to answer the question.[12] While this improves the dataset's discrimination, it can hurt popularity because you cannot copy/paste code from other QA tasks. The complicated pyramidal structure complicates[13] some questions (e.g., what is log base four of sixty-four).

5 A Call to Action

You may disagree with the superiority of Quizbowl as a QA framework (*de gustibus non est disputandum*). In this final section, we hope to distill our advice into a call to action regardless of your question format or source. Here are our recommendations if you want to have an effective leaderboard.

---

[12]This complex methodology can be an advantage. The underlying mechanisms of systems that can play Quizbowl (e.g., reinforcement learning) share properties with other tasks, such as simultaneous translation (Grissom II et al., 2014; Ma et al., 2019), human incremental processing (Levy et al., 2008; Levy, 2011), and opponent modeling (He et al., 2016).

[13]But not necessarily preclude, as the Illinois High School Scholastic Bowl Coaches Association shows:

> This is the smallest counting number which is the radius of a sphere whose volume is an integer multiple of π. It is also the number of distinct real solutions to the equation $x^7 - 19x^5 = 0$. This number also gives the ratio between the volumes of a cylinder and a cone with the same heights and radii. Give this number equal to the log base four of sixty-four.

Talk to Trivia Nerds You should talk to trivia nerds because they have useful information (not just about the election of 1876). Trivia is not just the accumulation of information but also connecting disparate facts (Jennings, 2006). These skills are exactly those that we want computers to develop.

Trivia nerds are writing questions anyway; we can save money and time if we pool resources.[14] Computer scientists benefit if the trivia community writes questions that aren't trivial for computers to solve (e.g., avoiding quotes and named entities). The trivia community benefits from tools that make their job easier: show related questions, link to Wikipedia, or predict where humans will answer.

Likewise, the broader public has unique knowledge and skills. In contrast to low-paid crowdworkers, public platforms for question answering and citizen science (Bowser et al., 2013) are brimming with free expertise if you can engage the relevant communities. For example, the Quora query "Is there a nuclear control room on nuclear aircraft carriers?" is purportedly answered by someone who worked in such a room (Humphries, 2017). As machine learning algorithms improve, the "good enough" crowdsourcing that got us this far may not be enough for continued progress.

Eat Your Own Dog Food As you develop new question answering tasks, you should feel comfortable playing the task as a human. Importantly, this is not just to replicate what crowdworkers are doing (also important) but to remove hidden assumptions, institute fair metrics, and define the task well. For this to feel real, you will need to keep score; have all of your coauthors participate and compare their scores.

Again, we emphasize that **human and computer skills are not identical**, but this is a benefit: humans' natural aversion to unfairness will help you create a better task, while computers will blindly optimize an objective function (Bostrom, 2003). As you go through the process of playing on your question–answer dataset, you can see where you might have fallen short on the goals we outline in Section 3.

---

[14]Many question answering datasets benefit from the efforts of the trivia community. Ethically using the data, however, requires acknowledging their contributions and using their input to create datasets (Jo and Gebru, 2020, Consent and Inclusivity criterion).

Won't Somebody Look at the Data? After QA datasets are released, there should also be deeper, more frequent discussion of actual questions within the NLP community. Part of every post-mortem of trivia tournaments is a detailed discussion of the questions, where good questions are praised and bad questions are excoriated. This is not meant to shame the writers but rather to help build and reinforce cultural norms: questions should be well-written, precise, and fulfill the creator's goals. Just like trivia tournaments, QA datasets resemble a product for sale. Creators want people to invest time and sometimes money (e.g., GPU hours) in using their data and submitting to their leaderboards. It is "good business" to build a reputation for quality questions and discussing individual questions.

Similarly, discussing and comparing the actual predictions made by the competing systems should be part of any competition culture—without it, it is hard to tell what a couple of points on some leaderboard mean. To make this possible, we recommend that leaderboards include an easy way for anyone to download a system's development predictions for qualitative analyses.

