Question Answering: From the basics to the state-of-the-art with PrimeQA

By: Avi Sil
Principal Research Scientist & Manager

https://github.com/primeqa/primeqa
Question Answering milestones at IBM Research AI

Natural Questions

2011

Rank | Model | Participant | Affiliation |
--- | --- | --- | --- |
1 | BERT-mnlp-ensemble | GAAMA | IBM Research AI |
2 | BERT-mnlp-ensemble | GAAMA | IBM Research AI |
3 | BERT-ensemble | RONOA | Anonymous |
4 | BERT-mnlp-v2-ensemble | DREAM | Anonymous |
5 | BERT-ensemble | GAAMA | IBM Research AI |
6 | BERT-ensemble | GAAMA | IBM Research AI |
7 | BERT-ensemble | GAAMA | IBM Research AI |
8 | BERT-ensemble | GAAMA | IBM Research AI |
9 | BERT-ensemble | GAAMA | IBM Research AI |
10 | BERT-ensemble | GAAMA | IBM Research AI |

2020

Rank | Model | Participant | Affiliation | Attempt Date | F1 |
--- | --- | --- | --- | --- | --- |
1 | GAAMA (XLM-R) with ARES system | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
2 | BERT with language-clustered vocab | Google-Research | Google Research | 6/3/2020 | 63.40 |
3 | mBERT-mnlp-single | GAAMA | IBM Research AI | 8/12/2020 | 53.19 |
4 | tydiqa-baseline | tydiqa-team | Google Research | 2/14/2020 | 52.69 |
5 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
6 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
7 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
8 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
9 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
10 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |

2021-2022

Rank | Model | Participant | Affiliation | Attempt Date | F1 |
--- | --- | --- | --- | --- | --- |
1 | GAAMA (ColBERT Ensemble with IBM NMT + Google MT) | IBM Research AI, NY | 6/18/2021 | 71.4 | 65.0 |
2 | DPR + Google Translate | University of Washington, AI2, Google, UT Austin | 4/11/2021 | 67.2 | 59.3 |
3 | Path Retriever + Google Translate | University of Washington, AI2, Google, UT Austin | 4/11/2021 | 61.7 | 58.2 |
4 | GAAMA (XLM-R) with ARES system | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
5 | BERT with language-clustered vocab | Google-Research | Google Research | 6/3/2020 | 63.40 |
6 | mBERT-mnlp-single | GAAMA | IBM Research AI | 8/12/2020 | 53.19 |
7 | tydiqa-baseline | tydiqa-team | Google Research | 2/14/2020 | 52.69 |
8 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
9 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
10 | BERT-ensemble | GAAMA | IBM Research AI | 11/12/2020 | 66.08 |
What is QA? Reading Comprehension vs Open-Retrieval QA

Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

833,500

Note: ORQA is aka Open Domain QA [Lee et al., 2019] and/or End-2-end QA [Reddy et al., 2021].
Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

833,500
Link to IBM Research’s GAAMA (public) demos

- GAAMA: Go Ahead Ask Me Anything
  - Reading Comprehension (English only): [http://ibm.biz/ibm_gaama](http://ibm.biz/ibm_gaama)
Retrievers – Nuts and bolts
A Traditional Retriever

- A TF-IDF [Robertson 2004] weighted term vector model over unigrams/bi-grams

\[ tf = \text{term frequency, idf = inverse document frequency} \]
\[ t: \text{term (uni/bi), } d: \text{document (= one Wiki. article), } D: \text{corpus (= Wikipedia)} \]

\[ tf-idf(t, d, D) = tf(t, d) \times idf(t, D) \]

\[ tf(t, d) = \log(1 + \text{freq}(t, d)) \]

\[ idf(t, D) = \log \left( \frac{|D|}{|\{d \in D : t \in d\}|} \right) \]

- However, this retriever is not trainable
However, they have limitations!

1. Can NOT answer questions when there’s little or no lexical overlap

   “Who is the bad guy in lord of the rings?”

   “Sala Baker is an actor and stuntman from New Zealand. He is best known for portraying the villain Sauron in the LOTR trilogy by Peter Jackson…”

2. Can NOT retrieve cross-lingual passages without translation (needs special models)

   Q (in Ja): “ロン・ポールの学部”

   Ron Paul  （en.wikipedia）
   Paul went to Gettysburg College, where he was a member of the Lambda Chi Alpha fraternity. He graduated with a B.S. degree in Biology in 1957.

