QUOREF: A Reading Comprehension Dataset with Questions Requiring Coreferential Reasoning

Pradeep Dasigi[♡] **Nelson F. Liu**[♡♣] **Ana Marasović**[♡]
Noah A. Smith[♡♣] **Matt Gardner**[♡]
[♡]Allen Institute for Artificial Intelligence
[♣]Paul G. Allen School of Computer Science & Engineering, University of Washington
{pradeepd,anam,mattg}@allenai.org, {nfliu,nasmith}@cs.washington.edu

Abstract

Machine comprehension of texts longer than a single sentence often requires coreference resolution. However, most current reading comprehension benchmarks do not contain complex coreferential phenomena and hence fail to evaluate the ability of models to resolve coreference. We present a new crowdsourced dataset containing more than 24K span-selection questions that require resolving coreference among entities in over 4.7K English paragraphs from Wikipedia. Obtaining questions focused on such phenomena is challenging, because it is hard to avoid lexical cues that shortcut complex reasoning. We deal with this issue by using a strong baseline model as an adversary in the crowdsourcing loop, which helps crowdworkers avoid writing questions with exploitable surface cues. We show that state-of-the-art reading comprehension models perform significantly worse than humans on this benchmark—the best model performance is 70.5 F_1, while the estimated human performance is 93.4 F_1.

1 Introduction

Paragraphs and other longer texts typically make multiple references to the same entities. Tracking these references and resolving coreference is essential for full machine comprehension of these texts. Significant progress has recently been made in reading comprehension research, due to large crowdsourced datasets (Rajpurkar et al., 2016; Bajaj et al., 2016; Joshi et al., 2017; Kwiatkowski et al., 2019, *inter alia*). However, these datasets focus largely on understanding local predicate-argument structure, with very few questions requiring long-distance entity tracking. Obtaining such questions is hard for two reasons: (1) teaching crowdworkers about coreference is challenging, with even experts disagreeing on its nuances (Pradhan et al., 2007; Versley, 2008; Recasens et al., 2011; Poesio et al., 2018), and (2) even if we can get crowdworkers to target coreference phenomena in their questions, these questions may contain giveaways that let models arrive at the correct answer without performing the desired reasoning (see §3 for examples).

Byzantines were avid players of tavli (Byzantine Greek: τάβλη), a game known in English as backgammon, which is still popular in former Byzantine realms, and still known by the name tavli in Greece. Byzantine nobles were devoted to horsemanship, particularly ***tzykanion***, now known as *polo*. The game came from Sassanid Persia in the early period and a Tzykanisterion (stadium for playing *the game*) was built by Theodosius II (r. 408–450) inside the Great Palace of Constantinople. Emperor Basil I (r. 867–886) excelled at **it**; Emperor Alexander (r. 912–913) died from exhaustion while playing, Emperor Alexios I Komnenos (r. 1081–1118) was injured while playing with Tatikios, and John I of Trebizond (r. 1235–1238) died from a fatal injury during a game. Aside from Constantinople and Trebizond, other Byzantine cities also featured tzykanisteria, most notably Sparta, Ephesus, and Athens, an indication of a thriving urban aristocracy.

Q1. What is the Byzantine name of the game that Emperor Basil I excelled at? **it → tzykanion**
Q2. What are the names of the sport that is played in a Tzykanisterion? *the game → tzykanion*; *polo*
Q3. What cities had tzykanisteria? cities → Constantinople; Trebizond; Sparta; Ephesus; Athens

Figure 1: Example paragraph and questions from the dataset. Highlighted text in paragraphs is where the questions with matching highlights are anchored. Next to the questions are the relevant coreferent mentions from the paragraph. They are bolded for the first question, italicized for the second, and underlined for the third in the paragraph.

We introduce a new dataset, QUOREF,[1] that contains questions requiring coreferential reasoning (see examples in Figure 1). The questions are derived from paragraphs taken from a diverse set of English Wikipedia articles and are collected using an annotation process (§2) that deals with the aforementioned issues in the following ways:

[1]Links to dataset, code, models, and leaderboard available at https://allennlp.org/quoref.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 5925–5932, Hong Kong, China, November 3–7, 2019. ©2019 Association for Computational Linguistics

First, we devise a set of instructions that gets workers to find anaphoric expressions and their referents, asking questions that connect two mentions in a paragraph. These questions mostly revolve around traditional notions of coreference (Figure 1 Q1), but they can also involve referential phenomena that are more nebulous (Figure 1 Q3). Second, inspired by Dua et al. (2019), we disallow questions that can be answered by an adversary model (uncased base BERT, Devlin et al., 2019, trained on SQuAD 1.1, Rajpurkar et al., 2016) running in the background as the workers write questions. This adversary is not particularly skilled at answering questions requiring coreference, but can follow obvious lexical cues—it thus helps workers avoid writing questions that shortcut coreferential reasoning.