Make Questions Discriminative We argue that questions should be discriminative (Section 2.3), and while Quizbowl is one solution (Section 4), not everyone is crazy enough to adopt this (beautiful) format. For more traditional QA tasks, you can maximize the usefulness of your dataset by ensuring as many questions as possible are challenging (but not impossible) for today's QA systems.

But you can use some Quizbowl intuitions to improve discrimination. In visual QA, you can offer increasing resolutions of the image. For other settings, create pyramidality by adding metadata: coreference, disambiguation, or alignment to a knowledge base. In short, consider multiple versions/views of your data that progress from difficult to easy. This not only makes more of your dataset discriminative but also reveals what makes a question answerable.

Embrace Multiple Answers or Specify Specificity As QA moves to more complicated formats and answer candidates, what constitutes a correct answer becomes more complicated. Fully automatic evaluations are valuable for both training and quick-turnaround evaluation. In the case annotators disagree, the question should explicitly state what level of specificity is required

(e.g., <u>September 1, 1939</u> vs. <u>1939</u> or <u>Leninism</u> vs. <u>socialism</u>). Or, if not all questions have a single answer, link answers to a knowledge base with multiple surface forms or explicitly enumerate which answers are acceptable.

Appreciate Ambiguity   If your intended QA application has to handle ambiguous questions, do justice to the ambiguity by making it part of your task—for example, recognize the original ambiguity and resolve it ("did you mean...") instead of giving credit for happening to 'fit the data'.

To ensure that our datasets properly "isolate the property that motivated [the dataset] in the first place" (Zaenen, 2006), we need to explicitly appreciate the unavoidable ambiguity instead of silently glossing over it.[15]

This is already an active area of research, with conversational QA being a new setting actively explored by several datasets (Reddy et al., 2018; Choi et al., 2018); and other work explicitly focusing on identifying useful clarification questions (Rao and Daumé III), thematically linked questions (Elgohary et al., 2018) or resolving ambiguities that arise from coreference or pragmatic constraints by rewriting underspecified question strings in context (Elgohary et al., 2019).

Revel in Spectacle   However, with more complicated systems and evaluations, a return to the yearly evaluations of TRECQA may be the best option. This improves not only the quality of evaluation (we can have real-time human judging) but also lets the test set reflect the build it/break it cycle (Ruef et al., 2016), as attempted by the 2019 iteration of FEVER (Thorne et al., 2019). Moreover, another lesson the QA community could learn from trivia games is to turn it into a spectacle: exciting games with a telegenic host. This has a benefit to the public, who see how QA systems fail on difficult questions and to QA researchers, who have a spoonful of fun sugar to inspect their systems' output and their competitors'.

In between full automation and expensive humans in the loop are automatic metrics that mimic the flexibility of human raters, inspired by machine translation evaluations (Papineni et al., 2002; Specia and Farzindar, 2010) or summarization (Lin, 2004). However, we should not forget that these

---

[15]Not surprisingly, 'inherent' ambiguity is not limited to QA; Pavlick and Kwiatkowski (2019) show natural language inference has 'inherent disagreements' between humans and advocate for recovering the full range of accepted inferences.

metrics were introduced as 'understudies'—good enough when quick evaluations are needed for system building but no substitute for a proper evaluation. In machine translation, Laubli et al. (2020) reveal that crowdworkers cannot spot the errors that neural MT systems make—fortunately, trivia nerds are cheaper than professional translators.

Be Honest in Crowning QA Champions Leaderboards are a ranking over entrants based on a ranking over numbers. This can be problematic for several reasons. The first is that single numbers have some variance; it's better to communicate estimates with error bars.