3. Lower performance on some benchmarks

<table>
<thead>
<tr>
<th>Model</th>
<th>NQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>BM25</td>
<td>59.1</td>
</tr>
<tr>
<td>DPR</td>
<td>79.4</td>
</tr>
</tbody>
</table>
Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

Retrieval top-k passages

Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

Perform Retrieval Methods:
- Neural: Dense Passage Retrieval (DPR) [Karpukhin et al., 2020]

Retriever score:

\[ h_q = W_q BERT_Q(q) [CLS] \]
\[ h_b = W_b BERT_B(b) [CLS] \]
\[ S_{retr}(b, q) = h_q^T h_b \]

Evidence block 1: \( s_{retr}(b_1, q) \)

Evidence block 2: \( s_{retr}(b_2, q) \)

Evidence block 3: \( s_{retr}(b_3, q) \)

Evidence block 4: \( s_{retr}(b_4, q) \)

Evidence block 5: \( s_{retr}(b_5, q) \)
Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

Perform Retrieval Methods:
- **Neural: ColBERT** [Khattab et al., 2021]

Demographics
Demographically, Warsaw was the most diverse city in Poland, with a large and thriving Jewish minority. According to the Imperial Census of 1897, out of the total population of 638,000, Jews constituted 219,000 (equivalent to 34%). Prior to the Second World War, Warsaw hosted the world’s second largest Jewish population after New York — approximately 30 percent of the city’s total population in the late 1930s.

In 1933, 833,500 out of 1,178,914 people declared Polish as their mother tongue. There was also a notable German community.

The ethnic composition of contemporary Warsaw is incomparable to the diversity that existed for nearly 300 years.

Most of the modern-day population growth is based on internal migration and urbanisation.

Other countries
In 1939, approximately 1,300,000 people resided in Warsaw; by 1945 the population had dropped to 420,000. During the first years after the war, the population growth rate was high and the city soon began to suffer from the lack of flats and dwellings to house new incomers. The first remedial measure was the enlargement of Warsaw’s total area (1951) — however the city introduce limitations; only the spouses and children of permanent residents as well as some renowned specialists, artists, engineers were permitted to stay. This negatively affected the image of an average Warsaw citizen, who was perceived as more privileged than those migrating from rural areas, towns or other cities. While all restrictions on residency registration were scrapped in 1990, the negative opinion of Varsovians in some form continues to this day.

Immigrant population
Much like most capital cities in Europe, Warsaw boasts a foreign-born population that is significantly larger than in other cities, although not coming close to the figures representing the likes of Madrid or Rome. In 2019, it was estimated that 40,000 people living in Warsaw were born overseas. Of those, Ukrainians, Vietnamese, Belarusians, Russians and Indians were the most prominent groups.

Other choices for Neural Retrievers
Soft-matching of query to document terms

when did the transformers cartoon series come out

The animated Transformers was released in August 1986
Situating ColBERT in the neural IR landscape

(a) Query–Document Interaction

Khattab and Zaharia 2020
PLAID ColBERT results: MS MARCO v1

<table>
<thead>
<tr>
<th>System</th>
<th>MRR@10</th>
<th>R@100</th>
<th>R@1k</th>
<th>Latency (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1-CPU</td>
</tr>
<tr>
<td>BM25 (PISA [34]; k = 1000)</td>
<td>18.7*</td>
<td>-</td>
<td>-</td>
<td>8.3*</td>
</tr>
<tr>
<td>SPLADEv2 (PISA; k = 1000)</td>
<td>36.8*</td>
<td>-</td>
<td>97.9*</td>
<td>220.3*</td>
</tr>
<tr>
<td>ColBERTv1</td>
<td>36.1</td>
<td>87.3</td>
<td>95.2</td>
<td>-</td>
</tr>
<tr>
<td>Vanilla ColBERTv2 (p=2, c=2^{13})</td>
<td>39.7</td>
<td>90.4</td>
<td>96.6</td>
<td>3485.1</td>
</tr>
<tr>
<td>Vanilla ColBERTv2 (p=4, c=2^{16})</td>
<td>39.7</td>
<td>91.4</td>
<td>98.3</td>
<td>-</td>
</tr>
<tr>
<td>PLAID ColBERTv2 (k = 10)</td>
<td>39.4</td>
<td>-</td>
<td>-</td>
<td>185.5</td>
</tr>
<tr>
<td>PLAID ColBERTv2 (k = 100)</td>
<td>39.8</td>
<td>90.6</td>
<td>-</td>
<td>222.3</td>
</tr>
<tr>
<td>PLAID ColBERTv2 (k = 1000)</td>
<td>39.8</td>
<td>91.3</td>
<td>97.5</td>
<td>352.3</td>
</tr>
</tbody>
</table>

Huge speed-ups

No performance loss over ColBERTv2
But do these work in other languages than English?
Problem statement

Cross-lingual Open-Retrieval Question Answering (XOR QA)

- Practical issue in QA: information scarcity and information asymmetry
- XOR QA: Enable questions from one language (non-Eng) to be answered via content from another language (English)
Multilingual Retriever – Training Algorithm

IBM / Google Translation Engine

XLM-RoBERTa

BERT (monolingual English)

Krasnodar has the lowest unemployment rate among the cities of the Southern Federal District at 0.3% of the total working-age population. In addition, Krasnodar holds the first place in terms of highest average salary — 21,742 rubles per capita.