QUOREF contains more than 24K questions whose answers are spans or sets of spans in 4.7K paragraphs from English Wikipedia that can be arrived at by resolving coreference in those paragraphs. We manually analyze a sample of the dataset (§3) and find that 78% of the questions cannot be answered without resolving coreference. We also show (§4) that the best system performance is 70.5% F_1, while the estimated human performance is 93.4%. These findings indicate that this dataset is an appropriate benchmark for coreference-aware reading comprehension.

2 Dataset Construction

Collecting paragraphs We scraped paragraphs from Wikipedia pages about English movies, art and architecture, geography, history, and music. For movies, we followed the list of English language films,[2] and extracted plot summaries that are at least 40 tokens, and for the remaining categories, we followed the lists of featured articles.[3] Since movie plot summaries usually mention many characters, it was easier to find hard QUOREF questions for them, and we sampled about 40% of the paragraphs from this category.

Crowdsourcing setup We crowdsourced questions about these paragraphs on Mechanical Turk. We asked workers to find two or more co-referring spans in the paragraph, and to write questions such that answering them would require the knowledge that those spans are coreferential. We did not ask them to explicitly mark the co-referring spans. Workers were asked to write questions for a random sample of paragraphs from our pool, and we showed them examples of good and bad questions in the instructions (see Appendix A). For each question, the workers were also required to select one or more spans in the corresponding paragraph as the answer, and these spans are not required to be same as the coreferential spans that triggered the questions.[4] We used an uncased base BERT QA model (Devlin et al., 2019) trained on SQuAD 1.1 (Rajpurkar et al., 2016) as an adversary running in the background that attempted to answer the questions written by workers in real time, and the workers were able to submit their questions only if their answer did not match the adversary's prediction.[5] Appendix A further details the logistics of the crowdsourcing tasks. Some basic statistics of the resulting dataset can be seen in Table 1.

	Train	Dev.	Test
Number of questions	19399	2418	2537
Number of paragraphs	3771	454	477
Avg. paragraph len (tokens)	384±105	381±101	385±103
Avg. question len (tokens)	17±6	17±6	17±6
Paragraph vocabulary size	57648	18226	18885
Question vocabulary size	19803	5579	5624
% of multi-span answers	10.2	9.1	9.7

Table 1: Key statistics of QUOREF splits.

3 Semantic Phenomena in QUOREF

To better understand the phenomena present in QUOREF, we manually analyzed a random sample of 100 paragraph-question pairs. The following are some empirical observations.

Requirement of coreference resolution We found that 78% of the manually analyzed questions cannot be answered without coreference resolution. The remaining 22% involve some form of coreference, but do not require it to be resolved for answering them. Examples include a paragraph that mentions only one city, "*Bristol*", and a sentence that says "*the city was bombed*". The associated question, *Which city was bombed?*, does not really require coreference resolution from a model

[2] https://en.wikipedia.org/wiki/Category:English-language_films

[3] https://en.wikipedia.org/wiki/Wikipedia:Featured_articles

[4] For example, the last question in Table 2 is about the coreference of {*she*, *Fania*, *his mother*}, but none of these mentions is the answer.

[5] Among models with acceptably low latency, we qualitatively found uncased base BERT to be the most effective.

Reasoning	Paragraph Snippet	Question	Answer
Pronominal resolution (69%)	Anna and Declan eventually make their way on foot to a roadside pub, where they discover **the three van thieves** going through Anna's luggage. **Declan fights *them***, displaying unexpected strength for a man of his size, and retrieves Anna's bag.	Who does **Declan get into a fight** with?	the three van thieves
Nominal resolution (54%)	Later, **Hippolyta** was granted a daughter, Princess Diana, . . . **Diana defies *her mother*** and . . .	What is the name of the person who is **defied by her daughter**?	Hippolyta
Multiple resolutions (32%)	The now upbeat collective keep **the toucan**, nicknaming it "**Amigo**" . . . When authorities show up to catch **the bird**, Pete and Liz spirit **him** away by **Liz hiding *him* in her dress** . . .	What is the name of the character who **hides in Liz's dress**?	Amigo
Commonsense reasoning (10%)	Amos reflects back on his early childhood . . . with **his mother *Fania* and father Arieh**. . . . One of **his mother**'s friends is killed while hanging up laundry during the war. . . . **Fania** falls into a depression. . . **she** . . . goes to . . . Tel Aviv, where **she** kills **herself** by overdose . . .	How does **Arieh's wife** die?	kills herself by overdose