While—particularly for leaderboards—it is tempting to turn everything into a single number, recognize that there are often different sub-tasks and types of players who deserve recognition. A simple model that requires less training data or runs in under ten milliseconds may be objectively more useful than a bloated, brittle monster of a system that has a slightly higher F_1 (Dodge et al., 2019). While you may only rank by a single metric (this is what trivia tournaments do too), you may want to recognize the highest-scoring model that was built by undergrads, took no more than one second per example, was trained only on Wikipedia, etc.

Finally, if you want to make human–computer comparisons, pick the right humans. Paraphrasing a participant of the 2019 MRQA workshop (Fisch et al., 2019), a system better than the average human at brain surgery does not imply superhuman performance in brain surgery. Likewise, beating a distracted crowdworker on QA is not QA's endgame. If your task is realistic, fun, and challenging, you will find experts to play against your computer. Not only will this give you human baselines worth reporting—they can also tell you how to fix your QA dataset... after all, they've been at it longer than you have.

References

Luis von Ahn. 2006. Games with a purpose. *Computer*, 39:92–94.

David Baber. 2015. *Television Game Show Hosts: Biographies of 32 Stars*. McFarland.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*.

Nick Bostrom. 2003. Ethical issues in advanced artificial intelligence. *Institute of Advanced Studies in Systems Research and Cybernetics*, 2:12–17.

Anne Bowser, Derek Hansen, Yurong He, Carol Boston, Matthew Reid, Logan Gunnell, and Jennifer Preece. 2013. Using gamification to inspire new citizen science volunteers. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications*, Gamification '13, pages 18–25, New York, NY, USA. ACM.

Jordan Boyd-Graber, Brianna Satinoff, He He, and Hal Daume III. 2012. Besting the quiz master: Crowdsourcing incremental classification games. In *Proceedings of Empirical Methods in Natural Language Processing*.

Seth Chaiklin. 2003. *The Zone of Proximal Development in Vygotsky's Analysis of Learning and Instruction*, Learning in Doing: Social, Cognitive and Computational Perspectives, page 39âĂŞ64. Cambridge University Press.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of Empirical Methods in Natural Language Processing*.

Anthony Cuthbertson. 2018. Robots can now read better than humans, putting millions of jobs at risk. *Newsweek*.

Paul Diamond. 2009. *How To Make 100 Pounds A Night (Or More) As A Pub Quizmaster*. DP Quiz.

Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. 2019. Show your work: Improved reporting of experimental results. In *Proceedings of Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur Güney, Volkan Cirik, and Kyunghyun Cho. 2017. SearchQA: A new Q&A dataset augmented with context from a search engine. *CoRR*, abs/1704.05179.

R. Dwan. 2000. *As Long as They're Laughing: Groucho Marx and You Bet Your Life*. Midnight Marquee Press.

Ahmed Elgohary, Denis Peskov, and Jordan Boyd-Graber. 2019. Can you unpack that? learning to rewrite questions-in-context. In *Empirical Methods in Natural Language Processing*.

Ahmed Elgohary, Chen Zhao, and Jordan Boyd-Graber. 2018. Dataset and baselines for sequential open-domain question answering. In *Empirical Methods in Natural Language Processing*.

Stephen Eltinge. 2013. Quizbowl lexicon.

Manaal Faruqui and Dipanjan Das. 2018. Identifying well-formed natural language questions. In *Proceedings of Empirical Methods in Natural Language Processing*.

Shi Feng, Eric Wallace, Alvin Grissom II, Pedro Rodriguez, Mohit Iyyer, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretation difficult. In *Proceedings of Empirical Methods in Natural Language Processing*.

David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefer, and Chris Welty. 2010. Building Watson: An Overview of the DeepQA Project. *AI Magazine*, 31(3).

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen, editors. 2019. *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*. Association for Computational Linguistics, Hong Kong, China.

Flight Standards Service. 2008. *Aviation Instrutor's Handbook*, volume FAA-H-8083-9A. Federal Aviation Administration, Department of Transportation.