Mayor of Neuilly-sur-Seine from 1983 to 2002, he was Minister of the Budget under Prime Minister Edouard Balladur (1993–1995).

Founded Hanei Maebashi Silk Mill and learned instrumental silk reeling techniques directly from Caspar Müller.


His family emigrated to Marseille in the mid-to-late 1930s. Consequently, it was common for him to serve as prime minister.
Results

We obtained the top position in the leaderboard.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model Description</th>
<th>R@5k</th>
<th>R@2k</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GAAMA (ColBERT ensemble with xlm-r + UW Translate) IBM Research AI, NY</td>
<td>59.9</td>
<td>52.7</td>
</tr>
<tr>
<td>2</td>
<td>DPR + Vanilla Transformer MT University of Washington, AI2, Google, UT Austin</td>
<td>50.0</td>
<td>42.7</td>
</tr>
<tr>
<td>3</td>
<td>Multilingual DPR University of Washington, AI2, Google, UT Austin</td>
<td>48.0</td>
<td>38.8</td>
</tr>
</tbody>
</table>

*Systems using external APIs

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model Description</th>
<th>R@5k</th>
<th>R@2k</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GAAMA (ColBERT Ensemble with IBM NMT + Google MT) IBM Research AI, NY</td>
<td>71.4</td>
<td>65.0</td>
</tr>
<tr>
<td>2</td>
<td>DPR + Google Translate University of Washington, AI2, Google, UT Austin</td>
<td>67.2</td>
<td>59.3</td>
</tr>
<tr>
<td>3</td>
<td>Path Retriever + Google Translate University of Washington, AI2, Google, UT Austin</td>
<td>61.7</td>
<td>58.2</td>
</tr>
</tbody>
</table>

XOR-TyDi
Cross-lingual Open-Retrieval Question Answering
## Multilingual Retriever – Results on XOR TyDi Retrieve

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model</th>
<th>R@5kt</th>
<th>R@2kt</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GAAMA (ColBERT ensemble with xlm-r + UW Translate) IBM Research AI, NY</td>
<td>59.9</td>
<td>52.7</td>
</tr>
<tr>
<td></td>
<td>June 19, 2021</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>DPR + Vanilla Transformer MT University of Washington, AI2, Google, UT Austin</td>
<td>50.0</td>
<td>42.7</td>
</tr>
<tr>
<td></td>
<td>April 11, 2021</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Multilingual DPR University of Washington, AI2, Google, UT Austin</td>
<td>48.0</td>
<td>38.8</td>
</tr>
<tr>
<td></td>
<td>April 11, 2021</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Systems using external APIs*

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model</th>
<th>R@5kt</th>
<th>R@2kt</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GAAMA (ColBERT Ensemble with IBM NMT + Google MT) IBM Research AI, NY</td>
<td>71.4</td>
<td>65.0</td>
</tr>
<tr>
<td></td>
<td>June 16, 2021</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>DPR + Google Translate University of Washington, AI2, Google, UT Austin</td>
<td>67.2</td>
<td>59.3</td>
</tr>
<tr>
<td></td>
<td>April 11, 2021</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Path Retriever + Google Translate University of Washington, AI2, Google, UT Austin</td>
<td>61.7</td>
<td>58.2</td>
</tr>
<tr>
<td></td>
<td>April 11, 2021</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Can I make the multilingual system as good as the monolingual system?

Then I won’t need to translate my incoming queries to English!
Solution: Perform Knowledge distillation

Monolingual ColBERT which needs translated data

F1: 75.0 (yes, we've a better model now!)

Monolingual ColBERT (does NOT need translated data)

F1: 54.7

Let me be the Teacher

Let's feed in parallel data!
Knowledge distillation with Dr. Decr

How?:

- English trained model teacher and have the cross-lingual model (student) learn from the teacher.
- Student: Dr. DECR (Dense Retrieval with Distillation-Enhanced Cross-lingual Representation)

MSE/KLDiv Loss
Knowledge distillation with Dr. Decr

Idea:
– Use English trained model as teacher and have the cross-lingual model (student) learn from the teacher.