Table 2: Phenomena in QUOREF. Note that the first two classes are not disjoint. In the final example, the paragraph does not explicitly say that *Fania* is *Arieh*'s wife.

that can identify city names, making the content in the question after *Which city* unnecessary.

Types of coreferential reasoning Questions in QUOREF require resolving pronominal and nominal mentions of entities. Table 2 shows percentages and examples of analyzed questions that fall into these two categories. These are not disjoint sets, since we found that 32% of the questions require both (row 3). We also found that 10% require some form of commonsense reasoning (row 4).

4 Baseline Model Performance on QUOREF

We evaluated two classes of baseline models on QUOREF: state-of-the art reading comprehension models that predict single spans (§4.1) and heuristic baselines to look for annotation artifacts (§4.2).

We use two evaluation metrics to compare model performance: exact match (EM), and a (macro-averaged) F_1 score that measures overlap between a bag-of-words representation of the gold and predicted answers. We use the same implementation of EM as SQuAD, and we employ the F_1 metric used for DROP (Dua et al., 2019). See Appendix B for model training hyperparameters and other details.

Method	Dev		Test	
	EM	F_1	EM	F_1
Heuristic Baselines				
passage-only BERT QA	19.77	25.56	21.25	29.27
passage-only XLNet QA	18.44	27.59	18.57	28.24
Reading Comprehension				
QANet	34.41	38.26	34.17	38.90
QANet+BERT	43.09	47.38	42.41	47.20
BERT QA	58.44	64.95	59.28	66.39
XLNet QA	**64.52**	**71.49**	**61.88**	**70.51**
Human Performance	-	-	86.75	93.41

Table 3: Performance of various baselines on QUOREF, measured by exact match (EM) and F_1. Boldface marks the best systems for each metric and split.

4.1 Reading Comprehension Models

We test four single-span (SQuAD-style) reading comprehension models: (1) **QANet** (Yu et al., 2018), currently the best-performing published model on SQuAD 1.1 without data augmentation or pretraining; (2) **QANet + BERT**, which enhances the QANet model by concatenating frozen BERT representations to the original input embeddings; (3) **BERT QA** (Devlin et al., 2019), the adversarial baseline used in data construction, and (4) **XLNet QA** (Yang et al., 2019), another large pretrained language model based on the Transformer architecture (Vaswani et al., 2017) that outperforms BERT QA on reading comprehension

benchmarks SQuAD and RACE (Lai et al., 2017).

We use the AllenNLP (Gardner et al., 2018) implementation of QANet modified to use the marginal objective proposed by Clark and Gardner (2018) and `pytorch-transformers`[6] implementation of base BERT QA[7] and base XLNet QA. BERT is pretrained on English Wikipedia and BookCorpus (Zhu et al., 2015) (3.87B wordpieces, 13GB of plain text) and XLNet additionally on Giga5 (Napoles et al., 2012), ClueWeb 2012-B (extended from Smucker et al., 2009), and Common Crawl[8] (32.89B wordpieces, 78GB of plain text).

4.2 Heuristic Baselines

In light of recent work exposing predictive artifacts in crowdsourced NLP datasets (Gururangan et al., 2018; Kaushik and Lipton, 2018, *inter alia*), we estimate the effect of predictive artifacts by training BERT QA and XLNet QA to predict a single start and end index given only the passage as input (**passage-only**).

4.3 Results

Table 3 presents the performance of all baseline models on QUOREF.

The best performing model is XLNet QA, which reaches an F_1 score of 70.5 in the test set. However, it is still more than 20 F_1 points below human performance.[9]

BERT QA trained on QUOREF under-performs XLNet QA, but still gets a decent F_1 score of 66.4. Note that BERT QA trained on SQuAD would have achieved an F_1 score of 0, since our dataset was constructed with that model as the adversary. The extent to which BERT QA does well on QUOREF might indicate its capacity for coreferential reasoning that was not exploited when it was trained on SQuAD (for a detailed discussion of this phenomenon, see Liu et al., 2019). Our analysis of model errors in §4.4 shows that some of the improved performance may also be due to artifacts in QUOREF.