Morris Freedman. 1997. The fall of Charlie Van Doren. *The Virginia Quarterly Review*, 73(1):157–165.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of Empirical Methods in Natural Language Processing*.

Alvin Grissom II, He He, Jordan Boyd-Graber, and John Morgan. 2014. Don't until the final verb wait: Reinforcement learning for simultaneous machine translation. In *Proceedings of Empirical Methods in Natural Language Processing*.

Mansi Gupta, Nitish Kulkarni, Raghuveer Chanda, Anirudha Rayasam, and Zachary C. Lipton. 2019. AmazonQA: A review-based question answering task. In *International Joint Conference on Artificial Intelligence*.

Thomas M. Haladyna. 2004. *Developing and Validating Multiple-choice Test Items*. Lawrence Erlbaum Associates.

Robert X. D. Hawkins, Andreas Stuhlmüller, Judith Degen, and Noah D. Goodman. 2015. Why do you ask? Good questions provoke informative answers. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society*.

He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. 2016. Opponent modeling in deep reinforcement learning. In *Proceedings of the International Conference of Machine Learning*.

R. Robert Hentzel. 2018. NAQT eligibility overview.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of Advances in Neural Information Processing Systems*.

Bryan Humphries. 2017. Is there a nuclear control room on nuclear aircraft carriers?

Shankar Iyer, Nikhil Dandekar, and Kornél Csernai. 2017. Quora question pairs.

Ken Jennings. 2006. *Brainiac: adventures in the curious, competitive, compulsive world of trivia buffs*. Villard.

Eun Seo Jo and Timnit Gebru. 2020. Lessons from archives: Strategies for collecting sociocultural data in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the Association for Computational Linguistics*.

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of Empirical Methods in Natural Language Processing*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*.

Samuel Laubli, Sheila Castilho, Graham Neubig, Rico Sennrich, Qinlan Shen, and Antonio Toral. 2020. A set of recommendations for assessing human-machine parity in language translation. *Journal of Artificial Intelligence Research*, 67.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the Association for Computational Linguistics*.

Roger Levy. 2011. Integrating surprisal and uncertain-input models in online sentence comprehension: Formal techniques and empirical results. In *Proceedings of the Association for Computational Linguistics*.

Roger P. Levy, Florencia Reali, and Thomas L. Griffiths. 2008. Modeling the effects of memory on human online sentence processing with particle filters. In *Proceedings of Advances in Neural Information Processing Systems*.

Chin-Yew Lin. 2004. Rouge: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*.

Zachary C. Lipton and Jacob Steinhardt. 2019. Troubling trends in machine learning scholarship. *Queue*, 17(1).

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, and Haifeng Wang. 2019. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In *Proceedings of the Association for Computational Linguistics*.

Kenny Malone. 2019. How uncle Jamie broke Jeopardy. *Planet Money*, (912).

Adam Najberg. 2018. Alibaba AI model tops humans in reading comprehension.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the Association for Computational Linguistics*.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the Association for Computational Linguistics*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of Empirical Methods in Natural Language Processing*.

Sudha Rao and Hal Daumé III. Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. In *Proceedings of the Association for Computational Linguistics*.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Pedro Rodriguez, Shi Feng, Mohit Iyyer, He He, and Jordan Boyd-Graber. 2019. Quizbowl: The case for incremental question answering. *CoRR*, abs/1904.04792.

Marc-Antoine Rondeau and T. J. Hazen. 2018. Systematic error analysis of the Stanford question answering dataset. In *Proceedings of the Workshop on Machine Reading for Question Answering*.

Andrew Ruef, Michael Hicks, James Parker, Dave Levin, Michelle L. Mazurek, and Piotr Mardziel. 2016. Build it, break it, fix it: Contesting secure development. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*.

Dan Russell. 2020. Why control-F is the single most important thing you can teach someone about search. *SearchReSearch*.