Input:
– Teacher: (Eng q, Eng d+, Eng d-)
– Student: (Non-Eng q, Eng d+, Eng d-)

<table>
<thead>
<tr>
<th>Baseline</th>
<th>XOR distillation</th>
<th>Teacher</th>
</tr>
</thead>
<tbody>
<tr>
<td>54.9</td>
<td>66.1</td>
<td>75.1</td>
</tr>
</tbody>
</table>

11.2 points improvement
Enhancement 1: Synthetic data

Idea:
- Using synthetic triples as extra training data in distillation

Created extra 6.5M triples

<table>
<thead>
<tr>
<th></th>
<th>Baseline</th>
<th>Baseline -&gt; XOR distillation</th>
<th>Baseline -&gt; Synthetic data -&gt; XOR distillation</th>
<th>Teacher</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>54.9</td>
<td>66.1</td>
<td>67.7</td>
<td>75.1</td>
</tr>
</tbody>
</table>

Another 1.6 points improvement with synthetic data
Enhancement 2: Parallel Corpus

General issue with IR dataset:
- High quality triples are difficult to make, which limits the training

Can other source of data be used to improve training?
- Parallel corpus? We have a lot!

Idea:
- Instead of teaching student to learn from teacher’s score, have the student to learn from teacher’s vector representation

If the student can produce same vectors as the teacher, their scores $S(q,d)$ will also be the same
Token alignment idea
When teacher and student see different languages, which vector to learn from which?

Idea:
- Align teacher’s output token with student, based on their cosine distances
- Will be noisy but hopefully can still work

During distillation, student sees both the Eng and Non-Eng version of the content
Summary of result
Distillation result Summary:

- 11.2 points improvement from XOR data
- 1.6 points improvement from synthetic data
- 4.4 points improvement from parallel corpus

<table>
<thead>
<tr>
<th>Rank</th>
<th>Model</th>
<th>R@5kt</th>
<th>R@2kt</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DrDeor (IBM Research AI)</td>
<td>70.3</td>
<td>63.0</td>
</tr>
<tr>
<td>2</td>
<td>Sentri 2.0 base (Huawei Noah’s Ark lab)</td>
<td>64.6</td>
<td>58.5</td>
</tr>
<tr>
<td>3</td>
<td>Contrastive Context-aware Pretraining Model (CCP) Anonymous</td>
<td>63.0</td>
<td>54.6</td>
</tr>
<tr>
<td>4</td>
<td>Single Encoder Retriever (Sentri) (Huawei Noah’s Ark lab)</td>
<td>61.0</td>
<td>52.7</td>
</tr>
<tr>
<td>5</td>
<td>Single Encoder Retriever (Sentri, resubmission) (Huawei Noah’s Ark lab)</td>
<td>60.7</td>
<td>55.5</td>
</tr>
</tbody>
</table>

• Another 4.4 points improvement
• In total, 17.2 points improvement
One Limitation of Neural Retrievers

- Problem: Neural retrievers do NOT attend to many important phrases in the passage, e.g., *academy of management* and *twentieth century*.

- Consequence: Low retriever scores for questions that are about these less-attended entities.

- Solution: Biases in retrievers can be overcome by generating synthetic data that is targeted towards these shortcomings.

---

Retrieval scores from DPR for different questions corresponding to the passage in left. Important terms in the question, that are also in the passage, are shown in *italics*.
Approach

Overall framework of our synthetic data generation process:

1. **Wikipedia Passage**
   - Input to the system

2. **NER Model**
   - Identifying Low Attention Entities

3. **DPR Passage Encoder**
   - Entity Attentions

4. **Question Generator**
   - Entity-Conditioned Question Generation
   - Low Attention Entities:
     - Entity 1
     - Entity 2
     - Entity 3
   - Synthetic Questions:
     - Question 1
     - Question 2
     - Question 3

5. **Remove low quality questions**
6. **Retain harder questions**
7. **Question Answering Model**
   - DPR
   - Output: Question 1
Entity-Conditioned Question Generation

- Given a passage and an entity in that passage, we aim to generate a synthetic question about that entity.

- While training the synthetic question generator, entities within questions in existing machine reading comprehension datasets are matched against the passage to identify the conditioning entities.

- While generating synthetic IR data, entities that get lowest attentions from the IR model are used as the conditioning entities.
Experiments

- The model that uses the entity-conditioned questions within its pre-training is named *Mixed-DPR*, and is compared with the baseline DPR.