We notice smaller improvements from XLNet QA over BERT QA (4.12 in F_1 test score, 2.6 in EM test score) on QUOREF compared to other reading comprehension benchmarks: SQuAD and RACE (see Yang et al., 2019). This might indicate the insufficiency of pretraining on more data (XL Net was pretrained on 6 times more plain text, nearly 10 times more wordpieces than BERT), for coreferential reasoning.

The passage-only baseline under-performs all other systems; examining its predictions reveals that it almost always predicts the most frequent entity in the passage. Its relatively low performance, despite the tendency for Wikipedia articles and passages to be written about a single entity, indicates that a large majority of questions likely require coreferential reasoning.

4.4 Error Analysis

We analyzed the predictions from the baseline systems to estimate the extent to which they really understand coreference.

Since the contexts in QUOREF come from Wikipedia articles, they are often either about a specific entity, or are narratives with a single protagonist. We found that the baseline models exploit this property to some extent. We observe that 51% of the QUOREF questions in the development set that were correctly answered by BERT QA were either the first or the most frequent entity in the paragraphs, while in the case of those that were incorrectly answered, this value is 12%. XLNet QA also exhibits a similar trend, with the numbers being 48% and 11%, respectively.

QA systems trained on QUOREF often need to find entities that occur far from the locations in the paragraph at which that the questions are anchored. To assess whether the baseline systems exploited answers being close, we manually analyzed predictions of BERT QA and XLNet QA on 100 questions in the development set, and found that the answers to 17% of the questions correctly answered by XLNet QA are the nearest entities, whereas the number is 4% for those incorrectly answered. For BERT QA, the numbers are 17% and 6% respectively.

5 Related Work

Traditional coreference datasets Unlike traditional coreference annotations in datasets like those of Pradhan et al. (2007), Ghaddar and Langlais (2016), Chen et al. (2018) and Poesio et al. (2018), which aim to obtain complete coref-

[6] https://github.com/huggingface/pytorch-transformers

[7] The large BERT model does not fit in the available GPU memory.

[8] https://commoncrawl.org/

[9] Human performance was estimated from the authors' answers of 400 questions from the test set, scored with the same metric used for systems.

erence clusters, our questions require understanding coreference between only a few spans. While this means that the notion of coreference captured by our dataset is less comprehensive, it is also less conservative and allows questions about coreference relations that are not marked in OntoNotes annotations. Since the notion is not as strict, it does not require linguistic expertise from annotators, making it more amenable to crowdsourcing. Guha et al. (2015) present the limitations of annotating coreference in newswire texts alone, and like us, built a non-newswire coreference resolution dataset focusing on Quiz Bowl questions. There is some other recent work (Poesio et al., 2019; Aralikatte and Søgaard, 2019) in crowdsourcing coreference judgments that relies on a relaxed notion of coreference as well.

Reading comprehension datasets There are many reading comprehension datasets (Richardson et al., 2013; Rajpurkar et al., 2016; Kwiatkowski et al., 2019; Dua et al., 2019, *inter alia*). Most of these datasets principally require understanding local predicate-argument structure in a paragraph of text. QUOREF also requires understanding local predicate-argument structure, but makes the reading task harder by explicitly querying anaphoric references, requiring a system to track entities throughout the discourse.

6 Conclusion

We present QUOREF, a focused reading comprehension benchmark that evaluates the ability of models to resolve coreference. We crowdsourced questions over paragraphs from Wikipedia, and manual analysis confirmed that most cannot be answered without coreference resolution. We show that current state-of-the-art reading comprehension models perform significantly worse than humans. Both these findings provide evidence that QUOREF is an appropriate benchmark for coreference-aware reading comprehension.

Acknowledgments

We thank the anonymous reviewers for the useful discussion. Thanks to HuggingFace for releasing `pytorch-transformers`, and to Dheeru Dua for sharing with us the crowdsourcing setup used for DROP.

References

Rahul Aralikatte and Anders Søgaard. 2019. Model-based annotation of coreference. arXiv:1906.10724.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *NeurIPS*.

Hong Chen, Zhenhua Fan, Hao Lu, Alan Yuille, and Shu Rong. 2018. PreCo: A large-scale dataset in preschool vocabulary for coreference resolution. In *Proc. of EMNLP*.