Lucia Specia and Atefeh Farzindar. 2010. A.: Estimating machine translation post-editing effort with HTER. In *In: AMTA 2010 Workshop, Bringing MT to the User: MT Research and the Translation Industry. The 9th Conference of the Association for Machine Translation in the Americas*.

Saku Sugawara, Kentaro Inui, Satoshi Sekine, and Akiko Aizawa. 2018. What makes reading comprehension questions easier? In *Proceedings of Empirical Methods in Natural Language Processing*.

Saku Sugawara, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2020. Assessing the benchmarking capacity of machine reading comprehension datasets. In *Association for the Advancement of Artificial Intelligence*.

Idan Szpektor and Gideon Dror. 2013. From query to question in one click: Suggesting synthetic questions to searchers. In *Proceedings of the World Wide Web Conference*.

David Taylor, Colin McNulty, and Jo Meek. 2012. Your starter for ten: 50 years of University Challenge.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal, editors. 2018. *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2019. The FEVER2.0 shared task. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*.

Ellen M. Voorhees. 2003. Evaluating the evaluation: A case study using the TREC 2002 question answering track. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

Ellen M. Voorhees. 2008. *Evaluating Question Answering System Performance*, pages 409–430. Springer Netherlands, Dordrecht.

Ellen M Voorhees and Dawn M Tice. 2000. Building a question answering test collection. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*.

Eric Wallace, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. 2019. Trick me if you can: Human-in-the-loop generation of adversarial question answering examples. *Transactions of the Association of Computational Linguistics*, 10.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Conference on Computational Natural Language Learning*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the Association for Computational Linguistics*.

Xingdi Yuan, Jie Fu, Marc-Alexandre Cote, Yi Tay, Christopher Pal, and Adam Trischler. 2019. Interactive machine comprehension with information seeking agents.

Annie Zaenen. 2006. Mark-up barking up the wrong tree. *Computational Linguistics*, 32(4):577–580.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the Association for Computational Linguistics*.

Appendix

Footnote numbers continue from main article.

A An Abridged History of Modern Trivia

In the United States, modern trivia exploded immediately after World War II via countless game shows including College Bowl (Baber, 2015), the precursor to Quizbowl. The craze spread to the United Kingdom in a bootlegged version of Quizbowl called *University Challenge* (now licensed by ITV) and pub quizzes (Taylor et al., 2012).

The initial explosion, however, was not without controversy. A string of cheating scandals, most notably the Van Doren (Freedman, 1997) scandal (the subject of the film *Quiz Show*), and the 1977 entry of Quizbowl into intercollegiate competition forced trivia to "grow up". Professional organizations and more "grownup" game shows like *Jeopardy!* (the "all responses in the form of a question" gimmick grew out of how some game shows gave contestants the answers) helped created formalized structures for trivia.

As the generation that grew up with formalized trivia reached adulthood, they sought to make the outcomes of trivia competitions more rigorous, eschewing the randomness that makes for good television. Organizations like National Academic Quiz Tournaments and the Academic Competition Federation created routes for the best players to help direct how trivia competitions would be run. In 2019, these organizations have institutionalized the best practices of "good trivia" described here.

B Simulating the Test Set Needed

We simulate a head-to-head trivia competition where System A and System B have an accuracy a (probability of getting a question right) separated by some difference: $a_A - a_B \equiv \Delta$. We then simulate this on a test set of size N—scaled by the effective dataset proportion ρ—via draws from two Binomial distributions with success probabilities of a_A and a_B:

$$R_a \sim \text{Binomial}(\rho N, a_A$$
$$R_b \sim \text{Binomial}(\rho N, a_B) \tag{1}$$

and see the minimum test set questions (using an experiment size of 5000) needed to detect the better system 95% of the time (i.e., the minimum N such that $R_a > R_b$ from Equation 1 in 0.95 of the experiments). Our emphasis, however is ρ: the smaller the percentage of discriminative questions (either because of difficulty or because of annotation error), the larger your test set must be.[16]

C Qualitative Analysis Examples

We provide some concrete examples for the classes into which we classified the XLNet-123 wins over XLNet. We indicate gold answer spans (provided by the human annotators) by underlining (there may be, **the XLNet answer span** by bold face, and *the* XLN*et-123 answer span* by italics, **combining** *for tokens shared* between spans as is appropriate.