- We also compare with a model pre-trained on data that contains synthetic questions generated without any conditioning (*UnCon-DPR*).

- We see that Mixed-DPR gives up to 2% more attention to latter sentences of the passage, compared to the baseline DPR model.

<table>
<thead>
<tr>
<th>Model</th>
<th>Natural Questions (NQ)</th>
<th>WebQuestions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Full test</td>
<td>No ans. overlap</td>
</tr>
<tr>
<td>TF-IDF</td>
<td>Top-1 14.2</td>
<td>Top-5 32.0</td>
</tr>
<tr>
<td>BM25</td>
<td>Top-1 22.7</td>
<td>Top-5 44.6</td>
</tr>
<tr>
<td>DPR (ours)</td>
<td>Top-1 44.3</td>
<td>Top-5 67.1</td>
</tr>
<tr>
<td>UnCon-DPR</td>
<td>Top-1 45.8</td>
<td>Top-5 68.4</td>
</tr>
<tr>
<td>Mixed-DPR</td>
<td><strong>Top-1 45.9</strong></td>
<td><strong>Top-5 69.0</strong></td>
</tr>
</tbody>
</table>
Reader – nuts and bolts
Popular choice: Add a fine-tuning layer on top of BERT [Devlin et al., 2019]

In 1933, out of 1,178,914 inhabitants 833,500 were of Polish mother tongue. World War II changed the demographics of the city, ...

How many of Warsaw’s inhabitants spoke Polish in 1933?
Popular choice: Add a fine-tuning layer on top of M-BERT [Bornea et al., 2021]

In 1933, out of 1,178,914 inhabitants 833,500 were of Polish mother tongue. World War II changed the demographics of the city, ...
We want our models to generalize to new, unseen domains.

We have access to multiple source domains with labeled training data.

But how do we train on them to do well on unseen target domains?

- Common advice: Regularize training (cross-domain); focus is on noise.
- We say: Learn your source domains well; focus is on signal.
How do we Learn the Source Domains Better?

Wait, isn’t this the most basic question in machine learning? 😃

We know many ways in which to approach it

How about knowledge distillation?

- Let a bigger model learn the source domains first
  - High capacity, low inductive bias \(\Rightarrow\) better in-domain and OOD generalization
- Then learn from this teacher model, not directly from the data
Domain Generalization: Results on MRQA

MRQA (Fisch et al., 2019):
- Reading comprehension DG benchmark
- 6 source (train, dev) and 6 target (eval) datasets

“Domain-Invariant Learning” Baselines:
- Domain-Adversarial Training (Ganin et al., 2016)
- Episodic Training (Li et al., 2019)
- Meta-learning for DG (Li et al., 2018)

Not underfitting does indeed seem more important than not overfitting!
Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

Retriever

Demographics
Demographically, Warsaw was the most diverse city in Poland, with significant numbers of foreign-born residents. Prior to the Second World War, Warsaw hosted the world’s second largest Jewish population after New York – approximately 30 percent of the city’s total population in the interwar period. Jewish population, which existed for nearly 200 years, was a part of the modern-day population growth is based on internal migration and urbanisation.

Other countries
In 1939, approximately 1,300,000 people resided in Warsaw; by 1945 the population had dropped to 420,000. During the first years after the war, the population growth rate was high and the city soon began to suffer from the lack of flats and dwellings to house new incomers. The first remedial measure was the enlargement of Warsaw’s total area (1951) – however the city authorities were still forced to introduce limitations; only the spouses and children of permanent residents as well as some persons of public importance (renowned specialists, artists, engineers) were permitted to stay. This essentially affected the immigrant population. Much like most capital cities in Europe, Warsaw boasts a foreign-born population, in 2019 estimated at 40,000 people. Of those, Ukrainians, Vietnamese, Belarusians, Russians and Indians were the most prominent groups.

Fusion in Decoder [Izacard & Grave, 2020]
What about QA over multimedia data e.g. images & text?
MuMuQA: Multimedia Multi-hop QA

Given a news article with an image-caption pair and a question, a system needs to answer the question by extracting a short span from the body text.

Answering the questions require multi-hop reasoning:

- The first hop requires cross-media grounding between image and caption to get the bridge item.

- The second hop requires reasoning over the news body text by using the bridge item to extract the final answer.

The benchmark reflects questions that news readers might have after looking at the visual information in the news article, without having read the relatively longer body text.

---

**Question:** What party does the person with the blue tie in the image belong to?

**Answer:** Likud
Speaking during a visit to Poland, Trump said he is not one to draw red lines or talk about his plans but that he has “some pretty severe things” he is thinking about.