Christopher Clark and Matt Gardner. 2018. Simple and effective multi-paragraph reading comprehension. In *Proc. of ACL*.

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

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 *Proc. of NAACL*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proc. of Workshop for NLP Open Source Software*.

Abbas Ghaddar and Phillippe Langlais. 2016. WikiCoref: An English coreference-annotated corpus of Wikipedia articles. In *Proc. of LREC*.

Anupam Guha, Mohit Iyyer, Danny Bouman, and Jordan Boyd-Graber. 2015. Removing the training wheels: A coreference dataset that entertains humans and challenges computers. In *Proc. of NAACL*.

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

Mandar S. Joshi, Eunsol Choi, Daniel S. Weld, and Luke S. Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proc. of ACL*.

Divyansh Kaushik and Zachary Chase Lipton. 2018. How much reading does reading comprehension require? A critical investigation of popular benchmarks. In *Proc. of EMNLP*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones,

Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: A benchmark for question answering research. In *TACL*.

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

Nelson F. Liu, Roy Schwartz, and Noah A. Smith. 2019. Inoculation by fine-tuning: A method for analyzing challenge datasets. In *Proc. of NAACL*.

Courtney Napoles, Matthew R. Gormley, and Benjamin Van Durme. 2012. Annotated Gigaword. Technical report.

Massimo Poesio, Jon Chamberlain, Silviu Paun, Juntao Yu, Alexandra Uma, and Udo Kruschwitz. 2019. A crowdsourced corpus of multiple judgments and disagreement on anaphoric interpretation. In *Proc. of NAACL*.

Massimo Poesio, Yulia Grishina, Varada Kolhatkar, Nafise Moosavi, Ina Roesiger, Adam Roussel, Fabian Simonjetz, Alexandra Uma, Olga Uryupina, Juntao Yu, and Heike Zinsmeister. 2018. Anaphora resolution with the ARRAU corpus. In *Proc. of CRAC Workshop*.

Sameer S. Pradhan, Lance Ramshaw, Ralph Weischedel, Jessica MacBride, and Linnea Micciulla. 2007. Unrestricted coreference: Identifying entities and events in OntoNotes. In *Proc. of ICSC*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. of EMNLP*.

Marta Recasens, Eduard Hovy, and M Antònia Martí. 2011. Identity, non-identity, and near-identity: Addressing the complexity of coreference. *Lingua*, 121(6):1138–1152.

Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proc. of EMNLP*.

Mark D. Smucker, Charles L. A. Clarke, and Gordon V. Cormack. 2009. Experiments with clueweb09: Relevance feedback and web tracks. In *Proc. of TREC*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. of NeurIPS*.

Yannick Versley. 2008. Vagueness and referential ambiguity in a large-scale annotated corpus. *Research on Language and Computation*, 6(3):333–353.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. arXiv:1906.08237.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. 2018. QANet: Combining local convolution with global self-attention for reading comprehension. In *Proc. of ICLR*.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *Proc. of ICCV*.

Appendices

A Crowdsourcing Logistics

A.1 Instructions

The crowdworkers were giving the following instructions:

"In this task, you will look at paragraphs that contain several phrases that are references to names of people, places, or things. For example, in the first sentence from sample paragraph below, the references Unas and the ninth and final king of Fifth Dynasty refer to the same person, and Pyramid of Unas, Unas's pyramid and the pyramid refer to the same construction. You will notice that multiple phrases often refer to the same person, place, or thing. Your job is to write questions that you would ask a person to see if they understood that the phrases refer to the same entity. To help you write such questions, we provided some examples of good questions you can ask about such phrases. We also want you to avoid questions that can be answered correctly by someone without actually understanding the paragraph. To help you do so, we provided an AI system running in the background that will try to answer the questions you write. You can consider any question it can answer to be too easy. However, please note that the AI system incorrectly answering a question does not necessarily mean that it is good. Please read the examples below carefully to understand what kinds of questions we are interested in."

A.2 Examples of Good Questions

We illustrate examples of good questions for the following paragraph.