C.1 Insignificant and significant span differences

> **QUESTION:** What type of vote must the Parliament have to either block or suggest changes to

---

[16]Disclaimer: This should be only one of many considerations in deciding on the size of your test set. Other factors may include balancing for demographic properties, covering linguistic variation, or capturing task-specific phenomena.

---

the Commission's proposals?
> **CONTEXT:** The essence is there are three readings, starting with a Commission proposal, where the Parliament must vote by **a *majority* of all MEPs** (not just those present) to block or suggest changes

a majority of all MEPs is as good an answer as *majority*, yet its Exact Match score is 0. Note that the problem is not merely one of picking a soft metric; even its Token-F1 score is merely 0.4, effectively penalizing a system for giving a more complete answer. The limitations of Token-F1 become even clearer in light of the following significant span difference:

> **QUESTION:** What measure of a computational problem broadly defines the inherent difficulty of the solution?
> **CONTEXT:** A problem is regarded as inherently difficult *if its solution requires **significant resources**,* whatever the algorithm used.

We agree with the automatic evaluation that a system answering **significant resources** to this question should not be given full (and possibly no) credit as it fails to mention relevant context. Nevertheless, the Token-F1 of this answer is 0.57, i.e., larger than for the insignificant span difference just discussed.

C.2 Missing Gold Answers

We also observed 7 (out of 100) cases of missing gold answers. As an example, consider

> **QUESTION:** What would someone who is civilly disobedient do in court?
> **CONTEXT:** Steven Barkan writes that if defendants *plead not guilty*, "they must decide whether their primary goal will be to win an acquittal and avoid imprisonment or a fine, or to use the proceedings as a forum to inform the jury and the public of the political circumstances surrounding the case and their reasons for breaking the law via civil disobedience." [. . .]
> In countries such as the United States whose laws guarantee the right to a jury trial but do not excuse lawbreaking for political purposes, some civil disobedients seek **jury nullification**.

While annotators did mark two distinct spans as gold answers, they ignored **jury nullification** which is a fine answer to the question and should be rewarded. Reasonable people can disagree whether this is a missing answer or if it is excluded by a subtlety in the question's phrasing. This is precisely the point—relying on a pre-collected answer strings without a process for adjudicating disagreements in official comparisons does not do justice to the complexity of question answering.

C.3 Bad Questions

We also observed 5 cases of genuinely bad questions. Consider

> **QUESTION:** What library contains the Selmur Productions catalogue?
> **CONTEXT:** Also part of **the library** is the aforementioned *Selznick* library, the Cinerama Productions/Palomar theatrical library and the Selmur Productions catalog that the network acquired some years back

This is simply an annotation error—the correct answer to the question is not available from the paragraph and would have to be (the American Broadcast Company's) Programming Library. While we have to live with annotation errors as part of reality, it is not clear that we ought to simply accept them for *official evaluations*—any human taking a closer look at the paragraph, as part of an adjudication process, would concede that the question is problematic.

Other cases of 'annotation' error are more subtle, involving meaning-changing typos, for example:

> **QUESTION:** Which French kind [sic] issued this declaration?
> **CONTEXT:** They retained the religious provisions of the Edict of Nantes until the rule of <u>*Louis XIV*</u>, who progressively increased persecution of them until he issued the Edict of Fontainebleau (1685), which abolished all legal recognition of **Protestantism** in France

While one could debate whether or not systems ought to be able to do 'charitable' reinterpretations of the question text, this is part of the point—cases like these warrant discussion and should not be silently glossed over when 'computing the score'.