On Wednesday, U.S. Ambassador to the United Nations Nikki Haley told Security Council members that the United States is prepare to use military means to defend against the threat posed by North Korea’s launch of an intercontinental ballistic missile.
## Results and Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-hop Text-only QA</td>
<td>25.6</td>
<td>24.8</td>
</tr>
<tr>
<td>End-to-end Multimedia QA</td>
<td>12.1</td>
<td>11.5</td>
</tr>
<tr>
<td>Pipeline-based Multimedia QA</td>
<td>37.3</td>
<td>32.6</td>
</tr>
<tr>
<td>Human Baseline</td>
<td>-</td>
<td>66.5</td>
</tr>
</tbody>
</table>

F1 Performance (%) of different baselines on the MuMuQA evaluation benchmark.

Coming soon to PrimeQA

An example where the grounding system failed to capture the gold bridge item (in **green**). The grounded entity is in **blue** in the caption and its corresponding bounding box is shown in **blue** in the image.
However, how can we perform these experiments easily?

- Is there a SINGLE repository that contains the latest & greatest in QA research already?
- Is that based on HuggingFace’s transformers library?
Welcome Cecilia! A new graduate student

- Cecilia has taken ML 101, NLP 101 and knows basic QA details.

- She has read about: the academic benchmarks for performing QA:
  - Multilingual Machine Reading Comprehension: TyDI [Clark2019]
  - Cross-lingual Open Retrieval: XOR-TyDI [Asai2020]
  - Table QA: WikiSQL [Zhong2017]
Cecilia does some literature survey!

• Cecilia wants to get the latest greatest SOTA models to start with!
  • She sees the following leaderboards ->
    • She reads the following papers: SOTA on the tasks
      • TAPAS --[Herzig2020_ACL] – SOTA on WikiSQL
      • Dr. Decr -- [Li2022_NAACL, Bornea2020_AAAI]– SOTA on XOR TyDI
Cecilia looks for the **source code** to replicate these models.
Wait! Cecilia finds the SOTA model’s source code on TableQA

But this is only Table QA: no other QA use-case

No TyDI
No XOR TyDI
No CoLBERT
Our Objective: Democratize & Replicate QA research

- We need to build a single OPEN-source repository for ALL QA problems
- End-user can use them as Lego blocks for QA problems
- End-user can modify them as per their own needs
- End-user can replicate advanced research papers and leaderboard submissions quickly
PrimeQA has 3 basic scripts

- Run ir.py
- Run mrc.py
- Run qg.py

Question Generators

- General multilingual QG
  - Supported Datasets:
    - SQUAD
    - Tydi

- Table QG
  - Supported Datasets:
    - WikiSQL

Document Retrievers

- BM25
  - Pyserini
  - Supported Datasets:
    - XOR-TyDI

- ColBERT
  - Multilingual support (Dr.Decr)
  - Knowledge Distillation
  - Supported Datasets:
    - XOR-TyDI
    - NQ

- DPR
  - Re-implemented to be license friendly
  - Supported Datasets:
    - NQ
    - XOR TyDI

Extractive

- General MRC (with confidence calibration)
  - Supported Datasets:
    - TyDI
    - NQ
    - SQuAD v1.0
    - MLQA
    - XQuAD

- Special MRC:
  - Supported Datasets:
    - Boolean QA
    - List QA

833,500

With multilingual support!

Q: How many of Warsaw’s inhabitants spoke Polish in 1933?
These are leaderboard winners!

Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

833,500

General multilingual QG:
- Supported Datasets:
  - SQUAD
  - Tydi

Table QG
- Supported Datasets:
  - WikiSQL

Extractive

- BM25: Pyserini
- ColBERT: Multilingual support (Dr. Decr)
- DPR: Re-implemented to be license friendly

General MRC (with confidence calibration):
- Supported Datasets:
  - TyDi
  - QAD v1.0
  - MLQA
  - XSQuAD

Special MRC:
- Supported Datasets:
  - Boolean QA
  - List QA

Open-Retrieval QA (ORQA)
- run_qg.py
- run_mrc.py
- run_ir.py
- Document Retriever

Question Generators

BM25
- ColBERT
- DPR

Supported Datasets:
- XOR-TyDI
PrimeQA full suite [yellow: indicates coming soon]

Open-Retrieval QA (ORQA)

Document Retriever

run_ir.py

Q: How many of Warsaw’s inhabitants spoke Polish in 1933?