"The Pyramid of Unas is a smooth-sided pyramid built in the 24th century BC for the Egyptian pharaoh Unas, the ninth and final king of the Fifth Dynasty. It is the smallest Old Kingdom pyramid, but significant due to the discovery of Pyramid Texts, spells for the king's afterlife incised into the walls of its subterranean chambers. Inscribed for the first time in Unas's pyramid, the tradition of funerary texts carried on in the pyramids of subsequent rulers, through to the end of the Old Kingdom, and into the Middle Kingdom through the Coffin Texts which form the basis of the Book of the Dead. Unas built his pyramid between the complexes of Sekhemket and Djoser, in North Saqqara. Anchored to the valley temple via a nearby lake, a long causeway was constructed to provide access to the pyramid site. The causeway had elaborately decorated walls covered with a roof which had a slit in one section allowing light to enter illuminating the images. A long wadi was used as a pathway. The terrain was difficult to negotiate and contained old buildings and tomb superstructures. These were torn down and repurposed as underlay for the causeway. A significant stretch of Djoser's causeway was reused for embankments. Tombs that were on the path had their superstructures demolished and were paved over, preserving their decorations."

The following questions link pronouns:

Q1: What is the name of the person whose pyramid was built in North Saqqara? **A:** Unas

Q2: What is significant due to the discovery of Pyramid Texts? **A:** The Pyramid of Unas

Q3: What were repurposed as underlay for the causeway? **A:** old buildings; tomb superstructures

The following questions link other references:

Q1: What is the name of the king for whose afterlife spells were incised into the walls of the pyramid? **A:** Unas

Q2: Where did the final king of the Fifth dynasty build his pyramid? **A:** between the complexes of Sekhemket and Djoser, in North Saqqara

A.3 Examples of Bad Questions

We illustrate examples of bad questions for the following paragraph.

"Decisions by Republican incumbent Peter Fitzgerald and his Democratic predecessor Carol Moseley Braun to not participate in the election resulted in wide-open Democratic and Republican primary contests involving fifteen candidates. In the March 2004 primary election, Barack Obama won in an unexpected landslidewhich overnight made him a rising star within the national Democratic Party, started speculation about a presidential future, and led to the reissue of his memoir, Dreams from My Father. In July 2004, Obama delivered the keynote address at the 2004 Democratic National Convention, seen by 9.1 million

viewers. His speech was well received and elevated his status within the Democratic Party. Obama's expected opponent in the general election, Republican primary winner Jack Ryan, withdrew from the race in June 2004. Six weeks later, Alan Keyes accepted the Republican nomination to replace Ryan. In the November 2004 general election, Obama won with 70 percent of the vote. Obama cosponsored the Secure America and Orderly Immigration Act. He introduced two initiatives that bore his name: LugarObama, which expanded the NunnLugar cooperative threat reduction concept to conventional weapons; and the Federal Funding Accountability and Transparency Act of 2006, which authorized the establishment of USAspending.gov, a web search engine on federal spending. On June 3, 2008, Senator Obamaalong with three other senators: Tom Carper, Tom Coburn, and John McCain—introduced follow-up legislation: Strengthening Transparency and Accountability in Federal Spending Act of 2008."

The following questions do not require coreference resolution:

Q1: Who withdrew from the race in June 2004? **A:** Jack Ryan

Q2: What Act sought to build on the Federal Funding Accountability and Transparency Act of 2006? **A:** Strengthening Transparency and Accountability in Federal Spending Act of 2008

The following question has ambiguous answers:

Q1: Whose memoir was called Dreams from My Father? **A:** Barack Obama; Obama; Senator Obama

A.4 Worker Pool Management

Beyond training workers with the detailed instructions shown above, we ensured that the questions are of high quality by selecting a good pool of 21 workers using a two-stage selection process, allowing only those workers who clearly understood the requirements of the task to produce the final set of questions. Both the qualification and final HITs had 4 paragraphs per HIT for paragraphs from movie plot summaries, and 5 per HIT for the other domains, from which the workers could choose. For each HIT, workers typically spent 20 minutes, were required to write 10 questions, and were paid US$7.

B Experimental Setup Details

Unless otherwise mentioned, we adopt the original published procedures and hyperparameters used for each baseline.

BERT QA and XLNet QA We use uncased BERT, and cased XLNet, but lowercase our data while processing. We train our model with a batch size of 10, sequence length of 512 wordpieces, and a stride of 128. We use the AdamW optimizer, with a learning rate of 3^{-5}. We train for 10 epochs, checkpointing the model after 19399 steps. We report the performance of the checkpoint which is the best on the dev set.

QANet Durining training, we truncate paragraphs to 400 (word) tokens during training and questions to 50 tokens. During evaluation, we truncate paragraphs to 1000 tokens and questions to 100 tokens.

Passage-only baseline We keep the HPs setup used for training BERT QA and XLNet QA and replace questions with empty strings.