Extractive

run_qg.py

run_mrc.py

833,500

BM25

ColBERT

• Multilingual support (Dr.Decr)

• Knowledge Distillation

• XOR-TyDi

• NQ

DPR

• Re-implemented to be license friendly

Supported Datasets:

• XOR-TyDi

• NQ

General MRC

with confidence calibration:

Supported Datasets:

• TyDI

• NQ

• SQuAID v1.0

• MLQA

• XSQuAD

Special MRC:

Supported Datasets:

• Boolean QA

• List QA

General MRC:

Supported Datasets:

• NQ (KILT)

• WoW (KILT)

Generative

Multi-modal

TableQA:

Supported Datasets:

• WikiSQL

• WikiTableQuestions

Vision + Text:

Supported Datasets:

• MumuQA

Question Generators

General multilingual QG:

Supported Datasets:

• SQUAD

• Tydi

Table QG

Supported Datasets:

• XOR-TyDi

Supported Datasets:

• XOR-TyDi

• NQ

• Pyserini

• XOR-TyDi

• XOR-TyDi
https://github.com/primeqa
PrimeQA

The prime repository for state-of-the-art Multilingual and Multimedia Question Answering research and development.

PrimeQA is a public open source repository that enables researchers and developers to train state-of-the-art models for question answering (QA). By using PrimeQA, a researcher can replicate the experiments outlined in a paper published in the latest NLP conference while also enjoying the capability to download pre-trained models (from an online repository) and run them on their own custom data. PrimeQA is built on top of the Transformers toolkit and uses datasets and models that are directly downloadable.

The models within PrimeQA supports End-to-end Question Answering. PrimeQA answers questions via:

- **Information Retrieval**: Retrieving documents and passages using both traditional (e.g. BM25) and neural (e.g. ColBERT) models
- **Multilingual Machine Reading Comprehension**: Extract and/or generate answers given the source document or passage.
- **Multilingual Question Generation**: Supports generation of questions for effective domain adaptation over tables and multilingual text.

Some examples of models (applicable on benchmark datasets) supported are:
Running MRC (predict mode)

- Step 1: Initialize your reader. You can choose any of the MRC models we currently have [here](#).

```python
import json
from primeqa.pipelines.extractive_mrc_pipeline import MRCPipeline
reader = MRCPipeline("PrimeQA/tydiqa-primary-task-xlm-roberta-large")
```

- Step 2: Execute the reader in inference mode:

```python
question = "Which country is Canberra located in?"
context = "Canberra is the capital city of Australia. Founded following the federation of the colonies of Australia as the seat of government for the new nation, it is Australia's largest inland city."
answers = reader.predict(question, context)
print(json.dumps(answers, indent=4))
```

The above statements will generate an output in the form of a dictionary:

```
[
    {
        "span_answer_text": "Australia",
        "confidence_score": 0.798851690240685
    },
    {
        "span_answer_text": "Australia. Founded following the federation of the colonies of Australia has the seat of",
        "confidence_score": 0.1072188935823319
    },
    {
        "span_answer_text": "Australia. Founded following the federation of the colonies of Australia",
        "confidence_score": 0.09392941361769835
    }
]
```
Running MRC (predict mode)

Step 1: Initialize your reader. You can choose any of the MRC models we currently have [here](https://huggingface.co/PrimeQA).

```python
import json
from primeqa.pipelines.extractive_mrc_pipeline import MRCpipeline
reader = MRCpipeline("PrimeQA/tydiqa-primary-task-xlm-roberta-large")
```

Step 2: Execute the reader in inference mode:

```python
question = "Which country is Canberra located in?"
context = "Canberra is the capital city of Australia. Founded following the federation of the colonies of Australia as the seat of government for the new nation, it is Australia's largest inland city."
answers = reader.predict(question, context)
print(json.dumps(answers, indent=4))
```

The above statements will generate an output in the form of a dictionary:

```python
[
    {
        "span_answer_text": "Australia",
        "confidence_score": 0.7988516980240685
    },
    {
        "span_answer_text": "Australia.\nFounded following the federation of the colonies of Australia\nhas the seat of government for the new nation, it is Australia's largest inland city.",
        "confidence_score": 0.1072188963823319
    },
    {
        "span_answer_text": "Australia.\nFounded following the federation of the colonies of Australia",
        "confidence_score": 0.09392941361769635
    }
]
```

You can upload your own MRC models for others to use!
Running MRC (full train + predict)

If your dataset has support for Boolean Questions (e.g. Yes/No) as in TyDI QA you can further run:

```
python examples/mrc/run_mrc.py --model_name_or_path xlm-roberta-large \
  --output_dir ${OUTPUT_DIR} --fp16 --learning_rate 4e-5 \
  --do_train --do_eval --per_device_train_batch_size 16 \
  --per_device_eval_batch_size 128 --gradient_accumulation_steps 4 \
  --warmup_ratio 0.1 --weight_decay 0.1 --save_steps 50000 \
  --overwrite_output_dir --num_train_epochs 1 \
  --evaluation_strategy no --overwrite_cache
```

```
python examples/mrc/run_mrc.py --model_name_or_path PrimeQA/tydiqa-primary-task-xlm-roberta-large \
  --output_dir ${OUTPUT_DIR} --fp16 --overwrite_cache \
  --per_device_eval_batch_size 128 --overwrite_output_dir \
  --do_boolean --boolean_config examples/boolqa/tydi_boolqa_config.json
```

Parameter to run Boolean questions
Running IR

- Training
- Indexing
- Search

```
python examples/ir/run_ir.py
   --engine_type ColBERT
   --do_train
   --triples <data_dir>/xorqa.train_ir_negs_100_pos_3.tsv
   --model_type xlm-roberta-base
   --root <my_dir>/experiments
   --experiment <my_expt>
```

```
python examples/ir/run_ir.py
   --engine_type ColBERT
   --do_search
   --queries <data_dir>/xorqa_dev.tsv
   --index_location <my_expt>_indname
   --model_name_or_path <my_dir>/experiments/<my_expt>/checkpoints/colbert-LAST.dnn
   --output_dir <my_dir>
```
Multilingual Question Generation: Usage

CLI – training and evaluation

```bash
python examples/qg/run_qg.py
--model_name_or_path t5-base
--modality passage
--dataset_name tydiqa
--do_train
--do_eval
--output_dir models/qg/$DIR_NAME
--learning_rate 0.0001
--num_train_epochs 4
```

CLI - generation

```bash
python examples/qg/run_qg.py
--model_name_or_path models/qg/wikisql
--modality table
--do_generate
--num_questions_per_instance 20
--data_path <path-to-json-file>
--generate_aggregate
--max_where_clauses 2
--gen_output_path /results/qg/$DIR_NAME
```

Using pretrained QG model in python code

```python
from primeqa.qg.models.qg_model import QGModel
table_qg_model = QGModel('ibm/t5-base-table-question-generator', modality='table')
table_qg_model.generate_questions(table_list,
    num_questions_per_instance = 10,
    agg_prob = [1.,0,0,0,0,0],
    num_where_prob = [0,1.,0,0,0],
    ineq_prob = 0.0)
```

- One can use QG over table/passage with only 1 primeqa import and 2 code lines.
Running TableQA

```python
from primeqa.tableqa.models.tableqa_model import TableQAModel
import pandas as pd
# Load the pre-trained tapas table-qa model
model = TableQAModel("google/tapas-base-finetuned-wtq")

# Load the Table
data = {"Actors": ["Brad Pitt", "Leonardo Di Caprio", "George Clooney"], "Number of movies": ["87", "53", "69"]}
print(pd.DataFrame.from_dict(data))

# Queries list:
queries = ["What is the name of the first actor?", "How many movies has George Clooney played in?", "Brad Pitt acted in how many movies"]
print(model.predict_from_dict(data, queries))
```

Inference Pipeline
- 2 PrimeQA imports, 3 lines of code.
Use PrimeQA to build your own QA app / search engine

- Head over to https://github.com/primeqa/create-primeqa-app
Conclusion

• To make information access really possible quickly we need to share code and models

• Our software needs to be compatible with one another

• PrimeQA: Let’s work on this together and make QA research move quicker than ever
<table>
<thead>
<tr>
<th>Stanford NLP</th>
<th>University of Illinois</th>
</tr>
</thead>
<tbody>
<tr>
<td>University of Stuttgart</td>
<td>University of Notre Dame</td>
</tr>
<tr>
<td>Ohio State University</td>
<td>Carnegie Mellon University</td>
</tr>
<tr>
<td>University of Massachusetts</td>
<td></td>
</tr>
</tbody>
</table>
Thank you!

- Clone the repo!
- Star and watch the repo
- Get regular updates
- Join the slack channel

https://github.com/primeqa/primeqa
A Simple Assignment: Extra credits

- Get a hands-on experience working with the PrimeQA toolkit/models
- solve an open retrieval question answering task over a real world dataset: covid-qa.
- The target domain is Covid19 related documents/publications, over which PrimeQA/models can answer natural language questions.
- We have designed experiments to focus on domain adaptation aspect of question answering.
- We will provide the Jupyter notebook – just provide the scores you get by running the models
  - Don’t forget to use GPUs 😊
- Office hours: Tuesday and Thursday (Time + Webex to be announced)